IPSJ Transactions on System LSI Design Methodology Vol.14 30–39 (Aug. 2021)

[DOI: 10.2197/ipsjtsldm.14.30]

Regular Paper

Energy-aware Routing of Delivery Drones under Windy Conditions

Satoshi Ito[1] Hiroki Nishikawa[1,2] Xiangbo Kong[1] Yusuke Funabashi[1] Atsuya Shibata[1] Shunsuke Negoro[1] Ittetsu Taniguchi[3,a)] Hiroyuki Tomiyama[1,b)]

Received: December 5, 2020, Accepted: April 21, 2021

Abstract: Drone is one of the promising vehicles that have exhibited the potential to reduce the cost and time in the field of logistics. However, due to the limitation of battery capacities, the flight time remains short. Therefore, energy consumption is one of the most critical concerns in drone delivery services. In order to reduce the energy consumption, drone generally needs to fly to the destination in as short a time as possible. For delivery services, the drone has loads to deliver and is exposed to weather effects such as windy conditions. This paper studies a routing problem for energy minimization of delivery drones under the assumption of windy conditions. This paper formally defines Energy Minimizing Vehicle Routing Problem (EMVRP) under windy conditions. Experimental scenarios with different wind velocities and the number of customers have been simulated, and demonstrate a comparison of the metrics in the energy consumption and the flight distance.

Keywords: EMVRP, dynamic programming, drone delivery

1. Introduction

In the domain of logistic operations, last-mile delivery by Unmanned Aerial Vehicles (i.e., drones) has been remarkably promising recently. Compared with a traditional regular delivery truck, a drone takes advantage of avoiding the congestion of road networks, faster delivery, and cutting the cost due to the flight without human operation. Regardless of the regulations for preventing the adoption of drone delivery, many countries try to relax the regulation for commercial companies. In 2013, Amazon announced to start a logistic service called Prime Air. The service assumes to utilize a number of drones to deliver packages to customers. At last, this is a first service that was realized in the UK [1]. In addition, another is that Google's Project Wing is currently testing food delivery drones in Australia [2].

For various usages as shown above, drones have been utilized in many companies. On the other hand, since most of drones are powered by batteries, the battery capacity is one of crucial issues to delivery by drones. Therefore, energy consumption is a serious constraint for drone delivery, and traditional drones can hardly exploit the full potential of providing the drone flight. In order to effectively utilize the full potential within the battery limitation, the determination of an effective delivery route is one of the principal actions.

Classically, routing problems start investigated from vehicle routing problems (VRPs [3]) that assume to use trucks for delivery. An extensive literature exists on the problems such as the works in Refs. [4], [5], [6]. One of typical problems of VRP is called as capacitated vehicle routing problem (CVRP) [4]. the vehicle starts and returns to a depot in such a way that the total traveling cost is minimized, where the total capacity of its vehicle is not exceeded. The work in Ref. [5] assumes to use vehicles powered by battery yet the vehicles have large capacities. In Ref. [6], the authors introduce energy consumption as a cost function, which is based on distance and load of the vehicle for the VRP with the capacity of loads. The work aims at minimization of the energy consumption during the delivery operation, and the problem is called as energy minimizing vehicle routing problems (i.e., EMVRP). Unlike these works, drones have much smaller capacity than the general vehicles. Therefore, routing problems for drone on such the assumptions as small capacity of loads, limited flight time, and energy consumption have been extensively researched for several years [7], [8]. The work [7] tries to minimize cost or delivery time under the consideration of battery- and load-weights, and also assumes to reuse drone. They proposed string-based simulated annealing algorithm compared with a MILP technique. The authors in Ref. [8] develop a DP-based algorithm for routing problem of drone. This work attempts to minimize the energy consumption of the flight on the assumption that the amount of energy consumption is dependent on the weight of packages to carry.

One characteristic of drone delivery unlike the trucks is that drone is easily affected by the effect of wind since the weight of drone is relatively much lighter than trucks even if the drone carries packages. Most of works for drones assume that the flight speeds are constant, however, this is not practical in the environ-

[1] Graduate School of Science and Engineering, Ritsumeikan University, Kusatsu, Shiga 525–8577, Japan
[2] JSPS Research Fellow
[3] Graduate School of Information Science and Technology, Osaka University, Suita, Osaka 565–0871, Japan
[a)] i-tanigu@ist.osaka-u.ac.jp
[b)] ht@fc.ritsumei.ac.jp

ments, where drone is flying under windy conditions.

This paper addresses one of VRPs for drone delivery, and this is an extended version of Ref. [8] which was published as a short paper in the same journal. This paper extends the previous work [8] in the following way: Our work takes into account the effect of wind, while the previous work assumes that drone flies in an ideal environment without the windy conditions. In our paper, given a set of items to deliver and wind velocity, the problem tries to find an optimal route which starts from a depot, delivers all of the items to customers, and comes back to the depot with the effect of wind in mind. To solve EMVRP, the paper presents a dynamic programming algorithm. dynamic programming algorithm efficiently finds exactly optimal routes in terms of energy consumption.

The reminder of this paper is organized as follows. Section 2 describes related work regarding from traditional routing problems to modern problems for drone delivery. Section 3 introduces our previous work presented in Ref. [8]. In this section, routing problem of minimizing energy consumption for drone delivery is formally defined. Based on the described problem, the section presents principles of dynamic programming approach, and describes its algorithm. Moreover, the evaluation of EMVRP for drone delivery with the traditional routing techniques is conducted. Section 4 is an extension of our work that is the routing problem to minimize the energy consumption under windy conditions. The overview of our proposed problem is described with a motivated example, and the effect of wind is formulated. The experimental scenarios have been conducted with instances of the problem, and compare the energy consumption, flight distance, and flight time, with or without the effect of wind. Finally, Section 5 concludes this paper.

2. Related Work

Classical routing problems are based on traveling salesman problem (i.e., TSP) which is well-known problem to ask the shortest route with visiting all customers and return back to the origin. TSP is one of NP-hard problems, and a large number of heuristic, meta-heuristic and exact algorithms have been developed for decades. One of the simplest yet efficient heuristic algorithms is the nearest neighbour (i.e., NN) algorithm, which selects the nearest customer one by one until all customers are visited. Exact algorithms for TSP include dynamic programming (i.e., DP) algorithms developed by Bellman [9] and by Held and Karp [10]. Based on such the works, a number of extensions for routing problems have been recently investigated.

