

Placement and Routing Optimization for Automated Inspection with UAVs: A Study in Offshore Wind Farm

Hwei-Ming Chung, *Student Member, IEEE*, Sabita Maharjan, *Senior Member, IEEE*, Yan Zhang, *Fellow, IEEE*, Frank Eliassen, *Member, IEEE*, and Kai Strunz

Abstract—Wind power is a clean and widely deployed alternative to reducing our dependence on fossil fuel power generation. Under this trend, more turbines will be installed in wind farms. However, the inspection of the turbines in an offshore wind farm is a challenging task because of the harsh environment (e.g., rough sea, strong wind, and so on) that leads to high risk for workers who need to work at considerable height. Also, inspecting increasing number of turbines requires long man hours. In this regard, unmanned aerial vehicles (UAVs) can play an important role for automated inspection of the turbines for the operator, thus reducing the inspection time, man hours, and correspondingly the risk for the workers. In this case, the optimal number of UAVs enough to inspect all turbines in the wind farm is a crucial parameter. In addition, finding the optimal path for the UAVs' routes for inspection is also important and is equally challenging. In this paper, we formulate a placement optimization problem to minimize the number of UAVs in the wind farm and a routing optimization problem to minimize the inspection time. Wind has an impact on the flying range and the flying speed of UAVs, which is taken into account for both problems. The formulated problems are NP-hard. We therefore design heuristic algorithms to find solutions to both problems, and then analyze the complexity of the proposed algorithms. The data of the Walney wind farm are then utilized to evaluate the performance of the proposed algorithms. Simulation results clearly show that the proposed methods can obtain the optimal routing path for UAVs during the inspection.

Index Terms—unmanned aerial vehicle (UAV), offshore wind farm, inspection, placement and routing problem, heuristic algorithm.

NOMENCLATURE

A. Sets and Indices

i, j	UAV index.
k, l	Turbine index.
T	Total number of turbines in the wind farm.
\mathcal{N}_i	Set of turbines for UAV i to inspect.

This work was supported by Norwegian Research Council under Grants 275106 (LUCS project), 287412 (PACE project), and 267967 (SmartNEM project). This work was also supported by German Federal Ministry for Economic Affairs and Energy (BMWi) project WindNODE: Echtzeitlabor Energiewende under Grant 03SIN539.

H.-M. Chung and F. Eliassen are with the Department of Informatics, University of Oslo, Oslo 0373, Norway, (e-mail: hweiminc@ifi.uio.no, frank@ifi.uio.no).

S. Maharjan and Y. Zhang are with Department of Informatics, University of Oslo, Oslo 0373, Norway; and Simula Metropolitan Center for Digital Engineering, Oslo 0167, Norway. (e-mail: yanzhang@ieec.org, sabita@ifi.uio.no).

K. Strunz is with the Department of Energy and Automation Technology, Technische Universität Berlin, Berlin 10587, Germany (e-mail: kai.strunz@tu-berlin.de).

N	Number of candidate UAVs in wind farm.
\mathcal{W}	Set of wind data.

B. Variables

$\mathbf{v}_{i,k,l}$	UAV velocity for UAV i flying from turbine k to turbine l .
$\ \mathbf{v}_{i,k,l}\ _2$	Airspeed of UAV i flying from turbine k to turbine l .
$\mathbf{s}_{i,k,l}$	Resultant velocity for UAV i flying from turbine k to turbine l .
$\ \mathbf{s}_{i,k,l}\ _2$	Groundspeed of UAV i flying from turbine k to turbine l .
$t_{i,k,l}$	Time for UAV i flying from turbine k to turbine l .
$\theta_{i,k,l}^{s,w}$	Angle between $\mathbf{s}_{i,k,l}$ and \mathbf{w} .
$\theta_{i,k,l}^{s,v}$	Angle between $\mathbf{s}_{i,k,l}$ and $\mathbf{v}_{i,k,l}$.
\mathbf{A}	Binary vector indicating the UAV state.
\mathbf{B}	Binary matrix indicating the link between UAVs and turbines.
\mathbf{C}	Binary matrix indicating the communication link between UAVs.
\mathbf{U}_i^m	Binary matrix to indicate the m -th route of UAV i .

C. Parameters

$\mathbf{q}_k(\mathbf{q}_i)$	Coordinates of turbine k (UAV i).
\mathbf{w}	Wind velocity.
θ_w^{met}	Wind direction obtained from wind data.
θ_w^{pol}	Wind direction represented in polar coordinates.
w_s	Wind speed obtained from $\ \mathbf{w}\ _2$.
u_i^{max}	Maximum flying speed of UAV i .
t_i^{max}	Maximum flight time of UAV i .
u_i^{wind}	Maximum resistance to wind of UAV i .
$d_{i,j}(d_{i,k})$	Distance between UAV i and UAV j (turbine k).
ρ_i	Flying distance of UAV i .
$B_i^w(\rho_i)$	Flying range of UAV i under given \mathbf{w} and ρ_i .
Z_i	Intersection of flying range under different \mathbf{w} .
p	Maximum number of turbines for UAVs to inspect.
d	Maximum distance for obtaining a communication link between two UAVs.
M	Number of routes to perform turbine inspection.
\mathbf{D}_i	Adjacency matrix for a graph constructed by UAV i and \mathcal{N}_i .
ϵ_v	Threshold of hourly average wind speed so that hourly wind gust does not exceed u_i^{wind} .
$path$	Optimal routing path without considering t_i^{max} .

D. Operators

- $|\cdot|$ Cardinality of set.
- $\|\cdot\|_2$ Two norm of a vector.
- $\mathbf{0}$ Zero vector.
- $\mathbf{C}_{\{f,g\}}$ Submatrix that is taken from \mathbf{C} with the row index in set f and the column index in set g .

Other notations are defined in the text.

I. INTRODUCTION

WITH increasing influence of wind power in the energy ecosystem, the capacity of the global wind power is expected to grow by 60% over the next 5 years [1]. The overall capacity of all wind turbines installed in the first half of 2019 in Europe has reached 4.9 GW [2]. This is the same amount as the wind power generation capacity installed for the whole year of 2018. However, with a large number of wind turbines, inspecting and maintaining the condition of the turbines becomes a challenging task. Turbines may suffer failures from different components, such as blades, gearbox, yaw system, and so on [3]. The authors in [4] reported that a blade failure could result in a downtime of more than seven days.

Advanced sensors have been introduced to monitor the operation and the health condition of wind turbines [5]–[8]. For instance, in [5], fiber optic sensors were used to monitor the operation of wind turbines. Lidar sensors were used to detect cracks on the blade in [6]. Other technologies used for monitoring and analyzing the conditions of wind turbines are thermal wave radar [7] and millimeter wave imaging [8], respectively. Other researchers have developed algorithms for detecting the failure of turbines. For instance, in [9], the authors utilized a deep neural network (DNN) to detect the failure of the gearbox. The convolutional neural network (CNN) was introduced to detect the icing on the blade in [10].

From [5]–[10], we can observe that the detection mechanisms are rather advanced. However, the main challenge is to optimally place the sensors [5]–[8] or to acquire data [9], [10]. For the onshore wind farm, the sensors can be installed near the turbines and then transmit the measurements back to the control center through an aggregator. However, this method is not suitable for an offshore wind farm because the sensors cannot be easily installed on the sea. Currently, it is necessary to dispatch qualified service personnel to manually inspect turbines in offshore wind farms. This procedure may take several days to several weeks, requiring intensive and costly efforts. Moreover, workers may be subjected to the risks associated with climbing wind turbines and working at height.

Automated inspection of turbines is a solution that can address these issues. In this case, the unmanned aerial vehicle (UAV) can play a crucial role. UAVs have been widely applied for the automated inspection for energy systems [11]–[17]. If there is dust on the solar panel, the power generation of the solar panels can be influenced. The authors in [11] proposed a framework for utilizing UAVs to monitor the condition of solar panels. Regarding wind turbine blades, cracks on the surface

can be detected with the help of images taken by the UAVs [12], [14]. Automatic meter reading is another application of UAVs in power systems, that was studied in [13]. The damage of a power line can also be detected by UAVs [15] so that workers do not have to climb the transmission line tower. Moreover, defects of power insulators can be detected by images captured by UAVs and CNN [16]. In [17], the authors combined UAVs with fault indicators. If there is a damage on the distribution line, fault indicators can send a signal to the UAV, and then the UAV can help to transmit the signal back to the operator. This is similar in wireless communication where the UAV can be regarded as a relay to transmit the signal [18]–[20].

Most of the studies mentioned above focused on how to route UAVs to collect the data. That is, there are several targets for UAVs to collect data from, and therefore the UAV has to find the optimal path to route. For example, the authors in [18] jointly minimized usage of time and bandwidth to find an optimal path for routing. Then, maximizing the average secrecy rate to secure the communication link between users and UAVs was considered in [19]. The authors in [20] further minimized energy consumption while ensuring that the throughput requirements of the users are met. However, the placement of UAVs can significantly influence the routing results. Therefore, the placement problem such as finding the optimal number of UAVs and deriving the optimal topology for UAV placement should be considered as discussed in [21]–[23]. The authors in [21] proposed two algorithms to minimize the time for UAVs flying to the specific locations to serve mobile users. Specifically, the first algorithm minimized the flight time, and the second algorithm incorporates fairness of allocating transmitting power to users while minimizing the flight time. In [22], K -means clustering was applied to find the optimal locations and the number of UAVs such that the desired area can be covered with a minimum number of UAVs. In [23], the authors applied an algorithm to predict the future throughput requirements of the users. Then, the optimal number and the topology of UAVs are determined based on the prediction.

