Integrated Routing and Charging Scheduling for Autonomous Electric Aerial Vehicle System

Jun Chen
Department of Aerospace Engineering
San Diego State University
San Diego, CA, 92182-1308

***Abstract*—Autonomous electric aerial vehicles (EAVs) are expected to bring fundamental changes to city infrastructures and daily commutes. Currently, the EAVs, including the delivery drones and the electric vertical takeoff and landing (eVTOL) air taxi, are all under limited battery endurance and vertiport capacity, which is insufficient for long-range commutes. In this paper, we propose a joint scheduling methodology to handle the optimal routing and charging tasks for the Autonomous Electric Aerial Vehicle System. By considering the unique characteristics of on-demand EAVs, the joint optimization problem can effectively utilize the system's resources. The problem is formulated by integrating charging features into the classic vehicle routing problem with time windows. To make it easy to solve, we further transform the nonlinear problem to a mixed integer linear program. The problem formulation and the operational constraints are both well validated through comprehensive simulations.**

I. Introduction

A. Motivation

With increasing industrial and government investigations on unmanned aircraft systems (UAS) traffic management (UTM), autonomous electric aerial vehicles (EAVs) are expected to play an important role in the future air transportation system [1]. Under UTM concept, two major UAS operations get most of the attention, which include cargo delivery proposed by Amazon and Google, as well as the electric vertical takeoff and landing (eVTOL) air taxi of urban air mobility (UAM) developed by Uber, Boeing and Airbus [2]. Currently, most of the UTM and UAM operations of the EAV are under limited battery endurance and vertiport capacity, which is insufficient for long-range commutes [1]. Moreover, some reserved battery time may be required for landing in case of possible emergencies.

Given limited battery endurance, it is critical for the autonomous electric aerial vehicle system (AEAVS) to generate the optimal energy efficient routes and the charging plans simultaneously. Typically EAVs are connected, we consider a control center to manage a fleet of EAVs to route and get charged in the transportation network. The control center gathers required information to generate mission plans for each EAV to follow.

In this paper, we propose a joint scheduling methodology to handle the routing and charging tasks for the AEAVS. We formulate an optimization problem to develop schedules for delivering logistic requests, in which vehicle routing and battery charging are jointly considered. Since it is not easy to solve the original nonlinear joint optimization problem, a linear transformation with big M method is applied to get a mix integer linear program (MILP). The major contributions are listed as follows:

- We formulate a joint scheduling problem to handle delivering logistic requests in AEAVS. The optimization problem will determine the optimal routes for EAVs, in which vehicle routing and battery charging are jointly considered.
- We transform the nonlinear formulation to a mix integer linear program to solve the proposed optimization problem.
- We validate the performance of the proposed scheduling problem with extensive simulations considering real-world operation scenarios by comparing with a naive method.

B. Related Work

In literature, delivery of small packages are often modeled as vehicle routing problem (VRP) [3], where a set of trucks are based on a depot and try to serve all delivery requests with minimum travel costs. When drone delivery concept comes out, most previous studies have focused on the routing problem by adopting drones in the logistic service, such as traveling salesman problem (TSP) with drone and truck-drone hybrid delivery system [4]. Inspired by the truck-drone delivery concept, some extensions and variants of this problem are further discussed with improved heuristic solving approaches, see this survey for details [5].

With the potential to significantly change the air transportation, electric drones are gaining more attentions. However, limited battery endurance is still the bottleneck for electrical drones, which makes charging process a key concern for the operation of AEAVS. For To the best of the author's knowledge, there exists little research considering battery charging during the routing process of EAVs. This paper aims to bridge this research gap.

II. Drone System Modeling

The concept of a typical on-demand drone (air taxi) pickup and delivery scenario is shown in Fig 1. Each EAV is initially hosted at a drone terminus/depot, where it is charged and maintained after daily mission. Each EAV starts its mission from the drone terminus and returns to the same terminus.

We assume that the control center will gather and distribute the logistic requests to EAVs using existing request allocation algorithms, such as, [6], [7]. Then based on the current charging station capacity, the State of Charge (SoC) of EAV and logistic requests, it determines the routing and charging plan for each EAV. After getting the plan, each EAV implements the plan until next update. We should notice that the plan can be re-determined if new information is updated.

