This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJVT.2021.3095467, IEEE Open Journal of Vehicular Technology

Twin-Component Near-Pareto Routing Optimization for AANETs in the North-Atlantic Region Relying on Real Flight Statistics

Jingjing Cui, *Member, IEEE,* Halil Yetgin, *Member, IEEE*, Dong Liu, *Member, IEEE*, Jiankang Zhang, *Senior Member, IEEE*, Soon Xin Ng, *Senior Member, IEEE* and Lajos Hanzo, *Fellow, IEEE*

***Abstract*—Integrated ground-air-space (IGAS) networks intrinsically amalgamate terrestrial and non-terrestrial communication techniques in support of universal connectivity across the globe. Multi-hop routing over the IGAS networks has the potential to provide long-distance highly directional connections in the sky. For meeting the latency and reliability requirements of in-flight connectivity, we formulate a multi-objective multi-hop routing problem in aeronautical *ad hoc* networks (AANETs) for concurrently optimizing multiple end-to-end performance metrics in terms of the total delay and the throughput. In contrast to single-objective optimization problems that may have a unique optimal solution, the problem formulated is a multi-objective combinatorial optimization problem (MOCOP), which generally has a set of trade-off solutions, called the Pareto optimal set. Due to the discrete structure of the MOCOP formulated, finding the Pareto optimal set becomes excessively complex for large-scale networks. Therefore, we employ a multi-objective evolutionary algorithm (MOEA), namely the classic NSGA-II for generating an approximation of the Pareto optimal set. Explicitly, with the intrinsic parallelism of MOEAs, the MOEA employed starts with a set of candidate solutions for creating and reproducing new solutions via genetic operators. Finally, we evaluate the MOCOP formulated for different networks generated both from simulated data as well as from real historical flight data. Our simulation results demonstrate that the utilized MOEA has the potential of finding the Pareto optimal solutions for small-scale networks, while also finding a set of high-performance nondominated solutions for large-scale networks.**

***Index Terms*—Aeronautical *ad hoc* networks (AANETs), in-flight connectivity, multi-objective combinatorial optimization problem (MOCOP), multi-objective evolutionary algorithm (MOEA).**

I. Introduction

In support of ubiquitous connectivity, non-terrestrial and terrestrial convergence has already been initiated by the third Generation Partnership Project (3GPP) [1] for improving the availability and reliability of next-generation wireless networks (NGWNs). Therefore, it is expected to provide seamless connectivity between the home, the airport terminal and the aircraft cabin in NGWNs. In contrast to enhancing a single one of the key performance metrics, most use cases of NGWNs are expected to find all optimal operating points in terms of latency, throughput, energy consumption and so on. For instance, in dense urban areas the networks are expected to support high speed, low latency as well as massive connectivity, while the latency and the reliability become more critical in smart factories. Given the limited spectrum and energy supply, these performance metrics are usually in conflict with each other, hence striking a trade-off by simultaneously optimizing multiple criteria relying multi-objective optimizations offers a way forward. Therefore, the goal of the paper is to provide connectivity during flights above the clouds by concurrently optimizing multiple performance metrics.

In this context, aircraft rely on satellite to aircraft communication (S2AC), direct aircraft to ground communication (DA2GC) and air to air communication (A2AC). Explicitly, geostationary earth orbit (GEO) satellites are capable of supporting longer-lasting connections than DA2GC and A2AC as a benefit of their near-global coverage, but they suffer from an excessive propagation delay. In particular, the links between the ground and the satellite are subject to approximately 120ms one-way propagation delay, which constitutes a challenge when aiming for interactive communications. Fortunately, the medium and low earth orbit (MEO/LEO) satellites are significantly closer to the earth, hence they have substantially lower propagation latency than GEO satellites. More specifically, the LEO satellites at 300km altitude are capable of offering the lowest latency of any of the satellite orbits at a propagation delay of 1ms. However, the LEO satellites can only offer limited coverage, hence imposing different technical challenges than those of GEO satellites. Compared to S2AC, DA2GC has the potential of providing limited-delay transmission as well as low-cost deployment, but suffers from limited coverage [1]. As a result, aeronautical *ad hoc* networks integrating S2AC, DA2GC and A2AC provide a promising solution for supporting high-speed and moderate-delay in-flight connectivity by relying on multi-hop communication techniques [2], [3]. In particular, the integration of A2AC and DA2GC has the promise of conveying long-distance data packets from the ground base station (BS) to the destination aircraft by avoiding S2AC links, whilst maintaining reliable high-speed connections. Another potential benefit of AANETs relying on A2AC with DA2GC is that AANETs are capable of reducing the latency as well as the spiralling communication cost of satellite communications.

Considering the characteristics of AANETs, communication between the source and destination nodes that are usually far apart relies on cooperative multi-hop transmission. The goal of multi-hop routing is to select a subset of intermediate nodes to construct a multi-hop path spanning from the source node (SN) to the destination node (DN). Regarding

This work would like to acknowledge the financial support of the Engineering and Physical Sciences Research Council projects EP/P034284/1 and EP/P003990/1 (COALESCE) as well as of the European Research Council's Advanced Fellow Grant QuantCom (Grant No. 789028). The authors also acknowledge the use of the IRIDIS High Performance Computing Facility, and associated support services at the University of Southampton, in the completion of this work. Halil Yetgin was funded by the Slovenian Research Agency (Grant no. P2-0016).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJVT.2021.3095467, IEEE Open Journal of Vehicular Technology

the multi-hop routing problem of AANETs, most of the research contributions focused on finding the optimal route in terms of a single performance metric. Specifically, in [4], a geographic load sharing based forwarding and handover strategy was proposed for airborne mesh networks relying on multi-server queueing models. As a further development, a genetic algorithm based joint routing and scheduling technique was proposed for minimizing the weighted hop count. In [5], the characteristics of heterogeneous airborne networks relying on military radios were interpreted and their field tests concerning their interaction with mobile *ad hoc* network (MANET) protocols were reported on. Inspired by the accurate geographic information available for aircraft, a trajectory density based routing algorithm was developed for maximizing the successful packet delivery probability in [6]. In practice, routing schemes are often designed for improving the overall system performance in terms of multiple quality of service (QoS) parameters. A position-aware secure routing technque was proposed in [7] with the objective of enhancing the security of drone-assisted wireless mesh networks, whilst a path discovery approach was invoked in [8] by incorporating multiple path parameters. In [9], an powerful greedy routing strategy relying on probabilistic neighbour selection was conceived for a context-aware MANET. Regarding dynamic MANETs, some stochastic techniques such as Q-learning based cognitive routing [10] and online routing [11] were proposed for finding the best transmission path.

Again, most of the existing routing designs rely on a single objective or on artificially reducing the number of conflicting goals into a single objective. However, the relationship amongst objectives is usually rather complex and also relies the solution space available. Let us consider the simple example of choosing a routing path spanning from a SN to a DN. If we have two solutions available, namely a path having a delay of 200ms and a throughput of 2Mbps, plus a path with 400ms and 20Mbps, the decision maker might prefer the path with 400ms and 20Mbps for the sake of providing more reliable services. On the other hand, if the choice is between a path with 150ms and 50Mbps as well as a path with 100ms and 45Mbps, the second path may be preferred for its lower-delay and moderate-throughput services. In general, it is quite a challenge to combine different objectives into a single objective function before the solution space is known. Similarly, it is also hard to specify the constraints imposed on the objectives before the solution space is known, since the resultant feasible region may become empty, hence making the optimization problem impossible to solve. In constrast to optimizing a single objective, multi-objective optimization problems (MOOPs) are capable of characterizing distinct performance metrics of a system, which may be independent of and/or conflicting with each other [12]. In contrast to single-objective optimization problems that may have a unique optimal solution, MOOPs often have a set of solutions representing the trade-offs among these objectives, and thus MOOPs provide new opportunities for defining problems. However, due to the uncertainty concerning the solution space and the objective space, generating the Pareto optimal set of MOOPs is computationally expensive and often it is even impracticable. As a consequence, bio-inspired stochastic search methods such as multi-objective evolutionary algorithms (MOEAs) [13], multi-objective tabu search [14] and multi-objective simulated annealing [15] etc, have been developed for approaching the optimal trade-offs. In particular, MOEAs possess several features that are desirable for MOOPs and make them preferable to other optimization methods.

Specifically, the main characteristics of MOEAs include: 1) the intrinsic parallelism, which lends MOEAs the potential of capturing multiple Pareto optimal solutions in a single simulation run; and 2) exploiting synergies among the solutions during evolution, which potentially allows MOEAs to converge to the Pareto optimal solutions. Since MOEAs operate on a set of candidate solutions, the so-called elitism strategy was introduced for ensuring that the hitherto best solution always survives for the next generation. In [16], a fast and elitist nondominated sorting genetic algorithm (NSGA-II) was proposed for handling MOOPs, which employed a fast nondominated approach for ordering the solutions found in each generation. In contrast to the continuous MOOPs of [16], the multi-objective routing problem considered in this paper is a multi-objective combinatorial optimization problem (MOCOP) that involves a number of discrete variables. The application of MOEAs in wireless networks has been investigated in [17], [18] for striking a trade-off between the delay, the energy consumption as well as the bit error rate (BER). However, these algorithms are only suitable for employment in networks having a small number of nodes. Against this background, in this paper, we employ a MOEA, namely the classic NSGA-II, that solves the multi-objective multi-hop routing problem of AANETs, which is eminently simulate for large-scale networks. The main contributions of the paper are summarized as follows, which are boldly and explicitly contrasted to the state-of-the art in **Table I**.