In this paper, UAVs are adopted to perform automated monitoring of wind farms. Specifically, UAVs equipped with sensors, such as Lidar, millimeter wave, or thermographic sensors, can monitor the surface condition of turbines. Then, we address the placement problem of UAVs in an offshore wind farm and find the optimal routing path for turbine inspection by utilizing UAVs. Some related research works have focused on visiting targets utilizing multiple UAVs [18]–[20] and placing UAVs for the optimal topology [21]–[23]. However, the placement and routing problems of UAVs in offshore wind farm have not been studied. Especially, the random realization of an extreme weather event in the offshore wind farm was not considered in those studies. That is, compared to the onshore situation, wind in offshore wind farms is known to be stronger on average. The influence of wind speed and wind direction in offshore wind farms is considerable and is important to consider for the UAV placement and routing problems. The UAVs may crash if the wind speed is over the maximum wind speed resistance of the UAVs during turbine

inspection. Also, wind speed and wind direction impact the flying speed and flying range of UAVs. We then introduce a mathematical model that addresses the relation between wind speed, wind direction, and UAVs. This model is essential to consider when formulating the placement and the routing problems for offshore wind farms. The formulated problems are NP-hard, and therefore we design heuristic algorithms to solve both problems that also take the mathematical model into account.

Overall, the main contributions of this paper are threefold:

- We present a novel framework for inspecting the wind turbines in the offshore wind farm by utilizing UAVs. Then, two optimization problems are formulated for the placement and routing problems, incorporating wind as it can considerably influence the flying range and the flying speed of UAVs.
- The formulated problems are NP-hard such that they cannot be solved directly. Therefore, we design heuristic algorithms to obtain the required number and the topology of UAVs in an offshore wind farm and the optimal path for the inspection.
- We also analyze the complexity of the proposed algorithms. With the proposed methods, the placement problem can be solved in polynomial time, and the routing problem can be solved with lower complexity compared to the brute-force method.

II. SYSTEM MODEL

A. Wind Farm and Wind Model

The total number of turbines in the offshore wind farm is T . The coordinates of the k -th turbine are $\mathbf{q}_k = [x_k, y_k]$. The wind velocity is denoted by $\mathbf{w} = [w^x, w^y]$. The projection of the wind velocity on the x-axis and the y-axis are w^x and w^y , respectively. Quantity w_s represents the wind speed, which can be calculated as $w_s = \|\mathbf{w}\|_2$. The wind direction in the polar coordinate system is denoted by θ_w^{pol} , which can be calculated from $\theta_w^{pol} = \arctan \frac{w^y}{w^x}$. The definition of θ_w^{pol} is different from the wind direction of the meteorological measurements. Therefore, the wind direction in the meteorological measurements is denoted by θ_w^{met} . In the meteorological measurements, $0, \frac{\pi}{2}, \pi,$ and $\frac{3\pi}{2}$ are used to represent the north, the east, the south, and the west wind, respectively. Thus, the phase is represented in a clockwise direction. In the polar coordinate system however, the phase is represented in a counterclockwise direction. Therefore, θ_w^{pol} and θ_w^{met} are related as

$$\theta_w^{pol} = \frac{3\pi}{2} - \theta_w^{met}. \quad (1)$$

B. UAV Model

In an offshore wind farm, we place UAVs to monitor the condition of the turbines. Each UAV should inspect the wind turbines assigned to it. UAV i will be placed at $\mathbf{q}_i = [x_i, y_i]$, and the set of the turbines assigned to UAV i is denoted by \mathcal{N}_i . Then, $|\mathcal{N}_i|$ represents the cardinality of set \mathcal{N}_i .

When UAV i flies to inspect a wind turbine, it may face two wind conditions, namely tail wind and head wind, as shown

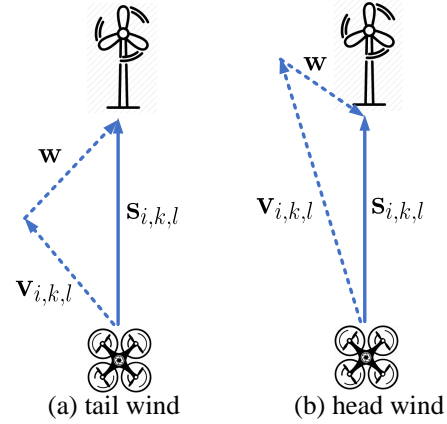


Fig. 1. The relation between UAV and wind.

in Fig. 1. The condition of the wind is to be considered in the decision-making of the UAV. We define $\mathbf{s}_{i,k,l} = [s_{i,k,l}^x, s_{i,k,l}^y]$ and $\mathbf{v}_{i,k,l} = [v_{i,k,l}^x, v_{i,k,l}^y]$ as the resultant velocity and the UAV velocity of UAV i flying from turbine k to turbine l , respectively. Components $s_{i,k,l}^x$ and $v_{i,k,l}^x$ are the projections on the x-axis, and $s_{i,k,l}^y$ and $v_{i,k,l}^y$ are the projections on the y-axis. The UAV velocity is the initial velocity of the UAV, and the resultant velocity is the velocity influenced by the wind. The relation between the UAV velocity, the wind, and the resultant velocity is expressed as

$$\mathbf{v}_{i,k,l} + \mathbf{w} = \mathbf{s}_{i,k,l}. \quad (2)$$

Quantities $\|\mathbf{v}_{i,k,l}\|_2$ and $\|\mathbf{s}_{i,k,l}\|_2$ are regarded as airspeed and groundspeed, respectively. UAV i has the maximum speed limit of u_i^{max} . Usually, u_i^{max} is referred to as the maximum value for the airspeed. However, UAVs may not remain stable, and the structural capacity of UAVs may degrade if UAVs fly at too high groundspeed. Therefore, airspeed and groundspeed are both limited to u_i^{max} in this paper. Specifically, for the tail wind situation in Fig. 1(a), the groundspeed is limited to this value. Then, when the UAV faces a head wind, the airspeed is limited to u_i^{max} . The angle between $\mathbf{s}_{i,k,l}$ and $\mathbf{v}_{i,k,l}$ is denoted by $\theta_{i,k,l}^{s,v}$. Also, $\theta_{i,k,l}^{s,w}$ is used to represent the angle between \mathbf{w} and $\mathbf{s}_{i,k,l}$. For UAV i , the maximum wind speed resistance is denoted by u_i^{wind} .

The time for UAV i to travel from turbine k to turbine l can be calculated as

$$t_{i,k,l} = \frac{\|\mathbf{q}_l - \mathbf{q}_k\|_2}{\|\mathbf{s}_{i,k,l}\|_2}. \quad (3)$$

There exists also a maximum flight time for UAV i denoted by t_i^{max} , which represents an upper limit of the total flight time during the inspection. The distance between UAV i and UAV j is denoted by $d_{i,j}$, and the distance between UAV i and turbine k is represented as $d_{i,k}$.

III. PROBLEM FORMULATION

In this section, we formulate a placement optimization problem to obtain the required number of UAVs in the wind farm and the corresponding topology. Then, a routing

optimization problem is formulated to find the optimal path for the inspection. Wind is a very important factor in determining the flying range and flying speed of UAVs. Therefore, the influence of the wind is incorporated in both problems.

A. Placement Problem Formulation

We introduce an important parameter of UAV i , namely the flying range. Here, the UAVs are assumed to fly at the same altitude, and therefore the z-axis can be ignored. The flying range of UAV i under the wind condition w can be expressed as

$$B_i^w(\rho_i) = \{x, y \in \mathbb{R} : \|\mathbf{r}\|_2 \leq \rho_i\}, \quad (4)$$

where

$$\mathbf{r} = [x - x_r, y - y_r]. \quad (5)$$

Quantity ρ_i is the actual flying distance of UAV i , which can be calculated by

$$\rho_i = \frac{u_i^{max} t_i^{max}}{2}, \quad (6)$$

assuming that the UAV can fly from and then back to the starting point at maximum flying speed, u_i^{max} , during maximum flight time, t_i^{max} . However, t_i^{max} is undetermined to the UAV because it is influenced by u_i^{max} . To obtain the value of ρ_i , an analytical model is introduced in Appendix A to determine u_i^{max} and t_i^{max} . The flying range of the UAV is regarded as a circle, with x_r and y_r as the center of the circle, which is calculated as

$$\begin{cases} x_r = x_i + w^x t_i^{max}, \\ y_r = y_i + w^y t_i^{max}. \end{cases} \quad (7)$$

The flying range of UAV i is influenced by different wind conditions. Thus, the flying range of the UAV is the intersection of the flying range under different wind conditions, i.e.,

$$Z_i = \bigcap_w B_i^w(\rho_i). \quad (8)$$

Initially, a total of N candidate UAVs are placed in the wind farm. We then introduce matrices $\mathbf{A} = [a_i]_{1 \times N}$, $\mathbf{B} = [b_{i,k}]_{N \times T}$, and $\mathbf{C} = [c_{i,j}]_{N \times N}$ to denote the states of candidate UAVs, the association between UAVs and turbines, and the communication link between UAVs, respectively. Here, a_i is set to 1 when this candidate UAV should be placed in the wind farm, and the candidate UAV should be removed if a_i is 0. Also, a docking station [24]¹ is placed together with the UAV so that the UAV can charge or swap its battery. If turbine k is assigned to UAV i , $b_{i,k}$ is set to 1; otherwise, $b_{i,k}$ is 0. When determining the topology of the UAVs, we need to ensure that the UAVs can maintain communication links with each other. By doing so, the collisions between UAVs can be prevented [25], [26]. Quantity $c_{i,j}$ is set to 1 if UAV i and j obtain a communication link.