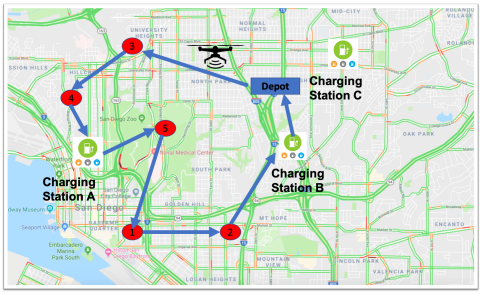


Fig. 1. Concept of flight mission plan with recharging

To model the above scenario, this section introduces the major components of the AEAVS. There are three basic components to model the AEAVS: transportation network (including charging station and terminus depot), EAV, and delivery request:

1) *Transportation Network*: The transportation network is defined as a directed graph $\mathbf{G}(\mathbb{V}, \mathbb{E})$. A node $i \in \mathbb{V}$ represents a service point, charging station or depot. Each edge $(i, j) \in \mathbb{E}$ represents a flight route from i to j, which is associated with a distance D_{ij} (km) and an average travel time T_{ij} (min). Since it is directed graph, D_{ij} and T_{ij} could be different from D_{ji} and T_{ji} to account for asymmetry flight routes.
–*Charging Stations*: There is a set of charging station $\mathbb{V}^c \subset \mathbb{V}$ distributed in the network (other than depots). Each charging station, located at V_c, can provide at most $\Omega_c \geq 0$ (kW) power to charge vehicle. In general, the charging power is limited at charging station and the EAV needs more time to get fully charged than depots. Also there are limited spots (capacity, C_i) for charging at each station.
–*Terminus Depots*: Terminus depots are the base of the EAVs, such as warehouses. The depot set is denoted as $\mathbb{V}^d \subset \mathbb{V}$. Each depot, located at V_d, can provide unlimited charging power to EAV, similar with Tesla supercharger. It is convenient to get charged at the distributed charge station since returning to depot for charging may be inefficient. However, the restricted charging power is a concern. It is an important problem to find an optimal strategy to allocate service requests while maintaining sufficient battery level for safe operation.
2) *Aerial Vehicles*: The set of EAVs in the system denoted as $\mathbb{K}$. Vehicles can have different settings in the system to model different types of EAVs, such as battery capacity and charging rate. Each EAV $k \in \mathbb{K}$ is an associated with a state vector $[L_k, \overline{B}_k, B_k^0, R_k^+, \eta_k, R_k^-]$, where $L_k \in \mathbb{V}$ is the assigned base depot for vehicle k, i.e. it is the initial location and also the final destination for k after all missions, $\overline{B}_k$ denote the battery capacity (kWh), B_k^0 is the initial energy (kWh), R_k^+ is the maximum charging rate (kW), η_k is the charging efficiency (%), and R_k^- is the average energy consumption rate (kWh/min).
3) *Delivery Request*: We denote the set of service request as $\mathbb{Q}$. After request allocation, each EAV has a request subset $\mathbb{Q}_k$. For each request $q \in \mathbb{Q}$, there is an attribute vector $[P_q^0, P_q^*, D_q^0, D_q^*, \underline{T}_q^0, \overline{T}_q^0, \underline{T}_q^*, \overline{T}_q^*]$, where P_q^0, P_q^* denote its pickup and delivery locations, D_q^0, D_q^* are the pickup and delivery service time, $\underline{T}_q^0, \overline{T}_q^0$ is the time window for pickup, and $\underline{T}_q^*, \overline{T}_q^*$ is the time window for delivery.

III. Optimization Problem Formulation

TABLE I
Decision Variables Description

Variable	*Description*
y_{ij}^k	Binary variable indicates if vehicle k will traverse edge (i,j)
$r_i^k(\tau)$	Averaged charging rate of k at node i in time slot τ (kW)
$s_i^k(\tau)$	Binary variable indicates if k is charging at nod i in slot τ
t_i^k	The time when vehicle k arrives at node i
d_i^k	The duration of vehicle k staying at node i
b_i^k	The battery level (SoC) of vehicle k when it arrives at node i

In this section, we formulate the joint request scheduling and charging optimization problem by considering realistic operational constraints. The proposed formulation is inspired by the author's previous work in stochastic vehicle routing problem [8]. By inheriting the idea from the Pickup and Delivery Problem with Time Windows [9] and the Green vehicle routing problem [3], we introduce the energy management process into the formulation. The planning time horizon is discretized into N time slots denoted as $\mathbb{T}$. Each time slots is indexed by τ and the duration of one unit time slot is $\Delta\tau$.