As the extensions of TSP, VRP have been developed for vehicle delivery scenarios [3], [11]. The VRP was first proposed by Dantizig and Ramser [3]. With the increasing demand for practical purposes, the trend of VRP has been shifted into realistic VRP, which is widely known as rich VRP (i.e., RVRP) [11]. One of the most popular VRP is called as CVRP presented in Refs. [3], [4]. The goal of VRP are focused on optimization of various costs for a vehicle on condition that vehicles must leave and return to delivery base after the accomplishment of visiting all the customers. EMVRP further extend VRP for energy minimization [6], [12], [13]. The energy for travel from a point to another is defined as a function of the distance between the two points and the weight of the vehicle and carrying items. Then, EMVRP asks the route which visits all customers with the minimum energy. In general, EMVRP as well as VRP are NP-hard, and can hardly be solved in a practical time. In Ref. [6], Kara et al. formally define EMVRP as an integer programming problem. The energy is assumed to be proportional to a product of the distance and the weight. In Ref. [14] Wang extended EMVRP for heterogeneous vehicles and presented an integer programming formulation. Then, in Ref. [15], Wang proposed a genetic algorithm for the EMVRP for heterogeneous vehicles. There have appeared many works for VRP with different variants, however, they can hardly be applied to routing problems for drone delivery since the VRP have not dealt with the issues specified to drone such as flight range and the amount of loads under the limitation of battery capacity.

Recently, one of routing problems unique to drone delivery is proposed, so called the flying sidekick traveling salesman problem (i.e., FSTSP). This work assumes to deliver packages by drones and trucks work together [16]. In the literature, Wang et al. [17] introduce VRP that multiple truck and drones are utilized for delivery. They conduct worst case scenarios analysis to maximize the advantages of drone usage. Chang and Lee attempt to search the effective route with truck carrying drones [18]. They develop an approach to cluster delivery locations for drones by K-means clustering, traveling route for truck is minimized based on TSP, and their proposal is to find shift-weights based on nonlinear programming. Jeong et al. [19] tackle the truck-drone hybrid delivery system with no-fly zone in mind. On the other hand, due to the battery capacity, energy-aware routing problems for drone delivery have been appealing. In Ref. [7], the work attempts to minimize the total cost under time window constraint during delivery operation. Moreover, sub-optimal solutions for the problem are presented by employing a simulated annealing heuristic approach. These works above have an assumption that the velocity of drone is basically constant. However, in the real world, the impact exerted by windy condition is not negligible, especially in terms of determining the flight path of drone [20].

The flight range and energy consumption of drones are greatly affected by the effect of wind [21], [22]. A simplest model of windy conditions is the steady wind [23]. It means the constant speed and direction, and which does not change during drone operation. Another model is based on the physically environmental information such as pressure, temperature, humidity, and so on. Kundu and Matis create a windy model derived from drag and lift ratio [24]. Moreover, statistically analyzing from historical data enables to estimate the current status of windy conditions. However, the models based on environments and statistics are much more complex than the simple model, and this paper assumes the simplest assumption that the wind blows constantly and does not change during delivery.

Their work assumes windy conditions, but does not assume the weight of packages to carry. Thus, their cost function does not include the weight of packages. On the other hand, in our work, the weight of packages and the wind affects the energy consumption may change the flight speed of drone.

To our best knowledge, this is the first paper to tackle EMVRP for drone delivery, where energy consumption is influenced by the weight of packages, under windy conditions.

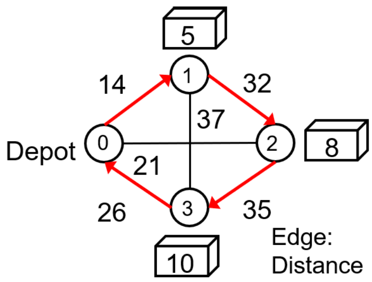
3. Energy-aware Routing Problem for Delivery Drones

This paper is the extension of the work in Ref. [8]. This section introduces the routing problem presented in Ref. [8] in order to compare with the proposed problem in this paper. Let us define the problem with an example of energy-aware routing problem for delivery drones, and describe the formulation based on an integer programming.

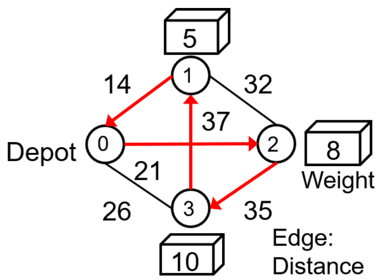
3.1 A Motivated Example

In this section, the definition of the routing problem for delivery drone is described with a motivated example. **Figure 1** shows the example of the routing problem for delivery drones. A set of customers with a depot is represented as a non-directed complete graph. In the example, the node labeled “0” denotes a depot, and the other three nodes denote customers (i.e., shipping destinations). The numbers in the boxes represent the weight of the items to deliver, and the numbers on the edges represent the distance between the two places. The optimal route for TSP is shown in Fig. 1 (a). The total distance of the route is 107 (= 14 + 32 + 35 + 26). However, this route may not be optimal for EMVRP. In general, the power consumption of a drone mainly depends on the total weight of loaded items.

The energy consumption of drones depends not only on the flight distance but also on the total weight of loaded items. The heavier the load is, the higher the energy consumption is. The optimal route for EMVRP is shown in Fig. 1 (b) [*1]. The total distance of the EMVRP route is 107 (= 21 + 35 + 37 + 14) as well as that of TSP, but the delivery order of the flight is not the same in terms of minimizing the energy consumption.



(a) TSP



(b) EMVRP

Fig. 1 Optimal routes.

[*1] Actually, the optimal EMVRP route depends on the drone.

3.2 Problem Definition

We are given N items to deliver. Without loss of generality, no two items are to be delivered to the same customer. The items or customers are numbered from 1 to N. The customer where item i $(1 \le i \le N)$ is to be delivered is called customer i. As already mentioned, the depot is numbered 0 as shown in Fig. 1.

This paper assumes that the items are delivered in a single trip by a drone. All of the items are uploaded onto the drone at the depot, and the drone starts a trip. If the total weight of the items exceeds the capacity of the drone, the items need to be partitioned into groups before routing and delivery, and how to partition the items is out of the scope of this paper.

Let $w(i)$ denote the weight of item i and $d(i1, i2)$ denote the distance between customers $i1$ and $i2$. Also, let $x(j)$ denote the j-th visited customer, which is the decision variable of the routing problem. Since a route starts and ends at the depot, we define:

$$x(0) = x(N+1) = 0 \tag{1}$$

Also, all of the customers are visited once, which is formally defined as follows:

$$1 \le x(j) \le N \quad (1 \le j \le N) \tag{2}$$

$$x(j_1) \ne x(j_2) \quad (1 \le j_1, j_2 \le N, j_1 \ne j_2) \tag{3}$$

Let $e(p(w), d(x(j), (j+1))/v_d)$ denote the energy consumption of the delivery drone, which is a function of payload weight w and flight distance d. Function $e(p(w), d(x(j), (j+1))/v_d)$ depends on the drone, and is assumed to be given. For example, $e(p(w), d(x(j), (j+1))/v_d)$ is defined as:

$$e\left(p(w), \frac{d(x(j), (j+1))}{v_d}\right) \propto p(w + W_{drone}) \times \frac{d(x(j), x(j+1))}{v_d} \tag{4}$$

where W_{drone} is the weight of drone itself and $p(w)$ is the power consumption and v_d is the speed of drone. v_d is assumed to be constant in this section. It should be noted that this work is not restricted to formula (4).