The objective function is to minimize the number of UAVs that should be placed in the wind farm. Therefore, the placement optimization problem takes the form:

$$\min_{\mathbf{A}, \mathbf{B}, \mathbf{C}, x_i, y_i} \sum_{i=1}^N a_i \quad (9a)$$

¹<https://www.airoboticsdrones.com/>

$$\text{subject to } a_i, b_{i,k}, c_{i,j} \in \{0, 1\}, \quad \forall i, k, j \quad (9b)$$

$$\sum_{i=1}^N b_{i,k} \leq 1, \quad \forall k \quad (9c)$$

$$\sum_{k=1}^T b_{i,k} \leq p, \quad \forall i \quad (9d)$$

$$[x_k, y_k] \in Z_i \quad \forall b_{i,k} = 1 \quad (9e)$$

$$[x_i, y_i] \in \{[x_k, y_k]\} \quad \forall b_{i,k} = 1 \quad (9f)$$

$$b_{i,k} \leq a_i, \quad \forall k \quad (9g)$$

$$\sum_{i=1}^N \sum_{k=1}^M b_{i,k} \geq T, \quad (9h)$$

$$\sum_{j=1, j \neq i}^N c_{i,j} \geq 1, \quad \forall a_i = 1 \quad (9i)$$

$$\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \leq d, \quad \forall i, j. \quad (9j)$$

Quantities a_i , $b_{i,k}$, and $c_{i,j}$ are defined as binary variables in (9b). Constraints (9c) and (9d) state that each turbine can only be assigned to one UAV, and each UAV can inspect up to p turbines, respectively. Here, p is defined by the wind farm operator. The location $[x_k, y_k]$ of turbine k assigned to UAV i must be in the flying range of UAV i as given by (9e). The UAVs cannot be placed on the sea, and therefore each UAV should be placed inside one of the turbines assigned to it as indicated in (9f). Also, constraint (9g) states that the turbine can only be assigned to a UAV that is actually placed in the wind farm. Constraint (9h) indicates that all T turbines must be assigned to the active UAVs. The minimum number of communication links one UAV must have is given by (9i). Finally, (9j) indicates that the distance between any two UAVs should be lower than d .

B. Routing Problem Formulation

For a given topology, we study how to route the UAVs to inspect the wind turbines. A UAV and the turbines assigned to it can be represented in a graph, defined as $\mathcal{G}_i = \{\mathcal{N}_i, \mathcal{E}_i\}$, where \mathcal{N}_i is the set of turbines assigned to UAV i , and will be the nodes in the graph, and then \mathcal{E}_i is the set of the edges which connects each turbine. With the graph structure, we can create an adjacency matrix for the graph denoted by \mathbf{D}_i . The value of the k -th column and the l -th row in \mathbf{D}_i is $t_{i,k,l}$ as the flight time from turbine k to l . The value of $t_{i,k,l}$ and the value of $t_{i,l,k}$ are not the same because of the wind, and therefore the adjacency matrix is not symmetric. Thus, \mathcal{G}_i is an asymmetric graph.

In the routing problem, M denotes the number of required routes to inspect the turbines. We introduce another matrix $\mathbf{U}_i^m = [U_{i,k,l}^m]_{|\mathcal{N}_i| \times |\mathcal{N}_i|}$ to denote the m -th route for UAV i . More specifically, $U_{i,k,l}^m$ is 1 when the UAV chooses to fly from turbine k to turbine l ; otherwise, $U_{i,k,l}^m$ is 0. The routing problem can then be formulated as

$$\min_{M, \mathbf{U}_i^m, \mathbf{v}_{i,k,l}, \mathbf{s}_{i,k,l}, \theta_{i,k,l}} \sum_{m=1}^M \sum_{k \in \mathcal{N}_i} \sum_{l \in \mathcal{N}_i \setminus \{k\}} t_{i,k,l} U_{i,k,l}^m \quad (10a)$$

$$\text{subject to } U_{i,k,l}^m \in \{0, 1\}, \quad \forall k, l \in \mathcal{N}_i \quad (10b)$$

$$\sum_{k \in \mathcal{N}_i} U_{i,s,k}^m = \sum_{k \in \mathcal{N}_i} U_{i,k,s}^m = 1, \quad \forall m \quad (10c)$$

$$\sum_{l \in \mathcal{N}_i \setminus \{l\}} U_{i,l,k}^m = \sum_{l \in \mathcal{N}_i \setminus \{k\}} U_{i,k,l}^m = 1, \forall m \quad (10d)$$

$$\sum_{k \in Q} \sum_{l \in Q} U_{i,k,l}^m \leq |Q| - 1, \forall Q \subsetneq \mathcal{N}_i, m, |Q| > 2 \quad (10e)$$

$$1 \leq M \leq |\mathcal{N}_i| - 1 \quad (10f)$$

$$\sum_{k,l \in \mathcal{N}_i} t_{i,k,l} U_{i,k,l}^m \leq t_i^{max}, \quad \forall m \quad (10g)$$

$$\|\mathbf{v}_{i,k,l}\|_2 \leq u_i^{max}, \quad \forall k, l \in \mathcal{N}_i, U_{i,k,l}^m = 1 \quad (10h)$$

$$\|\mathbf{s}_{i,k,l}\|_2 \leq u_i^{max}, \quad \forall k, l \in \mathcal{N}_i, U_{i,k,l}^m = 1 \quad (10i)$$

$$\mathbf{v}_{i,k,l} + \mathbf{w} = \mathbf{s}_{i,k,l}, \quad \forall k, l \in \mathcal{N}_i, U_{i,k,l}^m = 1 \quad (10j)$$

In (10), the objective is to minimize the flight time and the number of routes for inspecting the turbines. Here, $U_{i,k,l}^m$ is a binary parameter as shown in (10b) to represent the path of routing during the inspection. Eq. (10c) indicates that the starting point of every route should be s , which is the position of the UAV, \mathbf{q}_i . Then, there can only exist one route between turbines as stated in (10d). Constraint (10e) ensures that a closed path does not exist in the subset Q of \mathcal{N}_i . Constraint (10f) states that the number of routes must be less than the number of turbines in \mathcal{N}_i . The summation of the flight times in every route is not to exceed t_i^{max} according to (10g). Constraints (10h) and (10i) enforce that the airspeed and groundspeed are bounded by the maximum speed, respectively. The relationship between the wind, the UAV velocity, and the resultant velocity is given by (10j).

IV. ALGORITHM DESIGN

The formulations in (9) and (10) cannot be solved directly as they contain binary parameters. Both problems are mixed-integer linear programming (MILP) problems. In this case, if the dimension of the problems increases, the problems may become NP-hard. We therefore design heuristic algorithms to solve the problems.

A. Flying Range Determination

For both placement and routing optimization problems, an important parameter is the flying range. The flying range can be influenced by the wind conditions in the wind farm. However, wind conditions in the future are unknown to UAVs, and different UAVs have different maximum wind speed resistance. Since the algorithm is used to determine how many UAVs are required for inspections and where to place the UAVs, the algorithm can be considered as part of a planning stage before operation begins. As such, the algorithm makes use of historic wind data.

The wind data, denoted by \mathcal{W} , provide the hourly average wind speeds and directions for several days. Selected data also provide the hourly wind gust and the corresponding direction. The UAV cannot perform inspection if the wind speed exceeds maximum wind speed resistance, u_i^{wind} , in that

hour. Therefore, we introduce an auxiliary parameter, ϵ_v . This parameter is a user input. Specifically, an appropriate value is assigned to ϵ_v such that, based on a stochastic analysis of the historic data, the hourly wind gust does not exceed u_i^{wind} if the hourly average wind speed is lower than ϵ_v . The assignment of ϵ_v is done prior to the start of Algorithm 1. For usage of ϵ_v in the loop starting in line 2, the angle between 0 and 2π is discretized into μ segments of equal size. For every segment, the wind velocity located in this segment is taken out to construct a subset \mathcal{W}_b . Then, we compare the maximum hourly average wind speed in \mathcal{W}_b with ϵ_v . The smallest value is retained to define the flying range according to (4). After executing the loop for all segments, the flying range for the UAV can be obtained from (8). The detailed steps are summarized in Algorithm 1.

Algorithm 1: Flying Range Determination

Input: wind data \mathcal{W} , ϵ_v , t_i^{max} , u_i^{max}
Output: Z_i

- 1 Calculate ρ_i based on t_i^{max} and u_i^{max} using (6)
- 2 **for** $b = 1$ **to** μ **do**
- 3 $\mathcal{W}_b = \left\{ \mathbf{w} \mid \mathbf{w} \in \mathcal{W}, \frac{2\pi \times (b-1)}{\mu} \leq \theta_w^{pol} \leq \frac{2\pi \times b}{\mu} \right\}$
- 4 $\hat{w}_s = \min\{\epsilon_v, \max_{\mathbf{w} \in \mathcal{W}_b} \|\mathbf{w}\|_2\}$
- 5 $\hat{\theta}_w^{pol} = \frac{2b\pi - \pi}{\mu}$
- 6 $\hat{\mathbf{w}} = \left[\hat{w}_s \cos(\hat{\theta}_w^{pol}), \hat{w}_s \sin(\hat{\theta}_w^{pol}) \right]$
- 7 Calculate the flying range with (4) and $\hat{\mathbf{w}}$
- 8 Perform the intersection as mentioned in (8)

B. Algorithms for Obtaining Topology of UAVs

With known flying range, we can design an algorithm to get the topology of UAVs in the wind farm. At the beginning of the search, T UAVs are placed in the wind farm; in this case, every turbine has a UAV assigned to it. If the distance between two UAVs is shorter than d , they establish a communication link. Then, the turbines inside the flying range of UAV i are assigned to UAV i . The detailed process of the initialization is provided in Algorithm 2.