A. Decision Variables

First of all, we define a binary variable y_{ij}^k to present if EAV k will traverse edge $(i, j) \in \mathbb{E}$.

$$y_{ij}^k = \begin{cases} 1 & \text{if k traverses edge (i,j)} \\ 0 & \text{otherwise} \end{cases}$$

To adopt the VRP formulation, we assume that an EAV can visit each node at most once. Actually, we can always create duplicated nodes for the same location if it needs to be visited multiple times, such as the depot.

To consider the time window constraints, two time-related decision variables are introduced as following:

- t_i^k : The time when vehicle k arrives at node i
- d_i^k : The duration of vehicle k staying at node i

Finally, to model charging process three decision variables are defined as following:

- $s_i^k(\tau)$: Binary variable indicates if k is charging at nod i in time slot τ
- b_i^k : The State of Charge (SoC) of vehicle k when it arrives at node i
- $r_i^k(\tau) \geq 0$ *(kW)* : The average charging rate of k at node i in time slot τ.

If k will not visit i in its route, the corresponding decision variables will be ignored. Table I summaries the decision variables.

B. Objective Function

A multi-objective function can be considered in this problem, including travel distance and delivery time. Here we choose to minimize the total travel distance, which is a key concern of classic vehicle routing problems. Longer travel distance implies more energy consumption and heavier traffic, shorter routes are preferred. We can get:

$$D = \sum_{k \in \mathbb{K}} \sum_{(i,j) \in \mathbb{E}} D_{ij} y_{ij}^k$$

Also we encourage to deliver the items as soon as possible and we get:

$$T = \sum_{k \in \mathbb{K}} \sum_{q \in \mathbb{Q}_k} t_{P_q^*}^k$$

Therefore the objective function is defined as:

$$\text{minimize} \quad D + \alpha T$$

where α is the weight coefficient to balance the two objectives.

C. Constraints

1) Routing Constraints:

To fulfill the assigned request, EAV k should reach the corresponding pickup and delivery locations, which is ensured by

$$\sum_{i \in \mathbb{V}} y_{ij}^k \geq 1, \forall k \in \mathbb{K}, q \in \mathbb{Q}_k, j \in \{P_q^0, P_q^*\} \tag{1}$$

where when request q is assigned to k, the incoming flows to pickup and delivery nodes must be greater than one.

Moreover, the vehicle routing based on the network flow model [8], [9] is formulated to ensure flow conservation:

$$\sum_{j \in \mathbb{V}} y_{ji}^k - \sum_{i \in \mathbb{V}} y_{ij}^k = 0 \quad \forall i \in \mathbb{V}, \quad \forall k \in \mathbb{K} \tag{2}$$

Where the $\sum_{j \in \mathbb{V}} y_{ji}^k$ is the number of incoming flows of node i and $\sum_{i \in \mathbb{V}} y_{ij}^k$ represents the outgoing flows. Since each drone starts from its drone terminus and return back to the same terminus, the number of incoming and outgoing flow should be the same for each node. Moreover, the initial terminus and final destination could be different for each drone. The following modifications are needed: If node i is the initial point or the final destination of k, there is an extra outgoing or incoming flow, which will change the 0 to be -1 or +1 respectively.

Moreover, there are some time constraints for the operations. First, the arrival time of each visit (pickup or delivery) should fall into the given time window constraints. Second the duration of stay should be long enough to load/unload the package. Thus we have:

$$\underline{T}_q^0 \leq t_i^k \leq \overline{T}_q^0, d_i^k \geq D_q^0, \quad i = P_q^0, \forall k \in \mathbb{K}, q \in \mathbb{Q}_k \tag{3}$$

$$\underline{T}_q^* \leq t_i^k \leq \overline{T}_q^*, d_i^k \geq D_q^*, \quad i = P_q^*, \forall k \in \mathbb{K}, q \in \mathbb{Q}_k \tag{4}$$

The constraints 5 ensure that the delivery can only happen after the pickup, which is an important condition for drone delivery and often this is ignored in previous related research.

$$t_j^k \geq t_i^k + d_i^k + T_{ij}, \quad i = P_q^0, j = P_q^*, \forall k \in \mathbb{K}, q \in \mathbb{Q}_k \tag{5}$$

To eliminate the sub-tours in the vehicle routing problem, the following travel time constraints should also be considered:

$$t_j^k \geq t_i^k + d_i^k + T_{ij}, \forall k \in \mathbb{K}, (i,j) \in \mathbb{E}, y_{ij}^k = 1 \tag{6}$$

It ensures that when k passes through (i,j), the arrival time at j should be no less than the arrival time at i with the duration of stay at i and the travel time on (i,j).