Let $W(j)$ denote the total payload when the drone leaves j-th visited customer. When the drone starts a trip, all of the items are loaded. Therefore, the following formula expresses the constraint.

$$W(0) = \sum_{i=1}^{N} w(i) \tag{5}$$

When the drone makes the j-th stop $(1 \le j \le N)$ at customer $x(j)$, an item of weight $w(x(j))$ is unloaded. Therefore, the total payload when the drone leaves customer $x(j)$ is defined as:

$$W(j) = W(j-1) - w(x(j)) \tag{6}$$

Then, the objective of our EMVRP is defined as:

$$\mathbf{min} \quad \sum_{j=0}^{N} e\left(p(W(j)), \frac{d(x(j), x(j+1))}{v_d}\right) \tag{7}$$

Algorithm 1 Dynamic Programming for EMVRP

```
W_init ← ∑ W
for next_customer ∈ 𝕍 do
    dp[1 << (next_customer − 1)][next_customer] ← energy(depot to next_customer with W_init)
    Weight[1 << (next_customer − 1)] ← (W_init − W_next_customer)
end for

for state in [0, 1, 2, ..., (2^N − 1)] do
    for next_customer ∈ 𝕍 do
        if next_customer has not been visited yet then
            for prev_customer ∈ 𝕍 do
                if prev_customer has been already visited then
                    dp[state|(1 << (next_customer − 1))][next_customer] ←
                     min(dp[state][prev_customer] + energy(prev_customer to next_customer with Weight[state]),
                           dp[state|(1 << (next_customer − 1))][next_customer])
                    Weight[state|(1 << (next_customer − 1))] ← Weight[state] − W_next_customer
                end if
            end for
        end if
    end for
end for

min_cost ← INFINITE
for prev_customer ∈ 𝕍 do
    min_cost ← min( dp[2^N − 1][prev_customer] + energy(prev_customer to depot without payload), min_cost)
end for
```

The EMVRP addressed in this paper is formally defined as follows. Given w, W_{drone}, d, v_d and e, find x for the objective (7) while meeting the constraints (1)–(3), (5) and (6).

3.3 Dynamic Programming Algorithm

This section presents a DP-based approach to the routing problem described in Section 3.2. Generally, the routing problems such as TSP and VRP as well as the problem in this section are known as NP-hard problems. Therefore, the routing problems can hardly be solved in practical time, and many heuristic algorithms are developed by researchers. In order to evaluate the quality of such the algorithms, exact solutions are very crucial for comparison. In this paper, DP-based algorithm, which is one of popular algorithms that can find an exact optimal solution, is utilized to the problem. The following section presents principles of the algorithm, and describes the overview of its algorithm.

3.4 Principles

This section outlines our DP-based algorithm for EMVRP [*2]. In general, dynamic programming is an approach to mathematical optimization problems. DP divides a given problem into smaller sub-problems in a recursive manner. Then, using the optimal solutions of the sub-problems, DP finds an optimal solution for the original problem. DP runs efficiently by avoiding re-computation of similar sub-problems, where the similar sub-problems denote the sub-problems whose optimal solutions are the same with each other. In the design of DP algorithms, it is crucial to derive a recurrence relation between an original problem and sub-problems.

Let $\mathbb{S}$ denote a set of customers who are already visited, and let i be the latest visited customer in $\mathbb{S}$. We call a pair $(\mathbb{S}, i)$ as a *state*. Obviously, the initial state is $(\{\phi\}, 0)$. Then, we define a problem asking the minimum energy consumption $E(\mathbb{S}, i)$ for delivery from the initial state to state $(\mathbb{S}, i)$. Now, we can derive a recurrence formula to calculate $E(\mathbb{S}, i)$ as follows.

*2 Due to the limited space, pseudo-code of the algorithm is not presented.

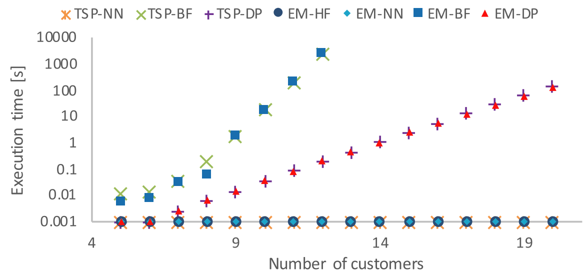


Fig. 2 Runtime of the routing algorithms.

$$E(\mathbb{S}, i) = \min\left\{ E(\mathbb{S}\backslash i, i') + e\Big(p(W'(\bar{\mathbb{S}}) + w(i)), \frac{d(i', i)}{v_d}\Big) \;\middle|\; i' \in \mathbb{S}\backslash i \right\} \quad (8)$$

Recall that i is the latest customer in $\mathbb{S}$. In the formula, i' denotes the second latest customer. $E(\mathbb{S}\backslash i, i')$ is the minimum energy consumption for flying from the depot to i', and $e(p(W'(\bar{\mathbb{S}}) + w(i)), d(i', i)/v_d)$ is the energy consumption for flying from i' to i. $W'(\bar{\mathbb{S}})$ denotes the total weight of items which are not yet delivered, which is formulated as:

$$W'(\bar{\mathbb{S}}) = \sum_{k \notin \mathbb{S}} w(k) \quad (9)$$

In formula (8), it should be noted that, when departing from i', item i is still loaded on the drone. Therefore, $w(i)$ is added to $W'(\bar{\mathbb{S}})$. Also, it is obvious that the energy consumption at the initial state, i.e., before leaving the depot, is zero.

$$E(\{\phi\}, 0) = 0 \quad (10)$$

The original routing problem asks the minimum energy consumption when the drone departs from the depot, visits all of N destinations, and comes back to the depot. Formally, the original problem asks:

