This is a preprint of the following article, which is available at: http://mdolab.engin.umich.edu
S. Kaneko and J. R. R. A. Martins. Fleet Design Optimization of Package Delivery UAVs Considering Operations. *2022 AIAA SciTech Forum*, January 2022.
The original article may differ from this preprint and is available at:
https://arc.aiaa.org/doi/10.2514/6.2022-1503.

Fleet Design Optimization of Package Delivery UAVs Considering Operations

Shugo Kaneko* and Joaquim R. R. A. Martins†
University of Michigan, Ann Arbor, MI, 48109

The conceptual design process of aircraft starts by deciding the representative mission requirements, followed by optimization of design variables to satisfy the given requirements. However, selecting appropriate mission requirements is not an obvious task, particularly when designing the package delivery UAVs because UAVs must accommodate the various combinations of package weight and delivery distances. The complexity increases further when designing a heterogeneous fleet of UAVs that serves a large number of customers. In this work, we tackle this problem by solving the coupled design-operation optimization to find the optimal mission requirements and the optimal UAV designs simultaneously. We formulate this problem as mixed-integer nonlinear optimization and propose a sequential heuristic algorithm to solve the coupled problem. The benchmark study of the proposed algorithm against a non-convex branch-and-cut solver shows that the sequential heuristics are effective. We also demonstrate that the simultaneous UAV design and routing optimization reduces the fleet acquisition cost by more than 10% on average compared to the conventional baseline.

1. Introduction

The unmanned aerial system (UAS) is a viable option for transporting commercial packages and medical supplies. Unmanned aerial vehicles (UAVs) deliver faster than trucks and are not dependent on ground road networks or traffic conditions. Among several UAV configurations, the electric vertical takeoff and landing (eVTOL) UAV has potential as a solution for lightweight package delivery. One of the active research fields on the eVTOL UAV is the conceptual design methodology, which is not yet established because of the lack of historical data and knowledge.

For package delivery applications, Sridharan et al. [1] performed the conceptual sizing study of quadrotor biplane tailsitter (QBiT) configuration [2] for various payload weights and mission ranges. Govindarajan and Sridharan [3] presented an optimization-based conceptual design approach and applied it to four eVTOL configurations including QBiT and hexarotor. Work on multirotor design with no wing includes Bershadsky et al. [4], Winslow et al. [5], and Delbecq et al. [6].

Regardless of the model or aircraft configuration, the conceptual design process starts by setting appropriate mission requirements, for which engineers optimize the sizing variables. The research question here is: how should engineers choose appropriate mission requirements? This is a challenging task for a package delivery UAV because the UAV needs to perform delivery missions of various package weights and delivery distances. The problem becomes even more complicated when we design a fleet of UAVs that serves a large number of customers. In this work, we tackle this challenge using multidisciplinary design optimization (MDO) to find the optimal mission requirements and the optimal UAV designs at the same time. We achieve this goal by simultaneously optimizing the UAV fleet design and delivery operations for the minimum fleet acquisition cost.

The optimization of delivery operations is called a vehicle routing problem (VRP), an integer optimization problem that seeks the optimal routes of a set of vehicles given the customer demands. The VRP has traditionally been studied for

*Ph.D. Candidate, Aerospace Engineering, AIAA Student Member.
†Professor, Aerospace Engineering, AIAA Fellow.

truck delivery, and recently, there have also been research efforts on UAV routing [7–9]. The recent literature includes truck-UAV collaborative delivery [10–12] as well as the UAV-only delivery [13–16]. One crucial aspect that arises in UAV routing problems is the energy consumption modeling. Because of the limited energy capacity of UAVs, the range or duration is a limiting factor for UAVs, unlike conventional trucks. Dorling et al. [13] solved a UAV routing problem using the linear approximation of the energy consumption model based on the momentum theory in hover. Cheng et al. [16] extended their work by incorporating a nonlinear energy model into UAV routing. There have also been research efforts that assess the sustainability impact of UAV delivery by solving the UAV routing with energy models [11, 17, 18]. Other energy consumption models used in the UAV routing literature are summarized in Zhang et al. [19].

The literature mentioned above all solved VRP to a make short-term tactical decision, i.e., to decide the delivery schedule using an existing fleet. Our goal in this paper is to make a long-term strategic decision on the UAV fleet design via VRP instead of tactical decisions. This type of VRP for a strategic decision is categorized as fleet size and mix VRP (FSMVRP) [20, 21], which aims at finding the optimal number and composition of vehicles in the fleet to maximize long-term profit. Accordingly, the customer information on which FSMVRP is solved should represent the demand over a life cycle of the fleet. Compared to the rich literature of ordinary VRPs for tactical routing, FSMVRP has drawn less attention from the research community; the previous work on UAV FSMVRP is limited to Troudi et al. [22] and Choi [23], where both work studied homogeneous fleet sizing. In the current work, we seek to expand the scope of FSMVRP by coupling it to the UAV conceptual design optimization. To the authors' best knowledge, there has been no literature on the truck or ship FSMVRP that includes the vehicle or vessel design variables. This is not a surprise given that the delivery or transportation company does not usually have the freedom to design the vehicle or vessel by themselves. However, because of the cheap unit cost of UAVs, it is possible for the delivery service providers to also design the UAVs for their own use. The only preceding work of the UAV design-routing optimization is by Choi [23]. They performed sequential optimization of multi-trip VRP and UAV design on a 20-node problem with up to 3 homogeneous UAVs. In the current work, we aim to expand the problem scope by considering a heterogeneous fleet, including more UAV design variables such as cruise speed and payload capacity, and solving larger-scale problems.

Although the application is not package delivery, there have been several works on simultaneous optimization of commercial aircraft design and operations. Taylor and de Weck [24] used simulated annealing in combination with a linear programming solver to optimize the aircraft conceptual design and cargo network flow. Mane et al. [25] proposed a sequential decomposition approach to solve the aircraft sizing and airline allocation optimization, and they compared their approach to a mixed-integer nonlinear programming (MINLP) exact solver. Davendralingam and Crossley [26] used the decomposition method by Mane et al. [25] to perform robust optimization of aircraft design and airline network design. Jansen and Perez [27] also optimized the aircraft design and fleet allocation by decomposing the design-allocation coupled problem. In contrast to the decomposition-based approaches in the aforementioned literature, Hwang et al. [28] used a monolithic MDO architecture [29] to solve the coupled optimization of computational fluid dynamics (CFD)-based aircraft design, airline allocation, and flight trajectory. They used continuous relaxation of the allocation problem to enable large-scale optimization. Roy et al. [30] proposed a surrogate-based monolithic optimization framework to solve mixed-integer design-allocation problems.

Our work shares the same overarching goal as the literature [23–26, 28, 30]: we concurrently optimize the vehicle design and fleet operation to achieve better system design. Our new contributions are: 1) a different aircraft configuration and application, i.e., we design eVTOL UAVs for package delivery; and 2) VRP for the operation problem instead of the resource allocation problems in the previous literature. The VRP is more complicated than the resource allocation problem because the VRP decision variables include route selection from a network, which often has a huge number of possible combinations. In contrast, the resource allocation problems typically do not include the routing variables.

The outline of the paper is as follows. Section 2 provides the problem statement of the design-operation optimization for package delivery UAVs. Section 3 summarizes the eVTOL conceptual design model used in this study. In Section 4, we present a MINLP formulation for the design-operation optimization problem and propose effective sequential heuristics. We then perform a benchmark study of the optimization approaches in Section 5. Section 6 compares the baseline and optimized solution to demonstrate the benefit of the coupled optimization.

2. Problem Description

2.1 Simultaneous Optimization of UAV Design and Operations

We minimize the fleet acquisition cost required to serve a given set of customers with respect to fleet design and operation variables. The acquisition cost is approximated by the summation of the takeoff weights of all UAVs in the

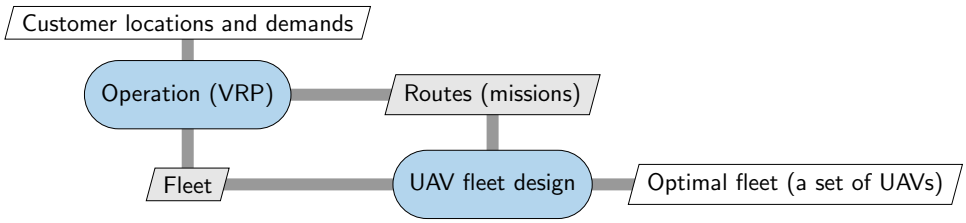


Fig. 1 Coupling of UAV fleet design and operations.

fleet. The operational optimization variables determine the allocation and routing; more precisely, the allocation of each UAV to a subset of customers, and the sequence of the visit within the subset. The fleet design variables consist of the number of UAVs in the fleet and the design of each UAV. We consider a heterogeneous fleet composed of multiple designs and two configurations, which will be described in Section 3.

The design-operation optimization is a mixed-integer optimization problem because we have both discrete routing variables and continuous design variables. Furthermore, the UAV design optimization is generally nonlinear, making the coupled optimization a mixed-integer nonlinear programming (MINLP) problem [31].

Figure 1 shows the coupling structure of the design-operation optimization. The UAV fleet determines the optimal operations because the routing solution is subject to the payload and energy capacity constraints of each UAV. The operational optimization outputs a set of routes, which gives the flight mission requirements for UAV design optimization. Here, we parametrize the missions in terms of the delivery distance, total package weight, and the number of customers on each route. The UAV design optimization outputs a set of UAV designs, which compose a fleet, given the mission requirements. Therefore there is a cycle of information between the operation and design optimization; thus, we need the coupled optimization to find the optimal fleet design.

2.2 Vehicle Routing Problem (VRP)

As operational optimization, we solve a fleet size and mix VRP (FSMVRP) [20, 21], which aims at making a strategic decision on the fleet size and composition. Because of its long-term objective, FSMVRP requires representative customer information over the life cycle of the fleet, which we assume to be available.

The key assumptions and constraints in our VRP model are as follows:

1) The fleet is heterogeneous and is composed of various UAV designs.
2) Not all UAVs in the fleet need to be used; the VRP solver finds the optimal number and composition of UAVs.
3) Customer locations and demands are deterministic and known a priori.
4) All customers must be served by exactly one UAV. Customers can neither be dropped nor served more than once (i.e., we prohibit splitting the demand between multiple vehicles).
5) We only consider a single depot. Each route must be a closed-loop that starts and ends at the depot.
6) Each UAV is used only once; we do not allow the multi-trip of each vehicle.
7) Each UAV must satisfy the payload capacity and energy consumption constraints.

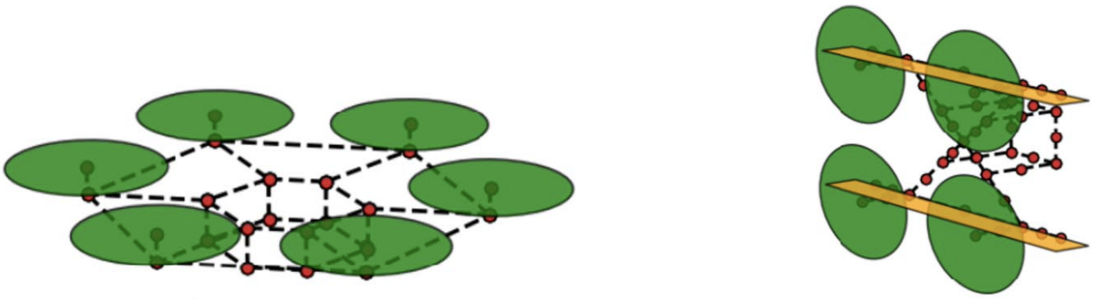
3. UAV Conceptual Design Model

This section presents an eVTOL UAV sizing model we used in this study. We consider two different configurations, a hexarotor and a quadrotor biplane tailsitter (QBiT) [1], as shown in Fig. 2. Both configurations are capable of vertical climb, descent, and hover. In cruise, the hexarotor performs the edgewise flight, whereas the QBiT uses wings to generate the lift that compensates its weight.

The goal of the sizing model is to find the minimum-weight UAV design given mission requirements. We parametrize the mission in terms of range R, total payload weight W_{payload}, and the number of customers n_{c} on the route. In general, we optimize a UAV for multiple missions, i.e., we find a single UAV design that can fly multiple delivery scenarios. In such cases, the mission inputs R, W_{payload}, and n_{c} are vectors. The sizing outputs are the UAV takeoff weight W_{total}, battery weight W_{battery}, and power consumption in hover P_{hover} and cruise P_{cruise}.

The inputs and outputs of the sizing module are summarized as follows:

$$\left(W_{\text{total}}, W_{\text{battery}}, P_{\text{hover}}, P_{\text{cruise}}\right) = f_{\text{sizing}}\left(R, W_{\text{payload}}, n_{\text{c}}\right) \ . \tag{1}$$



(a) Hexarotor. **(b) Quadrotor biplane tailsitter (QBiT)**

Fig. 2 Delivery UAV configurations considered in this study. Figures are adapted from Ref. [3]

Inside the above function $f_{\text{sizing}}(\cdot)$, we perform UAV design optimization.

Figure 3 shows the extended design structure matrix (XDSM) [32] of the QBiT sizing model. The hexarotor sizing model also has a similar structure.