1) We concurrently minimize the total delay and maximize the throughput of routing paths in in-flight connectivity at the same time. As the two objectives are usually in conflict in AANETs, there exists a number of Pareto optimal solutions since no single solution is optimal simultaneously for each objectives. Furthermore, for the multi-objective routing problem formulated, counting the number of solutions is #P-complete [21]. In addition to the potentially excessive solution space, another challenge of the MOCOP formulated is the uncertainty concerning the number of Pareto optimal solutions. To address these challenges, we employ the MOEA for approximating the set of the Pareto optimal solutions.
2) We present a novel technique for generating and reproducing new solutions to our multi-objective routing problem via genetic operators during its evolution. Specifically, a variable-length chromosome encoding approach is conceived, where a routing path is represented uniquely and unambiguously by a sequence of node indices (IDs). Then, a common-node based crossover approach and a chromosome-wise mutation approach are exploited for producing new solutions by exploiting the interaction and cooperation among the solutions,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJVT.2021.3095467, IEEE Open Journal of Vehicular Technology

Table I: Comparison of routing optimization approaches

	[4]	[6]	[7]	[8]	[9]	[11]	[17]	[18]	[19]	[20]	This article
Single objective	√	√	√	√	√	√			√	√	
Multiple objectives							√	√			√
Pareto optimization							√	√			√
Stochastic search						√	√	√	√	√	√
In-flight connectivity	√	√	√	√		√			√		√
Large-scale networks	√			√					√		√

where the knowledge of network topology is exploited for avoiding the generation of infeasible paths.

3) We develop a multi-objective routing solution based on NSGA-II. First, a set of solutions is generated randomly and the solutions are encoded into variable-length chromosomes for performing crossovers and mutation. With the solutions at hand, the new solutions of the next generation are selected by sorting them into a number of sets based on their dominance in the objective space and then the crowding distances are calculated within each set. Then, the best solutions associated with higher ranks and lower density are selected for generating a diverse set of solutions and for guaranteeing closer convergence to the true Pareto optimal solutions.
4) We evaluate the multi-objective routing problem formulated in different networks that are generated both by artificial simulated data and by real flight data collected from the North-Atlantic region. Specifically, a pair of metrics are introduced for assessing the quality of the trade-off solutions found by the MOEA employed. Then an extensive simulation study is provided for characterizing the networks having different sizes. Explicitly, the number of possible solutions spans from a small number to an extremely large number in the real historical flight data set. The results demonstrate that the MOEA is capable of obtaining a set of beneficial solutions of the multi-objective routing problem formulated in terms of their spread and their convergence to the optimal trade-off solutions.

A. Basic Concepts and Organization

In this paper, a *directed graph* (or *digraph*) is defined as a graph that is made up of a set of vertices connected by edges, where the edges have a direction associated with them. A *complete digraph* is a *digraph* in which every pair of distinct vertices is connected by a pair of unique edges (one in each direction). A *simple path* is a path that contains no repeated vertices. Inspired by biological evolution, in MOEAs, any solution candidate is termed as an individual and all possible solutions are referred to as individuals. The set of possible solutions is called the population. To be more precise, the *population* in MOEA is a multiset, which may contain multiple copies of the same individual. Note that an *individual* is generally not a decision vector, but rather encodes it based on an appropriate representation. Thus an *individual* is also called a *chromosome* in MOEAs. The *mating pool* is formed by candidate solutions that are used for creating our next-generation population. *Solutions* that are included in the mating pool are referred to as parents. Every two parents selected from the mating pool will generate two *offspring* (*children*). In MOOPs, the set of optimal solutions in the decision space is referred to as *Pareto (optimal) set* and its image in the objective space is called *Pareto (optimal) front*. Moreover, the key notations used in the paper are given in **Table II**.

The rest of this article is organized as follows. Section II presents the system model, followed by MOOP modelling and the basic concepts of MOOPs in Section III. Section IV is devoted to the MOEA employed in terms of genetic operators and its implementation procedure for solving the MOOP considered. The size of solution space and performance metrics of assessing the MOEA are introduced in Section V. Section VI offers our simulation results and discussions, followed by our conclusions in Section VII.

II. System model

We consider an AANET comprised of the ground layer and the aerial layer, where the aircraft want to connect to a certain ground BSs either via direct communication or multi-hop communication techniques. In particular, the aircraft can build communication links with other aircraft and ground BSs via A2AC and DA2GC techniques, respectively, in order to improve the onboard Internet experience of aircraft passengers. Denote the nodes in the system encompassing aircraft and ground BSs as a set of nodes with $\mathcal{N} = \{1, 2, \cdots, N\}$, where each node $n \in \mathcal{N}$ uniquely corresponds to a specific entity in the system. Moreover, the network is operated in half-duplex mode and in an interference-free scenario such as in [4]–[6]. To guarantee that the signal received at the destination node has a sufficiently high reliability, the tele-traffic emanating from the source on the ground is expected to pass through the links with adequate quality, but it may encounter increased delay by passing through several relaying nodes. Therefore, there are trade-offs between the reliability and the delay.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJVT.2021.3095467, IEEE Open Journal of Vehicular Technology

Table II: Table of parameters

List of Key Symbols			
s	Source node	$\mathbf{X}$	Matrix representation of a path
d	Destination node	$\mathcal{X}$	Decision space
n_k	Node k	$\mathcal{Y}$	Objective space
N	Total number of nodes	$\mathcal{F}$	Set of ordered fronts
$\mathcal{P}/\mathcal{C}$	Population	$\mathcal{F}^{PF}$	Pareto front
P	Population size	$\tilde{\mathcal{F}}^*$	Obtained nondominated front
G	Generations	$f_m(\cdot)$	The i-th objective function
$\langle k \rangle$	Average degree	γ	Pareto front acquisition fraction
ρ	Network density	δ	Pareto front acquisition success rate

III. MOOP Modelling

In Section III-A, we will formulate the objective functions (OFs) in terms of the end-to-end delay as well as the end-to-end throughput of in-flight connectivity. Then in Section III-B, we present the definition of our multi-objective routing problem, followed by the related concepts of Pareto optimality.

A. Formulation of OFs

Let $e_{i,j}$ and R denote the link spanning from node i to node j and a route from the SN s to the DN d. Let $x_{i,j}$ be a binary indicator of the link (i,j) in a route R, where $x_{i,j}=1$ if the link $e_{i,j}$ exits in R, otherwise $x_{i,j}=0$. Hence for any routing path, there is a unique indicator matrix $\mathbf{X}=(x_{i,j}) \in \mathbb{B}^{N\times N}$ with $\mathbb{B}=\{0,1\}$ and the feasible set of all legitimate routes spanning from the source to the destination is $\mathcal{X}=\{\mathbf{X} \mid \mathbf{x} \text{ subjects to (1)}\}$, where (1) is given by

$$\sum_{\substack{j\neq i\\ j\in\mathcal{N}}} x_{i,j} - \sum_{\substack{j\neq i\\ j\in\mathcal{N}}} x_{j,i} = \begin{cases} 1, & \text{if } i=s,\\ -1, & \text{if } i=d,\\ 0, & \text{otherwise,}\end{cases} \tag{1a}$$

$$\sum_{j\neq i, j\in\mathcal{N}} x_{i,j} \begin{cases} \leq 1, & \text{if } i\neq d,\\ = 0, & \text{if } i=d,\end{cases} \tag{1b}$$

$$x_{i,j}\in\{0,1\},\ \forall i,j\in\mathcal{N}, i\neq j, \tag{1c}$$

where (1a) and (1b) ensure that the solution found for the problem formulated does indeed represent a legitimate path from the SN on the ground to the DN in the air. More specifically, (1a) represents that the ground BS acting as the SN and the target aircraft as the DN have a single outgoing link and a single incoming link, respectively, while the number of incoming and outgoing links for the other intermediate nodes (i.e., aircraft, ships or satellites in the system considered) are the same in a legitimate route. Furthermore, (1b) represents that all aircraft except for the target aircraft have at most one outgoing link in a legitimate route.

We assume that the channel between a pair of nodes is symmetric and the channel between the transmitter and receiver nodes is perfectly known. Since aircraft typically fly 10km above the ground level, they benefit from negligible scatterers and shadowing effects. Hence, we assume that communication links in the AANET networks considered have a line of sight (LoS) propagation model [2], [22]. As a result, the data rate in the link $e_{i,j}$ can be expressed as

$$\begin{aligned} C_{i,j} =& B\log_2\left(1+\frac{P_i\cdot G_i^t\cdot G_j^r\cdot H_{i,j}}{\sigma^2}\right),\\ =& B\log_2\left(1+\frac{P_i\cdot G_i^t\cdot G_j^r\lambda^\alpha}{(4\pi)^\alpha\sigma^2 d_{i,j}^\alpha}\right),\end{aligned} \tag{2}$$

where $\lambda=\frac{c}{f_c}$ is the wavelength, $c=3\times10^8$m/s and f_c is the carrier frequency, while B is the bandwidth allocated to the link. Furthermore, G_i^t and G_j^r are the transmit and the receive antenna gain, respectively. Finally, $d_{i,j}$ is the distance between aircraft i and aircraft j, α is the path loss exponent and σ^2 denotes the noise power.

According to the stability condition of queuing theory, the end-to-end throughput is bounded by the specific hop having the lowest throughput [23]. Therefore, the throughput of a route from the source to the destination is

$$C(\mathbf{X}) = \min_{x_{i,j}\in\mathbf{X}} x_{i,j}C_{i,j}. \tag{3}$$

The total delay D_{tot} is defined as the sum of the propagation delay, relaying delay and file-transfer delay of D_{pr}, D_{tr} and D_{df}, respectively. Therefore, the delay from node i to node j can be expressed as

$$D_{i,j} = \begin{cases} D_{i,j}^{tr}+D_{i,j}^{pr}, & \text{if } j \text{ is the destination node,}\\ D_{i,j}^{tr}+D_{i,j}^{pr}+D_{df}, & \text{otherwise,}\end{cases} \tag{4}$$

where $D_{i,j}^{tr}=\frac{L}{C_{i,j}}$ denotes the file-transfer delay for a file size of L from node i to node j, $D_{i,j}^{pr}=\frac{d_{i,j}}{c}$ denotes the propagation delay of link $e_{i,j}$, and D_{df} denotes the decode-and-forward (DF) relaying delay over link $e_{i,j}$. It should be pointed out that the DF relaying delay depends on decoding and encoding techniques as well as the hardware performance, etc, which would alter the DF relaying time. In this paper , the DF relaying delay is assumed to be 20ms for simplicity. Correspondingly, the total delay for a route R is given by

$$D(\mathbf{X}) = \sum_{x_{i,j}\in\mathbf{X}} x_{i,j}D_{i,j}. \tag{5}$$

Note that the throughput and the delay of a routing path usually cannot reach their individual best value simultaneously. To elaborate a little further, Fig. 1 provides a concrete example for illustrating that the delay and the throughput constitutes a pair of conflicting objectives. Specifically, there are four nodes in the network considered in Fig. 1, where the source node n_0