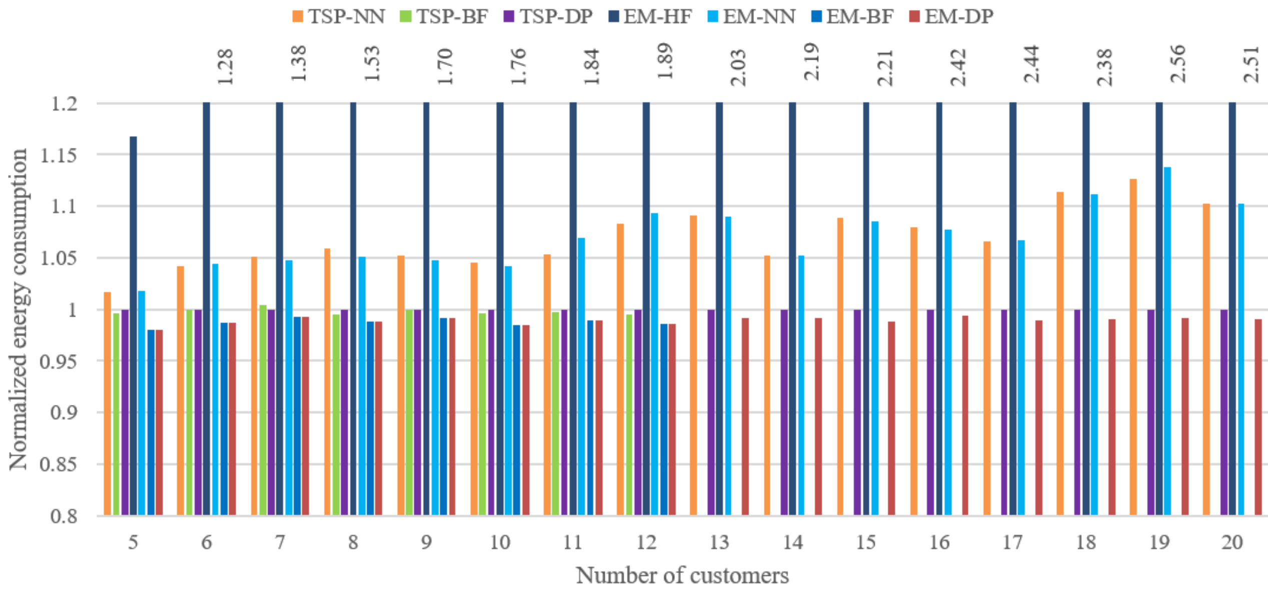


Fig. 3 Energy consumption of the obtained routes.

$$E(\{0, 1, 2, \ldots, N\}, 0) \tag{11}$$

The problem in expression (11) is recursively partitioned into sub-problems according to formula (8), reaching formula (10), and then, the optimal route with the minimum energy consumption is obtained.

3.5 Algorithms

A dynamic programming algorithm for the routing problem is presented in the previous work [8], based on the principles above. The pseudo code of our algorithm is outlined in Algorithm 1.

W_{init} is the total weight of all items to be delivered, and $\mathbb{V}$ is a set of customers. *state* represents a set of customers who are already visited. Actually, *state* is a bit-vector of length N, where N is the number of customers. If customer i is visited, the $(i-1)$-th bit is set. $dp[state][customer]$ is a two-dimensional array which stores the energy consumption, and it corresponds to $E(\mathbb{S}, i)$ in Section 3.4. For example, dp[3][2] means that the drone already visited customers 1 and 2 [*3], and the drone is now at 2. Lines 2–4 calculates the energy consumption from the depot to the first customer. Then, Lines 7–18 travel all of remaining customers, and finally in Lines 21–23, the drone comes back to the depot and the minimum energy consumption is calculated. Lines 7–18 are the main part of the DP algorithm. Instead of recursive procedure calls, the algorithm calculates the energy with three-level nested loops. The computational complexity of our DP algorithm is $O(2^N \times N^2)$, which is much faster than an exhaustive search of $O(N!)$.

3.6 Evaluation

Our DP algorithm as well as several existing algorithms are implemented in Python, and are compared in terms of the runtime of the algorithms and the quality of solutions (i.e., the energy consumption of the obtained routes). Seven routing algorithms shown below are compared in the experiments.

TSP-NN: The nearest-neighbor algorithm for TSP.
TSP-BF: A brute-force algorithm for TSP.
TSP-DP: A dynamic programming algorithm for TSP [10].
EM-HF: A heaviest-first algorithm for EMVRP. It iteratively selects the heaviest item one after another.
EM-NN: A NN-like algorithm for EMVRP. It iteratively selects the minimum-energy neighbor one after another.
EM-BF: A brute-force algorithm for EMVRP.
EM-DP: Our DP algorithm proposed in this paper.

Expression (4), which is presented in Ref. [6], is used for energy calculation. Based on the power measurement results in Ref. [25], W_{drone} is set to 300 and the maximum payload is set to 48. A total of 320 instances of EMVRP are randomly generated, where the number of customers ranges from 5 to 20. For each number of customers, there are 20 problem instances.

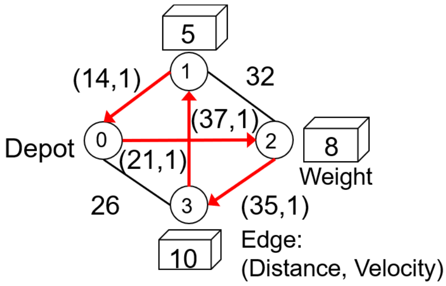
Figure 2 shows the results on the runtime of the seven routing algorithms on Intel Core i5 processor. The complexity of our TSP-NN, EM-HF and EM-NN implementations are $O(N^2)$, and their runtime is less than 1 millisecond in any cases. The complexity of TSP-BF and EM-BF is $O(N!)$, and they fail to find optimal solutions within an hour for delivery to more than 12 customers. Although the complexity of TSP-DP and EM-DP is exponential, they are practical for delivery to 20 customers.

Figure 3 shows the results on energy consumption obtained by the seven algorithms, where the results are normalized to TSP-DP. As mentioned above, there exist 20 problems for each number of customers. In the figure, the average of the seven results is depicted. The results show that the three heuristic algorithms, TSP-NN, EM-HF and EM-NN, are not effective. As long as EM-BF finds solutions, it is confirmed that our EM-DP algorithm successfully finds the same solutions as EM-BF does. Also, it is observed that TSP-DP is not optimal in terms of energy consumption. EM-DP outperforms TSP-DP by 1.08% on average.

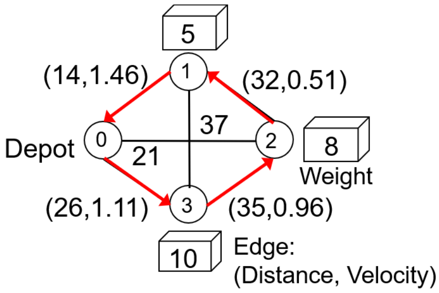
4. A Drone Flight under Windy Conditions

In Section 3, the routing problem of a drone has been formu-

[*3] Note that 3 is 011 in binary notation, where the first and second bits from the right are 1.



(a) Routing without the windy effect



(b) Routing with the windy effect

Fig. 4 Optimal routes.

lated without weather conditions. However, the flight condition for drones often depends on the weather in the real world. One of most significant weather conditions to drone flight is windy condition, as it may affect the flight path, resulting in the increased energy consumption and the longer flight time.