3.1 Weight Estimation

We breakdown the UAV weight into the payload, battery, and empty weight as follows:

$$W_{\text{total}} = W_{\text{payload}} + W_{\text{battery}} + W_{\text{empty}} \,, \tag{2}$$

where the empty weight is further broken down into

$$W_{\text{empty}} = W_{\text{motor}} + W_{\text{ESC}} + W_{\text{rotor}} + W_{\text{wing}} + W_{\text{frame}} \,. \tag{3}$$

The motor and electronic speed controller (ESC) weights are estimated using the regression models [3],

$$\begin{aligned} W_{\text{motor}}\,(\text{lb}) &= 0.412P\,(\text{hp})\,, \\ W_{\text{ESC}}\,(\text{lb}) &= 0.591P\,(\text{hp})\,, \end{aligned} \tag{4}$$

where P is the installed power of the motor. We assume a 50% power margin on top of the maximum power required during the missions.

The rotor weight is given by the regression formula [4],

$$W_{\text{rotor}}\,(\text{g}) = 0.1207(2r)^2 - 0.5122(2r), \tag{5}$$

where r is the rotor radius in inches.

We assume the wing weight to be linear with respect to the wing area:

$$W_{\text{wing}}\,(\text{kg}) = -0.0802 + 2.2854 S_{\text{wing}}\,(\text{m}^2)\,. \tag{6}$$

We also assume that the frame weight, including miscellaneous weights such as wires and recovery parachutes, is linear with respect to the takeoff weight:

$$W_{\text{frame}} = 0.5 + \beta W_{\text{total}}\,(\text{kg})\,, \tag{7}$$

where $\beta = 0.2$ for the hexarotor and $\beta = 0.18$ for the QBiT. We determined the linear coefficients in Eqs. (6) and (7) so that our sizing outputs agree with the results of Govindarajan and Sridharan [3], where they used a finite element analysis-based weight estimation. The smaller value of β for the QBiT accounts for the wings that we compute the weight separately, which can also carry the structural load.

The battery weight is computed based on the battery energy density and the total energy required for the mission. We assume a simple mission profile composed of vertical takeoff, cruise, and landing. Furthermore, we approximate the power in vertical climb and descent to be equal to the power in hover. The energy required for the mission is then

$$E_{\text{req}} = P_{\text{hover}}\Big(2(n_c + 1)t_{\text{hover}}\Big) + P_{\text{cruise}}\frac{R}{V_\infty}\,, \tag{8}$$

where V_∞ is the cruise speed; P_{hover} and P_{cruise} are the power consumption in hover and cruise, respectively. The first term is the energy for takeoff and landing, and the second term the energy is for a cruise. We assume the hovering time t_{hover} of 60 s for a takeoff or landing operation; therefore, a UAV hovers for 120 s at each customer or the depot. We assume that P_{hover} and P_{cruise} are constant across the mission, which means that we ignore the UAV weight change after unloading the payload. The battery weight is then given by

$$W_{\text{battery}} = \frac{E_{\text{req}}}{0.85\rho_b} \,. \tag{9}$$

The battery density ρ_b is assumed to be 158 Wh/kg, which is conservative but a typical value in commercial battery packs including the safety casing [3]. The factor of 0.85 accounts for the losses in battery and transmission as well as the power required by onboard systems other than motors.

3.2 Power Consumption

3.2.1 Power in hover and vertical flight

Based on the momentum theory, the shaft power required by each rotor is,

$$P_{\text{hover}} = \frac{1}{\eta_{\text{hover}}} \frac{T^{1.5}}{\sqrt{2\rho A}} \,, \tag{10}$$

where T is the thrust of each rotor, ρ is the air density, and A is the rotor disk area. The trim condition in hover yields $T = W_{\text{total}}/n_{\text{rotor}}$. We assume the hover figure of merit of $\eta_{\text{hover}} = 0.75$ for hexarotors and 0.65 for QBiTs; we set lower efficiency for the QBiTs because the QBiT rotors (propellers) are generally designed to be efficient in cruise at the sacrifice of hover efficiency. On the other hand, the hexarotor's rotor operates in similar conditions in hover and edgewise forward flight, which allows it to have hover-efficient rotors.

3.2.2 Power in cruise

In forward flight, we use the relation [33] based on the momentum theory to compute the power required as follows:

$$P_{\text{cruise}} = TV_\infty \sin\alpha + \kappa T V_i + P_0 \,, \tag{11}$$

where V_∞ is the freestream velocity, V_i is the induced velocity, α is the shaft tilt angle, and P_0 is the profile power. Accordingly, $V_\infty \sin\alpha$ is the component of the freestream velocity normal to the rotor disk. Following the method by Govindarajan and Sridharan [3], the induced power factor κ is given by

$$\kappa = \min\left(1.15, \frac{1}{\eta_{\text{hover}}} - \frac{\sqrt{2\rho A}}{T^{1.5}} P_0\right) . \tag{12}$$

To compute the induced velocity V_i, we first solve the inflow equation

$$\lambda = \mu \tan\alpha + \frac{C_T}{2\sqrt{\mu^2 + \lambda^2}} \tag{13}$$

for the rotor inflow λ. Then, we use the following relation to obtain V_i:

$$\frac{V_\infty \sin\alpha + V_i}{\Omega r} = \lambda \,. \tag{14}$$

In Eq. (13), μ, C_T are the rotor advance ratio and thrust coefficient, given by

$$\begin{aligned} \mu &= \frac{V_\infty \cos\alpha}{\Omega r} \,, \\ C_T &= \frac{T}{\rho A (\Omega r)^2} \,, \end{aligned} \tag{15}$$

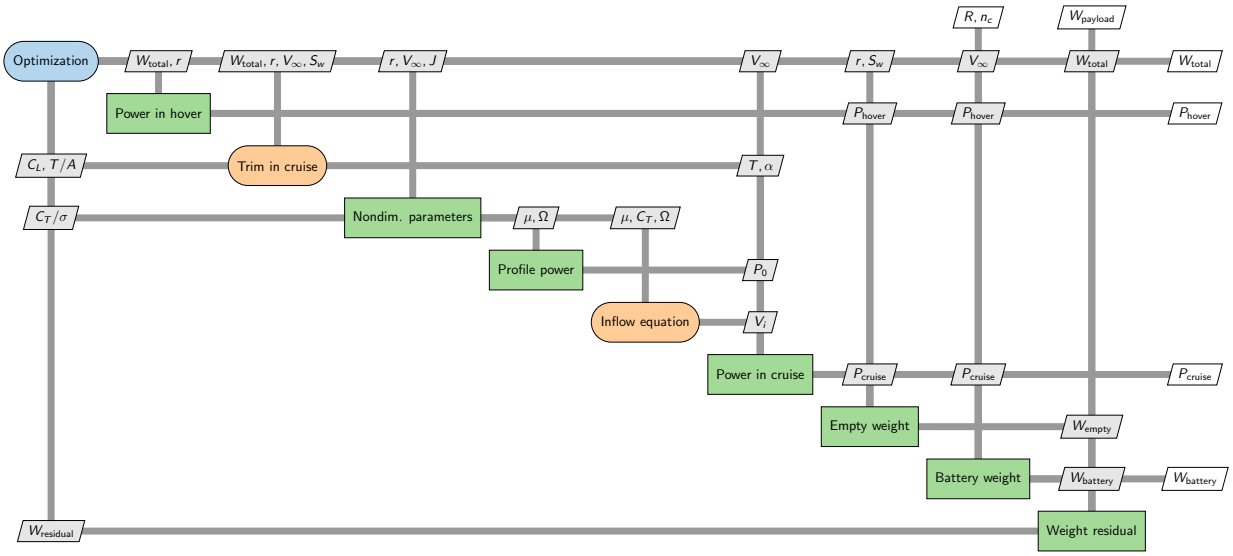


Fig. 3 Extended design structure matrix (XDSM) of QBiT sizing optimization.

where Ω, r are the rotor angular velocity and radius, respectively. For the winged cruise of the QBiT, we also introduce the propeller advance ratio

$$J = \frac{V_\infty}{2nr}\,, \tag{16}$$

where n is the number of propeller revolutions per second.

The profile power is given by the following formula [3]:

$$P_0 = \frac{\sigma C_{d_0}}{8}\left(1 + 4.65\mu^2\right)(\rho A)(\Omega r)^3\,, \tag{17}$$

where C_{d_0} is the airfoil zero-lift drag coefficient and σ is the rotor solidity; we use $C_{d_0} = 0.012$ and $\sigma = 0.13$ [33].

3.2.3 Trim conditions in cruise

The trim conditions determine the thrust and shaft tilt angle. For the hexarotor without wing,

$$\begin{aligned} n_{\text{rotor}} T &= \sqrt{W_{\text{total}}^2 + D_{\text{body}}^2}\,, \\ \tan\alpha &= \frac{D_{\text{body}}}{W_{\text{total}}}\,, \end{aligned} \tag{18}$$

where D_{body} is the body drag, and n_{rotor} is the number of rotors, which is six for the hexarotor. For the body drag estimation, we use the method of Sridharan et al. [1] and Govindarajan and Sridharan [3]. Their method assumes the UAV body to be a cylinder whose radius is 58% of the rotor radius, and the length-to-diameter ratio is 2.5. Then the drag is given by

$$\begin{aligned} D_{\text{body}} &= \frac{1}{2}\rho V_\infty^2 S_b C_{D,b}\,, \\ C_{D,b} &= 0.1 + 0.2\cos^3\alpha\,, \end{aligned} \tag{19}$$

where the body reference area S_b is the cylinder frontal area, i.e., the radius times the length.

For the QBiT with wings, the trim conditions are simply $n_{\text{rotor}} T = D$ and $L = W$. The drag is

$$\begin{aligned} D &= D_{\text{body}} + D_{\text{wing}} \\ &= D_{\text{body}} + \frac{1}{2}\rho V_\infty^2 S_w \left(C_{D_{0,w}} + C_{D_i}\right)\,, \end{aligned} \tag{20}$$

Table 1 UAV conceptual design optimization problem

	Function/variable	Description	Bounds
minimize	W_{total}	Takeoff weight, kg	
given	R, W_{payload}, n_{c}	One or multiple mission requirements	
with respect to	W_{total}	Takeoff weight, kg	[0.5, 50]
	V_∞	Cruise speed, m/s	[10, 50]
	r	Rotor radius , m	[0.05, 1.0]
	μ	Edgewise advance ratio, hexarotor only	[0.01, 0.5]
	J	Propeller advance ratio, QBiT only	[0.01, 1.3]
	S_w	Wing area, m^2, QBiT only	[0.05, 5.0]
subject to	$W_{\text{total}} - W_{\text{payload}} - W_{\text{battery}} - W_{\text{empty}} = 0$	Equality constraint to satisfy Eq. (2)	
	$T/A \leq 250$ N/m^2	Disk loading	
	$C_T/\sigma \leq 0.14$	Blade loading	
	$C_L \leq 0.6$	Cruise lift coefficient, QBiT only	

where S_w is the wing area, $C_{D_{0,w}}$ is the zero-lift drag coefficient of the wing (set to 0.01), and C_{D_i} is the induced drag coefficient given by

$$C_{D,i} = \frac{C_L^2}{\pi \text{AR} e} \,. \tag{21}$$

We assume a fixed aspect ratio of AR = 8 and Oswald efficiency of $e = 0.8$ [3]. For the body drag, we use the same method as the hexarotor, with the angle of attack of 5 deg (i.e., $\alpha = 85$ deg) assumed.

3.3 Sizing Optimization Problem

Given the mission requirements, we minimize the takeoff weight with respect to the cruise speed, rotor radius, edgewise or propeller advance ratio in cruise, and wing area (only in QBiT cases). The UAV sizing optimization problem is summarized in Table 1. We impose the upper limit on disk loading, blade loading, and the cruise lift coefficient to avoid poor maneuverability and poor gust responses. The upper bound values are the same as those of Govindarajan and Sridharan [3], who explains these constraints.

The sizing model has three implicit variables that need an iterative solver: the rotor inflow λ in Eq. (13); shaft tilt angle α in Eqs. (18) and (19); and takeoff weight W_{total} in Eq. (2). We use a Newton solver for λ and α, whereas we impose the weight residual as an equality constraint instead of Eq. (2).

We implemented the sizing models using the OpenMDAO framework [34] with analytical derivatives, and we used SNOPT [35] via pyOptSparse [36] as an optimizer.

3.4 Characteristics of Hexarotor and QBiT

This section explains the key characteristics of the UAV models in the context of the design-routing coupled problem; the detailed discussion about the UAV conceptual design and its design space can be found in Sridharan et al. [1], Bershadsky et al. [4] and Govindarajan and Sridharan [3].

The most important distinction between the hexarotor and QBiT is their different suitability on various mission settings. The hexarotor is more efficient in hover than the QBiT, therefore, it is preferable for short-range flights as well as multi-customer delivery. On the other hand, the QBiT is more efficient in cruise because of its wings, making it a viable option for long-range delivery with a few customers.