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJVT.2021.3095467, IEEE Open Journal of Vehicular Technology

wants to send a message to the destination node n_3, where n_1 and n_2 are the intermediate nodes between n_0 and n_3. There are two optional routes from n_0 to n_3: $R_1 : n_0 \rightarrow n_3$ and $R_2 : n_0 \rightarrow n_1 \rightarrow n_2 \rightarrow n_3$. In our manuscript, the total delay is defined as the sum of the propagation delay, relaying delay and file-transfer delay. Since the relaying detection and retransmission delay are taken into account, a route will have a much longer delay when it contains intermediate nodes. This is because each relay will add a certain extra delay, depending on whether amplifying-forward or decode-forward relaying is used. As for the throughput, it is determined by that of the lowest-throughput link of a complete route. Let us consider a concrete example, where $d_{0,1} = d_{1,2} = d_{2,3} = 300$km and $d_{0,3} = 700$km for the simplicity of calculations. Correspondingly, the throughput and the delay of the two routes R_1 and R_2 can be obtained based on equations of (3) and (5) in the manuscript, respectively. Based on the parameter setups used in the manuscript, we can obtain the the delay and the throughput on each edge given in Fig. 1, where the units of the delay and the throughput are ms and km, respectively. We can see that the delay and the throughput over route R_1 of our example are 60ms and 35Mbps, respectively, while route R_2 has the delay of 15ms and the throughput of 16Mbps. This indicates that the delay and the throughput are a pair of conflicting objectives, hence generally they cannot simultaneously reach their best value. Furthermore, note that the relationship between the delay and the throughput relies the topology of the AANET to be constructed. When the the solution space is unknown, it is quite a challenge to combine different objectives into a single objective function or to specify the constraints imposed on the objectives, since the resultant feasible region may become empty. Therefore, we formulate the optimization of the delay and the throughput as a MOOP.

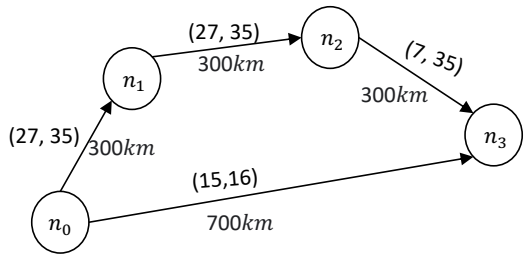


Figure 1: An example of illustrating two conflicting objectives in terms of the throughput and the delay of an AANET.

B. Problem Model

To provide low-delay and high-reliability in-flight connections, we construct the MOOP of minimizing the total delay while maximizing the throughput of the selected route, which can be formulated as follows:

$$\begin{aligned} &\min\ D(\mathbf{X}) \text{ and } \max\ C(\mathbf{X}) \\ &\text{s.t. } \mathbf{X} \in \mathcal{X}. \end{aligned} \tag{6}$$

Note that Problem (6) is a MOCOP, where $\mathbf{X}$ denotes the matrix of decision variables, while $\mathcal{X}$ represents the decision space. It is challenging to solve Problem (6) due to the combinatorial nature of $D(\mathbf{X})$ and $C(\mathbf{X})$, especially for a non-monotonic $C(\mathbf{X})$.

For solving Problem (6), we transform it into the standard MOCOP having a minimization-type objective as follows:

$$\begin{aligned} &\min\ \mathbf{f}(\mathbf{X}) = [f_1(\mathbf{X}), f_2(\mathbf{X})] \\ &\text{s.t. } \mathbf{X} \in \mathcal{X},\ \mathbf{f}(\mathbf{X}) \in \mathcal{Y}. \end{aligned} \tag{7}$$

where $f_1(\mathbf{X}) = D(\mathbf{X})$ and $f_2(\mathbf{X}) = -C(\mathbf{X})$, while $\mathbf{f}(\mathbf{X})$ denotes the vector of OFs. Furthermore, the values of OFs are stated in an objective vector, which constitutes the objective space $\mathcal{Y}$, i.e., $\mathbf{f}(\mathbf{X}) \in \mathcal{Y}$ for any $\mathbf{X} \in \mathcal{X}$. In contrast to single-objective optimization problems, there exist multiple optimal objective vectors representing different trade-offs between the objectives. In particular, there is no single globally optimal solution, and it is often necessary to determine a set of points that all fit a predetermined definition for an optimum. Therefore, we introduce a few fundamental concepts in terms of optimality used in MOOPs.

Definition 1. *Pareto dominance: Let $\mathbf{X}_1, \mathbf{X}_2 \in \mathcal{X}$ be a pair distinct feasible solutions of Problem* (6)*, $\mathbf{X}_1$ dominates $\mathbf{X}_2$, also denoted as $\mathbf{X}_1 \prec \mathbf{X}_2$, if and only if*

1) *$f_i(\mathbf{X}_1) \leq f_i(\mathbf{X}_2)$ for any $i \in \{1, 2\}$, and*
2) *there is at least one OF value satisfying the strict inequality, i.e. at least for one component $f_i(\mathbf{X}_1) < f_i(\mathbf{X}_2), i \in \{1, 2\}$ is true.*

Accordingly, we can say that the objective $\mathbf{f}(\mathbf{X}_1)$ dominates $\mathbf{f}(\mathbf{X}_2)$, denoted as $\mathbf{f}(\mathbf{X}_1) \prec \mathbf{f}(\mathbf{X}_2)$, if $\mathbf{X}_1$ dominates $\mathbf{X}_2$.

Therefore, we have $\mathbf{X}_1 \prec \mathbf{X}_2 \Leftrightarrow \mathbf{f}(\mathbf{X}_1) \prec \mathbf{f}(\mathbf{X}_2)$. Note that there may exist several optimal solutions in the decision space having the same objective vector, corresponding to a single optimum in the objective space.

Definition 2. *Pareto optimal: A solution, $\mathbf{X}^*$ is Pareto optimal, if and only if there doesn't exist another point, $\mathbf{X} \in \mathcal{X}$, such that $\mathbf{f}(\mathbf{X}) \prec \mathbf{f}(\mathbf{X}^*)$*

That is a solution is said to be Pareto optimal if it is not dominated by any other solution in the decision space. The set of Pareto optimal solutions in the decision space $\mathcal{X}$ is termed as the *Pareto optimal set* $\mathcal{X}^*$ with $\mathcal{X}^* \in \mathcal{X}$, while the corresponding objective vectors in the objective space constitute the *Pareto front* $\mathcal{Y}^* = \{\mathbf{f}(\mathbf{X}^*),\ \forall \mathbf{X}^* \in \mathcal{X}^*\} \in \mathcal{Y}$.

Remark 1. *Note that Problem* (6) *has the same Pareto optimal set as Problem* (7)*, while the Pareto front of Problem* (6) *is $\mathcal{Y}^* = [f_1(\mathbf{X}^*), -f_2(\mathbf{X}^*)]$.*

Due to the discrete structure of Problem (7), it is not sufficient to determine the set of all Pareto solutions (or nondominated vectors in objective space) by aggregating the objectives through weighted sums, because usually there exist Pareto solutions, which are not optimal for any weighted sum of the objectives [12]. In fact, generating the Pareto set is often computationally expensive and may even be infeasible since the complexity prevents exact methods from being applicable. For this reason, we employed NSGA-II based on stochastic search strategies for approximating the Pareto optimal set.

IV. Proposed Solutions

To mimic biological evolution, a MOEA generally includes the evolutionary operators of crossover, mutation and selection

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJVT.2021.3095467, IEEE Open Journal of Vehicular Technology

[24]. In particular, the crossover and mutation operators aim for generating new solutions within the search space by altering the existing ones. Specifically, crossover operator recombines the genetic information of two parents for generating new offspring and the mutation operator modifies individuals by transforming a part according to a given mutation rate.

A. Path Encoding

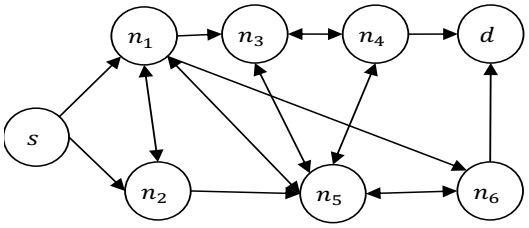
How to encode a path in the graph is key for developing a MOEA to Problem (6). From (1), we can see that each path includes N^2 binary variables, hence the length of a chromosome is extremely high when using binary encoding. On the other hand, in contrast to the permutation encoding methods used for the 01 knapsack problem and the travelling salesman problem (TSP) [25], the lengths of routing paths in our problem vary. Hence we opt for directly encoding the chromosomes based on node IDs [26], [27], which results in variable-length chromosomes.

Explicitly, the node-ID based path representation uses a sequence of integers for representing a chromosome, where each gene represents the index of a node which a routing path passes through. As a result, each locus of the chromosome denotes the hop index of a node in a path. Note that in all chromosomes, the first and the last genes are reserved for the source node and the destination, respectively. Furthermore, the length of the chromosome is variable, but the maximum length of a chromosome is N. Explicitly, the number of the nodes in the longest path spanning from the source node to the destination is no more than N. Having said that, we can see that any routing path of Problem (6) can be encoded by a chromosome according to the topology of the network. More specifically, considering a specific chromosome, the first gene is the SN, and the second gene is randomly selected from the nodes that are directly connected with the SN. Then the node chosen is removed from the routing table to prevent the node from being repeatedly selected, so that having loops in the routing path can be avoided. Fig. 2 illustrates a simple network as well as the chromosome of a routing path. Fig. 2(b) illustrates the representation of a chromosome that encodes a routing path from s to d via nodes $n_1 \cdots n_k$. Note that each node in the chromosome denotes a gene and there are $(l+1)$ genes in total.