Algorithm 2: Initialization

Input: The coordinates of T turbines, Z_i
Output: \mathbf{A} , \mathbf{B} , \mathbf{C} , x_i , y_i

- 1 Set $N = T$, set every $[x_i, y_i]$ to $[x_k, y_k]$
- 2 **for** $i = 1$ **to** N **and** $j = 1$ **to** N **do**
- 3 $a_i = 1$
- 4 Calculate the distance between UAV i and j , $d_{i,j}$
- 5 **if** $d_{i,j} \leq d$ **then**
- 6 $c_{i,j} = 1$
- 7 **else**
- 8 $c_{i,j} = 0$
- 9 **for** $i = 1$ **to** N **and** $k = 1$ **to** T **do**
- 10 **if** $[x_k, y_k] \in Z_i$ **then**
- 11 $b_{i,k} = 1$
- 12 **else**
- 13 $b_{i,k} = 0$

After initialization, constraint (9d) should be validated. That is, some UAVs may have to inspect more than p turbines.

Redundant connections between the UAV and the turbines assigned to it should be deleted. For that, the distance between the UAV and the turbines assigned to it is sorted in a decreasing order. Then, turbines in \mathcal{N}_i are reassigned if UAV i is assigned more than p turbines; otherwise, we go to next UAV. The deletion starts from the turbine which has the longest distance to UAV i . At the same time, we have to ensure that this turbine is inspected by another UAV. Then, the procedure of deleting the turbines is repeated until $|\mathcal{N}_i| \leq p$. The details are provided in Algorithm 3.

Algorithm 3: Restrict Inspection Limit of UAVs

Input: \mathbf{B}, p
Output: \mathbf{B}

```

1 for  $i = 1$  to  $N$  do
2   Sort turbines in  $\mathcal{N}_i$  based on their distance to UAV  $i$  in
   a decreasing order as  $e_1, e_2, \dots, e_{|\mathcal{N}_i|}$ 
3    $k = e_1$ 
4   while  $|\mathcal{N}_i| > p$  do
5     if  $\sum_{j, j \neq i} b_{j,k} \geq 1$  then
6       Set  $b_{i,k} = 0$  and remove  $k$  from  $\mathcal{N}_i$ 
7     else
8       Set  $k$  to next turbine
```

After executing Algorithm 3, the current solution satisfies constraints (9b) and (9d)-(9j). However, some turbines may be assigned to more than one UAV. Also, the current number of placed UAVs is not minimized and still equal to the number of turbines, i.e., $\sum_i a_i = T$. Therefore, we need to reduce the number of the placed UAVs and then fix the issue of one turbine being assigned to multiple UAVs given by (9c). The details of this process are summarized in Algorithm 4. In this algorithm, two auxiliary parameters are introduced, namely cur and $step$. Parameter cur is a set to represent the set of the UAVs with $a_i = 1$. Then, $step$ indicates the current iteration step, and it will be used in the simulation part. In line 2, the UAVs in cur are sorted based on the number of turbines intersecting with other UAVs in a decreasing order as $f_1, f_2, \dots, f_{|cur|}$. Then, i is set to f_1 in line 4. In lines 5 and 6, we need to find if any turbine within \mathcal{N}_i can also be served by other UAVs. If that is true, UAV i is removed as described in lines 7 to 9. Otherwise, the connection between the UAV and the turbine is deleted based on the distance in lines 10 to 16.

C. Algorithms for Finding Optimal Routing Path

With the topology of the UAVs, we can now introduce how to route the UAVs to inspect the wind turbines. In (10), the optimal values of several parameters should be found. Moreover, the variables at the upper bounds of summation and the binary parameters make the problem difficult to solve. To address the challenge, solving (10) is separated into three stages and each of them is solved individually.

In the first stage, the adjacency matrix, \mathbf{D}_i , should be constructed. Prior to calculating \mathbf{D}_i , we have to calculate $\mathbf{s}_{i,k,l}$, $\mathbf{v}_{i,k,l}$, and $t_{i,k,l}$ for all $k, l \in \mathcal{N}_i$. The calculation of $\mathbf{s}_{i,k,l}$ and $\mathbf{v}_{i,k,l}$ differs depending on whether the UAV is facing head

Algorithm 4: UAV Number Minimization

Input: $\mathbf{A}, \mathbf{B}, \mathbf{C}$
Output: $\mathbf{A}, \mathbf{B}, \mathbf{C}$

```

1 step = 0, cur = {1, 2, \dots, N}
2 Sort UAV  $i$  in cur based on  $\sum_{j \in cur \setminus \{i\}} |\mathcal{N}_i \cap \mathcal{N}_j|$  in a
   decreasing order as  $f_1, f_2, \dots, f_{|cur|}$ 
3 while  $|\mathcal{N}_i \cap \mathcal{N}_j| > 0$  do
4   diff = {}, delete = True, i =  $f_1$ 
5   for j = 1 to N and  $a_j = 1$  do
6     diff = diff  $\cup (\mathcal{N}_i \cap \mathcal{N}_j)$ 
7   if diff =  $\mathcal{N}_i$  and  $\mathbf{C}_{\{j \in cur \setminus \{i\}, cur \setminus \{i\}\}} \neq \mathbf{0}$  and delete
   then
8      $a_i = b_{i,k} = 0 \forall k \in \mathcal{N}_i$ , cur = cur  $\setminus \{i\}$ , step = step + 1
9      $c_{i,j} = c_{j,i} = 0 \forall j \in cur$ , delete = False
10  if delete then
11    for j =  $f_1$  to  $f_{|cur|}$  do
12      if  $d_{j,k} < d_{i,k}$  then
13         $b_{i,k} = 0$ 
14      else
15         $b_{j,k} = 0$ 
16    step = step + 1
17  Repeat 2
```

wind or tail wind. Quantity $\theta_{i,k,l}^{s,w}$ is utilized to determine the wind condition, and it can be obtained by calculating the inner product of $\mathbf{s}_{i,k,l}$ and \mathbf{w} . If $\theta_{i,k,l}^{s,w}$ is between 0 and $\frac{\pi}{2}$, the UAV is facing a tail wind; otherwise, the UAV i is facing a head wind. The resultant velocity of UAV i flying from turbine k to turbine l can be obtained from

$$\mathbf{s}_{i,k,l} = \begin{cases} [u_i^{max} \cos(\theta_s), u_i^{max} \sin(\theta_s)], & 0 \leq \theta_{i,k,l}^{s,w} \leq \frac{\pi}{2}, \\ [u_i^s \cos(\theta_s), u_i^s \sin(\theta_s)], & \frac{\pi}{2} < \theta_{i,k,l}^{s,w} \leq \pi, \end{cases} \quad (11)$$

where θ_s is given by $\arctan((y_l - y_k)/(x_l - x_k))$. In (11), u_i^s can be calculated using

$$u_i^s = u_i^{max} \cos(\theta_{i,k,l}^{s,v}) - w_s \cos(\pi - \theta_{i,k,l}^{s,w}), \quad (12)$$

where

$$\theta_{i,k,l}^{s,v} = \arcsin \frac{w_s \sin(\pi - \theta_{i,k,l}^{s,w})}{u_i^{max}}. \quad (13)$$

Then, the relation in (2) can be used to calculate the UAV velocity of UAV i , $\mathbf{v}_{i,k,l}$. With $\mathbf{s}_{i,k,l}$, \mathbf{q}_l , and \mathbf{q}_k , $t_{i,k,l}$ can be obtained by (3), and then used to construct \mathbf{D}_i .

In the next stage, we relax the constraints (10f)-(10j) and solve the following optimization problem

$$\min \sum_{m=1}^1 \sum_{k \in \mathcal{N}_i} \sum_{l \in \mathcal{N}_i \setminus \{k\}} t_{i,k,l} U_{i,k,l}^m \quad (14a)$$

$$\text{subject to } (10c) - (10e). \quad (14b)$$

In the above problem, the optimal routing paths for the UAVs are searched without considering the time limit. A heuristic algorithm is designed to solve (14), and it is presented as Algorithm 5. At the beginning of the algorithm, the number of routes is set to 1, and the starting point, s , is set to the location of UAV i , $[x_i, y_i]$. Then, we create three sets, namely $From$, To , and To_{next} . The possible set of the current locations of the UAV is defined as $From$. The sets To and To_{next} denote

the possible sets of remaining turbines for a UAV to inspect in the next step and next two steps, respectively. Sets $From$, To , and To_{next} are initialized by $\mathcal{N}_i \setminus \{s\}$, $\{s\}$, and an empty set, respectively. Another vector, $path$, is used to denote the optimal routing path for the UAV. Then, $g(k, To)$ denotes the minimal time to finish the inspection of remaining turbines in the set To when the UAV is at the k -th turbine. At Lines 2 to 9 in Algorithm 5, the elements in the set $From$ is moved to set To , and then we calculate the $g(k, To)$ with $k \in From$. If $To = \{s\}$, $To_{next} = \{\}$, and the UAV is at turbine k , $g(k, To)$ is obtained as

$$g(k, To) = t_{i,k,s}, \quad (15)$$

where the minimal time for finishing the inspection is to fly back to the starting point from turbines k .

$$g(k, To) = \min_{l \in To, l \neq s, To_{next} = To \setminus \{l\}} t_{i,k,l} + g(l, To_{next}). \quad (16)$$

With all $g(k, To)$, a path with minimal routing time, $path$, can be obtained as shown in lines 10 to 15 in Algorithm 5.