4.1 The Routing Problem under Windy Condition

This section describes the routing problem for drone delivery under windy condition with a motivated example. **Figure 4** shows the example. Figure 4 (a) represents the routing without windy effect, on the other hand, Fig. 4 (b) represents the routing with its windy effect. In this work, the wind is assumed to be steady and does not change during delivery operation. As well as the earlier mentioned example in Section 3.1, the node labeled "0" denotes a depot, and the other nodes denote customers. The numbers in the boxes represent the weight of the items to deliver. Unlike the prior example, there are two numbers on the edges in the parentheses. While the left number in the parentheses is represented as the distance, the right number represents the velocity of the drone under the windy condition. In this example, the velocity of a drone is assumed to set to 1. If the wind blows against the heading direction of the drone, the velocity of the drone becomes lower than one. On the other hand, it becomes larger than one if tailwind blows.

The route of a delivery drone without the wind is shown in Fig. 4 (a). In a practical case, the delivery drone is assumed to usually flight outdoor, and the effect of the windy condition can not be ignored. Under the windy condition, the optimal route for the flight is shown in Fig. 4 (b). As the result, the distance to flight is the same each other, but the optimal routes aware of energy consumption are different. Moreover, the flight times in the figures are also different. In Fig. 4 (a), the flight time of the drone is 107 (= $21/1 + 35/1 + \cdots + 14/1$), though that in Fig. 4 (b) is

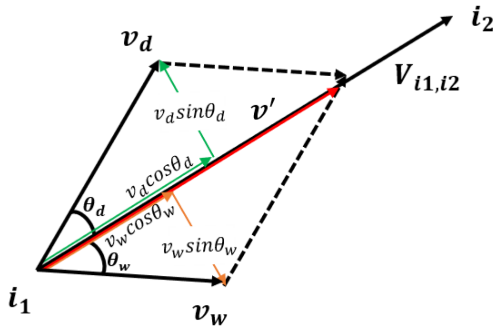


Fig. 5 The effect of wind.

132.215 (= $26/1.11 + 35/0.96 + \cdots + 14/1.96$). The results imply that not only the total flight time but also the delivery order are crucial to minimize the energy consumption of delivery.

4.2 The Effect of Wind

In this section, the velocity of a drone under windy conditions is calculated.

Let us consider the flight of a drone from the customer $i1$ to the customer $i2$. **Figure 5** illustrates the concept of the flight from the customer $i1$ to the customer $i2$ under windy condition. In the figure, $\boldsymbol{v_w}$ denotes the velocity vector of the wind. In this paper, the wind is assumed to be steady, and does not change during the delivery. $\boldsymbol{V_{i1,i2}}$ denotes the position vector from customer $i1$ to $i2$. If the drone straightforwardly reaches $i2$ from $i1$, the velocity vector of the drone $\boldsymbol{v_d}$ should head towards $\boldsymbol{V_{i1,i2}}$. However, the wind have an effect to $\boldsymbol{v_d}$, and change the direction and the velocity. Therefore, the direction of $\boldsymbol{v_d}$ should head in such a way that the synthetic vector $\boldsymbol{v'}$ head towards the vector $\boldsymbol{V_{i1,i2}}$. In this example, we are given $\boldsymbol{v_w}$ and $\boldsymbol{V_{i1,i2}}$ as well as the velocity of drone v_d. From a customer to another customer, we need to find the direction of $\boldsymbol{v_d}$ and the flight time for calculating energy consumption.

Let θ_w denote the angle between $\boldsymbol{v_w}$ and $\boldsymbol{V_{i1,i2}}$. The relationship between $\boldsymbol{v_w}$ and $\boldsymbol{V_{i1,i2}}$ is expressed with the internal product of these vectors.

$$\cos\theta_w = \frac{\boldsymbol{v_w} \cdot \boldsymbol{V_{i1,i2}}}{|\boldsymbol{v_w}||\boldsymbol{V_{i1,i2}}|} \tag{12}$$

Let θ_d denote the angle with $\boldsymbol{v_d}$ and $\boldsymbol{V_{i1,i2}}$. In order to direct the drone to head towards $\boldsymbol{V_{i1,i2}}$, the following formula must be met:

$$|\boldsymbol{v_d}|\sin\theta_d = |\boldsymbol{v_w}|\sin\theta_w \tag{13}$$

The norm of the synthetic vector $\boldsymbol{v'}$ of $\boldsymbol{v_w}$ and $\boldsymbol{v_d}$ can be expressed as follows:

$$|\boldsymbol{v'}| = |\boldsymbol{v_d}|\cos\theta_d + |\boldsymbol{v_w}|\cos\theta_w \tag{14}$$

According to formulas (13) and (14), the scalar of the vector $\boldsymbol{v'}$ is led in the following formula:

$$|\boldsymbol{v'}| = |\boldsymbol{v_d}|\cos(\arcsin\frac{|\boldsymbol{v_w}|\sin\theta_w}{|\boldsymbol{v_d}|}) + |\boldsymbol{v_w}|\cos\theta_w \tag{15}$$

Finally, we can obtain the velocity vector of the drone $\boldsymbol{v'}$ as referred to the formulas (12) and (15). Based on the formulas presented above, unlike the formulas presented in Section 3, the routing problem presented in this section is necessary to be redefined with taken into account the wind effect. Thus, the energy

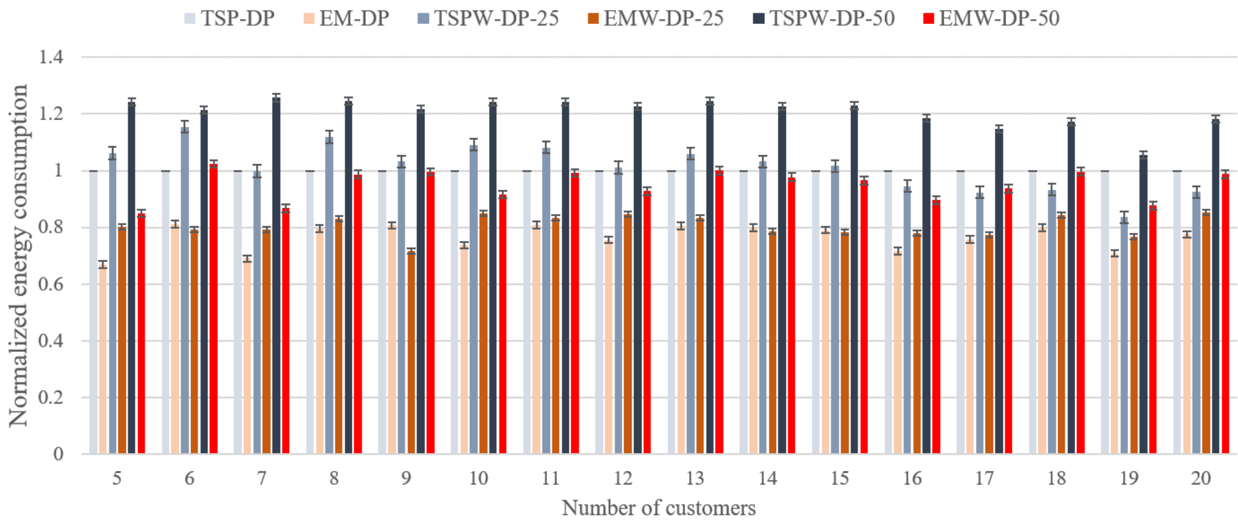


Fig. 6 Energy consumption of the routes.