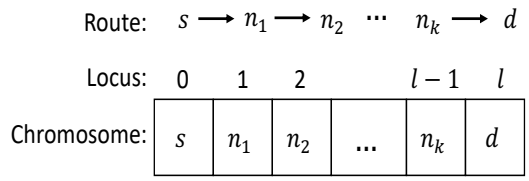
B. Population Initialization

Defining the population initialization of MOEAs requires that of the initial population size and the specific procedure of initializing the population. Since the population size relates to the nature of the problem, a very large population will slow down the algorithm, while a smaller population might result in a local solution. As a result, deciding a adequate population size is crucial for approaching the true Pareto front. However, due to the problem-dependent and metaheuristic natures of MOEA [28], determining the best population for general MOOPs is challenging and unattainable. For this reason, in this paper the population is initialized by several hundreds of possible solutions [16], [25], [29].

Furthermore, there are two popular ways of generating the initial population [24]: heuristic initialization and random initialization. Although the heuristic initialization may help the MOEA to find solutions faster, it may just explore a small part of the solution space and never find globally optimal solutions because of the lack of diversity in the population. Therefore, we opt for the random initialization approach that are used in most literature such as [16], [25]. It should be noted that the randomly initial population only contains a set of possible legitimate paths, but exclude the infeasible paths[1].



(a) A simple digraph of an AANET.



(b) Node ID based encoding

Figure 2: An example of a routing path and its chromosome.

C. Genetic Operators

The task of genetic operators is to create new populations from the existing solutions, where crossover operation is used for generating offspring. Then the new offspring are mutated with a small probability, which helps avoid getting trapping in local optima. Due to the variable-length nature of the chromosomes, specific crossover and mutation techniques are required.

1) Crossover: The crossover operations are used for generating new offspring from the current population in order to find better ones. In the crossover procedure of the multi-objective routing problem, two chosen chromosomes (parents) exchange their partial routes to generate new offspring. In particular, the resultant offspring must represent one of the routes from the source node to the destination node, otherwise it is a lethal gene (infeasible route). In contrast to the conventional on-site crossover and to the binary crossover [16], in this paper we adopt the common node based crossover method [27], where the pair of chromosomes selected for the crossover operation must have at least one common gene (node), except for the source and destination nodes, but they can be located at different locus. Furthermore, the crossing point will be selected randomly if there are multiple common genes between two chromosomes. Note that due to the sparsity of adequate-quality links in AANETs, the uniform crossover using a mask is likely to produce illegal routing paths. For instance, some airplanes in the routing path generated are not connected in the AANET or some airplanes will be considered more than once. To avoid these issues, we adopted the common node based

[1]In this paper, there are two types of infeasible paths: the paths that include at least one link that is nonexistent in the network and the paths with loops.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJVT.2021.3095467, IEEE Open Journal of Vehicular Technology

crossover method in our work, which guarantees that any path generated by the crossover corresponds to a legitimate route in the AANET. More sophisticated design strategies can be developed for further enhancing the attainable performance of the networks considered, but this is beyond the scope of this manuscript.

Fig. 3 highlights the crossover procedure, where a pair of chromosomes, Parent 1 and Parent 2, are selected, which contains the pair of common nodes n_1 and n_3. Therefore, there are two pairs of possible crossing points $(2,1)$ and $(3,2)$, which are also called potential crossing sites in MOEAs. Then, one pair of the crossing sites is randomly selected such as $(2,1)$ to generate a pair of new chromosomes, namely Child 1 and Child 2. It is possible that loops are generated during crossover, which violates the constraints of (1a). Although such chromosomes (routes with loops) will be gradually removed from the population, a repair approach is applied in this paper, which is capable of removing all loops in a route. The detailed implementation of the crossover procedure is given in **Algorithm 1**. The newly created offspring $\mathcal{Q}$ can then be mutated, as discussed in the following section.

2) Mutation: The offspring created by crossover are mutated by modifying small parts of the individual with a mutation probability p_m. In this paper, mutation alters one or more gene values in a chromosome except for the source and the destination nodes, thus a routing path may change entirely from the previous path. Explicitly, the mutation procedure first randomly selects a mutation point from the locus of $[1, l-1]$ for a chromosome to be mutated. As a result, the chromosome is divided into two portions at the mutation point, where the former partial-path right before the mutation point (excluding the mutation node) will be passed on to the next generation. Then the latter portion (including the mutation node) will be replaced by an alternative partial-path, which is randomly generated between the mutation node to the destination node according to the network topology information. Note that due to the random effects in the genetic operators, some individuals in the mating pool may not be affected by the variation and simply represent a copy of a previously generated solution.

Algorithm 1: The crossover procedure

Function `crossover(`$\mathcal{P}$`):` `/* crossover */`

Set the offspring set $\mathcal{Q} = \emptyset$;
Evenly divide $\mathcal{P}$ into two random subsets $\mathcal{P}_1$ and $\mathcal{P}_2$;
while $\mathcal{P}_1 \neq \emptyset$ **do**
  Choose $Parent_1$ and $Parent_2$ from $\mathcal{P}_1$ and $\mathcal{P}_2$, respectively;
  Calculate $\mathcal{C} = (Parent_1 \cap Parent_2)/\{s, d\}$;
  if $\mathcal{C} \neq \emptyset$ **then**
    Randomly select a pair of crossing sites (c_1, c_2);
    `/* "+" here means concatenate the two list. */`
    $Child_1 = Parent_1[0 : c_1] + Parent_2[c_2, :]$;
    $Child_2 = Parent_2[0 : c_2] + Parent_1[c_1, :]$;
    if *There exists loops in* $Child_1$ *or* $Child_2$ **then**
      Repair them by removing the loops;
  else
    $Child_1 = Parent_1;\ Child_2 = Parent_2$;
  Add $Child_1$ and $Child_2$ into $\mathcal{Q}$;
return $\mathcal{Q}$;

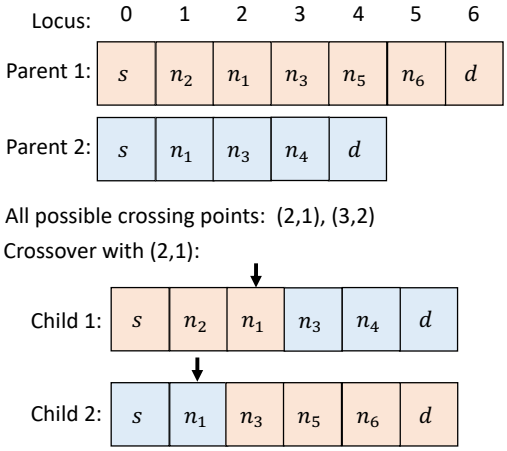


Figure 3: An example of the crossover procedure for a pair of routes.

More specifically, Fig. 4 illustrates an example of the mutation procedure, where the chromosome selected to be mutated contains four possible mutation points $\{1, 2, 3, 4\}$. Then one of the possible mutation point is picked randomly, such as point 2, which corresponds to node n_1. From the network topology information of Fig. 2(a), we can find that there are three possible candidates $\{n_3, n_5, n_6\}$ of n_1 to be mutated, where one of them will be selected randomly such as n_5. The above procedures are then repeated until the destination node is picked. The implementation procedure of mutation is given in **Algorithm 2**.

3) Naive Genetic Operations: In the naive genetic operators, the offspring are mutated randomly without exploiting any specific network-related constraints. Explicitly, in the naive genetic operators, the node selected to be mutated will be randomly changed to an arbitrary node. As a result, infeasible chromosomes may be generated, for which no routing path exist in the network and these chromosomes can be gradually removed with the aid of natural evolution. Note however that these infeasible paths may in fact generate more feasible routing paths than the specific genetic operators advocated. On the other hand, these infeasible paths may continue to produce infeasible paths via the genetic operators, at the expense of evaluating a reduced number of feasible routing paths. The performance of the MOEA using naive genetic operations will be detailed in the results of Section VI.

D. Selection

The goal of selection is to pick out the good solutions from the entire population in order to create the offspring of the next generation. Additionally, maintaining a beneficial solution-diversity in the population during selection is also

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJVT.2021.3095467, IEEE Open Journal of Vehicular Technology

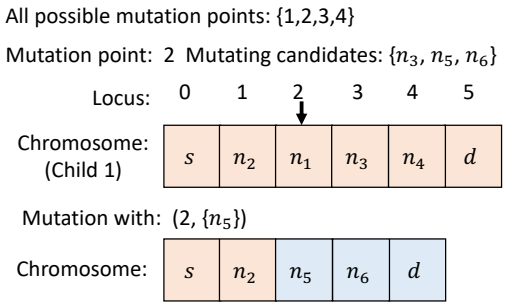


Figure 4: An example of the mutation procedure for a route.

Algorithm 2: The mutation procedure

Function `mutation(`$\mathcal{Q}$`):` `/* mutation */`
1 **for** *each chromosome* $c \in \mathcal{Q}$ **do**
2 Random generate a mutation probability ϵ;
3 **if** $\epsilon \leq$ *the mutation rate* **then**
4 Remove c from $\mathcal{Q}$;
5 $c' = c$;
6 Randomly select a valid mutation point i from c';
7 **while** *the node* $c'[i] \neq d$ **do**
8 Get the set S_i of the potential nodes that in $c[i]$;
9 **if** $S_i \neq \emptyset$ **then**
10 Randomly choose a node n' from S_i;
11 $c[i] = n'$
else
12 $c' = c$;
13 Add c' to $\mathcal{Q}$;
14 **return** $\mathcal{Q}$;

critical to the success of MOEAs, so that they converge to the Pareto optimal front. To this end, various selection methods have been developed [25] by incorporating different concepts such as elitism and niche, which are independent of the fitness assignment method used in conventional GAs. In this paper, we employ the fast elitist nondominated sorting approach of [16] for locating a series of nondominated fronts based on the domination count. The corresponding MOEA is termed as NSGA-II. Explicitly, for a chromosome c, the domination count n_c is defined as the number of points that dominates the point $\mathbf{f}(c)$ in the objective space with $c \in \mathcal{C}$. Explicitly, the domination count of the points in the first nondominated front $\mathcal{F}_1$ is zero, i.e. $n_c = 0$ for $\mathbf{f}(c) \in \mathcal{F}_1$. Then we have $n_c = i - 1$ for any point $\mathbf{f}(c)$, $c \in \mathcal{C}$ that belongs to the i-th nondominated front $\mathcal{F}_i$. Upon assuming that the maximum of the domination count in $\mathcal{C}$ is M, there are $M+1$ sorted fronts in total. The procedure of the nondominated sorting approach in NSGA-II is illustrated in **Algorithm A.1** of Appendix B, where the size of the vector is $\mathbf{n}$ is $|\mathcal{C}| = 2P$.