Algorithm 5: Search Optimal Routing Path

Input: D_i , the coordinate of UAV i $[x_i, y_i]$
Output: $path$

- 1 Set $M = 1$; $s = [x_i, y_i]$; $From = \mathcal{N}_i \setminus \{s\}$; $To = \{s\}$;
 $To_{next} = \{\}$; $path = [0]_{(|\mathcal{N}_i|+1) \times 1}$
- 2 **for** $num = 0$ **to** $|\mathcal{N}_i| - 2$ **do**
- 3 **for** $K = 1$ **to** $\binom{|\mathcal{N}_i|-1}{num}$ **do**
- 4 Move different num elements in the set $From$ to
 the set To
- 5 **for** $k \in From$ **do**
- 6 **if** $num = 0$ **then**
- 7 Calculate (15)
- 8 **else**
- 9 Calculate (16)
- 10 $From = \{s\}$; $To = \mathcal{N}_i \setminus \{s\}$; $path[1] = s$
- 11 **for** $num = 2$ **to** $|\mathcal{N}_i|$ **do**
- 12 $k = path[num - 1]$
- 13 $l = \underset{l \in To, To_{next} = To \setminus \{l\}}{\operatorname{argmin}} t_{i,k,l} + g(l, To_{next})$
- 14 $path[num] = l$, $From = From \cup \{l\}$, $To = To \setminus \{l\}$
- 15 $path[|\mathcal{N}_i| + 1] = s$

The results of Algorithm 5 may yield a total flight time that exceeds the time limit. In this case, the routing path is modified based on the maximum flight time. More specifically, the UAV should be able to fly back to the starting point to charge or swap its battery that is low or empty. Therefore, it is tested if the UAV is able to fly back to s when it decides to inspect the l -th turbine from the k -th turbine. In this case, two auxiliary parameters, t_{accu} and $t_{compare}$, are introduced. Quantity t_{accu} is used to denote the accumulative flight time of flying from s to turbine l via turbine k . Then, the time for the UAV to fly from turbine l back to s is denoted by $t_{compare}$. If t_{accu} and $t_{compare}$ are all below t_i^{max} , UAV i can fly from turbine k to turbine l , $U_{i,k,l}^m = 1$. Otherwise, UAV i needs to fly back to s from the k -th turbine and then add another round of inspection starting from turbine l . The detailed procedure is presented in Algorithm 6. Quantities $s_{i,k,l}$, $\theta_{i,k,l}^{s,v}$ and $v_{i,k,l}$ can be obtained

from (11), (13), and (2), respectively. The output of Algorithm 6 is our final routing path.

Algorithm 6: Maximum Flight Time Check

Input: $path$, t_i^{max}
Output: $U_{i,k,l}^m$, M

- 1 $m = 1$; $t_{accu} = 0$, $t_{compare} = 0$
- 2 **for** $t = 1$ **to** $|\mathcal{N}_i|$ **do**
- 3 $k = path[t]$, $l = path[t + 1]$
- 4 $t_{accu} = t_{accu} + t_{i,k,l}$
- 5 $t_{compare} = t_{accu} + t_{i,l,s}$
- 6 **if** $t_{accu} < t_i^{max}$ **and** $t_{compare} < t_i^{max}$ **then**
- 7 $U_{i,k,l}^m = 1$
- 8 **else**
- 9 $U_{i,k,s}^m = 1$
- 10 $m = m + 1$, $t_{accu} = t_{i,s,l}$, $U_{i,s,l}^m = 1$
- 11 $M = m$

D. Algorithm Complexity Analysis

In what follows, the complexity of the proposed algorithms is analyzed. The computational complexity for the worst-case scenario is provided in our analysis.

First, we analyze the complexity of the algorithms for finding the optimal topology of the UAVs. Then, a sorting algorithm with the complexity of $n \log(n)$ is utilized. The complexity of the initialization in Algorithm 2 is $\mathcal{O}(N^2) + \mathcal{O}(NT)$. Since the number of turbines is the same as the number of UAVs, $\mathcal{O}(N^2) + \mathcal{O}(NT)$ can be simplified to $\mathcal{O}(N^2)$. On Algorithm 3, we reduce the number of connections down to p . The complexity of Algorithm 3 is $\mathcal{O}(N(N \log(N) + (T - p)))$ considering the worst case is then every UAV is assigned to all turbines. Line 2 in Algorithm 4 has the complexity of $\mathcal{O}(N^2 p + N \log(N))$, where the complexity of the intersection and union is $\mathcal{O}(p)$. Each iteration has a complexity of $\mathcal{O}(2Np + Np + N^2 p + N \log(N))$. Thus, the complexity of Algorithm 4 is $\mathcal{O}(4N^2 p + N^3 p + N^2 \log(N) + N \log(N))$. The proof of the optimality of Algorithms 2-4 is provided in Appendix B.

Then, we check the complexity of solving the routing problem. The complexity of executing lines 2 to 9 in Algorithm 5 is $\mathcal{O}(|\mathcal{N}_i|^2 \times 2^{|\mathcal{N}_i|})$. The complexity of lines 11 to 14 in Algorithm 5 is $\mathcal{O}(|\mathcal{N}_i|)$. Thus, the complexity of Algorithm 5 is $\mathcal{O}(|\mathcal{N}_i|^2 \times 2^{|\mathcal{N}_i|} + |\mathcal{N}_i|)$. Algorithm 6 has the complexity $\mathcal{O}(|\mathcal{N}_i|)$, which depends on the number of turbines assigned to the UAV. The complexity of applying brute force to find the optimal path is $\mathcal{O}(|\mathcal{N}_i|!)$. The brute-force method is one of the typical methods used for solving the problem which is NP-hard. Therefore, the proposed method results in lower complexity compared to brute-force method. The optimality of Algorithms 5 and 6 will be proved by comparing with the brute-force method in Section V.

V. NUMERICAL RESULTS

In this section, the performance of the proposed method is evaluated based on a real-world dataset. For the wind farm, we choose the Walney offshore wind farm in the United Kingdom (UK). The wind farm has an area of 218 square

kilometer (km), a generation capacity of around 1 Gigawatt (GW), and 189 turbines. The data are obtained from Centre for Environmental Data Analysis (CEDA) [27] and Kingfisher Information Service - Offshore Renewable Cable Awareness (KIS-ORCA) [28]. Specifically, the data from [27] contain the meteorological measurements in the UK, and then the wind data at Walney Island are utilized in the simulation since wind data inside the wind farm cannot be obtained. The layout of the Walney wind farm is collected from [28], which contains the longitude and the latitude of each turbine in the wind farm. The layout of the wind farm obtained from [28] is referred to as Walney. Each turbine is assigned with a code. The longitude and the latitude of wind turbines are transformed to Cartesian coordinates by using the Mercator projection. However, a diagram showing the topology of the UAVs with 189 turbines could lack clarity. Therefore, we pick 47 out of 189 turbines to create another dataset denoted by Walney-1.

The UAV used in the simulation is AscTec Falcon 8². This UAV can carry diverse sensors, namely Lidar, ultrasonic sensor, and camera, to inspect turbines. The maximum speed limit, u_i^{max} , is set to 16 m/s. Then, in the specification of the UAV, it has the maximum flight time between 12 to 22 minutes; we set t_i^{max} to 20 minutes. The model introduced in Appendix A can be used to verify these settings. The maximum resistance to the wind speed of the UAV is $u_i^{wind} = 15$ m/s. The UAVs need to communicate in order to prevent collision. Therefore, the maximum distance of the communication between UAVs is set to 5 km. Every UAV can be assigned to inspect up to 5 turbines, i.e., $p = 5$. The range between 0 and 2π is discretized into $\mu = 36$ segments.

A. The Topology of the UAVs in the Wind Farm

To solve the placement problem, we need to determine the flying range and then use it as input. The histogram in Fig. 2 counts the occurrences of the quotient of hourly wind gust and hourly average wind speed. Values below 2 account for 93.14% of the data. Thus, there is a higher than 90% chance for the peak value not to exceed twice the average value. Given a maximum wind speed resistance of up to 15 m/s for this UAV, it is plausible to set the hourly average wind speed up to which the UAV is allowed to perform inspection to $\epsilon_v = 8$ m/s. Wind speeds above 15 m/s are rarely encountered. With ϵ_v , Algorithm 1 is applied to determine the flying range.