Figure 4 shows the optimized takeoff weight of each configuration for single- and 4-customer delivery with various payload weights and mission ranges. The takeoff weight was smaller with hexarotors for shorter ranges, whereas the hexarotors became heavier than QBiTs on longer-range missions. For single-customer delivery, the "switch" of the takeoff weight occurred between 30 and 50 km range, depending on the payload weight. For 4-customer delivery, the hexarotor weight was always smaller except 1 kg-60 km delivery, although the hexarotor weight was more sensitive to the range than the QBiT weight. In deciding the optimal design, the hover performance becomes more dominant

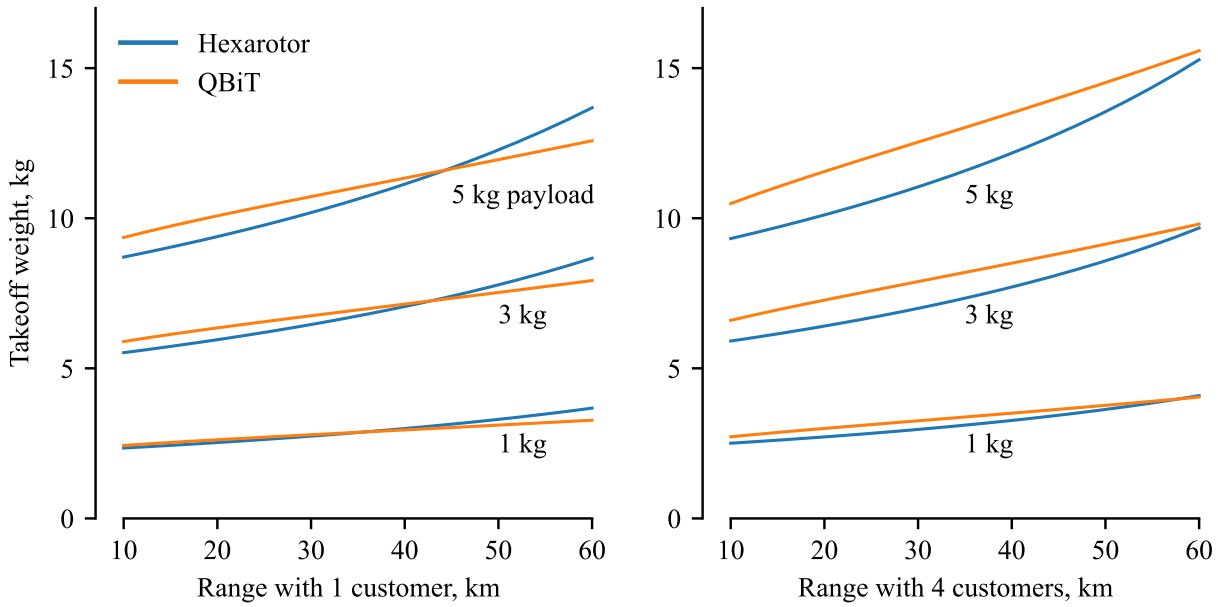


Fig. 4 Optimized takeoff weight of the hexarotor and QBiT under various mission settings.

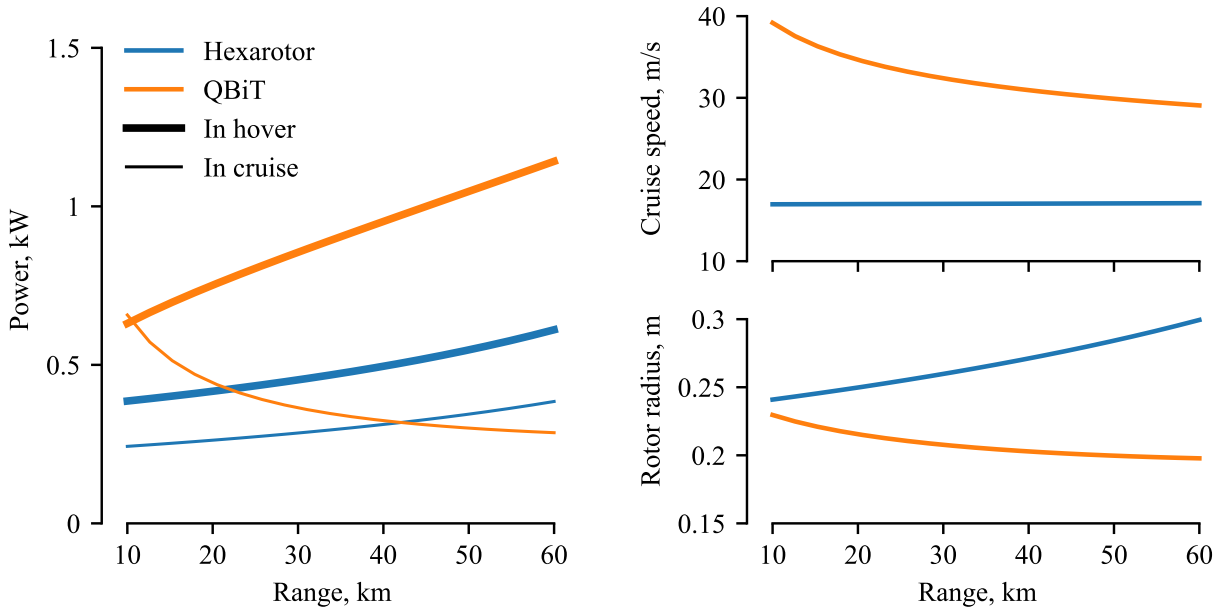


Fig. 5 Optimized design variables and power performances for single-customer missions with 3 kg payload. In the power plot, the thicker lines represent the hover conditions, and the thinner lines represent cruise.

than the cruise when the number of customers increases. This is because the increased number of customers requires a longer hovering time.

Figure 5 shows the power consumption, cruise speed, and rotor radius of the optimized UAVs with a single customer with 3 kg payload. The QBiT consistently required more power in hover than the hexarotor because of its smaller rotor radius, the fewer number of rotors, and the lower hover figure of merit. On the other hand, the QBiT consumed less power in cruise than the hexarotor when the mission range was longer than 43 km, which led to the smaller takeoff weight, as shown in Fig. 4. In cruise, the wings achieve a low thrust-to-weight ratio that reduces the cruise power.

For the hexarotor, the optimized rotor radius increased as the UAV total weight increased for longer ranges, but

Table 2 Additional constraints and assumptions for design model linearization

Variable	Additional constraint or assumptions to fix the variable
Cruise speed V_∞	Constant at 18 m/s for hexarotor, 33 m/s for QBiT
Rotor radius r	Fixed hover disk loading of 120 N/m^2 for hexarotor, 180 N/m^2 for QBiT
Edgewise advance ratio μ	Fixed at 0.3, hexarotor only
Propeller advance ratio J	Fixed at 1.3, QBiT only
Wing area S_w	Constant lift coefficient of 0.6 in cruise, QBiT only

the cruise speed was nearly constant regardless of the range. The optimal speed implies the best balance between the following two effects: 1) the slower cruise speed results in a longer cruise duration, which increases the energy consumption; 2) on the other hand, the slower speed reduces the profile power because the profile power is proportional to $(\Omega r)^3$ according to Eq. (17), where (Ωr) is in inverse proportion to the speed under the constant advance ratio. The optimal advance ratio was constant because it was always constrained by the blade-loading constraint; the detail of the blade-loading constraint is discussed by Govindarajan and Sridharan [3]. The hexarotor power ratio, $P_{\text{cruise}}/P_{\text{hover}}$ was less sensitive to the range than the QBiT. The lower power consumption in cruise than in hover is mainly because of the translational lift.

The QBiT design was more sensitive to the mission range than the hexarotor, and it showed a trade-off between hover performances and cruise performances. A smaller rotor radius benefits the cruise efficiency by reducing the profile power as well as the empty weight. On the other hand, in hover, a larger rotor is preferable to reduce the power, according to Eq. (10). As a result of this trade-off, the optimizer chose a smaller rotor radius for long-range missions where the cruise performance is dominant. In contrast, it preferred a larger radius for short-range missions where the hovering is more critical than the cruise. In long-range missions, the cruise efficiency was further benefitted by flying slowly and reducing the drag and parasite power. This benefit of reducing the power was more significant than the detriment of the increased cruise duration on the long-range missions. In short-range missions, in contrast, the higher cruise speed resulted in a smaller takeoff weight by shortening the cruise duration at the sacrifice of the cruise power. Note that the propeller advance ratio always reached its upper bound of $J = 1.3$. This upper bound prevented too small a rotor in the long-range missions, or too fast a cruise speed in the short-range missions.

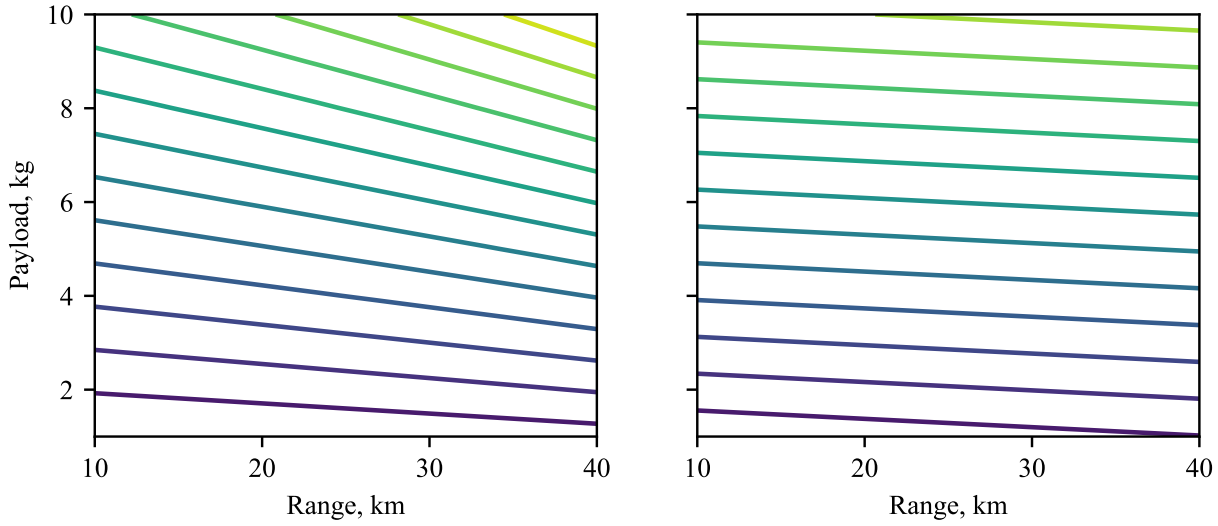
3.5 Linear Surrogate Models

The sizing module, which is a mapping from the mission inputs to the sizing outputs as summarized in Eq. (1), is nonlinear because we perform design optimization inside this mapping. When solving the coupled design-routing optimization, the heuristic approach (which will be described in Section 4.2) is capable of directly incorporating this nonlinear design optimization. However, the mixed-integer nonlinear programming (MINLP) approach (in Section 4.1) is not compatible with the design nonlinearity. This section builds linear surrogates of the sizing model to enable the coupled optimization by a MINLP branch-and-cut solver.

We first introduce new equality constraints on the disk loading and lift coefficient to determine the rotor radius and wing area. We also fix the cruise speed and advance ratio; the fixed values are summarized in Table 2. The sizing optimization now has zero degrees of freedom with these new assumptions.

Next, we train the linear fitting models for W_{total}, P_{hover}, and P_{cruise} in the $(R, W_{\text{payload}}, n_c)$ three-dimensional space. The input domains are $10 \le R \le 40$ km, $1 \le W_{\text{payload}} \le 10$ kg, and $1 \le n_c \le 9$. Here, we set the range upper bound at 40 km to exclude the increasing nonlinearity above 40 km, as shown in Fig. 4. We then build the surrogate models by solving the least square problems with non-negativity constraints on the weight and power predictions. We do not need a surrogate for W_{battery} because the MINLP formulation directly incorporates the energy capacity constraint. The trained surrogates have the "switching point" of the takeoff weight under the 40 km range, which implies that the design-routing solution would include both hexarotor and QBiT in the optimal fleet.

Figure 6 shows an example of the surrogate model compared to the original nonlinear model (after applying the assumptions in Table 2). We evaluate the surrogate prediction error using 100 random test points. The average and maximum errors were 6.0% and 24.6% for the hexarotor, and 3.4% and 12.1% for the QBiT, respectively. Note that our primary purpose of linearization is not to have accurate and inexpensive surrogates; instead, we are restricted to the linear models because of the limitation of the MINLP solver, regardless of the surrogate accuracy.



(a) Nonlinear model **(b) Linear surrogate model**

Fig. 6 The nonlinear model and its linear surrogate for hexarotor W_{total}, sliced at $n_c = 1$.

4. Optimization Approaches

This section first presents a mixed-integer nonlinear programming (MINLP) formulation of the design-operation coupled optimization, which will be solved by a commercial branch-and-cut solver. However, the MINLP solver cannot incorporate nonlinear UAV design models and requires linearized surrogates instead. The MINLP solver is also limited to the small problem size. To tackle these challenges, in Section 4.2, we propose a novel heuristic approach that can efficiently solve large-scale problems and incorporate nonlinear design models.

4.1 Mixed-Integer Nonlinear Programming (MINLP)

The first approach solves the coupled design-routing optimization using a mathematical MINLP solver. This monolithic optimization approach corresponds to the simultaneous analysis and design (SAND) in terms of the MDO architecture [29]. We formulate the coupled optimization as a non-convex mixed-integer quadratic constrained program (MIQCP) using the three-index vehicle flow model [37] for routing, which allows us to consider a heterogeneous UAV fleet. We use the non-convex branch-and-cut solver by Gurobi 9.5.0 [38]. This solver utilizes the bilinear transformation and spatial branching to handle non-convex quadratic constraints.