2) Battery Charging Constraints: Recall that the planning time horizon is discretized and we defined a new variable $s_i^k(\tau)$ to indicate whether EAV k is charging at node i during time slot τ. Obviously, $s_i^k(\tau)$is conditioned on variables t_i^k and d_i^k:

$$s_i^k(\tau) = \begin{cases} 1 & t_i^k \leq \tau\Delta\tau \leq t_i^k + d_i^k - \Delta\tau \quad i \in \mathbb{V}^c \cup \mathbb{V}^d \\ 0 & \text{otherwise} \end{cases}$$
$$\forall k \in \mathbb{K}, \tau \in \mathbb{T} \tag{7}$$

In other words, charging can only start after arrival time of k at i and need to stop before departure time.

The battery energy should be balanced for charging and consumption. Also there are some charging restrictions. We have the following constraints:

$$\overline{B}_k b_i^k + \sum_{\tau \in \mathbb{T}} r_i^k(\tau)\eta_k\Delta\tau - T_{ij}R_k^- = \overline{B}_k b_j^k, \tag{8}$$
$$\forall k \in \mathbb{K}, (i,j) \in \mathbb{E}, y_{ij}^k = 1$$

$$\overline{B}_k b_i^k = B_k^0, \quad \forall k \in \mathbb{K}, i = L_k^0 \tag{9}$$

$$\overline{B}_k b_i^k + \sum_{\tau \in \mathbb{T}} r_i^k(\tau)\eta_k\Delta\tau \leq \overline{B}_k, \quad \forall k \in \mathbb{K}, i \in \mathbb{V}^c, \tag{10}$$

$$r_i^k(\tau) \leq \Omega_c, \quad \forall k \in \mathbb{K}, i \in \mathbb{V}^c, \tau \in \mathbb{T} \tag{11}$$

$$r_i^k(\tau) \leq R_k^+, \quad \forall k \in \mathbb{K}, i \in \mathbb{V}^d, \tau \in \mathbb{T} \tag{12}$$

$$r_i^k(\tau) = 0, \quad \forall k \in \mathbb{K}, i \in \mathbb{V}^c, \tau \in \mathbb{T}, s_i^k(\tau) = 0 \tag{13}$$

$$r_i^k(\tau) = 0, \quad \forall k \in \mathbb{K}, i \in \mathbb{V} \backslash \mathbb{V}^c, \tau \in \mathbb{T} \tag{14}$$

Constraints 8 established how to update the battery level according to flight consumption and charging profile. Constraints 9 is the initial battery level and Constraints 10 ensure the battery will not be overcharged. The charging power should be limited by the maximum charge rate, which is Constraints 11 and 12. Usually, $\Omega_c \leq R_K^+$, means charging station can only provide limited power. Constraints 13 and 14 means that EAVs cannot be charged if they are not charging or not staying at a charging node.

In addition, to ensure there is enough reserved battery for landing in case of possible emergencies, the battery level constraints are formulated:

$$\underline{b^k} \leq b_i^k \leq 1, \quad \forall k \in \mathbb{K}, i \in \mathbb{V} \tag{15}$$

where $\underline{b^k}$ is the required reserved battery level for safety concerns (often 20% to 30%) [2].

Lastly, there is a limited charging capacity at each charge station for each time slot, we have:

$$\sum_{k \in \mathbb{K}} s_i^k(\tau) \leq C_i \quad \forall i \in \mathbb{V}^c, \tau \in \mathbb{T} \tag{16}$$

IV. Methodology

A. Mixed Integer Linear Program Transform

The formulation in Section III cannot be easily handled [10] because of the nonlinear constraints (6), (7), (8) and (13). In specific, Constraints (6), (8) are conditioned on $y_{ij}^k = 1$. Moreover, in (7) $s_i^k(\tau)$ is conditioned on variables t_i^k and d_i^k and Constraints (13) depend on $s_i^k(\tau) = 0$. Therefore, inspired by our previous work with MILP [11], we will adopt the classic "big-M" formulation to transform those constraints into equivalent linear form [12].