The flying range is then applied to determine the topology. Fig. 3 shows the iteration process and the final placement results by applying the proposed algorithms to the Walney-1 dataset. At the beginning of the iteration, 47 UAVs are placed in the wind farm as shown in Fig. 3a. In Fig. 3b and 3c, redundant UAVs are deleted based on the proposed algorithms. After 30 iterations, the algorithm stops and outputs the final placement results as shown in Fig. 3d. According to our results, only 17 UAVs are required to cover all wind turbines in the offshore wind farm. We also observe that all the turbines are assigned to a UAV, and every UAV serves no more than 5 turbines. The same setting are applied to solve

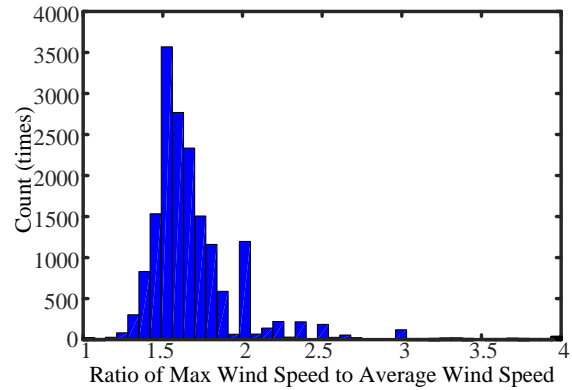


Fig. 2. The ratio of hourly wind gust to hourly average wind speed

TABLE I
THE PLACEMENT RESULTS OF CHANGING THE NUMBER A UAV CAN SERVE

Dataset	T	t_i^{max}	ϵ_v (m/s)	p	$\sum_i^N a_i$
Walney-1	47	20	6	5	16
				8	15
				10	15
	47	20	8	5	17
				8	16
				10	16
Walney	189	20	6	5	61
				8	45
				10	42
	189	20	8	5	63
				8	56
				10	54

the placement problem with the Walney dataset. The results reveal that we need to place 63 UAVs to cover all turbines in the wind farm.

In the proposed algorithms, the number of the placed UAVs can be further reduced if one UAV can serve more turbines by increasing the value of p . On the other hand, increasing the flying range by decreasing the value of ϵ_v can also reduce the number of the deployed UAVs. Therefore, we compare the influence of p and ϵ_v to the placement results in Table I. For Walney-1, there is no significant difference because this dataset includes only 47 turbines. However, for Walney itself, 63 UAVs are required according to the previous setting, $p = 5$ and $\epsilon_v = 8$. The number of placed UAVs can be cut down to 42 if we change p and ϵ_v to 10 and 6 m/s, respectively. However, with more turbines to be served per UAV, it will be more challenging to find an optimal routing solution. This issue will be discussed in the next section.

B. Routing Result

The results of the placement is applied to show how to route the UAVs to inspect the wind turbines. In this case, w_s is set to 8 m/s and θ_w^{met} is set to $\frac{\pi}{2}$ (east wind). We take UAV 15 as an example. The UAV is placed at the turbine whose code is B110. The result is shown in Fig. 4. To minimize inspection time, the UAV should avoid facing the head wind.

²<http://www.ascotec.de/en/uav-uas-drones-rpas-roav/ascotec-falcon-8/#pane-0-1>

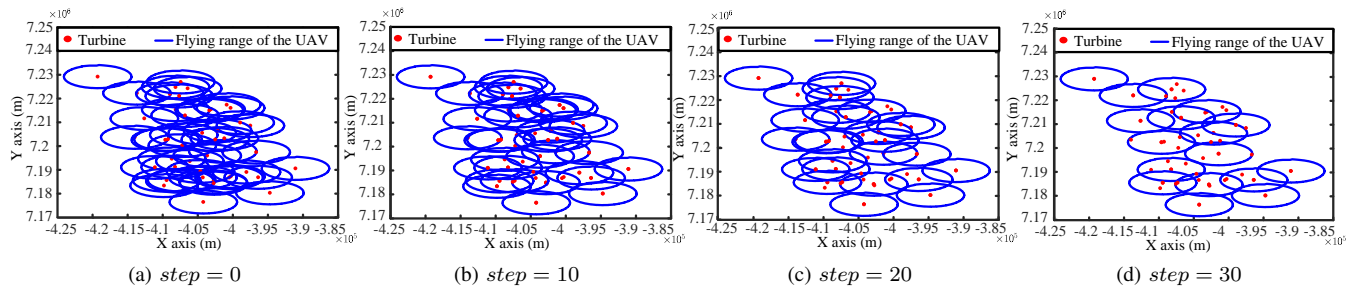


Fig. 3. UAV placement process.

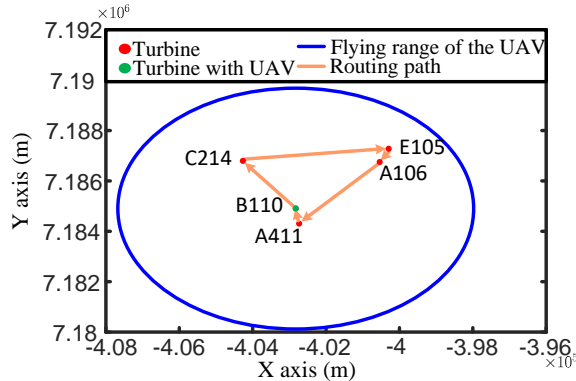


Fig. 4. The routing path of UAV 15

TABLE II
TOTAL FLIGHT TIME UNDER DIFFERENT SOLUTIONS

Method	Total flight time (mins)	path
Branch-and-bound	15.7343	B110>A106>E105>C214>A411>B110
Brute-force	15.3049	B110>C214>E105>A106>A411>B110
Algorithm 5 and 6	15.3049	B110>C214>E105>A106>A411>B110

Therefore, the UAV goes to C214 first and then chooses E105 afterwards. After E105, the UAV uses the tail wind to go to A106 and A411. The total flight time for the inspection is 15.30 minutes. The proposed method is compared with the brute-force method and the branch-and-bound method, which are common algorithms to solve MILP problem, in Table II. According to the results, Algorithms 5 and 6 obtain the same result as the brute-force method. However, this is not true for the branch-and-bound method. In this case, the optimality of Algorithms 5 and 6 can be proved.

Fig. 4 and Table III show that the UAV can finish the inspection in one round. This is because t_i^{max} is set to 20 minutes. Of interest is also the performance of Algorithm 6 when reducing t_i^{max} . For that, we decrease the maximum flight time to 15 and 12 minutes. The results are recorded in Table III. The UAV needs two rounds if t_i^{max} is set to 15 minutes. For the first round, it takes 14.44 minutes; it is 1.27 minutes

TABLE III
THE ROUTING RESULTS OF CHANGING THE MAXIMUM FLIGHT TIME

t_i^{max} (mins)	m	path
20	1	B110>C214>E105>A106>A411>B110
15	1	B110>C214>E105>A106>B110
	2	B110>A411>B110
12	1	B110>C214>B110
	2	B110>E105>A106>A411>B110

for the second round. The UAV still needs two rounds to finish the inspection if t_i^{max} is further reduced to 12 minutes. The first and the second rounds take 6.54 and 10.73 minutes, respectively. However, the turbines assigned to each route vary in each round compared to $t_i^{max} = 15$ minutes as shown in Table III. By applying the proposed algorithm, we can ensure that the UAV does not spend more than t_i^{max} on inspection.

As mentioned in Section V-A, increasing p may cause an issue for routing. This issue is discussed here, and p is set to 10. Two UAVs, with $i = 15$ and $i = 5$, are taken out to compare the required number of routes in Table IV. UAV 15 serves 5 and UAV 5 serves 9 turbines. We consider t_i^{max} to be 15 and 20 minutes. Then, w_s is set to 8 m/s and θ_w^{met} to 0 and π . For UAV 15, it only needs one round to finish the inspection under four different settings. However, UAV 5 needs 2 rounds if t_i^{max} is 20 minutes. The number of the required routes doubles to 4 if t_i^{max} is reduced to 15 minutes. Therefore, the UAV needs to spend more time and number of routes on routing if one UAV is to inspect more turbines. Also, in the wind data, 83.02% of the hourly average wind speed is lower than 8 m/s ($\epsilon_v = 8$). However, the value is 63.06% for the hourly average wind speed lower than 6 m/s ($\epsilon_v = 6$). Therefore, if we reduce ϵ_v , it means the UAVs have less chance to start the inspection. From these results, it is clear that there exists a fundamental tradeoff between maximum flight time, number of turbines one UAV can serve, and the flying range. In the case of Walney wind farm, $p = 5$ and $\epsilon_v = 8$ m/s is better than $p = 10$ and $\epsilon_v = 6$ m/s.

VI. CONCLUSION

In this paper, we presented a framework for utilizing UAVs to inspect the wind turbines in an offshore wind farm. An optimization problem was formulated to minimize the number of UAVs to be placed in the offshore wind farm with consideration of the challenging offshore wind condition faced by the

TABLE IV
THE ROUTING RESULTS OF CHANGING t_i^{max} AND p

i	$ \mathcal{N}_i $	p	ϵ_v	$t_i^{max} (min)$	$w_s (m/s)$	θ_w^{met}	M
15	5	5	8	15	8	0	1
					8	π	1
				20	8	0	1
					8	π	1
5	9	10	8	15	8	0	4
					8	π	4
				20	8	0	2
					8	π	2

UAVs. Another formulated optimization problem was to find an optimal route for the UAVs to inspect the wind turbines. We designed heuristic algorithms to solve both problems and analyzed the complexity of the proposed algorithms. For the purpose of validation, real-world data were utilized (meteorological measurements recorded by Centre for Environmental Data Analysis (CEDA), and positions of the turbines through Kingfisher Information Service - Offshore Renewable Cable Awareness (KIS-ORCA)). With the proposed methods, we can discover how many UAVs are needed to automatically inspect the turbines in an offshore wind farm. The optimal routing path can also be obtained for the inspection under different wind conditions. With the proposed framework, more efficient and more frequent inspection of wind turbines can be achieved for the wind farm operators. By doing so, the loss due to failures of the wind turbines can be reduced. In our future work, we will study how UAVs can bring more benefits to the operation of offshore wind farms.