Let $\mathcal{N}$ be a set of customers, $\mathcal{N} = \{1, 2, ..., n\}$, where n is the total number of customers. Define a set $\mathcal{V}$ of the nodes, which consists of the customers and depot, $\mathcal{V} = \mathcal{N} \bigcup \{o, d\}$. Here, we consider a single depot but distinguish the start point o and endpoint d. We also define a set of arcs between the nodes, $\mathcal{A} = (\mathcal{V} \backslash \{d\}) \times (\mathcal{V} \backslash \{o\})$. For the UAV fleet, we consider a heterogeneous fleet $\mathcal{K}$, $\mathcal{K} = \{1, 2, ..., k_{\max}\}$, where $k_{\max}$ is the *maximum* number of UAVs available for delivery. Furthermore, we define a subset of the fleet for each configuration, $\mathcal{K}_{\text{hexa}}$ and $\mathcal{K}_{\text{QBiT}}$. The two subsets are the disjoint sets and satisfy $\mathcal{K} = \mathcal{K}^{\text{Hexa}} \bigcup \mathcal{K}^{\text{QBiT}}$. In the MINLP formulation, we assume that each UAV has its own design variables, and therefore, we may have up to $k_{\max}$ different designs of UAVs. Since each UAV serves only one route, the number of routes in the solution equals the number of distinct designs. Note that the solution does not need to (and in most cases, it does not) use all $k_{\max}$ UAVs; rather than that, the solver determines the optimal number of UAVs required for delivery.

The optimization variables are binary variables x, y, and α for routing; continuous variables R, W_{payload}, and integer variables n_c to represent flight missions; and continuous variables W_{total}, P_{hover}, and P_{cruise} for UAV design. The routing variable $x_{ijk} = 1$ if vehicle k travels from node i to j, 0 otherwise. The indicator variable $y_{ik} = 1$ if node i is served by vehicle k; $\alpha_k = 1$ if vehicle k is active (i.e., serves at least one customer) and 0 otherwise (i.e., vehicle k is not used). The mission and design variables are assigned to each UAV.

To avoid unrealistically large-sized UAVs, we impose upper bounds on payload and energy capacity. The upper bound of the payload capacity is 10 kg. For the energy capacity, we impose a constraint such that the UAV's designed

energy capacity is less than the energy it consumes to perform a single-customer 40 km-range flight. In other words, we limit the maximum UAV size by not allowing any UAVs to fly more than 40 km on a single-customer delivery. This implies that the range upper bound R_{UB} is 40 km, which is only reachable when a vehicle serves only one customer. When a UAV serves more than one customer, the maximum range is less than 40 km because it consumes additional energy in hover.

The MINLP formulation is summarized as follows:

$$\text{minimize} \quad \sum_{k\in\mathcal{K}} \alpha_k W_{\mathrm{total},k}\,, \tag{22}$$

$$\text{w.r.t.} \quad x_{ijk},\ y_{ik},\ \alpha_k \in \{0,1\},\ n_{c,k} \in \mathcal{I} \qquad \forall i,j \in \mathcal{V},\ k \in \mathcal{K}, \tag{23}$$

$$W_{\mathrm{total},k}, W_{\mathrm{payload},k},\ P_{\mathrm{hover},k}, P_{\mathrm{cruise},k},\ R_k \in \mathcal{R} \qquad \forall k \in \mathcal{K}, \tag{24}$$

$$\text{subject to} \quad \sum_{k\in\mathcal{K}} y_{ik} = 1 \qquad \forall i \in \mathcal{N}, \tag{25}$$

$$\sum_{j\in\mathcal{V}\setminus\{i,o\}} x_{ijk} - \sum_{j\in\mathcal{V}\setminus\{i,d\}} x_{jik} = 0 \qquad \forall i \in \mathcal{N},\ \forall k \in \mathcal{K}, \tag{26}$$

$$\sum_{j\in\mathcal{V}\setminus\{i,o\}} x_{ijk} = 1 \qquad i = o,\ \forall k \in \mathcal{K}, \tag{27}$$

$$\sum_{j\in\mathcal{V}\setminus\{i,o\}} x_{ijk} = y_{ik} \qquad \forall i \in \mathcal{V}\setminus\{d\},\ \forall k \in \mathcal{K}, \tag{28}$$

$$\sum_{j\in\mathcal{V}\setminus\{i,d\}} x_{jik} = y_{ik} \qquad i = d,\ \forall k \in \mathcal{K}, \tag{29}$$

$$\sum_{i\in\mathcal{N}} y_{ik} q_k = W_{\mathrm{payload},k} \qquad \forall k \in \mathcal{K}, \tag{30}$$

$$\sum_{i,j\in\mathcal{A}} x_{ijk} d_{ij} = R_k \qquad \forall k \in \mathcal{K}, \tag{31}$$

$$\sum_{i\in\mathcal{N}} y_{ik} = n_{c,k} \qquad \forall k \in \mathcal{K}, \tag{32}$$

$$\sum_{i\in\mathcal{N}} y_{ik} \le \alpha_k M \qquad \forall k \in \mathcal{K}, \tag{33}$$

$$\sum_{\forall i,j\in\mathcal{S}} x_{ijk} \le |\mathcal{S}| - 1, \qquad \forall \mathcal{S} \subseteq \mathcal{N},\ \forall k \in \mathcal{K}, \tag{34}$$

$$P_{\mathrm{hover},k}\Big(2(n_{c,k}+1)\Big)t_{\mathrm{hover}} + P_{\mathrm{cruise},k}\frac{R_k}{V_{\infty,k}} \le P_{\mathrm{hover},k}(4t_{\mathrm{hover}}) + P_{\mathrm{cruise},k}\frac{R_{\mathrm{UB}}}{V_{\infty,k}} \qquad \forall k \in \mathcal{K}, \tag{35}$$

$$W_{\mathrm{total},k} = f_{\mathrm{w}}^{\mathrm{H}}\left(R_k, W_{\mathrm{payload},k}, n_{c,k}\right) \qquad \forall k \in \mathcal{K}^{\mathrm{Hexa}}, \tag{36}$$

$$P_{\mathrm{hover},k} = f_{\mathrm{ph}}^{\mathrm{H}}\left(R_k, W_{\mathrm{payload},k}, n_{c,k}\right) \qquad \forall k \in \mathcal{K}^{\mathrm{Hexa}}, \tag{37}$$

$$P_{\mathrm{cruise},k} = f_{\mathrm{pf}}^{\mathrm{H}}\left(R_k, W_{\mathrm{payload},k}, n_{c,k}\right) \qquad \forall k \in \mathcal{K}^{\mathrm{Hexa}}, \tag{38}$$

$$W_{\mathrm{total},k} = f_{\mathrm{w}}^{\mathrm{Q}}\left(R_k, W_{\mathrm{payload},k}, n_{c,k}\right) \qquad \forall k \in \mathcal{K}^{\mathrm{QBiT}}, \tag{39}$$

$$P_{\mathrm{hover},k} = f_{\mathrm{ph}}^{\mathrm{Q}}\left(R_k, W_{\mathrm{payload},k}, n_{c,k}\right) \qquad \forall k \in \mathcal{K}^{\mathrm{QBiT}}, \tag{40}$$

$$P_{\mathrm{cruise},k} = f_{\mathrm{pf}}^{\mathrm{Q}}\left(R_k, W_{\mathrm{payload},k}, n_{c,k}\right) \qquad \forall k \in \mathcal{K}^{\mathrm{QBiT}}. \tag{41}$$

As a problem setup, we generate a set of customers with known locations and demands q_i, which can be non-uniform. We then compute the distance of each arc d_{ij} based on the customer locations prior to running the MINLP solver.

The objective (22) minimizes the summation of the active UAV weights, which we assume approximates the fleet acquisition cost. Constraints (25) impose that each customer is served by only one vehicle. Constraints (26) and (27) ensure the vehicle flow conservations, that is, each route is a closed-loop. Constraints (28) and (29) are for the consistency between x_{ijk} and y_{ik}. Constraints (30) are the vehicle capacity constraints. Constraints (31) relate the active arcs and the flight range of each route. Constraints (32) relate the route and $n_{c,k}$. Constraints (33) relate y_{ik} and α_k, where M is a large constant (so called "big M") that needs to satisfy $M \ge n$.

Constraints (34) are the subtour elimination constraints (SECs), where $\mathcal{S}$ is a customer subset. The SECs prohibit infeasible closed loops that do not include the depot. The number of constraints (34) is exponential with respect to the number of customers, which is prohibitive for large-scale problems. Therefore, we employ the separation procedure [37] for SECs to avoid imposing the exponential number of constraints. This procedure first solves a relaxation problem without SECs, and every time the optimizer finds a feasible solution to the relaxation problem, it identifies all the subtours in the solution and adds the corresponding SECs. The optimizer then returns to the updated relaxation problem with new SECs, and repeats the process until no subtour is found in the solution.

Constraints (35) are the energy capacity constraints. The right-hand side is the energy consumed by vehicle k if it performed a single-customer 40 km-range delivery. The left-hand side is the actual energy required by vehicle k to serve $n_{c,k}$ customers in the route of distance R_k. The energy constraint requires the energy consumption (left-hand-side) to be less than the maximum energy capacity (right-hand-side). Constraints (35) are bilinear because P_{hover}, n_c, P_{cruise}, R_k in the left-hand side are all optimization variables. This non-convex nonlinearity is the consequence of the design-routing coupling. In conventional routing problems with fixed vehicle designs, P_{hover} and P_{cruise} are constant, and therefore, the energy constraints are linear. However, this bilinearity is inevitable when we include the vehicle design variables.

Finally, constraints (36)–(41) are the design surrogate models from Section 3.5. These constraints are linear because we use linear surrogates. If we directly used the original nonlinear design model instead, the constraints would become general nonlinear equalities. To the authors' best knowledge, any existing mixed-integer programming solvers, including Gurobi 9.5.0, cannot practically deal with such general nonlinear constraints.

4.2 Decomposition and Sequential Heuristics

The second optimization approach, a new approach we propose, decomposes the design-routing problem into design optimization and VRP, then solves them sequentially. Such decomposition has been successful in the coupled optimization of aircraft design and network flow [24–26]. Furthermore, we propose a modification to the conventional sequential optimization [23] to avoid local minima and increase the probability of finding a coupled optimal solution.

The optimization procedure is summarized as follows:

1) Prepare an initial fleet. The fleet is parametrized by variables $[W_{\text{total}}, W_{\text{battery}}, P_{\text{hover}}, P_{\text{cruise}}]$ of each UAV.
2) Solve FSMVRP given the fleet for the minimum fleet acquisition cost (Eq. (22)).
3) Given the routing solution, generate a set of flight missions flown by each UAV.
4) Perform multi-mission design optimization of each UAV for the assigned set of missions to fine-tune the designs.
5) Perform single-mission design optimization for each mission (not for a set of missions) to exploit the VRP solution.
6) Introduce exploratory UAV designs to escape from local minima.
7) Repeat Step 2) with the new fleet composed of the UAVs from Steps 4), 5), and 6).

Figure 7 shows the XDSM of the sequential heuristic algorithm. In Step 2), we solve FSMVRP using heuristics while fixing the UAV designs in the fleet, i.e., fixing the designs variables W_{total}, W_{battery}, P_{hover}, and P_{cruise}. The VRP here is equivalent to solving Eqs. (22)–(35), whereas we use a different form of the energy constraints instead of Eq. (35)

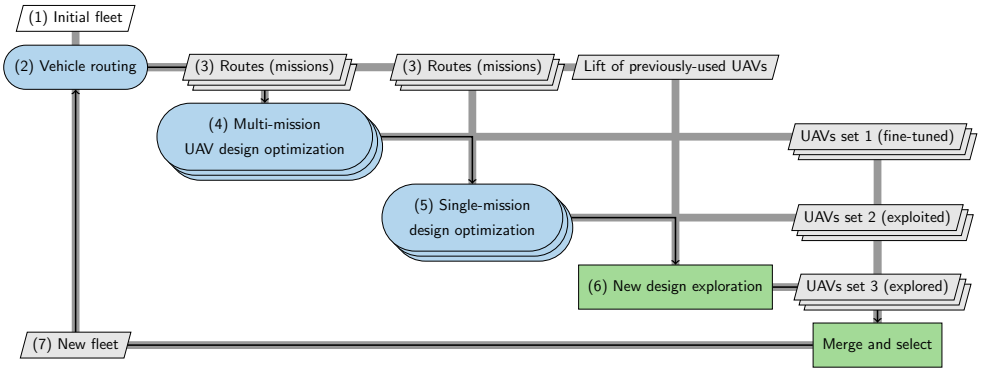


Fig. 7 Data flow and processes of the proposed sequential heuristic algorithm.