consumed by the drone flight under the wind is redefined as follows:

$$e(w, d, \boldsymbol{v}') \propto p(w) \times t(d, \boldsymbol{v}') \tag{16}$$

Similar to Section 3, Let $\mathbb{S}$ denote a set of customers who are already visited, and let i be the last-visited customer in $\mathbb{S}$. Also, where $t(d, \boldsymbol{v}')$ is flight time between the customers. Given i as the latest customer in $\mathbb{S}$ and i' as the second latest customer, $E(\mathbb{S}\backslash i, i')$ is the minimum energy consumption for flying from the depot to i'. $e(p(W'(\bar{\mathbb{S}}) + w(i)), t(d(i', i), \boldsymbol{v}'))$ is the energy consumption for flying from i' to i. $p(W'(\bar{\mathbb{S}}) + w(i))$ represents the power consumption dependent on the total weight of items and the drone. $W'(\bar{\mathbb{S}})$ denotes the total weight of items which are not yet delivered. $t(d(i', i), \boldsymbol{v}')$ is the flight time from i' to i under windy conditions. Now, the recurrence formula (8) presented in Section 3 is formulated to take into account the wind as follows:

$$E(\mathbb{S}, i) = \max\{\ E(\mathbb{S}\backslash i, i') + e(p(W'(\bar{\mathbb{S}}) + w(i)), t(d(i', i), \boldsymbol{v}')) \\ \mid i' \in \mathbb{S}\backslash i\ \} \tag{17}$$

The additional complexity for considering the wind is $O(N^2)$ since flight speed needs to be recalculated for any two customers. Therefore, the overall computational complexity of our dynamic programming algorithm is still $O(N^2 \times 2^N)$.

4.3 Experiments

In order to evaluate the extended problem to take into account windy conditions, this section presents experiments that have been conducted in terms of energy consumption, flight time, and distance.

4.3.1 Experimental Setup

The weight of a drone is assumed to be 300 and the maximum load is set to 48. For the comparison, a total of 320 instances of EMVRP are randomly generated, where the number of customers is varied from 5 to 20. In other words, twenty problem instances are included for each number of customers. The default velocity of a drone is set to 1. The wind is assumed to blow in random directions, and the speed of the wind sets to 0.25 and 0.5 of drone's speed.

In order to evaluate differences between the problems, dynamic programming is utilized to the problems. In the experiments, the following nine techniques are employed:

TSP-DP: A DP-based algorithm for TSP [10] without the wind.

EM-DP: A DP-based algorithm for EMVRP without the wind.

TSPW-DP-25: A DP-based algorithm for TSP with the wind. The wind speed is set to 0.25.

EMW-DP-25: A DP-based algorithm for EMVRP with the wind. The wind speed is set to 0.25.

TSPW-DP-50: A DP-based algorithm algorithm for TSP [10] with the wind. The wind speed is set to 0.5.

EMW-DP-50: A DP-based algorithm for EMVRP with the wind. The wind speed is set to 0.5.

4.3.2 Effects of Windy Conditions on Energy Consumption

This section evaluates the energy consumption obtained by TSP-DP, EM-DP, TSPW-DP-25, EMW-DP-25, TSPW-DP-50, and EMW-DP-50.

The experimental results for the comparison of energy consumption are shown in **Fig. 6**. Figure 6 shows the energy consumption with the compared techniques. The X-axis represents the number of customers to deliver, and the Y-axis represents the energy consumption normalized to TSP-DP.

According to the results, EM-DP, EMW-DP-25, and EMW-DP-50 totally obtain the lower energy consumption than TSP-DP, TSPW-DP-25, and TSP-DP-50, respectively. Compared to TSP-DP, EM-DP achieves the lower energy consumption by 23.6% on average. EMW-DP-25 also reduces the energy consumption by up to 20.8% towards the results of TSPW-DP-25. In the cases with 0.25 of the wind speed, EMW-DP-25 reduces the energy consumption by 36% on average towards TSP-DP, and the wind effect may imply to degrade the performance of routing problems for energy minimization. On average, the energy consumption tends to increase in the cases where the wind blows hard. Compared with TSPW-DP-50 and EMW-DP-50, the results are different by 25.9% on average.

4.3.3 Flight Distance Comparison

This section compares the total flight distance of the TSP and the EMVRP under windy conditions.

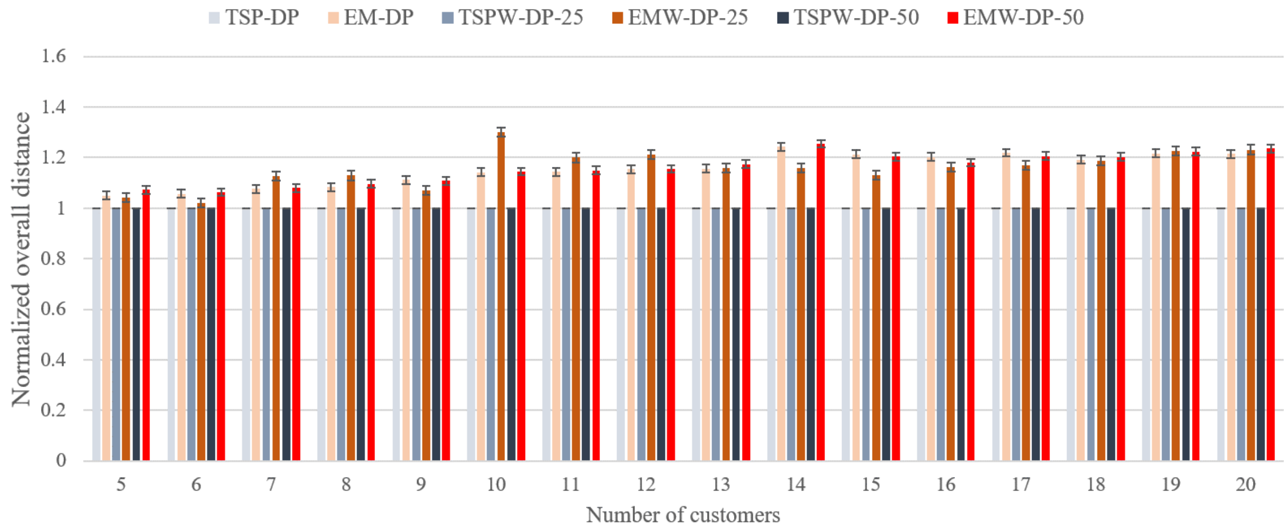


Fig. 7 Total flight distance of the routes.