First of all, let's define a sufficiently large constant number M, then Constraints (6) and (8) can re-written as:

$$t_j^k \geq t_i^k + d_i^k + T_{ij} - (1 - y_{ij}^k)M, \forall k \in \mathbb{K}, (i,j) \in \mathbb{E} \tag{17}$$

$$\begin{cases} \overline{B}_k b_i^k + \sum_{\tau \in \mathbb{T}} r_i^k(\tau)\eta_k \Delta\tau - T_{ij}R_k^- - (1 - y_{ij})M \leq \overline{B}_k b_j^k \\ \overline{B}_k b_i^k + \sum_{\tau \in \mathbb{T}} r_i^k(\tau)\eta_k \Delta\tau - T_{ij}R_k^- + (1 - y_{ij})M \geq \overline{B}_k b_j^k \end{cases}$$
$$\forall k \in \mathbb{K}, (i,j) \in \mathbb{E} \tag{18}$$

The charge index variable can be re-defined in this way:

$$\begin{cases} s_i^k(\tau) \leq 1 - (t_i^k - \tau\Delta\tau)/M \\ s_i^k(\tau) \leq 1 + (t_i^k + d_i^k - \Delta\tau - \tau\Delta\tau)/M \end{cases} \tag{19}$$
$$\forall k \in \mathbb{K}, \tau \in \mathbb{T}, i \in \mathbb{V}$$

Finally, based on the binary variable $s_i^k(\tau)$, we can remove the conditions on $s_i^k(\tau) = 0$ for Constraints (13) as following:

$$r_i^k(\tau) \leq s_i^k(\tau)M, \quad \forall k \in \mathbb{K}, i \in \mathbb{V}^c, \tau \in \mathbb{T}, \tag{20}$$

Therefore, the original problem is transformed into a MILP problem as following:

Optimization 1 (joint scheduling and charging problem):

$$\begin{aligned} &\text{minimize} \qquad D + \alpha T \\ &\text{subject to (1-5), (9-12), (14-16) and (17-20)} \end{aligned}$$

The above MILP problem can be modeled and solved by any MILP optimization solver, such as Gurobi [10] and CPLEX [13]. This paper will adopt Gurobi as the solver for this MILP problem. We notice that the problem can be decomposed vehicle by vehicle if we relax the coupling constraints (16). The decomposition methods and heuristic algorithms will be implemented soon in the coming work.

B. Naive Method

We also described a rule based naive method to operate the EAVs individually to demonstrate how joint scheduling with charging plan will benefit the whole system. The naive method simulates the way how people drive their own vehicles without a centralized plan. After receiving the assigned requests, each EAV will rank the requests by the earliest pickup time and then implement them sequentially. Before moving to the next node, two conditions about battery SoC will be checked. First, it will make sure the SoC will be above the required reserve level (30% in this paper) after arriving; second it will ensure the SoC is enough to travel to the nearest charge station after arriving if the next node is not the final destination. If any of the conditions cannot be met, the EAV will move to the nearest charge station in next step. Then it will continue its missions until enough energy is charged.

V. Simulation Results

A. Test Settings

Some case studies are conducted to evaluate the proposed joint routing and charging model in optimization 1. To make the cases close to realistic operations, the simulation parameters are adopted from some real world examples, such as Amazon Prime Air[1], google Wing [2] and Airbus Vahana [3]. Since most of them are still in the prototype stage, we will adopt the available parameters and calculate the missing ones with reasonable assumptions by taking references from electric cars.

We consider the map to be an urban area, where the nodes of network are randomly generated in a 50km X 50km area, which is a reasonable operational area considering the range of EAVs. All requests are randomly generated where the pickup and delivery locations are set as a node in the network. We randomly set 25% of the nodes as charge stations in the network and 5% of the nodes as depots. The maximum charging rate at charge station is 1.0 kWh/min.

The planning time horizon is set as 1 hour, each time slot is $\Delta\tau = 1min$. The time window for pickup and delivery are generated by randomly getting two values as earliest time from 0 to 45 minutes. Then the latest time is set as 10, 15 or 20 minutes later. The infeasible time window will be discard by checking needed flying time. The loading and unloading time for reach request is set as 5 minutes. Each request are considered as a single package.

The EAVs are considered to have different configurations. The speed of each EAV is set as 4km/min for air taxi and 2km/min for delivery drones. The battery capacity is randomly selected from 30 kWh, 35 kWh and 40kWh. The charge rate is set as 1Kw/min and the charge efficiency is randomly chosen from 0.8 to 0.9. The initial Battery level (SoC) is randomly set between 50% to 100%. The energy consumption rate is randomly get between 1.0 kWh/min to 1.2 kWh/min, which is

[1] https://en.wikipedia.org/wiki/Amazon_Prime_Air
[2] https://wing.com/how-it-works/
[3] https://vahana.aero/

calculated based on the normal battery capacity and maximum flight duration.