The selection process is based on the sorted front set $\mathcal{F}$ obtained from **Algorithm A.1**. To get a beneficial uniform spread of the Pareto front, the crowding distance is used for quantifying the density of points surrounding a particular point of $\mathbf{f}(c)$, $c \in \mathcal{C}$. For a point $\mathbf{f}(c)$ in the front $\mathcal{F}_i$, the crowding distance can be expressed as

$$D_{cd} = \sum_{m=1}^{2} \frac{|f_m(c'') - f_m(c')|}{f_m^{\max} - f_m^{\min}}, \tag{8}$$

where $\mathbf{f}(c'')$ and $\mathbf{f}(c')$ are the points that are closest to the point $\mathbf{f}(c)$ at $\mathcal{F}_i$ from either side of $\mathbf{f}(c)$, respectively. Furthermore, $f_m^{\max}$ and $f_m^{\min}$ are the maximum and minimum values of the objective function $f_m(\cdot)$. Therefore, the crowding distance D_{cd} of point $\mathbf{f}(c)$ can be viewed as the average side length of the rectangle with the two opposite corner points of $\mathbf{f}(c'')$ and $\mathbf{f}(c')$. The procedure of selection is stated in **Algorithm A.2** of Appendix B, where $\mathcal{D}_{cd}$ is the crowding distance set of all points at the front $\mathcal{F}_i$. From **Algorithm A.2**, we can see that the first front has the top priority to be chosen for later evolution. The crowding distance based selection is required for $\mathcal{F}_i$ only when the cumulative population from $\mathcal{F}_0$ to $\mathcal{F}_i$ is larger than the population size P. Note that as discussed in [16], the computational complexity of **Algorithm A.2** is dominated by that of the nondominated sorting, that is $\mathcal{O}(GP^2)$.

E. The Procedure of The MOEA Employed

The procedure is summarized in **Algorithm 3**, which determines a nondominated front $\mathcal{F}^*$ based on the network topology information. It commences by generating an initial population $\mathcal{P}_0$ having P chromosomes and then the crossover and mutation operators are applied to $\mathcal{P}_0$ for producing P offspring $\mathcal{Q}_0$. At t-th generation, all chromosomes both in $\mathcal{P}_t$ and $\mathcal{Q}_t$ would be evaluated in terms of both OFs, and thus each element in $\mathcal{C}_t$ has two records: the solution and its OF values. Afterwards, $\mathcal{C}_t$ is partitioned into a sequence of subsets/fronts based on their ranks. For constructing the new front from $\mathcal{C}_t$, the elitist and density inspired selection procedure is performed, which selects P best solutions for storing in $\mathcal{P}$, so as to create the next-generation population. When the stopping criterion is satisfied, that is the maximum number of generations has been considered, the best nondominated front obtained would be output. Furthermore, the goal of the comparison in line 9 - line 11 is to obtain the best spread of the front from the union chromosomes of the last two generations. Note that for the sake of improving the search efficiency, **Algorithm 3** confines its search within the valid solution space by avoiding the introduction of infeasible solutions into the population based on the generic operators discussed in **Algorithm 1** and **Algorithm 2**. Since the complexity of both the initialization and the genetic operators have the order of $\mathcal{O}(P)$, hence the overall complexity of **Algorithm 3** is dominated by the fast nondominated sorting of [16], that is $\mathcal{O}(GP^2)$.

V. Solution Space And Performance Metrics

In this section, we first present the network characteristics from the perspective of complex networks. Along with these characteristics, the approximations of the solution space size are provided. Finally, the performance measures of assessing the solution of MOOPs are discussed.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJVT.2021.3095467, IEEE Open Journal of Vehicular Technology

Algorithm 3: The NSGA-II based MOEA harnessed for locating the nondominated solutions

Input : Network topology information
Output: $\mathcal{F}^*$
Init. : the population size P and the number of generations G.

Randomly generate parent population $\mathcal{P}_0$ based on the network topology information;
for $t = 0 : G - 1$ **do**
  $\mathcal{Q}_t$ = `crossover` ($\mathcal{P}_t$);
  $\mathcal{Q}_t$ = `mutation` ($\mathcal{Q}_t$);
  Evaluate the objective values of all chromosomes in $\mathcal{C}_t = \mathcal{P}_t \cup \mathcal{Q}_t$;
  $\mathcal{F}$ = `Fast-nondominated-sort` ($\mathcal{C}_t$);
  $\mathcal{P}_{t+1}$ =`selection` ($\mathcal{F}$, $\mathcal{C}_t$);
  `/* Comparison */`
  if $t > 0$ **then**
    $\mathcal{C}_t = \mathcal{P}_{t+1} \cup \mathcal{F}^*$;
    $\mathcal{F}$ = `Fast-nondominated-sort` ($\mathcal{C}_t$);
    $\mathcal{F}^*$ =`selection` ($\mathcal{F}$, $\mathcal{C}_t$);
  else
    $\mathcal{F}^* = \mathcal{P}_{t+1}$;

A. Network Characteristics

The AANET can be defined as a weighted digraph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ of order N, composed of $N = |\mathcal{N}|$ nodes and $E = |\mathcal{E}|$ edges. As aircraft fly from one airport to another, the distances amongst aircraft gradually change, which results in a dynamically evolving network versus time. We characterize the networks considered in terms of their average degree and network density.

Explicitly, the node degree is given by the number of edges connected to it. In an AANET, the node degree denotes the number of aircraft connected to a specific aircraft and thus reflects its accessibility. The average degree $\langle k \rangle$ of $\mathcal{G}$ is the average number of edges per node in the graph, which is given by

$$\langle k \rangle = \frac{E}{N}, \tag{9}$$

The network density ρ of $\mathcal{G}$ is defined as the actual number of edges E divided by the total number of possible edges in $\mathcal{G}$ having N nodes. Hence it quantifies how many edges are in a set $\mathcal{E}$ compared to the maximum possible number of edges among all vertices in that set, which is given by

$$\rho = \frac{E}{N(N-1)}, \tag{10}$$

where $N(N-1)$ is the total number of possible edges of $\mathcal{G}$. If the number of edges obeys $E << N(N-1)$, it is referred to as a sparse network. Naturally, having a higher density for a network implies having a higher complexity of finding the best routing path.

B. Size of the Solution Space

In this section, we study the size of the solution space, which indicates whether it is possible to find the true Pareto front by exhaustive search. In a graph G, a path without repeated nodes is termed as a simple path. Hence the valid routing path in our problem are thereby simple paths emanating from s to d. However, determining the number of simple paths leading from s to d is challenging, since the problem of counting the number of simple paths between s and d in a graph is #P-complete [21]. More explicitly, this means that counting the number of simple paths between s and d in AANETs is at least as hard as an NP-complete problem. It is known that for a complete digraph, the formula of the total number of simple paths between s and d can be expressed as

$$K = \sum_{k=0}^{N-2} \frac{(N-2)!}{k!}. \tag{11}$$

Note that we have $\sum_{k=0}^{N-2} \frac{1}{k!} = e$ when $N \to \infty$. Therefore, (11) can be approximated as follows:

$$K \approx (N-2)!e. \tag{12}$$

Note that using a modified depth-first search to generate the paths, a single path can be found in $\mathcal{O}(N+E)$ time at most but no efficient algorithm exists for counting the number of such paths for general graphs [30]. As a result, we present the approximations of the number K in general graphs. We first introduce the method proposed in [31] to roughly estimate the size of the solution space, which is given by

$$K = \sum_{k=0}^{N-2} \frac{(N-2)!}{k!} \rho^{N-1+\delta(N,\rho)} \approx (N-2)!e\rho^{N-1}, \tag{13}$$

where $\delta(N, \rho)$ is a function of N and d. In particular, in the second equation, we used the fact that $\delta(N, \rho) \to 0$ as $N \to \infty$. Note that the approximation of (13) becomes not accurate when $\rho \leq 0.1$. Therefore, we present another estimation as follows. Since $\mathcal{G}$ has E edges, it is equivalent a complete digraph with N' vertices with $N'(N'-1) = E$. By computation, we have $N' = \lceil \frac{1+\sqrt{1+4E}}{2} \rceil$. As a consequence, the number of simple paths between s and d is $(N'-2)!e$.

C. Performance Metrics in MOOP

In contrast to single-objective optimizations, there are three goals in multi-objective optimization [16], [25]: 1) A good distribution of the points at the obtained front is desirable. Generally, we expect that the points uniformly distributed at the front obtained. 2) The distance of the resultant front to the Pareto optimal front should be minimized, which relies on a certain distance metric. 3) The extent of the resultant front should be maximized. That is for each objective, a wide range of values should be covered by the front obtained. Note that this paper aims to illustrate the performance of the MOEA with the designed genetic operators and the naive genetic operators in the multi-objective routing problem formulated in small-scale AANETs. Therefore, the $\mathcal{C}$-measure[2] of [25] is adopted, which can reflects the convergence of the obtained solutions. The definition of $\mathcal{C}$-measure of [25] is given as follows.

[2] More comprehensive performance metrics like HV and IGD+ [32] associated with more sophisticated multi-objective optimization algorithms will be designed for the challenging multi-objective routing problem of AANETs in our future research.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJVT.2021.3095467, IEEE Open Journal of Vehicular Technology

Definition 3. *$\mathcal{C}$-measure : Let $\mathcal{Y}_1, \mathcal{Y}_2 \in \mathcal{Y}$ be two sets of objective vectors. The $\mathcal{C}$ function maps the ordered pair $(\mathcal{Y}_1, \mathcal{Y}_2)$ to the interval $[0, 1]$:*

$$\mathcal{C}(\mathcal{Y}_1, \mathcal{Y}_2) = \frac{|\{\mathbf{y}_2 \in \mathcal{Y}_2 \mid \exists\ \mathbf{y}_1 \in \mathcal{Y}_1 : \mathbf{y}_1 \preceq \mathbf{y}_2\}|}{|\mathcal{Y}_2|}. \quad (14)$$

Here $\mathcal{C}(\mathcal{Y}_1, \mathcal{Y}_2) = 1$ entails that all points in $\mathcal{Y}_2$ are weakly dominated by $\mathcal{Y}_1$, while $\mathcal{C}(\mathcal{Y}_1, \mathcal{Y}_2) = 0$ represents that none of the points in $\mathcal{Y}_2$ are weakly dominated by $\mathcal{Y}_1$. In our problem, the Pareto front is constructed by a finite number of discrete points in the objective space. Therefore, the $\mathcal{C}$-measure can be transformed into a simpler form, when the Pareto front is known.