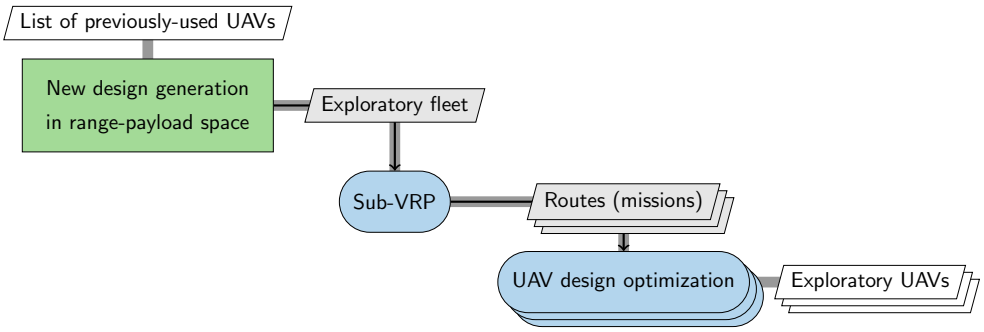


Fig. 8 Procedure of new design exploration in the sequential heuristic algorithm.

as follows:

$$P_{\text{hover},k}\Big((2(n_{c,k}+1)\Big)t_{\text{hover}} + P_{\text{cruise},k}\frac{R_k}{V_{\infty,k}} \leq \rho_b W_{\text{battery},k} \quad \forall k \in \mathcal{K} \ . \tag{42}$$

The left-hand side is the energy consumed, and the right-hand side is the battery energy capacity of the vehicle. We use heuristics to solve VRP because they are the only practical way to tackle large-scale problems, given VRP is NP-hard. In the current work, we use heuristic solvers implemented in the Google OR-Tools package [39], specifically, the combination of local cheapest insertion (LCI) to find an initial feasible solution, and guided local search (GLS) to escape from local minima and improve the solution globally. LCI and GLS were the most robust and accurate for our FSMVRP among the other algorithms available in Google OR-Tools.

The UAV design optimization in Step 4) achieves same-route improvement, or local improvement, by fine-tuning the design of each UAV for the assigned routes. In designing a fleet, it is not practical to have a distinct specialized design for each of the routes; otherwise, we may have to have hundreds of different UAV designs. Instead, we prefer a fleet with a limited number of different designs while allowing to have multiple same-design UAVs in the fleet. This leads to the multi-mission design optimization, which designs a UAV given a set of missions assigned to it. We use efficient gradient-based optimization as described in Section 3.3 for the design subproblem, which is summarized in Table 1.

In Step 5), we perform a series of single-mission design optimization for all missions ignoring the previous assignments by VRP (i.e., regardless of the mission grouping). This step intends to improve the solution by exploiting the routes generated by the previous VRP. For example, suppose UAV type-A served routes 1, 2, and 3 in the previous VRP solution. Then, Step 4) optimizes the design of UAV type-A while requiring it to fly all of route 1, 2, and 3. Whereas in Step 5), we optimize three new UAVs for each of the routes and generate new UAVs type-B, C, and D. The fleet acquisition cost would become lower if we use UAV type-B, C, and D instead of three UAVs type-A because type-B, C, and D are specialized for each route. However, it may be infeasible to actually use all type-B, C, and D in the fleet when we limit the maximum number of different UAV designs.

We also seek new-route improvement, or global improvement, in addition to the same-route improvement. Here, we propose a method for new design exploration in Step 6) to achieve the global improvement. The procedure of Step 6) is summarized in Fig. 8. In this step, we first generate a set of new UAVs whose designs are different from the previously-searched UAVs. For the design exploration purpose, we represent UAV design by the payload capacity and the range on a single-customer delivery, i.e., $X = [W_{\text{payload}}, R|_{n_c=1}]$, where X is a vector representing the UAV design. We then choose a new exploratory design by repeatedly solving the following optimization subproblem:

$$X_{\text{new}} = \arg\max_X \left[\min_i \|X - X_i\|\right] , \tag{43}$$

where X_i is a previously-used UAV. This subproblem finds a point that is most isolated from the previously-used designs in the range-payload 2D space. For the UAVs with $n_c > 1$, we compute an equivalent single-customer range using Eq. (8) to map the design into the range-payload space. After generating a set of new UAVs, we compute $[W_{\text{total}}, W_{\text{battery}}, P_{\text{hover}}, P_{\text{cruise}}]$ for each UAV using Eq. (1) and prepare an exploratory fleet. We then solve a sub-VRP using this exploratory fleet. In this sub-VRP, we allow the routing solution not to serve all the customers because the exploratory fleet is not necessarily capable of satisfying all the demands. This setting is different from the original VRP in Step 2), where we require the fleet to serve all customers. The exploration step is completed by optimizing the UAV designs for the mission outputs from the sub-VRP.

We now have a pool of candidate UAV designs for the next iteration generated from Steps 4), 5), and 6). At this point, we may need to select UAVs from the pool when the maximum number of different designs is limited. The selection rule, i.e., the portion of the UAVs from each step to be used in the next fleet, is a tunable hyperparameter. In this work, we use the following strategy: in the early sequential iterations, we prioritize the fine-tuned designs as elitism while keeping 30–50% of the spots for exploratory designs; we then reduce the exploration portion and focus more on the local improvement and exploitation toward the end.

5. Benchmark Results of the Optimization Approaches

We compare the accuracy, robustness, and computational cost of the two optimization approaches proposed in Section 4 by solving a set of benchmark problems. In this chapter, we use the linearized surrogate models for UAV sizing due to the limitation in the MINLP solver.

5.1 Benchmark problems

We created a benchmark set of the design-routing optimization problems with various problem sizes of 5, 10, 15, 30, and 60 customers. For each problem size, we generate 20 problem instances by randomly locating the customers in a 30 km × 20 km region. Figure 9 shows examples of 10 and 60-customer instances. Each customer has either 1 kg or 2 kg demand in 5, 10, and 15-customer problems, and 0.5 kg or 1 kg demand in 30 and 60-customer problems. The depot is located at the center of the region.

For the fleet settings, the maximum number of UAVs in the fleet is equal to the number of customers. Since the linearized design surrogate was trained based on a single-mission design model and cannot take multi-mission into account, we allow each UAV to have its own design (i.e., the number of different designs is equal to the number of UAVs in the fleet).

5.2 Accuracy and Robustness of the Sequential Heuristics

We first benchmark the accuracy and robustness of the sequential heuristic algorithm we proposed. We measure the accuracy by computing the errors of the optimized objective value as

$$\text{Objective error} = \frac{f^*_{\text{heuristics}} - f^*_{\text{global}}}{f^*_{\text{global}}}, \tag{44}$$

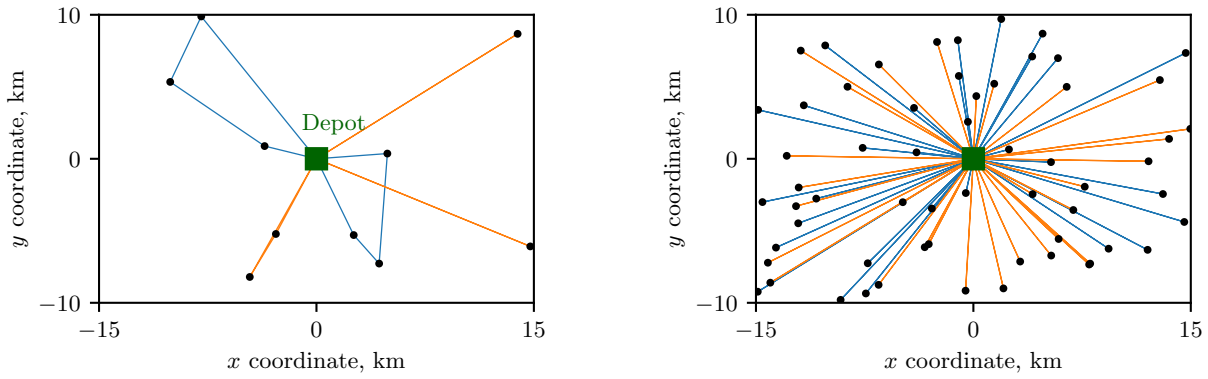
where f^*_{global} is the proven global optimal value obtained by the Gurobi MINLP solver.

Figure 10 shows the objective errors in the 5, 10, and 15-customer benchmark problems. Note that the 15 customers were the practical limit of the problem size on which we can fully converge the MINLP solver: until the optimality gap converges to less than 0.001%, it took up to 48 hours in parallel on a 48-core 1.4–3.7 GHz computing node. We solved each of the 60 test problems from 20 different randomly-generated initial fleets; for each run, we continued optimization for 30 sequential iterations. Each translucent dot in Fig. 10 corresponds to the optimized objective value starting from each fleet. The opacity in the plot indicates that multiple optimization runs converged to a similar value: the more frequent runs converged to a point, the more opaque the point becomes.

Most of the dots converged to the zero error on 5-customer problems, which indicates that most optimization runs found the global minimum. In 10-customer problems, some optimization runs resulted in a few percent of errors,

Table 3 Design-routing optimization problem formulation

		UAV sizing model incorporated	
	Function/Variable	Surrogate (Sec. 5)	Design optimization (Sec. 6)
minimize	fleet acquisition cost ($\sum W_{\text{total}}$)	✓	✓
with respect to	routing and allocation variables	✓	✓
	number of different UAV designs, n_{design}	✓	✓
	total number of UAVs in the fleet, n_{UAV}	$n_{\text{UAV}} = n_{\text{design}}$	$n_{\text{design}} \leq n_{\text{UAV}} \leq n_{\max}$
	config., W_{total}, W_{battery}, W_{payload}, P_{hover}, P_{cruise}	✓	✓
	V_∞, r, μ or J, S_w	fixed	✓



(a) 10 customers and the optimal routes.

(b) 60 customers and the MINLP initial routes.

Fig. 9 Examples of the benchmark problem instances. The black points show the customer; the green square at the center is the depot; the blue and orange lines show the routes flown by hexarotors and QBiTs, respectively.

although the majority converged to the global minimum. In 15-customer problems, fewer points converged to the global minimum, and the majority of the runs resulted in 0–1% errors. The worst-case error in the 15-customer problems was no more than 3%.

Figure 11 shows the probability of finding an acceptable solution within a given tolerance when performing the multi-start sequential heuristics. We computed the probability of success based on the multi-start optimization results

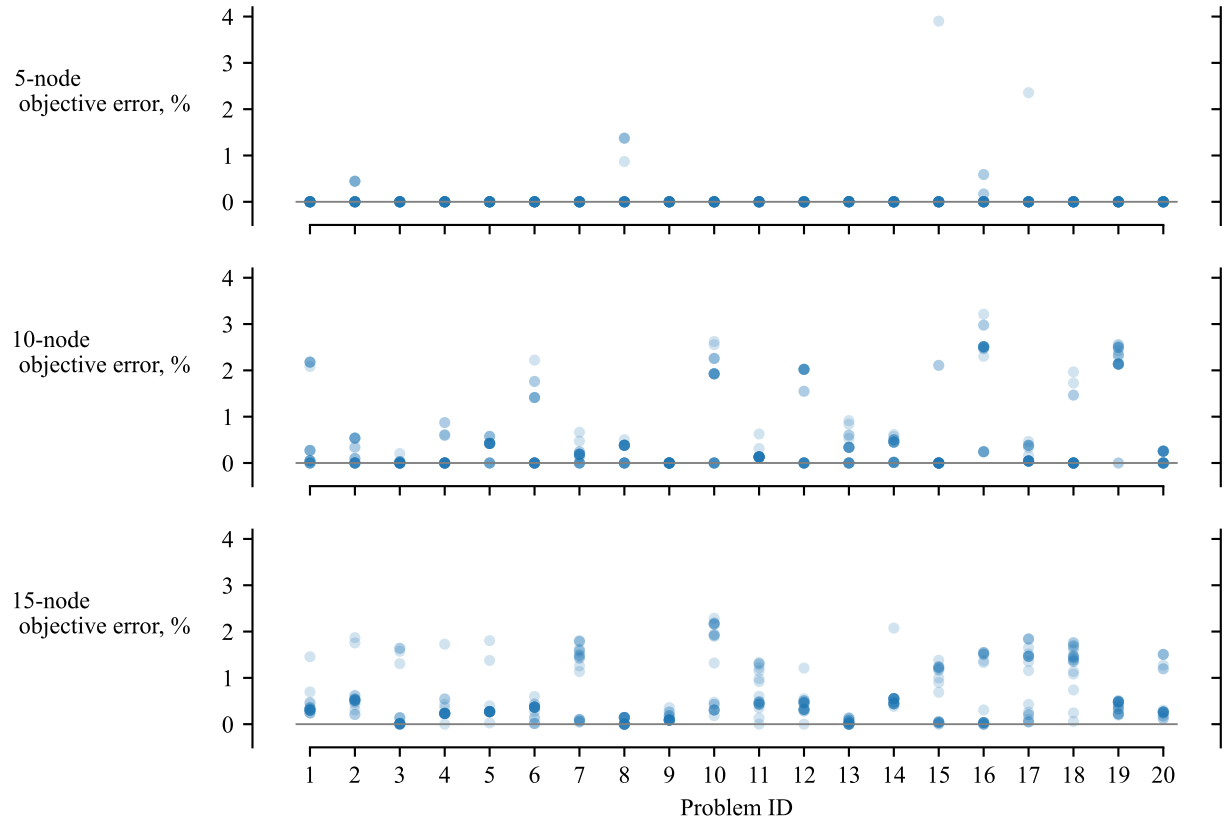


Fig. 10 Errors between the global minimum and the objective value found by the multi-start sequential heuristics.