The proposed simulations are all conducted on a computer with Intel i7-8700k CPU and 16GB RAM. The code is written in Python. Without specific setting, the default α is set as 1.

B. A simple Case

To clearly demonstrate how the model works, a simple scenario with 4 drones (eVTOL) and 8 requests are given in Figure 2. All drones have the same battery capacity (30kWh) and the initial SoC is 1. Each drone is assigned with 2 requests and each request has required pickup and delivery time window. All four drones are based in depot 0, which means they start from depot 0 and need to fly back to depot 0 after all assigned missions. Additionally, there are two distributed charging stations, node 17 and 18. Each charging station can hold only one drone at each time slot.

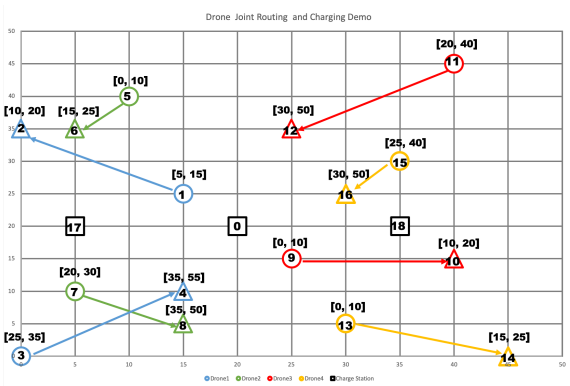


Fig. 2. Case study: pickup and delivery requests with time window

The optimized flight plan with charging schedule is clearly shown in Figure 3, where the exact arrive time is marked along the trajectory of each drone. Each drone find a feasible and optimal routing and charging strategy within the required time window. For comparison, the flight plan with the naive method is also provided in Figure 4, where the corresponding arrive time is also marked. Comparing the two plan, the actual routing plans are quite different, especial the timing to go to charge station. In naive method, EAVs will go to get charged only when they have to. Often they will travel more extra distance and wait more time for a charging slot if it is occupied. However, the optimal method will jointly plan the routing and charge together. EAVs can go to get charged before it is actually needed to avoid waiting and save more travel distance. The comparison results about travel distance and travel time of each EAV are summarized in Table II. The optimal results are better on both distance and time.

Based on the flight plans in Figure 3 and 4, we find drone 1 and 2 both get charged on station 17, and drone 3 and 4 get charged on station 18. The battery level along time for each drone and the corresponding charging profile is demonstrated in Figure 5 and Figure 6. Specifically, we can find two interesting points: 1) All drones are exactly planed to ensure