Proposition 1. *Pareto front acquisition fraction (γ): Let $\mathcal{F}^{PF}$ denote the complete Pareto front set. Given an approximated Pareto front set $\tilde{\mathcal{F}}^*$ obtained by* **Algorithm 3**, *$\mathcal{C}(\tilde{\mathcal{F}}^*, \mathcal{F}^{PF})$ can be expressed as*

$$\gamma = \mathcal{C}(\tilde{\mathcal{F}}^*, \mathcal{F}^{PF}) = \frac{|\tilde{\mathcal{F}}^* \cap \mathcal{F}^{PF}|}{|\mathcal{F}^{PF}|}, \quad (15)$$

where $\tilde{\mathcal{F}}*$ denotes the set of points at the front obtained by the MOEA employed and $\mathcal{F}^{PF}$ represents the set of points at the true Pareto front.

Proof. See Appendix B. □

The values of γ indicate the coverage extent of the obtained nondominated front to the Pareto optimal front. In parlance, γ is the fraction of the Pareto optimal front that is covered by nondominated front obtained. In addition, due to the stochastic nature of the genetic operators, multiple runs are usually performed for mitigating the randomness of the solution. Therefore, the notion of the success probability is introduced for characterizing the nondominated front obtained.

Definition 4. *Pareto front acquisition success rate (δ) is defined as the number of the event where the nondominated front obtained $\tilde{\mathcal{F}}^*$ is indeed the true Pareto front $\mathcal{F}^{PF}$ divided by the total number of events, which can be expressed as*

$$\delta = \frac{\sum_{r=1}^{R} \mathbf{1}_{\tilde{\mathcal{F}}^* = \mathcal{F}^{PF}}(r)}{R}, \quad (16)$$

where R is the total number of events.

VI. Results and Discussions

In this section, we evaluate the solutions obtained by computer simulations relying both on simulated data and on our real flight data. Specifically, we first study the performance of a series of small-scale networks, where the true Pareto front is available by the exhaustive search. Then, in Section VI-B we test our algorithms using three large datasets collected from real flights. It is worth pointing out that due to the stochastic nature of MOEAs, confidence intervals may be involved for characterizing the average performance of the MOEAs in the specific context of AANETs.

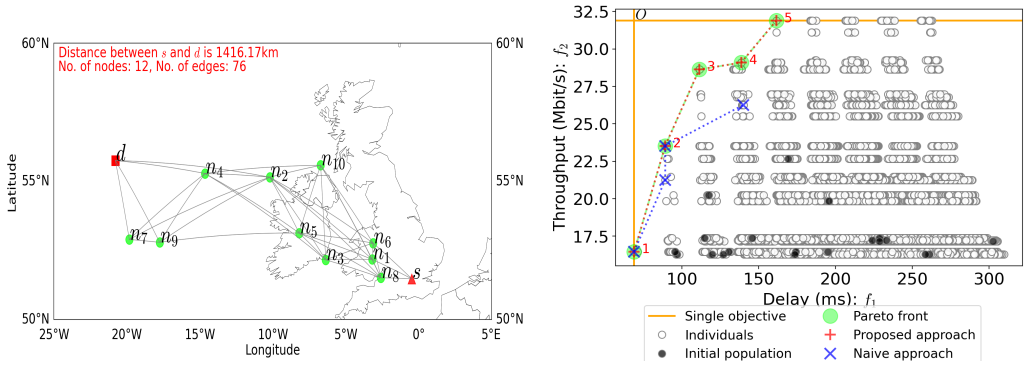
A. Simulation Results

In this section, we evaluate the performance of the proposed solutions based on computer simulations. The nondominated front obtained by **Algorithm 3** is compared to the true Pareto optimal front obtained by exhaustive search, when the solution space is not very large. Moreover, we also considered a naive approach termed Naive approach, where the naive genetic operations discussed in SectionIV-C3 are adopted. For visualization, the boundary of the objective space is also plotted. In all simulations, the mutation rate is 0.7, the population size is 20 and the maximum number of generations is 100, unless specified otherwise. Note that the mutation rate in the paper refers to the chromosome-wise mutation rate. The values of communication parameters are $L = 200$Kbit, while the system operates at the mm-Wave frequency of $f_c = 31$ GHz and the noise power is $\sigma^2 = -132$ dBm [22]. The other parameters used in our simulations are the same as those in [2]. Specifically, the transmit powers of the ground BS and of the aircraft are 45dBm and 30dBm, respectively. Furthermore, we have $G_i^t = G_i^r = 25$dB for the ground BS and the aircraft along with $B = 200$MHz and $\phi_0 = 0$dB. The heights of the ground BS, the aircraft as well as the satellite are 50m and 10.7 km, respectively.

We consider an aircraft flying from London Heathrow (LHR) to John F. Kennedy (JFK) international airport as shown in Fig. 5, in the region having the latitude range of $lat \in [51.47°, 55.75°]$ and the longitude range of $lon \in [-20.78°, -0.46°]$. The ground BS is at $(lon, lat) = (-0.46°, 55.75°)$ and the target aircraft is located at $(lon, lat) = (-20.78°, 51.47°)$. Consequently, the flight distance between the ground BS to the target aircraft is around 1416km, hence they cannot communicate directly over the horizon. There are $N_i = 10$ intermediate aircraft that are randomly generated in the region of $lon_i \in (-20.78°, -0.46°)$ and $lat_i \in (51.47°, 55.75°)$. The height of the ground BS is 50m and the aircraft altitude is 10.7km, respectively. In the example of Fig. 5 termed Network-1, we have 12 nodes and 76 edges. Furthermore, the total number of possible paths from s to d is $K = 56,514$, which can be computed by conventional enumerating.

Fig. 5(b) shows the Pareto front found by different algorithms as well as all the possible individuals. We can see from Fig. 5(b) that the individuals are distributed in strips since the throughput is not a monotonically increasing function of the routing path length, which remains constant for different route length and delays. For instance, we assume that the link $n_8 \to n_2$ has the minimum rate C_0 of all links in the network. Then we have the two routing paths $s \to n_8 \to n_2 \to d$ and $s \to n_1 \to n_6 \to n_8 \to n_2 \to d$, which have the same throughput but different delays. Furthermore, we also plot the minimum delay and maximum throughput of the network attained by single-objective optimization, respectively. Here the crossing point O at the top corner of Fig. 5(b) denotes the conceptual optimum that is actually unachievable in most networks in practice. For convenience, the points of the fronts attained by different approaches are connected by dashed lines. We observe that the Pareto front contains five points numbered as $\{1, \cdots, 5\}$ in Fig. 5(b), which are found by exhaustive search for comparison. Explicitly, given an AANET constructed, all possible solutions can be generated by efficient path-finding methods, such as the family of depth-first search based algorithms [30]. Then, comparisons of Pareto domi-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJVT.2021.3095467, IEEE Open Journal of Vehicular Technology



(a) Network-1 with 12 nodes and 76 edges.

(b) Nondominated front obtained by different approaches.

Figure 5: Comparisons of different solutions in a specific network.

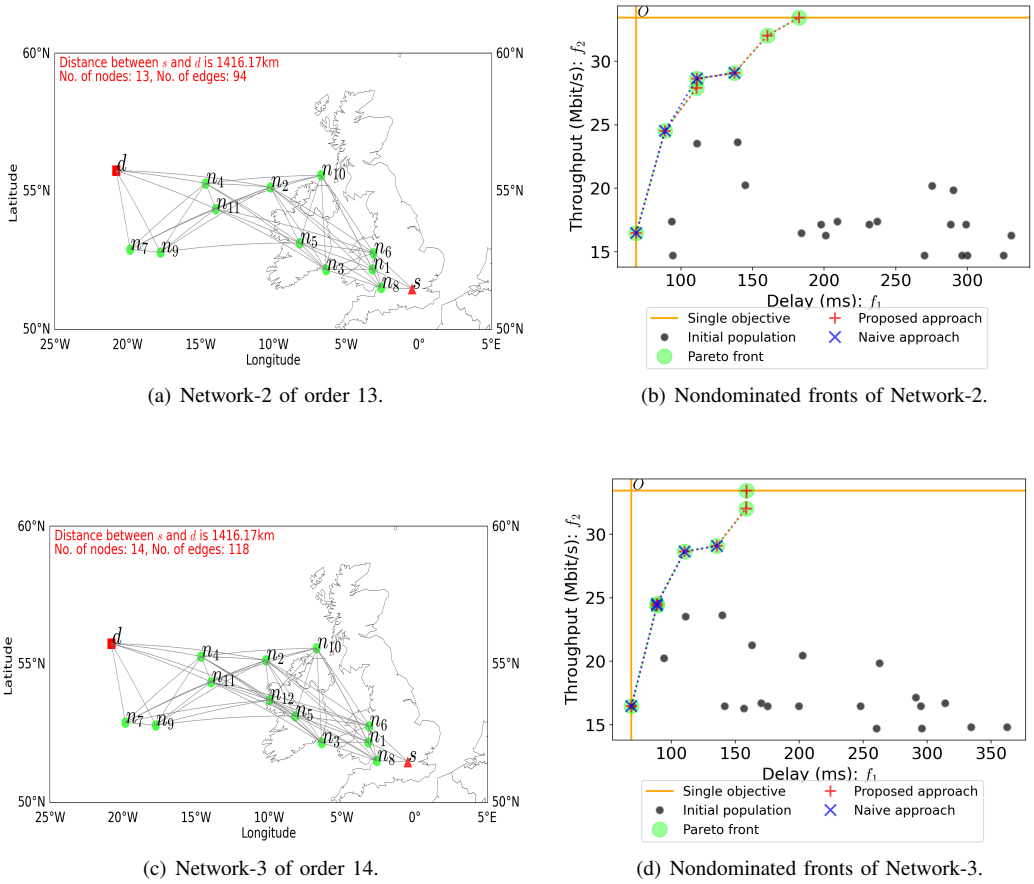
nance between a specific solution and all the other solutions in the solution space can be carried out exhaustively for the sake of checking whether the specific solution is Pareto optimal. For instance, given a solution c of the entire solution space $\mathcal{C}$, if c is nondominated by any $c' \in \mathcal{C}_{-c}$ with $\mathcal{C}_{-c}$ being the set that excludes c, then the nondominated solution c is Pareto optimal and the point in the objective space associated with c is on the true Pareto front. As a result, the true Pareto front can be obtained by repeating the comparisons until all solutions are checked. Each point at the Pareto front represents a certain trade-off between the delay and the throughput. Explicitly, the network has the minimum delay at point 1, but suffers from the lowest throughput, while the network at point 5 has highest throughput but also the maximum delay. The points at the orange solid line are the OF values computed by using single-objective optimization. Moreover, we observe from Fig. 5(b) that the proposed approach finds the complete Pareto front, while the naive approach finds four points at the nondominated front obtained, but only two of them are at the Pareto front. This is because infeasible paths might be searched in the naive approach while the proposed approach is capable of controlling its search within the valid solution space. Note that the nondominated fronts found by the proposed approach and the naive approach are based on the same initial population.