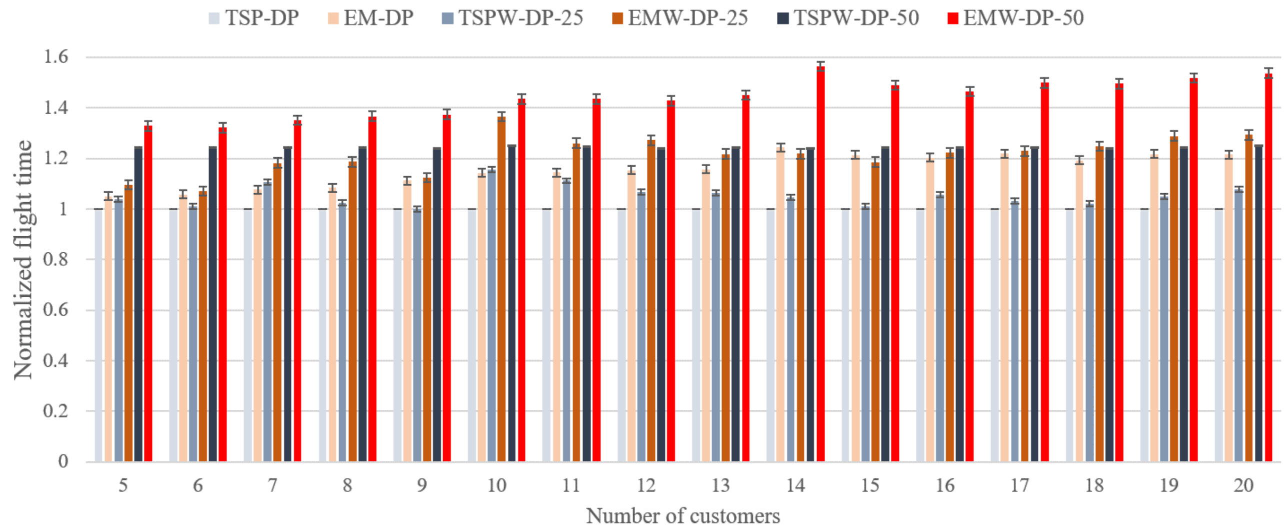


Fig. 8 Total flight time of the routes.

Figure 7 shows the results of the total flight distance. Since the objective of the TSP is to minimize the total distance, the obtained distance by TSP-DP, TSPW-DP-25, and TSPW-DP-50 should be the shortest. Therefore, when normalizing the results, the results of TSP are shown as 1, and the results of TSP are superior to EM for all the cases. On average, the results of EM-DP are longer by 15.5% than TSP-DP. The result of EMW-DP-25 for 10 customers is achieved by 15.8% of the long distance compared with TSPW-DP-25. Compared TSPW-DP-50 when the wind speed is 0.5, EMW-DP-50 obtains 16.0% longer routes. Despite the longer distance in these results, EM-based problems are totally superior to TSP in terms of energy consumption. Therefore, the energy consumption is significantly attributed not to the flight distance but to the flight time and the delivery order.

4.3.4 Flight Time under Windy Conditions

This section shows the results of the flight time for TSP and EMVRP under windy conditions.

Figure 8 shows the flight time for each number of customers from 5 to 20. As shown the results, the total flight time become long when the wind strongly blows. In particular, the results of EMW-DP-50 are obviously longer than TSPW-DP-50 by 19.6% on average. EMW-DP-25 also obtains the longer flight time by up to 17.2% for 6 customers, compared with TSPW-DP-50. TSP aims to simply minimize the total distance so that the route does not change if there is headwind against the direction drone heading. On the other hand, EM find a energy-minimum route, and both delivery order of packages and the total flight time should be in mind. However, regarding the energy consumption in Fig. 6, the results imply that long flight time does not affect the energy consumption.

Therefore, the delivery order and the weight of loads have significantly impact on the energy consumption.

5. Conclusions

In this paper, we have presented a dynamic programming algorithm for an energy-minimizing routing problems for delivery drones, and extended the problems to take into account windy conditions. The wind is assumed to be steady, and does not change during delivery operation.

Our experimental results show that the dynamic programming based algorithm for the EMVRP can efficiently find optimal routes with a significant reduction of the energy consumption.

In terms of the results of total flight distance, the experimental results imply that the wind may have strong impact on the length of flight routes.

The routing problem addressed in this paper is static that a set of items to deliver and the windy condition are fixed in advance. However, in the real world, the windy condition generally changes even during delivery operation. In future, we plan to further extend to dynamic problems where delivery orders arrive at the depot next from next over time under dynamically changing windy conditions.

Acknowledgments This work is in part supported by KAKENHI 20H04160 and 20J21208.

References

[1] Amazon.com Inc.: *Amazon Prime Air*, available from ⟨www.amazon.com/primeair⟩ (accessed 2016-04-02).

[2] The Irish News: *Google's Project Wing is testing food delivery drones in Australia*, available from ⟨www.irishnews.com/magazine/technology/2017/11/06/news/google-s-project-wing-is-testing-food-delivery-drones-in-australia-1180920/⟩ (accessed 2017-11-06).

[3] Dantzig, G.B. and Ramser, J.H.: The Truck Dispatching Problem, *Management Science*, Vol.6, No.1, pp.80–91 (1959).

[4] Mavrovouniotis, M. and Ynag, S.: Ant algorithms with immigrants schemes for the dynamic vehicle routing problem, *Inf. Sci.*, Vol.294, pp.456–477 (2015).

[5] Erdogan, S. and Miller-Hooks, E.: A Green Vehicle Routing Problem, *Transportation Research*, Vol.48, No.1, pp.100–114 (2012).

[6] Kara, I., Kara, B.Y. and Yetis, M.K.: Energy Minimizing Vehicle Routing Problem, *Proc. International Conference on Combinatorial Optimization and Applications*, pp.62–71, Springer (2007).

[7] Dorling, K., Heinrichs, J., Messier, G.G. and Magierowski, S.: Vehicle Routing Problems for Drone Delivery, *IEEE Trans. Systems, Man, and Cybernetics: Systems*, Vol.47, pp.70–85 (2017).

[8] Funabshi, Y., Shibata, A., Negoro, S., Taniguchi, I. and Tomiyama, H.: A Dynamic Programming Algorithm for Energy-aware Routing of Delivery Drones, *IPSJ Trans. System LSI Design Methodology*, Vol.13, pp.65–68 (2020).

[9] Bellman, R.: Dynamic Programming Treatment of the Travelling Salesman Problem, *Journal of the ACM*, Vol.9, No.1, pp.61–63 (1962).

[10] Held, M. and Karp, R.M.: A Dynamic Programming Approach to Sequencing Problems, *Journal of the Society for Industrial and Applied Mathematics*, Vol.10, pp.196–210 (1962).