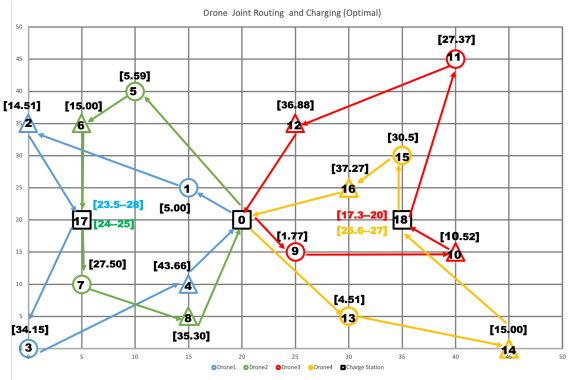


Fig. 3. Optimized flight plan with charging schedule

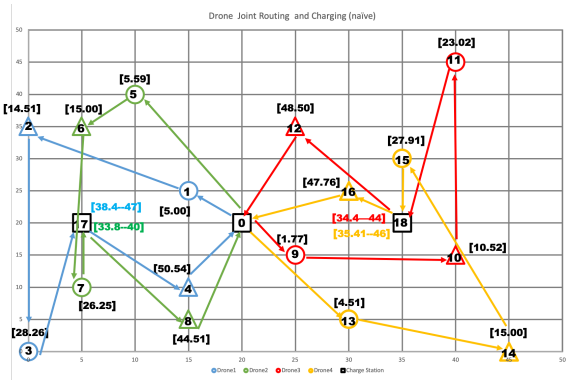
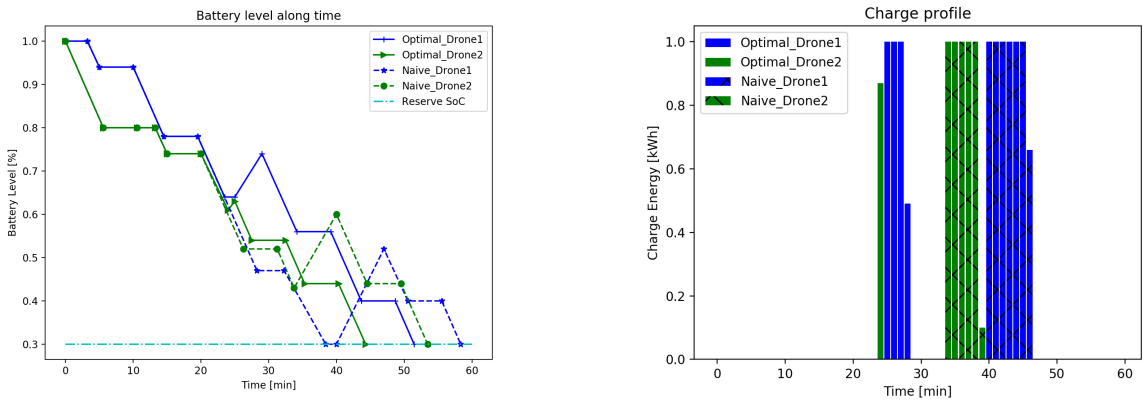


Fig. 4. Naive flight plan with charging schedule

TABLE II
COMPARISON OF NAIVE AND OPTIMAL SOLUTION

Method	ID	Route	Distance(km)	Time(min)
Naive	1	0-1-2-3-17-4-0	106.04	58.34
	2	0-5-6-7-17-8-0	98.27	53.46
	3	0-9-10-11-18-12-0	111.41	57.45
	4	0-13-14-15-18-16-0	93.71	55.56
Optimal	1	0-1-2-17-3-4-0	90.73	51.46
	2	0-5-6-17-7-8-0	81.42	44.31
	3	0-9-10-18-11-12-0	88.48	45.83
	4	0-13-14-18-15-16-0	84.45	45.07

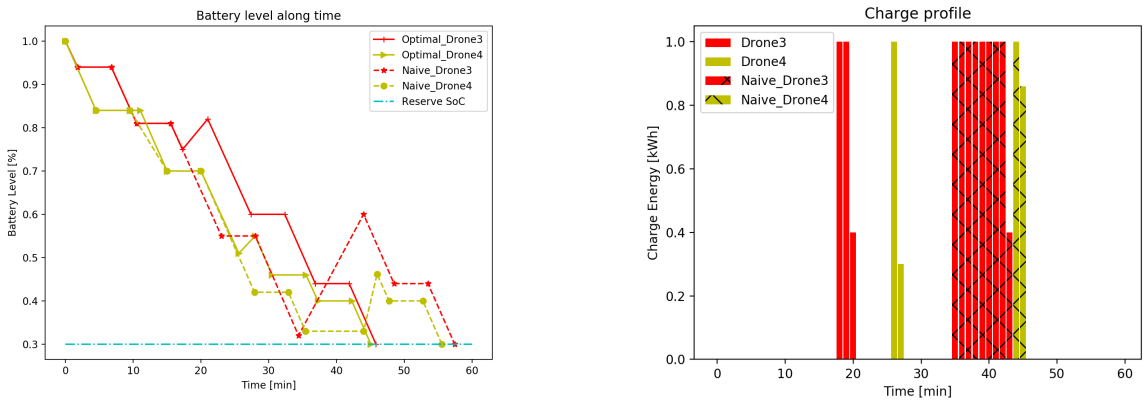
that they will have necessary reserved SoC (30%) when they arrive at their final destination, see Figure 5a and 6a. The optimal method saves unnecessary redundant energy with early and short charge, which proves that our scheduling model can help improve the system's efficiency without wasting limited resources, such as charging spots and charging time. 2) The capacity of the charge station is well allocated in optimal method, see Figure 5b and 6b. Since each time slot only one drone can get charged, drone 1 and drone 4 are hold for several more minutes in the naive plan to ensure drone 2 and drone 3 get the necessary SoC for their whole mission. However, optimal plan coordinates the charging plan to avoid charging peak time.



(a) Battery level along time

(b) Charge profile at charge station 17

Fig. 5. Battery and charging results for drone 1 and 2.



(a) Battery level along time

(b) Charge profile at charge station 18

Fig. 6. Battery and charging results for drone 3 and 4.