Fig. 6 illustrates two further small-scale networks generated by randomly adding further nodes into Network-1 of Fig. 5(a), which are called Network-2 and Network-3, respectively. By computation, we can obtain that Network-2 of Fig. 6(a) contains 13 nodes, 94 edges and $561,070$ possible simple paths from s to d, whilst Network-3 of Fig. 6(c) contains 14 nodes, 118 edges and $8,724,558$ simple paths. We can see that the solution spaces of Network-2 and Network-3 in Fig. 6 are much greater than that of Network-1 in Fig. 5(a). For avoiding obfuscating details, the individuals are not shown in Fig. 6(b) and Fig. 6(d). Furthermore, in these networks 500 generations are used for increasing the probability of finding the Pareto front. Observe from Fig. 6(b) and Fig. 6(d) that the proposed approach outperforms the naive approach in both networks. Furthermore, from Fig. 6(b), we can find that there are two close points at the Pareto front, which may limit the success rate of the proposed approach, because the goal of the nondorminaed sorting is to pursue a uniformly spread over the Pareto optimal front [16].

Fig. 7 investigates the performance of the average fraction $\bar{\gamma}$ as well as the success rate δ of Network-1 characterized in 5(a) over 100 runs, where the initial population is generated randomly for each run and the same initial populations are used for both the proposed approach and the naive approach. Three different pairs of population sizes and generations $(P, G) = (20, 500), (40, 200)$ and $(100, 100)$ are considered, all having 10,000 function evaluations in total. Note that $\bar{\gamma}$ is the average of γ calculated over 100 runs, representing the average Pareto front acquisition fraction for the nondominated front obtained. Naturally, the larger of $\bar{\gamma}$ the higher fraction of the Pareto front is obtained. The first two figures of Fig. 7, i.e. Fig. 7(a) and Fig. 7(b), characterizing the proposed approach using different population sizes convergence to the Pareto front (i.e. $\bar{\gamma} = 1$ and $\delta = 1$) upon increasing generations. Fig. 7(c) and Fig. 7(d) investigate the impact of the algorithmic complexity on the performance of the resultant front, where the search ratio τ is defined as the number of the function evaluations divided by the total number of possible solutions. Thus, we can see that the maximum of the search ratio $\tau_{\max}$ approaches 18%. It should be noted that the number of function evaluations over distinct individuals might be smaller than the number of solutions it evaluated, since some repeated solutions may be generated by the genetic operators. Observe from Fig. 7(c) and Fig. 7(d), that $\bar{\gamma}$ and δ converge to one when τ approaches 8%, which indicates that the proposed approach has the potential of locating all optimal solutions of the Pareto front, if the search ratio is high enough. Finally, we also can observe from Fig. 7(c) that given a value of $\bar{\gamma}$, the proposed approach is capable of finding more points on the Pareto front than the naive approach using the same initial population. This is because that the proposed approach exploits the search by avoiding the introduction of infeasible solutions into the population.

Fig. 8 illustrates the $\bar{\gamma}$ and δ vs. the number of populations relationship in Network-2 and Network-3 of Fig. 6 using $(P, G) = (100, 500)$. This scenario has an increased complexity due to the larger solution space. Thus, 50,000 functions evaluations are performed in total. Note that both

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJVT.2021.3095467, IEEE Open Journal of Vehicular Technology



(a) Network-2 of order 13.

(b) Nondominated fronts of Network-2.

(c) Network-3 of order 14.

(d) Nondominated fronts of Network-3.

Figure 6: Results on different networks.

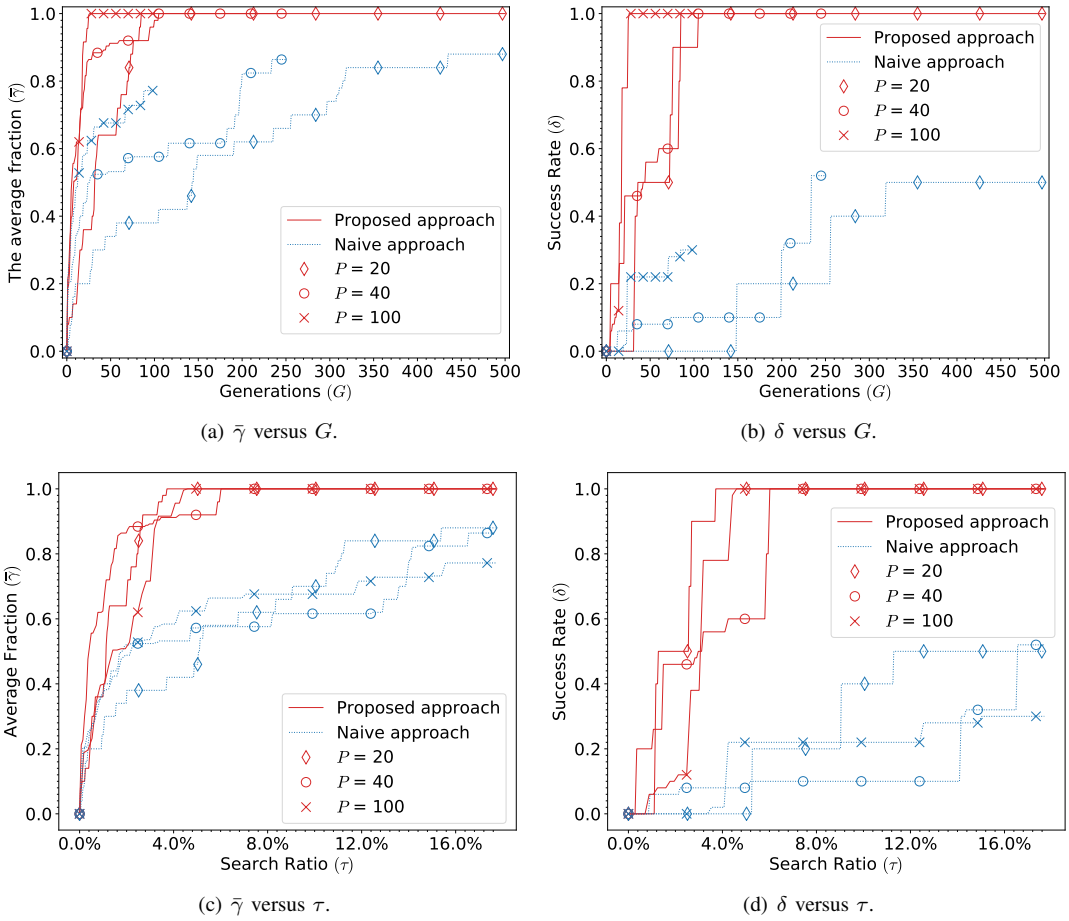
$\bar{\gamma}$ and δ represent a ratio belonging to $[0, 1]$, hence they can share the same y-axis. From Fig. 8, we can see that the success rate of the proposed approach reaches about 60% and 50% in terms of finding the Pareto front in Network-2 and Network-3, respectively. Furthermore, comparing the curves of $\bar{\gamma}$ to the curves of δ both in Fig. 7 and in Fig. 8, we can see that the evolution of $\bar{\gamma}$ is smoother than that of δ. This is because $\bar{\gamma}$ represents the number of points that are found at the Pareto front, while δ concerns the final results. For instance, in Fig. 8(a), the success rate of the proposed approach does not increase beyond 40 generations, while the values of $\bar{\gamma}$ increase gradually until 100 generations. This means that more points that are on the Pareto front are found by the proposed approach during the iterations from 40 generations to 100 generations. Moreover, we can see from Fig. 8 that the performance of Network-3 is better than that of Network-2. This trend is observed for several reasons, such as the stochastic nature of the genetic operators as well as the number and the distribution of the points at the Pareto front etc. Finally, by jointly observing the results of Fig. 7 and Fig. 8, we can infer that the proposed approach requires less generations than the naive approach for attaining a specific target performance. Explicitly, it implies a lower complexity.

B. Flight Data Based Results

In this subsection, we evaluate the performance of the proposed approach relying on real historical flight data. Explicitly, we use three datasets, termed as Data-1, Data-2 and Data-3, respectively, which contain historical flight data of the five busiest TransAtlantic airlines in the North-Atlantic region, i.e., Delta Airline, United Airline, American Airline, British Airways and Lufthansa. Specifically, these datasets contain the historical flight information of the area recorded at sampling intervals of 10s, where each entry of the flight contains the following information: timestamp, longitude, latitude, altitude and speed. The entries of Data-1 and Data-2 were collected from 00:00 on 24 Dec. 2017 to 00:00 on 26 Dec.2017, thus there are 17,281 entries for each flight. Data-1 having 57 flights only contains the TransAtlantic flights between LHR Airport and JFK Airport, while Data-2 having 381 flights contains all TransAtlantic flights of the five busiest TransAtlantic airlines. Moreover, the entries of Data-3 were collected from 00:00 on 29 Jun. 2018 to 00:00 on 30 Jun. 2018, which is the busiest day of the year having the most flights. Specifically, Data-3 contains 649 flights and 8,641 entries for each flight.