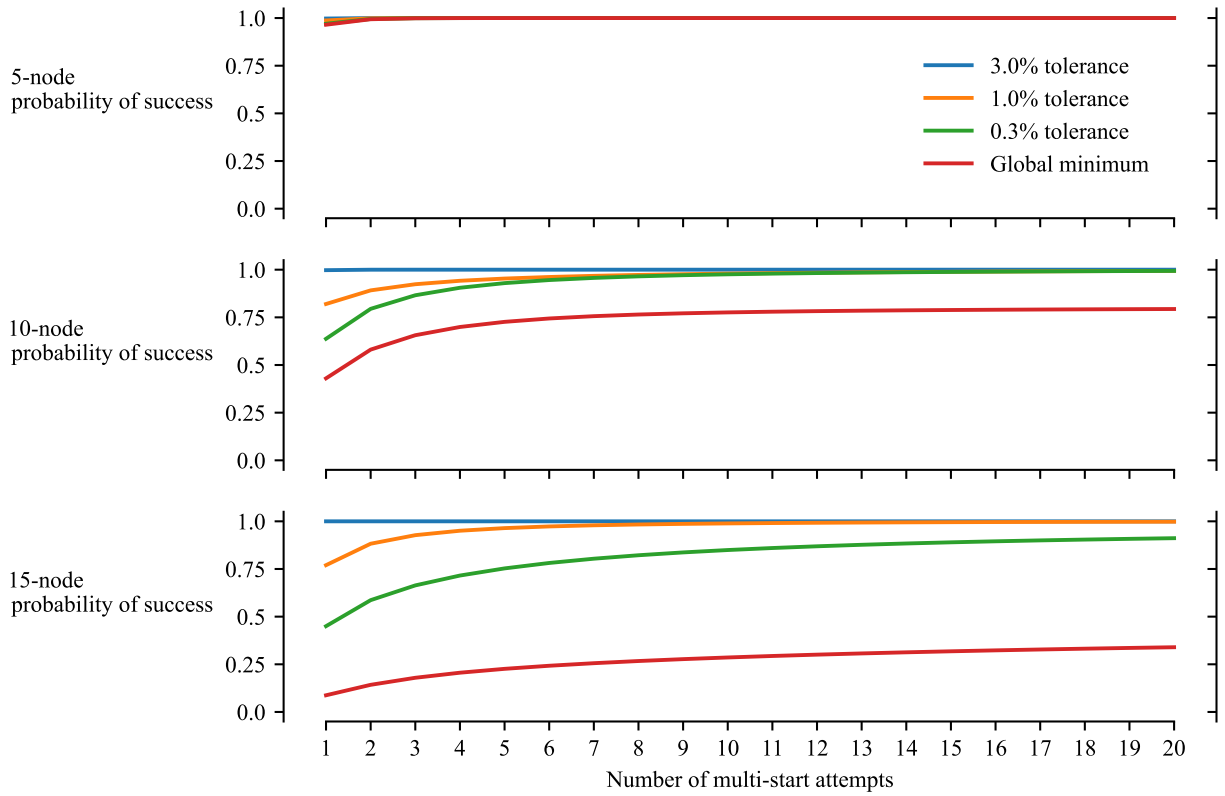


Fig. 11 Probability of finding an acceptable solution within a given tolerance when using the multi-start heuristics.

shown in Fig. 10. For example, suppose we will solve a new instance of the 10-customer problem. When we run the sequential heuristics once, Fig. 11 estimates that the probability of finding the global minimum is 42%; if we compromise and allow an error less than 0.3% with respect to the global minimum, then the probability of finding a solution within the 0.3% tolerance is 63%. When we run the sequential heuristics ten times from different initial fleets, the probability of finding the global minimum rises to 77%. If we allow an error less than 0.3% and perform ten multi-starts, then the probability of finding a solution within the 0.3% tolerance is 97%.

To summarize, the sequential heuristics can almost always find the global minimum in 5-customer problems. The algorithm is also robust on 10-customer problems if we accept an error of 0.3%. The probability of finding the global minimum in 15-customer problems is lower; however, it still is reliable in finding a near-optimal solution if we allow an error of 0.3-1.0%. Although it is impossible to validate the accuracy on larger-scale problems because the MINLP solver never converges, these benchmark results demonstrate the accuracy and robustness of the sequential heuristic algorithm we propose.

5.3 Scalability

We also benchmark the scalability of the optimization approaches with respect to the problem size. Here, we solve each benchmark problem by the two approaches under the same amount of the computational resource, and compare the best solutions found within the given resource. Table 4 lists the wall time limit for each problem size. For a fair comparison, we run both the sequential heuristics and Gurobi MINLP solver in series. Table 4 summarizes the resource allocation strategy of the sequential heuristics, i.e., the number of multi-start and sequential iterations that can be done within the time limit. Since the MINLP solver is not benefitted from a multi-start, we run a single optimization until the time limit. Note that on 30 and 60-customer problems, we provide a manual initialization to the MINLP solver; otherwise, the solver fails to find a feasible solution within the time limit. The initial solution we set is the simplest feasible routes where all customers are directly connected to the depot. An example of the initial solutions is shown in Fig. 9b. The MINLP solver fully converged on all of the 5 and 10-customer problems, whereas it did not converge on 15,

Table 4 Scalability benchmark study settings

No. of customers	Max. no. of UAVs	Wall time limit, s	Heuristics resource allocation	MINLP initialization
5	5	250	5 multi-starts, 10 iterations	No
10	10	1,000	10 multi-starts, 10 iterations	No
15	15	2,250	10 multi-starts, 10 iterations	No
30	30	9,000	15 multi-starts, 15 iterations	Manual initialization
60	60	36,000	20 multi-start, 20 iterations	Manual initialization

30, and 60-customer problems. The optimality gap achieved within the given time limit was 6.1–10.4%, 17.6–32.9%, and 31.1–36.3% on 15, 30, and 60-customer problems, respectively.

To compare the two optimization approaches, we define the objective ratio

$$\text{Objective ratio} = \frac{f^*_{\text{heuristics, best}}}{f^*_{\text{MINLP, best}}}, \tag{45}$$

which is the ratio of the best objective value found by the heuristics to the best objective value found by the MINLP solver (i.e., incumbent solution). The objective ratio of 1 means the heuristics and MINLP solver found the same solution. The objective ratio greater than 1 means the MINLP solver is more accurate, while the ratio less than 1 indicates that the heuristics finds a better solution.

Figure 12 shows the scatter plot of the objective ratio on all 100 benchmark problems. Each translucent dot in Fig. 12 corresponds to each problem. The opacity indicates that multiple benchmark problems resulted in a similar objective ratio. All 20 problems with five customers converged to the objective ratio of 1. On 10 and 15-customer problems, the objective ratio was 1 for most problems, although the heuristic solutions were slightly inferior to the MINLP solution for some problems. These results are consistent with the benchmark results shown in Fig. 10, where we observed a few percent of errors. On 30 and 60-customer problems, the sequential heuristics always results in a better objective value than the MINLP solution. The difference was 3–20% on 30 customers and 18–23% on 60 customers. The Gurobi MINLP solver showed poor performance even with the manual route initialization because of the NP-hardness of VRP and the non-convexity of the UAV energy constraints. To summarize, the sequential heuristic algorithm we proposed is scalable for large-scale design-routing problems as opposed to the MINLP branch-and-cut solver, which is limited to the problem size of 15 customers or less.

6. Results with Nonlinear UAV Design Optimization

This section shows the coupled design-routing optimization results that incorporate the UAV sizing optimization described in Section 3.3. We only use the sequential heuristics in this section because the MINLP solver cannot handle the nonlinearity of sizing optimization.

The differences of the optimization setup with and without the UAV sizing optimization is summarized in Table 3. We now include the cruise speed, rotor radius, cruise advance ratio, and wing area as design variables, which were fixed in Section 5. The other difference is that we now limit the number of different UAV designs to be less than the total number of UAVs in the fleet: in practice, it is preferable to have a small number of different designs considering the cost for the development, certification, and manufacture. Therefore, we seek to find the optimal fleet with potentially a large number of UAVs but a limited number of different designs.

6.1 Conventional Baseline

We first describe the procedure to obtain a baseline solution, which will be compared to the optimized solution. As a conventional design process, we assume a sequence of design optimization followed by a routing problem with no further iterations. The baseline design process is summarized as follows:

1) Generate m representative missions.
2) Optimize UAV designs for each mission. This generates m different designs.
3) Prepare p UAVs of each design and add them to the fleet; the total number of UAVs in the fleet is mp.

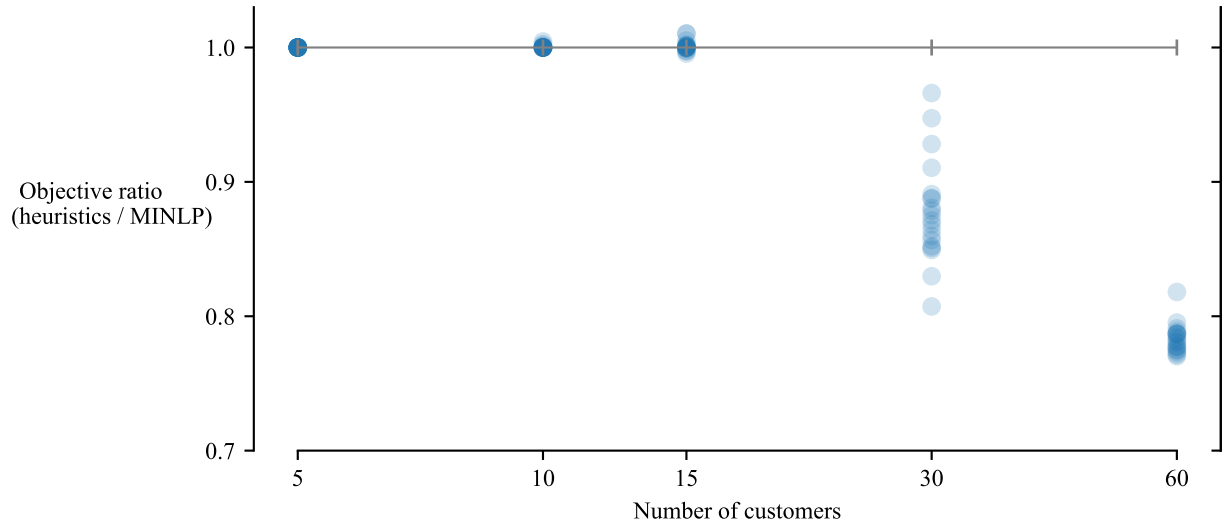


Fig. 12 Comparison of the best objective values found by the MINLP solver and sequential heuristics under the same wall time limit.

4) Solve VRP to determine the number of UAVs required to serve the customers.
5) Compute the baseline objective value f_0 by Eq. (22).

In this work, we generate five hexarotors and five QBiT designs, in a total of $m = 10$ different designs regardless of the number of customers. As representative missions, we assume the 40 km single-customer delivery with 1, 2, 5, 8, or 11 kg payload by the hexarotor, and the 60 km single-customer delivery with the same payload weights by the QBiT. We set $p = 2$ on 5, 10, and 15-customer problems, $p = 3$ on 30-customer problems, and $p = 6$ on 60-customer problems.

6.2 Comparison of Optimized and Baseline Objective Values

Figure 13 shows the comparison of the baseline solutions and the optimized objective values. We solved the same benchmark problems from Section 5, except we now extend the upper bound of the single-customer-trip range to 60 km and the delivery domain to 45 × 30 km. This modification intends to include the "switch point" of the takeoff weight between the hexarotor and QBiT (as shown in Fig. 4), which occurs at 30–50 km depending on the payload weight and number of customers on a route.

On each problem, we continue the sequential iterations starting from the baseline fleet until it satisfies the following convergence criteria: 1) at least 30 sequential iterations; 2) maximum of 100 iterations; 3) if the best objective value is unchanged for ten consecutive iterations. After the convergence, we evaluate the relative improvement of the optimized objective value with respect to the baseline, given by $|f^* - f_0|/f_0$. The scatter plot in Fig. 13 shows the distribution of the improvement for all benchmark problems. The mean improvement was 12–14% on 5, 10, and 15-customer problems, 21.8% on 30-customer problems, and 19.0% on 60-customer problems. This improvement in the objective value, which means the reduction of the fleet acquisition cost, demonstrates the importance of the design-routing coupling compared to the conventional baseline method.

6.3 Optimized Routes and Designs

In this section, we illustrate the optimized routes and UAV designs on a 60-customer problem. This problem has the same customer locations as one of the benchmark problems, whereas for better visualization, we now set the uniform demand of 0.5 kg per customer; the maximum number of different UAV designs to 6 ($m = 6$); and the maximum number of the same-design UAVs to 10 ($p = 10$).