C. Computing Time with Different Size

As a MILP model, scalability is often a concern when handling large size problem. To study the scalability, we test the computing time with different size of the problem from two aspects. First, the fleet size is increased and each EAV will serve two requests. An additional node is added as the common drone depot for all drones. The results are shown in Table III. We find the problem quickly becomes difficult to be solved when the fleet size reaches 10. Then we keep the fleet size to be three and different number of requests (2 to 5) are assigned to each EAV. The results are shown in Table IV. Again it is hard to find the solution when the requests per EAV reaches 5. Therefore, it is easy to find that this model cannot easily handle large scale problems. Therefore, some decomposition method with heuristic algorithm need to be developed to improve the scalability in the future work. As we discussed in Section IV, there is only one coupling constraints (16) for all EAVs. A decomposition-based method can be developed by relaxing this constraints. This paper only focuses on the formulation of the joint scheduling and charging problem itself.

TABLE III
COMPUTING TIME FOR PROBLEMS WITH DIFFERENT SIZES OF VEHICLES

Number of Vehicle	Number of requests	Network Size	Time
2	4	9	0.2s
3	6	13	0.5s
4	8	17	1.7s
5	10	21	21.2s
6	12	25	46.5s
7	14	29	82.1s
8	16	33	128s
10	20	41	>1000s

VI. CONCLUSION

In this paper, we introduce an integrated model for routing and charging of autonomous electric aerial vehicle system. This is a general system that can be used for delivery drone or air taxi scheduling problem. By integrating the routing strategy with charging scheduling, we can make this system more efficient to save total travel time, and to use the limited resource,

TABLE IV
COMPUTING TIME FOR PROBLEMS WITH DIFFERENT SIZES OF REQUESTS

Number of Vehicle	Number of requests	Network Size	Time
3	3	7	0.04s
3	6	13	0.5s
3	9	17	1.7s
3	12	25	177.5s
3	15	31	>1000s

such as charging spots. Even though, the current version with MILP model has some computational limitation for large scale problems, it provides a fundamental modeling framework for future research, which will handle the decomposition methods and heuristic algorithms.

REFERENCES

[1] T. Prevot, J. Rios, P. Kopardekar, J. E. Robinson III, M. Johnson, and J. Jung, "Uas traffic management (utm) concept of operations to safely enable low altitude flight operations," in *16th AIAA Aviation Technology, Integration, and Operations Conference*, 2016, p. 3292.

[2] J. Holden and N. Goel, "Fast-forwarding to a future of on-demand urban air transportation," *San Francisco, CA*, 2016.

[3] S. Erdoğan and E. Miller-Hooks, "A green vehicle routing problem," *Transportation Research Part E: Logistics and Transportation Review*, vol. 48, no. 1, pp. 100–114, 2012.

[4] C. C. Murray and A. G. Chu, "The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery," *Transportation Research Part C: Emerging Technologies*, vol. 54, pp. 86–109, 2015.

[5] A. Otto, N. Agatz, J. Campbell, B. Golden, and E. Pesch, "Optimization approaches for civil applications of unmanned aerial vehicles (uavs) or aerial drones: A survey," *Networks*, vol. 72, no. 4, pp. 411–458, 2018.

[6] J.-B. Sheu, "A novel dynamic resource allocation model for demand-responsive city logistics distribution operations," *Transportation Research Part E: Logistics and Transportation Review*, vol. 42, no. 6, pp. 445–472, 2006.

[7] T. Huth and D. C. Mattfeld, "Integration of vehicle routing and resource allocation in a dynamic logistics network," *Transportation Research Part C: Emerging Technologies*, vol. 17, no. 2, pp. 149–162, 2009.

[8] L. Chen, W.-C. Chiang, R. Russell, J. Chen, and D. Sun, "The probabilistic vehicle routing problem with service guarantees," *Transportation Research Part E: Logistics and Transportation Review*, vol. 111, pp. 149–164, 2018.

[9] S. Ropke and D. Pisinger, "An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows," *Transportation science*, vol. 40, no. 4, pp. 455–472, 2006.

[10] Gurobi, *Gurobi optimizer reference manual, (accessed 27/03/19) [Online]. Available: www. gurobi. com*, 2019.

[11] J. Chen, D. DeLaurentis, and D. Sun, "Dynamic stochastic model for converging inbound air traffic," *Journal of Guidance, Control, and Dynamics*, pp. 2273–2283, 2015.

[12] G. Dantzig, *Linear programming and extensions*. Princeton university press, 2016.

[13] CPLEX, *Cplex optimizer reference manual, (accessed 27/03/19) [Online]. Available:https://www.ibm.com/analytics/cplex-optimizer*, 2019.