[11] Caceres-cruz, J., Arias, P., Guimarans, D., Riera, D. and Juan, A.A.: Rich Vehicle Routing Problem: Survey, *Journal of ACM Computing Surveys*, Vol.47, No.2, article32 (2014).

[12] Cordeau, J.-F., Laporte, G., Savelsbergh, M.W.P. and Vigo, D.: Chapter 6 Vehicle Routing, *Proc. Handbooks in Operations Research and Management Science*, Vol.14, pp.367–428 (2007).

[13] Eksioglu, B., Vural, A.V. and Reisman, A.: The Vehicle Routing Problem: A Taxonomic Review, *Computers & Industrial Engineering*, Vol.57, No.4, pp.1472–1483 (2009).

[14] Wang, S. and Liu, X.: Energy Minimization Vehicle Routing Problem with Heterogeneous Vehicles, *International Conference on Service Systems and Service Management* (2016).

[15] Wang, S. and Wu, Y.: A Genetic Algorithm for Energy Minimization Vehicle Routing Problem, *International Conference on Service Systems and Service Management* (2017).

[16] Chase C.M. and Amanda, G.C.: The Flying Sidekick Traveling Salesman Problem: Optimization of Drone-assisted Parcel Delivery, *Transportation Research*, Vol.54, pp.86–109 (2015).

[17] Wang, X., Poikonen, S. and Golden, B.: The vehicle routing problem with drones: Several worst-case results, *Optimization Letters*, Vol.11, pp.679–697 (2017).

[18] Chang, Y.S. and Lee, H.J.: Optimal delivery routing with wider drone-delivery areas along a shorter truck-route, *Expert Systems with Applications*, Vol.104, pp.307–317 (2018).

[19] Jeong, H.Y., Song, B.D. and Lee, S.: Truck-drone hybrid delivery routing: Payload-energy dependency and no-fly zones, *International Journal of Production Economics*, Vol.214, pp.220–233 (2019).

[20] Doshi, A.A., Postula, A.J., Fletcher, A. and Singh, S.P.N.: Development of micro-uav with integrated motion planning for open-cut mining surveillance, *Microprocessors & Microsystems*, Vol.39, pp.829–835 (2015).

[21] Fu, Y., Ding, M., Zhou, C. and Hu, H.: Route planning for unmanned aerial vehicle (uav) on the sea using hybrid differential evolution and quantum-behaved particle swarm optimization, *IEEE Trans. Systems Man & Cybernetics Systems*, Vol.43, pp.1451–1465 (2013).

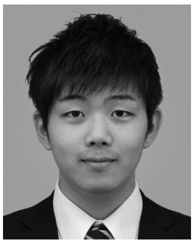
[22] Otte, M., Silva, W. and Frew, E.: Any-time path-planning: Time-varying wind field plus moving bbstacles, *IEEE International Conference on Robotics and Automation*, pp.2575–2582 (2016).

[23] Takegami, K., Negoro, S., Funabashi, Y., Taniguchi, I. and Tomiyama, H.: Low-Energy Routing for Deadline-Constrained Delivery Drones under Windy Conditions, *Proc. Bulletin of Networking, Computing, Systems, and Software* (2019).

[24] Kundu, A. and Matis, T.I.: A Delivery Time Reduction Heuristic Using Drones Under Windy Conditions, *Proc. IIE Annual Conference, Institute of Industrial and Systems Engineers* (2017).

[25] Negoro, S., Sukezane, D., Shibata, A., Maekawa, K., Taniguchi, I. and Tomiyama, H.: Measurement and Modeling of Quadcopter Energy with ROS, *Workshop on Synthesis and System Integration of Mixed Information Technologies* (2018).

Satoshi Ito received his B.E. degree in electronic and computer engineering from Ritsumeikan University in 2020. He is in the Master's degree program at Ritsumeikan University. His research interests include cyber-physical systems.

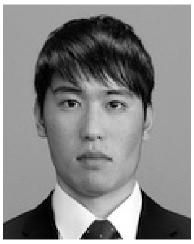


Hiroki Nishikawa received his B.E. and M.E. degrees from Ritsumeikan University in 2018 and 2020, respectively. At present, he is in Ph.D. program in Ritsumeikan University. His research interests include system level design methodologies, design methodologies for cyber-physical systems, etc. He is a member of IEEE.

Xiangbo Kong received B.E. degree from Nankai University in 2012 and he received M.E. and Ph.D. degrees from Ritsumeikan University in 2018 and 2020, respectively. In 2020, he joined the College of Science and Engineering, Ritsumeikan University as an assistant professor. His research interests include artificial intelligence, image processing, embedded system, etc. He is a member of IEEE and IPSJ.

Yusuke Funabashi received his B.E. and M.E. degrees from Ritsumeikan University in 2018 and 2020, respectively. At present, he works for Konami Digital Entertainment. His research interests include mathematical optimization for cyber-physical systems.

Atsuya Shibata received his B.E. degree from Ritsumeikan University in 2017 and his M.E. degree from Nara Institute of Science and Technology in 2019. At present, he works for Fixstars Corporation. His research interests include algorithm design and software programming.



Shunsuke Negoro received his B.E. and M.E. degrees from Ritsumeikan University in 2017 and 2020, respectively. At present, he works for ISHIDA. His research interests include cyber-physical systems.

Ittetsu Taniguchi received B.E., M.E., and Ph.D. degrees from Osaka University in 2004, 2006, and 2009, respectively. From 2007 to 2008, he was an international scholar at Katholieke Universiteit Leuven (IMEC), Belgium. In 2009, he joined the College of Science and Engineering, Ritsumeikan University as an assistant professor, and became a lecturer in 2014. In 2017, he joined the Graduate School of Information Science and Technology, Osaka University as an associate professor. His research interests include system level design methodology, design methodologies for cyber-physical systems, etc. He is a member of IEEE, ACM, IEICE, and IPSJ.

Hiroyuki Tomiyama received his B.E., M.E. and D.E. degrees in computer science from Kyushu University in 1994, 1996 and 1999, respectively. He worked as a visiting researcher at UC Irvine, as a researcher at ISIT/Kyushu, and as an associate professor at Nagoya University. Since 2010, he has been a full professor with College of Science and Engineering, Ritsumeikan University. He has served on program and organizing committees for a number of premier conferences including DAC, ICCAD, DATE, ASP-DAC, CODES+ISSS, CASES, ISLPED, RTCSA, FPL and MPSoC. He has also served as editor-in-chief for IPSJ TSLDM, as an associate editor for ACM TODAES, IEEE ESL and Springer DAEM, and as chair for IEEE CS Kansai Chapter and IEEE CEDA Japan Chapter. His research interests include, but not limited to, design methodologies for embedded and cyber-physical systems.

(Recommended by Associate Editor: *Kazuyoshi Takagi*)