Figure 14 shows the optimized routes. Each color corresponds to a different design; multiple routes of the same color indicate that we had multiple UAVs of the same design that served different routes. The blue colors show the routes flown by hexarotors, whereas the red colors represent QBiTs. Figure 15 shows the distribution of the routes in terms of the distance and number of customers along each route. Since the customer demand is uniform at 0.5 kg, the payload weight is obtained by multiplying 0.5 kg to the number of customers. The QBiTs served long-range missions

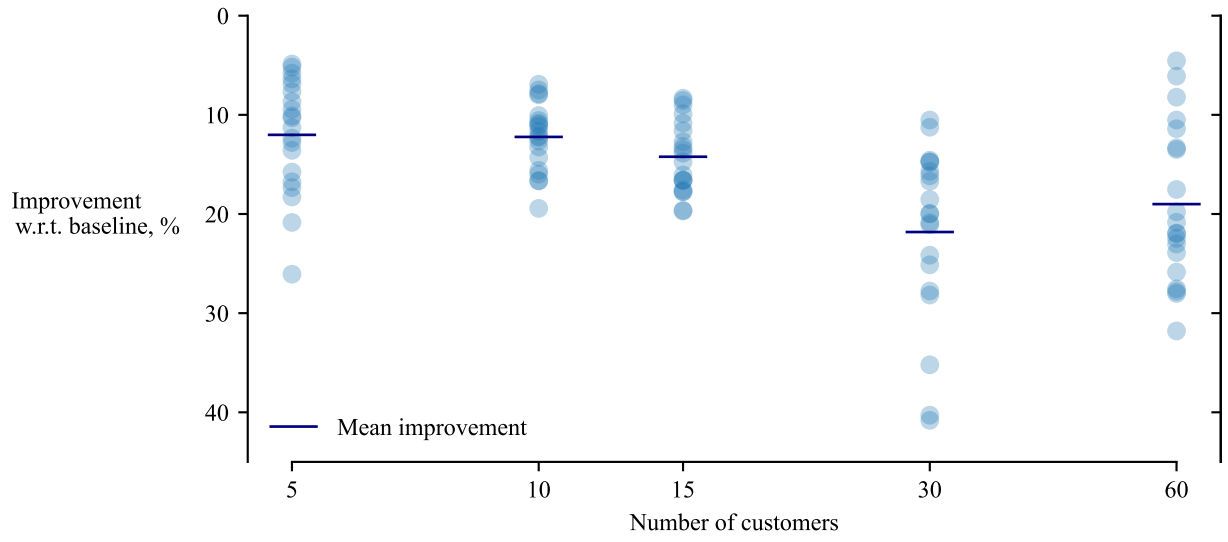


Fig. 13 Improvement of the optimized objective value compared to the conventional baseline solution.

with one or two customers, whereas the hexarotors performed 2- to 4-customer delivery with shorter ranges. This trend agrees with the mission suitability of each configuration discussed in Section 3.4.

Figure 16 shows the optimized design variables of all UAVs in the fleet. The optimized fleet has 27 UAVs (13 hexarotors and 14 QBiTs) of six different designs. The coordinates in the figure show the UAV takeoff weight and cruise speed; the UAV illustrations visualize the rotor size and wing area. As we observed in the design-only optimization study in Fig. 5, the hexarotors resulted in larger rotor radiuses than QBiT, whereas the QBiTs had higher cruise speeds. The larger rotor radius implies the potential for carrying large-volume cargo, although this study does not consider the cargo volume constraint.

On the 2-customer delivery with 28–37 km range, we observe the mixed pattern of the hexarotors (blue circles in Fig. 15) and QBiTs (brown triangles) performing similar missions. Both UAV designs were capable of flying these 28–37 km missions, because the hexarotor was designed for the 2-customer 43 km-range mission (the rightmost blue circle) that requires more energy; so as the QBiT, which was designed for the single-customer 49 km-range mission (the rightmost brown triangle). However, the QBiT cannot perform the 2-customer with more than 37 km range because of its inefficiency in hover. The takeoff weight was 3.16 kg for the hexarotor and 3.10 kg for the QBiT, therefore, the optimizer preferred to use the QBiTs as much as possible. As a result, the optimizer used 10 brown QBiTs, which reached the maximum number of the same-design UAVs. Then, the optimizer had to use the hexarotor for the remaining 28–37 km missions, which resulted in 7 hexarotors in total. If we removed the upper bound on p and allowed to use more than 10 QBiTs, the optimizer would assign QBiTs for all 28–37 km-range missions instead of the heavier hexarotors. Once the designs and number of each UAV were fixed, the assignment of UAVs to routes does not matter for the objective value, as long as the solution satisfies the energy constraints. In other words, even if we shuffled the assignment of the hexarotors and QBiTs within the 2-customer 28–37 km-range missions, the objective value would remain the same because the objective is the summation of the UAV takeoff weight in the fleet. This explains the mixed pattern of the hexarotor and QBiTs in the 2-customer 28–37 km missions. The additional factors that would narrow these assignments are the cargo volume and delivery speed consideration, although the current work does not include these in the optimization formulation.

6.4 Solution with Recharging Stations

The modularity of the proposed sequential heuristics allows us to upgrade the disciplinary models, i.e., the routing model or conceptual design model, without changing the top-level algorithm. To demonstrate this advantage, we solve a design-routing problem with an upgrade routing model with battery recharging. Here, we solve for the same customer locations and demands as Fig. 14, and we now add four recharging stations in the region. A UAV can charge its battery state to 100% every time it visits a recharging station to extend the range.

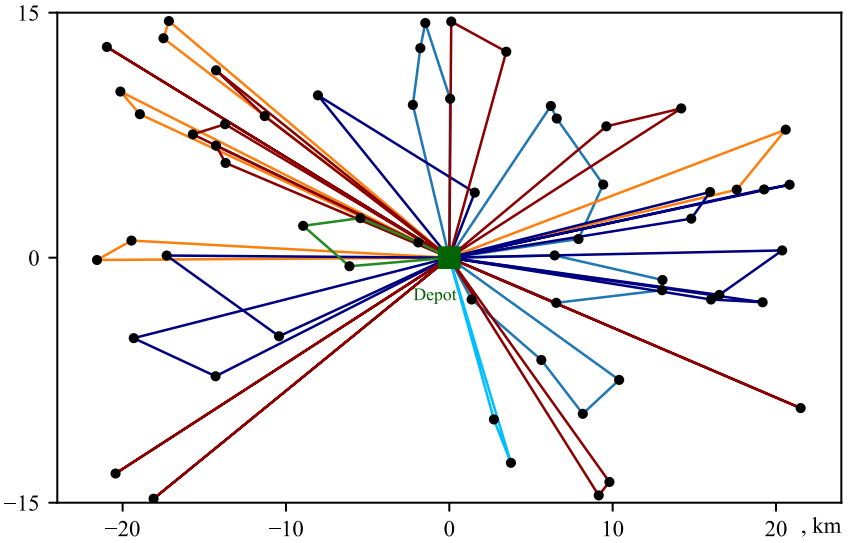


Fig. 14 Optimized routes for a 60-customer problem without recharging. The colors correspond to the different UAV designs shown in Fig. 16.

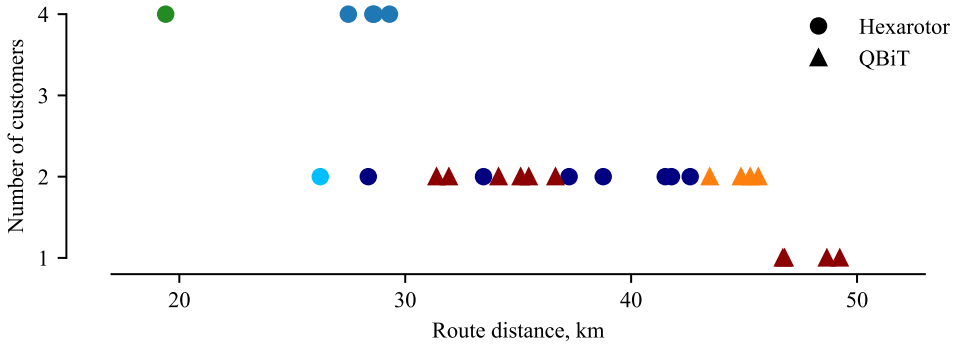


Fig. 15 Distribution of the distance and number of customers on each route in Fig. 14

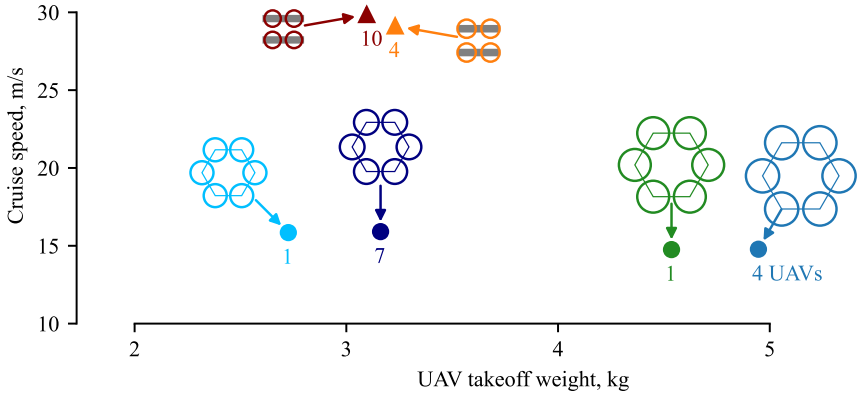


Fig. 16 Optimized UAV designs for the problem in Fig. 14. The UAV illustration size scales to the rotor size and wing area variables, and the number label corresponds to the number of each design of UAVs in the fleet.

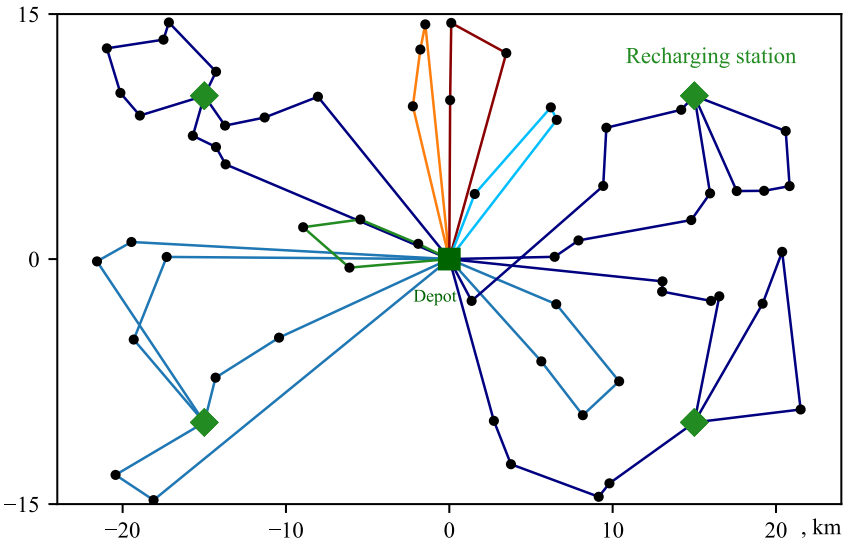


Fig. 17 Optimized routes for a 60-customer problem with recharging stations. The colors correspond to the different designs of UAVs shown in Fig. 18.

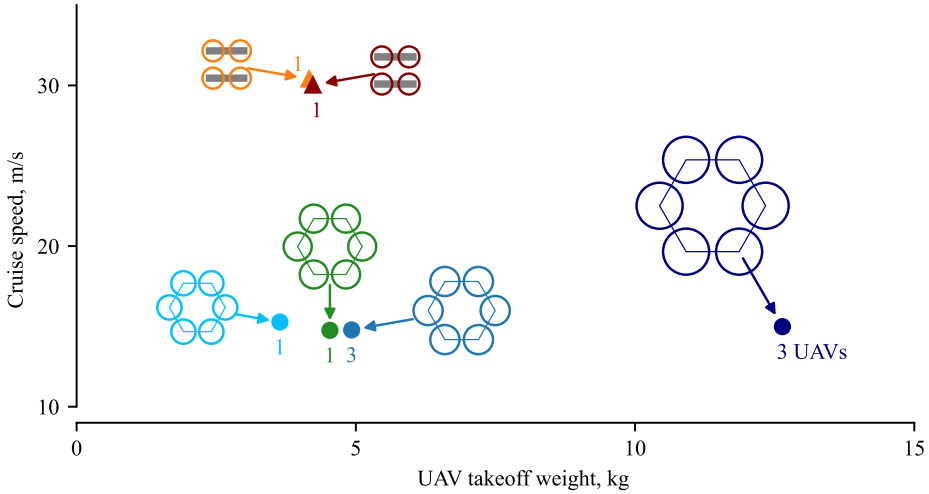


Fig. 18 Optimized UAV designs for the recharging problem.

Figures 17 and 18 show the optimized routes and UAV designs for the recharging problem. Compared to the original problem without recharging, the solution with recharging had fewer UAVs in the fleet and introduced larger-capacity hexarotors. The optimized fleet had only 10 UAVs in contrast to 27 UAVs for the problem without recharging; the largest UAV weight was 12.64 kg with recharging, 4.95 kg without recharging. Five out of 10 UAVs utilized the recharging stations; three UAVs served 11–12 customers each by recharging twice, and the other two UAVs served four customers by recharging once. The other five UAVs served customers far from the recharging stations or near the depot without recharging. The objective function value, which is the summation of the takeoff weights of all UAVs in the fleet, decreased by 25.6% compared to the original problem without recharging.

7. Conclusions

In this work, we solved the simultaneous optimization of UAV conceptual design and delivery operations to minimize the fleet acquisition cost. This MDO problem enabled us to find a set of optimal mission requirements for which the

UAV fleet should be designed, and the corresponding optimal UAV designs.