APPENDIX

A. Flying Distance Determination

In (6), the flying distance is mainly determined by the product of u_i^{max} and t_i^{max} . However, u_i^{max} and t_i^{max} may influence each other. Therefore, for u_i^{max} , it is reasonable to be assigned with the maximum flying speed listed in the specification of the UAV. Quantity t_i^{max} should be calculated based on u_i^{max} , and it can be obtained from calculating the energy consumption of the UAV. Before introducing the model, V is used to represent the airspeed, $\|\mathbf{v}_{i,k,l}\|_2$, for the sake of notational simplicity. The power consumption of a UAV flying with airspeed V can then be modeled [29], [30] as

$$\begin{aligned}
 P(V) &= \underbrace{P_0 \left(1 + \frac{3V^2}{U_{tip}^2}\right)}_{\text{blade profile}} + \underbrace{P_i \left(\sqrt{1 + \frac{V^4}{4v_0^4}} - \frac{V^2}{2v_0^2}\right)}_{\text{induced power}} \\
 &\quad + \underbrace{\frac{1}{2}d_0sol\rho A_{disc}V^3}_{\text{parasite}},
 \end{aligned} \tag{17}$$

where P_0 and P_i are two constants defined in (18) representing the blade profile power and induced power in hovering status,

TABLE V
PARAMETERS AND THEIR MEANINGS FOR CALCULATING MAXIMUM FLIGHT TIME

Parameter	Physical Meaning	Value
V	Air speed in in meter per second (m/s)	16
ω	UAV weight in Newton	16
R	Rotor radius in m	0.1016
ρ	Air density in kg/m ³	1.2250
A_{disc}	Rotor disc area in m ² , $A_{disc} \triangleq \pi R^2$	0.0314
Ω	Angular velocity of UAV blade in radian per second (rad/s)	300
U_{tip}	Tip speed of the rotor blade, $U_{tip} \triangleq \Omega R$	30
b_{num}	Number of blade	8
$cord$	Chord length of UAV blade in m	0.09
sol	Rotor solidity, $sol \triangleq \frac{b_{num}cord}{\pi R}$	2.5464
k_{cor}	Incremental correction factor to induced power	0.1
v_0	Mean rotor induced velocity in hover, $v_0 \triangleq \sqrt{\omega/(2\rho A_{disc})}$	14.4179
δ	Profile drag coefficient	0.0120
S_{fp}	Fuselage equivalent flat plate area in m ²	0.0063
d_0	Fuselage drag ratio, $d_0 \triangleq \frac{S_{fp}}{A_{sol}}$	0.0787
P_{bat}	Battery capacity of the UAV in Ampere hour (Ah)	6.25

respectively.

$$\begin{cases} P_0 = \frac{\delta}{8}sol\rho A_{disc}\Omega^3 R^3, \\ P_i = (1 + k_{cor})\frac{W^{3/2}}{\sqrt{2\rho A_{disc}}}. \end{cases} \tag{18}$$

Quantity U_{tip} denotes the tip speed of the rotor blade, and v_0 is known as the mean rotor induced velocity in hover. The fuselage drag ratio and rotor solidity are denoted by d_0 and sol , respectively. The air density is denoted by ρ , and A_{disc} is the rotor disc area. The maximum flight time can then be obtained from

$$t_i^{max} = \frac{P_{bat}}{P(V)}, \tag{19}$$

where P_{bat} is the battery capacity. The parameters and the meanings used in the calculation are provided in Table V. Moreover, the values of these parameters are based on the values on the specification of the UAV that is used in the simulation. Based on (17), (18), and Table V, the energy consumption of the UAV is 212.82 Watt. Then, according to (19), t_i^{max} is 20.02 minutes. Therefore, t_i^{max} is set to 20 minutes in Section V.

B. Proof of Optimality of Algorithm 2-4 for Solving Placement Problem

Suppose the optimal solution of the placement problem is $\mathbf{A}^* = [a_i^*]_{1 \times N}$, $\mathbf{B}^* = [b_{i,k}^*]_{N \times T}$, $\mathbf{C}^* = [c_{i,j}^*]_{N \times N}$, and $[x_i^*, y_i^*]$ for $i \in \{i | a_i^* = 1\}$. Also, $n^* = \sum_{i=1}^N a_i^*$ is the optimal number of UAVs that should be deployed in the wind farm. On the other hand, the solution obtained from the proposed algorithms is denoted by \mathbf{A} , \mathbf{B} , \mathbf{C} , and $[x_i, y_i]$ for

$i \in \{i | a_i = 1\}$. Then, n is defined by $\sum_{i=1}^N a_i$. With these notations, this implies that $n > n^*$ and at least one UAV can be removed from the solution obtained from the proposed algorithms. That is, UAV j can be removed if constraints in (9) are still satisfied without UAV j . Thus, it is necessary to recheck if there is still an overlap between \mathcal{N}_i and \mathcal{N}_j . However, the end condition of Algorithm 4 is that there is no overlap between \mathcal{N}_i and \mathcal{N}_j . This contradiction implies that n is the same as n^* . Also, $\mathbf{A} = \mathbf{A}^*$, $\mathbf{B} = \mathbf{B}^*$, $\mathbf{C} = \mathbf{C}^*$, and $[x_i, y_i] = [x_i^*, y_i^*]$. Finally, we can conclude that the algorithm would converge to the optimal solution.

REFERENCES

- [1] Wood Mackenzie, "Global wind power market outlook update: Q2 2019," Wood Mackenzie Power & Renewables, Tech. Rep., Jun. 2019. [Online]. Available: <https://www.woodmac.com/reports/power-markets-global-wind-power-market-outlook-update-q2-2019-318297/>
- [2] Wind Europe, "Europe Installs 4.9 GW of New Wind Energy Capacity in First Half of 2019," Tech. Rep., Jul. 2019. [Online]. Available: <https://windeurope.org/newsroom/press-releases/europe-installs-4-9-gw-of-new-wind-energy-capacity-in-first-half-of-2019/>
- [3] J. Ribrant and L. M. Bertling, "Survey of Failures in Wind Power Systems With Focus on Swedish Wind Power Plants During 1997–2005," *IEEE Trans. Energy Convers.*, vol. 22, no. 1, pp. 167–173, Mar. 2007.
- [4] K. Branner and A. Ghadirian, "Database about Blade Faults," DTU Wind Energy E-0067, Tech. Rep. 978-87-93278-09-7, Dec. 2014.
- [5] J.-K. Lee, J.-Y. Park, K.-Y. Oh, S.-H. Ju, and J.-S. Lee, "Transformation Algorithm of Wind Turbine Blade Moment Signals for Blade Condition Monitoring," *Renew. Energy*, vol. 79, pp. 209–218, Jul. 2015.
- [6] B. E. Schäfer, D. Picchi, T. Engelhardt, and D. Abel, "Multicopter Unmanned Aerial Vehicle for Automated Inspection of Wind Turbin," in *Proc. Medit. Conf. Control Autom. (MED)*, Athens, Greece, Jun. 2016, pp. 244–249.
- [7] R. Yang, Y. He, A. Mandelis, N. Wang, X. Wu, and S. Huang, "Induction Infrared Thermography and Thermal-Wave-Radar Analysis for Imaging Inspection and Diagnosis of Blade Composites," *IEEE Trans. Ind. Informat.*, vol. 14, no. 12, pp. 5637–5647, Dec. 2018.
- [8] J. R. Gallion and R. Zoughi, "Millimeter-Wave Imaging of Surface-Breaking Cracks in Steel With Severe Surface Corrosion," *IEEE Trans. Instrum. Meas.*, vol. 66, no. 10, pp. 2789–2791, Oct. 2017.
- [9] L. Wang, Z. Zhang, H. Long, J. Xu, and R. Liu, "Wind Turbine Gearbox Failure Identification With Deep Neural Networks," *IEEE Trans. Ind. Informat.*, vol. 13, no. 3, pp. 1360–1368, Jun. 2017.
- [10] B. Yuan, C. Wang, F. Jiang, M. Long, P. S. Yu, and Y. Liu, "WaveletFCNN: A Deep Time Series Classification Model for Wind Turbine Blade Icing Detection," *arXiv*, Feb. 2019. [Online]. Available: <https://arxiv.org/abs/1902.05625>
- [11] M. Aghaei, F. Grimaccia, C. A. Gonano, and S. Leva, "Innovative Automated Control System for PV Fields Inspection and Remote Control," *IEEE Trans. Ind. Electron.*, vol. 62, no. 11, pp. 7287–7296, Nov. 2015.
- [12] L. Wang and Z. Zhang, "Automatic Detection of Wind Turbine Blade Surface Cracks Based on UAV-Taken Images," *IEEE Trans. Ind. Electron.*, vol. 64, no. 9, pp. 7293–7303, Sep. 2017.
- [13] J. R. T. Neto, A. Boukerche, R. S. Yokoyama, D. L. Guidoni, R. I. Meneguette, J. Ueyama, and L. A. Villas, "Performance Evaluation of Unmanned Aerial Vehicles in Automatic Power Meter Readings," in *Proc. Ad Hoc New.*, vol. 60, no. C, Amsterdam, The Netherlands, May 2017, pp. 11–25.
- [14] L. Wang, Z. Zhang, and X. Luo, "A Two-Stage Data-Driven Approach for Image-Based Wind Turbine Blade Crack Inspections," *IEEE/ASME Trans. Mechatronics*, vol. 24, no. 3, pp. 1271–1281, Jun. 2019.
- [15] G. J. Lim, S. Kim, J. Cho, Y. Gong, and A. Khodaei, "Multi-UAV Pre-Positioning and Routing for Power Network Damage Assessment," *IEEE Trans. Smart Grid*, vol. 9, no. 4, pp. 3643–3651, Jul. 2018.
- [16] X. Tao, D. Zhang, Z. Wang, X. Liu, H. Zhang, and D. Xu, "Detection of Power Line Insulator Defects Using Aerial Images Analyzed With Convolutional Neural Networks," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 50, no. 4, pp. 1486–1498, Apr. 2020.
- [17] S. Y. Derakhshandeh, Z. Mobini, M. Mohammadi, and M. Nikbakht, "UAV-Assisted Fault Location in Power Distribution Systems: An Optimization Approach," *IEEE Trans. Smart Grid*, vol. 10, no. 4, pp. 4628–4636, Jul. 2019.