We first presented the conceptual design and sizing model of two eVTOL configurations, the hexarotor and quadrotor biplane tailsitter (QBiT), and investigated their key characteristics. The QBiT is efficient in cruise because of its wings, and therefore, it is suitable for long-range missions with a small number of customers. The hexarotor is efficient in hover and preferable for short-range and multi-customer delivery. We then coupled the eVTOL sizing model to a fleet size and mix VRP (FSMVRP) to optimize the UAV fleet design considering delivery operations. We presented a mixed-integer nonlinear programming (MINLP) formulation and solved it using a branch-and-cut solver. The MINLP solver found the proven global minimum on small-scale problems of up to 15 customers; however, it failed to find a feasible solution for a reasonable computational resource on larger-scale problems. The MINLP formulation was also restrictive in the model fidelity for UAV design, where we had to use the linear surrogate models.

To tackle large-scale problems and design nonlinearity, we proposed an effective sequential heuristic algorithm that combines specialized heuristics for VRP, gradient-based optimization for UAV design, and design exploration strategies to increase the probability of finding the global minimum. We benchmarked the accuracy and scalability of the sequential heuristics against the MINLP solver on 100 benchmark problems of the size of 5 to 60 customers. The proposed algorithm found the global minimum in the majority of 5- and 10-customer benchmark problems. It also converged to a near-optimal solution within a 0.3–1.0% error with respect to the global minimum for most of the 15-customer problems. On the scalability with respect to the problem size, the sequential heuristics consistently found a better solution than the MINLP solver for all of the 30- and 60-customer problems. We also demonstrated the benefit of incorporating VRP in the conceptual design process of a UAV fleet. Compared to the conventional baseline method, the design-routing optimization reduces the fleet acquisition cost by 21.8% on average for 30-customer problems; and 19.0% on average for 60-customer problems.

In addition to the scalability, modularity is another advantage of the proposed sequential heuristics. Because VRP and UAV design optimization are decomposed, the selection of the disciplinary models (i.e., an operational model or design model) is flexible. In other words, the sequential heuristics can accommodate a higher-fidelity design model or different VRP formulation, as long as the coupling interface remains the same. As an example, this paper demonstrated that the sequential heuristics can switch between two design models, the linear surrogate and the nonlinear sizing optimization, without modifying the top-level algorithm. We also showed the flexibility in the VRP model: the same algorithm solved the design-routing problems with and without battery recharging.

The simultaneous design-routing optimization is useful in two ways. First, it provides a model-based estimate of the minimum fleet size and the investment required for a prospective UAV delivery service. Second, the coupled optimization outputs a set of flight missions, which may be used as target mission requirements in the following preliminary and detailed design processes.

Acknowledgments

This work was funded by Israel's Ministry of Defense. The first author was also supported by Japan Student Services Organization (JASSO) Fellowship Program. Any opinions, findings, and conclusions, or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of Israel's Ministry of Defense or JASSO. The authors thank Max Opgenoord, Sicheng He, and Tomer Rokita for their feedback on the research methodology. The authors also acknowledge the Texas Advanced Computing Center (TACC) at The University of Texas at Austin for providing HPC resources.

References

[1] Sridharan, A., Govindarajan, B., and Chopra, I., "A Scalability Study of the Multirotor Biplane Tailsitter Using Conceptual Sizing," *Journal of the American Helicopter Society*, Vol. 65, 2019. doi:10.4050/JAHS.65.012009.

[2] Phillips, B., Hrishikeshavan, V., Yeo, D., and Chopra, I., "Flight Performance of a Package Delivery Quadrotor Biplane," 2017.

[3] Govindarajan, B., and Sridharan, A., "Conceptual sizing of vertical lift package delivery platforms," *Journal of Aircraft*, Vol. 57, No. 6, 2020, pp. 1170–1188. doi:10.2514/1.C035805.

[4] Bershadsky, D., Haviland, S., and Johnson, E. N., "Electric multirotor propulsion system sizing for performance prediction and design optimization," *57th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 2015, pp. 1–22. doi:10.2514/6.2016-0581.

[5] Winslow, J., Hrishikeshavan, V., and Chopra, I., "Design Methodology for Small-Scale Unmanned Quadrotors," *Journal of Aircraft*, Vol. 55, No. 3, 2018, pp. 1062–1070. doi:10.2514/1.C034483.

[6] Delbecq, S., Budinger, M., Ochotorena, A., Reysset, A., and Defay, F., "Efficient sizing and optimization of multirotor drones based on scaling laws and similarity models," *Aerospace Science and Technology*, Vol. 102, 2020, p. 105873. doi:10.1016/j.ast.2020.105873.

[7] Coutinho, W. P., Battarra, M., and Fliege, J., "The unmanned aerial vehicle routing and trajectory optimisation problem, a taxonomic review," *Computers and Industrial Engineering*, Vol. 120, 2018, pp. 116–128. doi:10.1016/j.cie.2018.04.037.

[8] Macrina, G., Di Puglia Pugliese, L., Guerriero, F., and Laporte, G., "Drone-aided routing: A literature review," *Transportation Research Part C: Emerging Technologies*, Vol. 120, No. July, 2020, p. 102762. doi:10.1016/j.trc.2020.102762.

[9] Poikonen, S., and Campbell, J. F., "Future directions in drone routing research," *Networks*, Vol. 77, No. 1, 2021, pp. 116–126. doi:10.1002/net.21982.

[10] Ulmer, M., and Thomas, B., "Same-Day Delivery with a Heterogeneous Fleet of Drones and Vehicles," *Networks*, 2018. doi:10.1002/net.21855.

[11] Chiang, W.-C., Li, Y., Shang, J., and Urban, T. L., "Impact of drone delivery on sustainability and cost: Realizing the UAV potential through vehicle routing optimization," *Applied Energy*, Vol. 242, 2019, pp. 1164–1175. doi:10.1016/j.apenergy.2019.03.117.

[12] Murray, C. C., and Raj, R., "The multiple flying sidekicks traveling salesman problem: Parcel delivery with multiple drones," *Transportation Research Part C: Emerging Technologies*, Vol. 110, 2020, pp. 368–398. doi:10.1016/j.trc.2019.11.003.

[13] Dorling, K., Heinrichs, J., Messier, G. G., and Magierowski, S., "Vehicle Routing Problems for Drone Delivery," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, Vol. 47, No. 1, 2017, pp. 70–85. doi:10.1109/TSMC.2016.2582745.

[14] Coelho, B. N., Coelho, V. N., Coelho, I. M., Ochi, L. S., Haghnazar K., R., Zuidema, D., Lima, M. S., and da Costa, A. R., "A multi-objective green UAV routing problem," *Computers & Operations Research*, Vol. 88, 2017, pp. 306–315. doi:10.1016/j.cor.2017.04.011.

[15] Choi, Y., Robertson, B., Choi, Y., and Mavris, D., "A Multi-Trip Vehicle Routing Problem for Small Unmanned Aircraft Systems-Based Urban Delivery," *Journal of Aircraft*, Vol. 56, No. 6, 2019, pp. 2309–2323. doi:10.2514/1.C035473.

[16] Cheng, C., Adulyasak, Y., and Rousseau, L. M., "Drone routing with energy function: Formulation and exact algorithm," *Transportation Research Part B: Methodological*, Vol. 139, 2020, pp. 364–387. doi:10.1016/j.trb.2020.06.011.

[17] Stolaroff, J. K., Samaras, C., O'Neill, E. R., Lubers, A., Mitchell, A. S., and Ceperley, D., "Energy use and life cycle greenhouse gas emissions of drones for commercial package delivery," *Nature Communications*, Vol. 9, No. 1, 2018, p. 409. doi:10.1038/s41467-017-02411-5.

[18] Kirschstein, T., "Comparison of energy demands of drone-based and ground-based parcel delivery services," *Transportation Research Part D: Transport and Environment*, Vol. 78, 2020, p. 102209. doi:10.1016/j.trd.2019.102209.

[19] Zhang, J., Campbell, J. F., Sweeney II, D. C., and Hupman, A. C., "Energy consumption models for delivery drones: A comparison and assessment," *Transportation Research Part D: Transport and Environment*, Vol. 90, 2021, p. 102668. doi:10.1016/j.trd.2020.102668.

[20] Golden, B., Assad, A., Levy, L., and Gheysens, F., "The fleet size and mix vehicle routing problem," *Computers & Operations Research*, Vol. 11, No. 1, 1984, pp. 49–66. doi:10.1016/0305-0548(84)90007-8.

[21] Hoff, A., Andersson, H., Christiansen, M., Hasle, G., and Løkketangen, A., "Industrial aspects and literature survey: Fleet composition and routing," *Computers & Operations Research*, Vol. 37, No. 12, 2010, pp. 2041–2061. doi:10.1016/j.cor.2010.03.015.

[22] Troudi, A., Addouche, S. A., Dellagi, S., and El Mhamedi, A., "Sizing of the drone delivery fleet considering energy autonomy," *Sustainability*, Vol. 10, No. 9, 2018, pp. 1–17. doi:10.3390/su10093344.

[23] Choi, Y., "A Framework For Concurrent Design and Route Planning Optimization of Unmanned Aerial Vehicle Based Urban Delivery Systems," PhD thesis, Georgia Institute of Technology, August 2019.

[24] Taylor, C., and de Weck, O. L., "Coupled Vehicle Design and Network Flow Optimization for Air Transportation Systems," *Journal of Aircraft*, Vol. 44, No. 5, 2007, pp. 1479–1486. doi:10.2514/1.27320.

[25] Mane, M., Crossley, W. A., and , N., "System-of-Systems Inspired Aircraft Sizing and Airline Resource Allocation via Decomposition," *Journal of Aircraft*, Vol. 44, No. 4, 2007, pp. 1222–1235. doi:10.2514/1.26333.

[26] Davendralingam, N., and Crossley, W., "Robust Approach for Concurrent Aircraft Design and Airline Network Design," *Journal of Aircraft*, Vol. 51, No. 6, 2014, pp. 1773–1783. doi:10.2514/1.C032442.

[27] Jansen, P. W., and Perez, R. E., "Coupled Optimization of Aircraft Families and Fleet Allocation for Multiple Markets," *Journal of Aircraft*, Vol. 53, No. 5, 2016, pp. 1485–1504. doi:10.2514/1.C033646.

[28] Hwang, J. T., Jasa, J., and Martins, J. R. R. A., "High-fidelity design-allocation optimization of a commercial aircraft maximizing airline profit," *Journal of Aircraft*, Vol. 56, No. 3, 2019, pp. 1165–1178. doi:10.2514/1.C035082.

[29] Martins, J. R. R. A., and Lambe, A. B., "Multidisciplinary Design Optimization: A Survey of Architectures," *AIAA Journal*, Vol. 51, No. 9, 2013, pp. 2049–2075. doi:10.2514/1.J051895.

[30] Roy, S., Crossley, W. A., Moore, K. T., Gray, J. S., and Martins, J. R. R. A., "Monolithic approach towards next generation aircraft design considering airline operations and economics," *Journal of Aircraft*, Vol. 56, No. 4, 2019, pp. 1565–1576. doi:10.2514/1.C035312.

[31] Martins, J. R. R. A., and Ning, A., *Engineering Design Optimization*, Cambridge University Press, 2021, Chap. 8, pp. 325–349. URL `https://mdobook.github.io`.

[32] Lambe, A. B., and Martins, J. R. R. A., "Extensions to the Design Structure Matrix for the Description of Multidisciplinary Design, Analysis, and Optimization Processes," *Structural and Multidisciplinary Optimization*, Vol. 46, 2012, pp. 273–284. doi:10.1007/s00158-012-0763-y.

[33] Chauhan, S. S., and Martins, J. R. R. A., "Tilt-wing eVTOL takeoff trajectory optimization," *Journal of Aircraft*, Vol. 57, No. 1, 2020, pp. 93–112. doi:10.2514/1.C035476.

[34] Gray, J. S., Hwang, J. T., Martins, J. R. R. A., Moore, K. T., and Naylor, B. A., "OpenMDAO: An open-source framework for multidisciplinary design, analysis, and optimization," *Structural and Multidisciplinary Optimization*, Vol. 59, No. 4, 2019, pp. 1075–1104. doi:10.1007/s00158-019-02211-z.

[35] Gill, P. E., Murray, W., and Saunders, M. A., "SNOPT: An SQP Algorithm for Large-Scale Constrained Optimization," *SIAM Review*, Vol. 47, No. 1, 2005, pp. 99–131. doi:10.1137/S0036144504446096.

[36] Wu, N., Kenway, G., Mader, C. A., Jasa, J., and Martins, J. R. R. A., "pyOptSparse: A Python framework for large-scale constrained nonlinear optimization of sparse systems," *Journal of Open Source Software*, Vol. 5, No. 54, 2020, p. 2564. doi:10.21105/joss.02564.

[37] Toth, P., and Vigo, D., *Vehicle Routing: Problems, Methods, and Applications*, 2nd ed., Society for Industrial and Applied Mathematics, 2014, Chap. 1, pp. 1–33. doi:10.1137/1.9781611973594.ch1.

[38] Gurobi Optimization, L., "Gurobi Optimizer Reference Manual," , 2021. URL `https://www.gurobi.com`.

[39] Perron, L., and Furnon, V., "OR-Tools," , 2019. URL `https://developers.google.com/optimization/`.