- [18] J. Zhang, Y. Zeng, and R. Zhang, "UAV-Enabled Radio Access Network: Multi-Mode Communication and Trajectory Design," *IEEE Trans. Sig. Proc.*, vol. 66, no. 20, pp. 5269–5284, Oct. 2018.
- [19] Y. Zeng, J. Xu, and R. Zhang, "Securing UAV Communications via Joint Trajectory and Power Control," *IEEE Trans. Wireless Commun.*, vol. 18, no. 2, pp. 1376–1389, Feb. 2019.
- [20] —, "Energy Minimization for Wireless Communication With Rotary-Wing UAV," *IEEE Trans. Wireless Commun.*, vol. 18, no. 4, pp. 2329–2345, Apr. 2019.
- [21] X. Zhang and L. Duan, "Fast Deployment of UAV Networks for Optimal Wireless Coverage," *IEEE Trans. Mobile Comput.*, vol. 18, no. 3, pp. 588–601, Mar. 2019.
- [22] J. Sun and C. Masouros, "Deployment Strategies of Multiple Aerial BSs for User Coverage and Power Efficiency Maximization," *IEEE Trans. Commun.*, vol. 67, no. 4, pp. 2981–2994, Apr. 2019.
- [23] Q. Zhang, W. Saad, M. Bennis, X. Lu, M. Debbah, and W. Zuo, "Predictive Deployment of UAV Base Stations in Wireless Networks: Machine Learning Meets Contract Theory," *arXiv*, Nov. 2018. [Online]. Available: <https://arxiv.org/abs/1811.01149>
- [24] L. Mathews, "Airobotics Drone Recharging Station will Function Forever without Humans," Jun. 2016. [Online]. Available: <https://www.geek.com/tech/airobotics-drone-recharging-station-will-function-forever-without-humans-1658919/>
- [25] X. Wang, V. Yadav, and S. N. Balakrishnan, "Emission-Aware and Cost-Effective Distributed Demand Response System for Extensively Electrified Large Ports," *IEEE Trans. Control Syst. Technol.*, vol. 15, no. 4, pp. 672–679, Jul. 2007.
- [26] F. Fabra, C. T. Calafate, J. C. Cano, and P. Manzoni, "A Collision Avoidance Solution for UAVs Following Planned Missions," in *Proc. IEEE Wireless Commun. Netw. Conf. Wksp. (WCNCW)*, Barcelona, Spain, Apr. 2018.
- [27] National Centre for Earth Observation and National Centre for Atmospheric Science, "The CEDA Archive: The Natural Environment Research Council's Data Repository for Atmospheric Science and Earth Observation." [Online]. Available: <http://archive.ceda.ac.uk/>
- [28] The Kingfisher Information Service - Offshore Renewable Cable Awareness, "Awareness Chart of Walney 1-4." [Online]. Available: <http://www.kis-orca.eu/downloads>
- [29] G. D. A. R. S. Bramwell and D. Balmford, *Bramwell's Helicopter Dynamics, 2nd ed.* American Institute of Aeronautics & Ast (AIAA), 2001.
- [30] A. Filippone, *Flight Performance of Fixed and Rotary Wing Aircraft.* American Institute of Aeronautics & Ast (AIAA), 2006.



Hwei-Ming Chung received the B.S. degree from the Department of Electrical Engineering and the M.S. degree from the Institute of Communications Engineering, National Sun Yat-sen University, Kaohsiung, Taiwan, in 2014 and 2016, respectively. He was a research assistant at the Wireless Communications Laboratory, Research Center for Information Technology Innovation, Academia Sinica, Taiwan in 2017. Then, he is currently working toward the Ph.D. degree with the Department of Informatics, University of Oslo, Norway. His current research

interests include power system monitoring, smart grid, and statistical signal processing.



Sabita Maharjan [M'09] received her Ph.D. degree in Networks and Distributed Systems from University of Oslo, and Simula Research Laboratory, Norway, in 2013. She is currently a Senior Research Scientist in Simula Metropolitan Center for Digital Engineering, Norway, and Associate Professor (adjunct position) in University of Oslo, Norway. She worked as a Research Engineer in Institute for Infocomm Research (I2R), Singapore in 2010. She was a Visiting Scholar at Zhejiang University (ZU), Hangzhou, China in 2011, and a Visiting

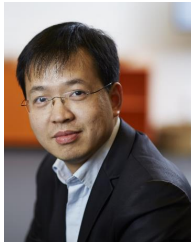
Research Collaborator in University of Illinois at Urbana Champaign (UIUC) in 2012. She was a Postdoctoral Fellow at Simula Research laboratory, Norway from 2014 to 2016. She publishes regularly in prestigious journals in her field such as IEEE Transactions on Smart Grid, IEEE Transactions on Vehicular Technology, IEEE Transactions on Intelligent Transportation Systems, IEEE Communications Magazine, IEEE Network Magazine, IEEE Wireless Communications Magazine and IEEE Internet of Things Journal. She serves/has served in the technical program committee of conferences including top conferences like IEEE INFOCOM and IEEE IWQoS. Her current research interests include vehicular networks and 5G, network security and resilience, smart grid communications, Internet of Things, machine-to-mac



Kai Strunz received the Dipl.-Ing. and Dr.-Ing. (summa cum laude) degrees from Saarland University, Saarbrücken, Germany, in 1996 and 2001, respectively. From 1995 to 1997, he pursued research at Brunel University in London. From 1997 to 2002, he worked at the Division Recherche et Développement of Electricité de France in the Paris area. From 2002 to 2007, he was an Assistant Professor of electrical engineering with the University of Washington in Seattle. Since September 2007, he has been Professor with the Department of Sustainable

Electric Networks and Sources of Energy, Technische Universität Berlin. He is Guest Professor of the Chinese Academy of Science, Beijing.

Dr. Strunz is the Secretary of the IEEE Power and Energy Society (PES) Committee on Energy Development and Power Generation, Chair of the Subcommittee on Distributed Energy Resources, and Vice Chair of the Working Group on Dynamic Performance and Modeling of HVDC Systems and Power Electronics for Transmission Systems. In 2012, Dr. Strunz was the General and Technical Program Chair of the IEEE PES Innovative Smart Grid Technologies (ISGT) Europe 2012 in Berlin. He has received the IEEE PES Prize Paper Award 2015 and the IEEE Journal of Emerging and Selected Topics.



Yan Zhang (IEEE Fellow'20) received the Ph.D. degree from the School of Electrical and Electronics Engineering, Nanyang Technological University, Singapore. He is currently a Full Professor with the Department of Informatics, University of Oslo, Oslo, Norway. His research interests include next-generation wireless networks leading to 5G beyond/6G, green and secure cyber-physical systems (e.g., smart grid and transport). Dr. Zhang is an Editor for several IEEE publications, including IEEE Communications Magazine, IEEE Network Magazine,

IEEE Transactions on Vehicular Technology, IEEE Transactions on Industrial Informatics, IEEE Transactions on Green Communications and Networking, IEEE Communications Survey and Tutorials, IEEE Internet of Things Journal, IEEE Systems Journal, IEEE Vehicular Technology Magazine, and IEEE Blockchain Technical Briefs. He is a symposium/track chair in a number of conferences, including IEEE ICC 2021, IEEE Globecom 2017, IEEE PIMRC 2016, IEEE SmartGridComm 2015. He is an IEEE Vehicular Technology Society Distinguished Lecturer during 2016-2020 and he is named as CCF 2019 Distinguished Speaker. He is the Chair of IEEE Communications Society Technical Committee on Green Communications and Computing (TCGCC). He is an elected member of CCF Technical Committee of Blockchain. In both 2019 and 2018, Prof. Zhang was a recipient of the global "Highly Cited Researcher" Award (Web of Science top 1% most cited worldwide).



Frank Eliassen is professor in the UiOs Department of informatics. He is an experienced researcher and a project manager for several decades, in the areas of distributed systems middleware and IoT/Cyber-Physical Systems (CPS) with experience from national and EU level projects. His present research interest include service-oriented IoT/edge/fog computing and CPS middleware and programming models in application areas including smart cities and smart grids, adaptive software systems, autonomic systems (self-*), peer-to-peer systems, and coopera-

tive micro-grids.