Fig. 9 illustrates some of the associated network topological characteristics versus the flight distance in Data-1, Data-2 and Data-3, respectively. Observe from Fig. 9(a), that the number of both the nodes and edges of the networks in Data-3 is about twice as higher as those in Data-2, while the networks in Data-1 have the least number of nodes and edges. We can also find that in Data-1, the target flight is unable to connect to the SN on the ground when the flight distance

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJVT.2021.3095467, IEEE Open Journal of Vehicular Technology



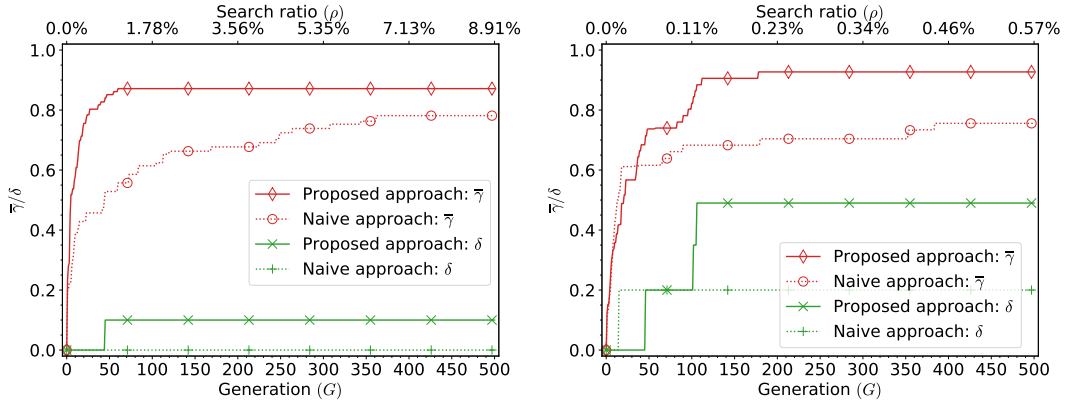
(a) $\bar{\gamma}$ versus G.

(b) δ versus G.

(c) $\bar{\gamma}$ versus τ.

(d) δ versus τ.

Figure 7: Performance of different approaches for Network-1 with the population size $P = \{20, 40, 100\}$.



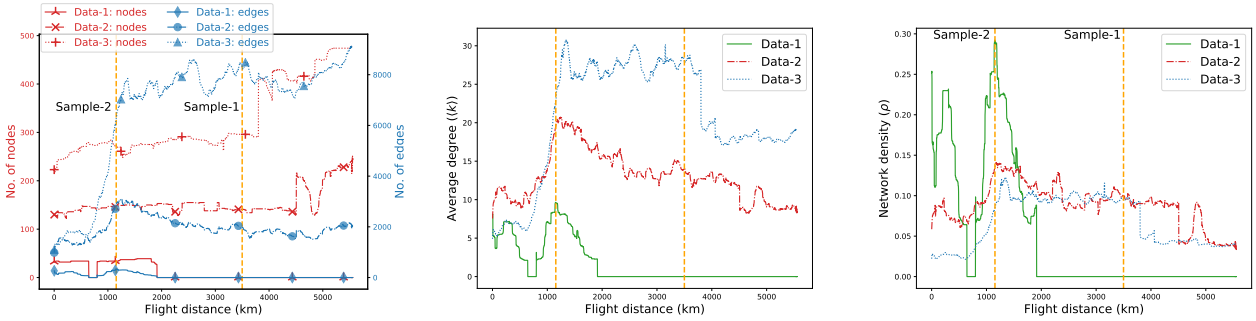
(a) Performance of Network-2.

(b) Performance of Network-3.

Figure 8: Results of Network-2 and Network-3, respectively, where $P = 100$.

is higher than 1912km, since there is no path connecting them. Furthermore, Fig. 9(b) shows the average degree of the networks over different flight distances, which reflects the average grade of connectivity for a network. A larger average degree indicates a higher probability of having more paths between two nodes. Fig. 9(c) shows the density of networks in different datasets, which reflects the connection rate of the networks at different time instants. Observe from Fig. 9(c), that the networks generated during the flight of the target plane are relatively sparse, especially for Data-2 and Data-3, where the maximum network density is below 0.15. From these characteristic information in Fig. 9, we can see that these networks become complex and non-trivial since patterns of connection between their nodes are neither purely regular nor purely random. Consequently, the full solution space becomes excessively large, hence it is computationally impractical to find the true Pareto front for these networks. Therefore, we use a population size 100 and 500 generations for generating a series of nondominated solutions to approach the Pareto front. Moreover, since the true Pareto front is unknown due to the excessive number of solutions, for limiting the objective space, the points associated with the minimum delay and the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJVT.2021.3095467, IEEE Open Journal of Vehicular Technology



(a) Data-1 (b) Data-2 (c) Data-3

Figure 9: Network characteristics in the Data-1, Data-2 and Data-3.

maximum throughput are computed, which are obtained by optimizing the single-objective function $D(\mathbf{X})$ and $C(\mathbf{X})$, respectively. Specifically, the optimum of $D(\mathbf{X})$ can be readily obtained by means of the conventional shortest path algorithm such as Dijkstra's algorithm. By contrast, as $C(\mathbf{X})$ is a max-min function, hence the single-objective optimization problem of $C(\mathbf{X})$ becomes a maximum-capacity path problem [33], which can be solved by the branch and bound algorithm. Therefore, the boundary of the objective space is also plotted in the figures.

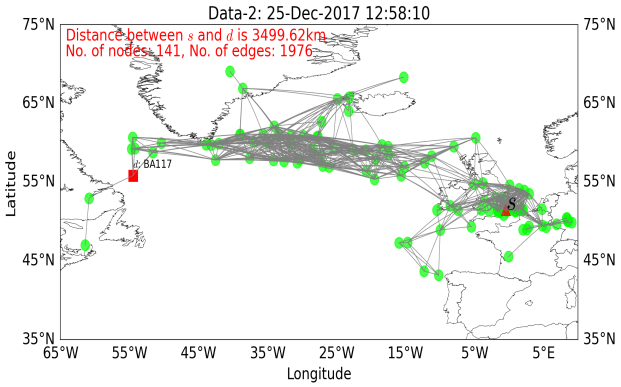
Fig. 10 illustrates the topology of the network and the non-dominated front obtained using different approaches, where the network features can be found from Fig. 9 marked by Sample-1. Fig. 10(a) and Fig. 10(c) show that the network of Data-2 at Sample-1 contains 141 nodes and 1976 edges, while the network of Data-3 contains 296 nodes and 8274 edges. This indicates that the number of aircraft at the busiest days is almost twice the number of that on the quietest days. Fig. 10(b) and Fig. 10(d) illustrate the nondominated front obtained for each dataset, where the points acquired by the single-objective optimization are marked by green stars. We can observe that the optimum single-component $C(\mathbf{X})$ may be dominated by the point obtained by the proposed two-component approach. This is because the path having the maximum throughput may suffer from a high delay, since some paths probably share the same maximum throughput in the AANET due to the non-monotonic nature of $C(\mathbf{X})$. Furthermore, in Fig. 10(b) and Fig. 10(d) the nondominated front of each approach is constituted by a union set of the results collected over 10 runs, where the dominated points are removed from the union set. Finally, we can see from Fig. 10(b) and Fig. 10(d) that the fronts found by the proposed approach in the both networks exhibits a better performance than that of the naive approach.

Fig. 11 illustrates the nondominated front obtained for the different datasets at Sample-2 of Fig. 9, where Data-1 is also involved, since the source BS on the ground is now able to communicate to the target aircraft via the AANET. Specifically, Fig. 11(b)-Fig. 11(f) illustrate the front obtained for the three datasets at Sample-2, while Fig. 11(a)-Fig. 11(e) illustrate the corresponding network topology. Furthermore, the same simulation configurations are used as in Fig. 10. We can see from Fig. 11(b) that the optima of $D(\mathbf{X})$ and $C(\mathbf{X})$ are overlapped in the network of Data-1, while both the naive approach and the proposed approach converged to the global optimum. Observe from Fig. 11(d), that the points obtained by the single-objective optimization are almost covered by the nondominated front obtained by the proposed approach. Furthermore, the proposed approach also outperforms the naive approach in terms of the nondominated front obtained. Finally, we can observe from Fig. 11(f) that the point O is also achievable in the network of Data-3, which is found by the proposed approach. This indicates that the proposed approach has the potential of locating the global optimum provided that it is feasible. Moreover, observe from Fig. 11(b) and Fig. 11(f), that there exist an optimal routing path that having the minimum delay and the maximum throughput simultaneously, which indicates that the relationship of $D(\mathbf{X})$ and $C(\mathbf{X})$ depends on the network's topology due to the substantial non-trivial topological features of these complex networks.

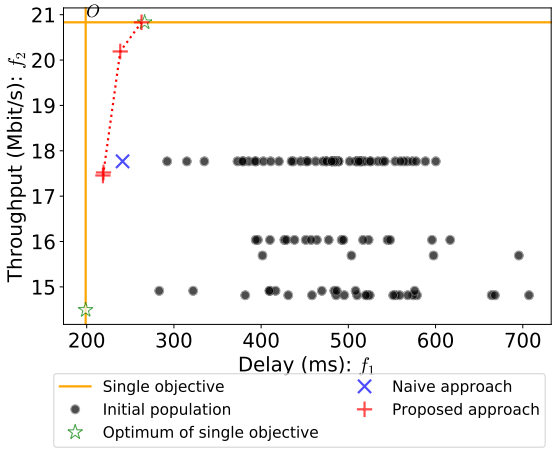
VII. Conclusions

In this paper, we have designed a multi-objective routing solution for enhancing multiple performance metrics of in-flight connectivity in terms of the total delay and the throughput. Therefore, the MOCOP formulated concurrently deals with two discrete, non-continuous problems, which results in a number of Pareto optimal solutions. As a consequence, obtaining the Pareto set imposes an excessive computational complexity and it is often infeasible, especially for networks having numerous possible paths. We employed the NSGA-II for generating an approximation of the Pareto optimal set of the MOCOP formulated. Our simulation results revealed that a set of beneficial trade-off solutions can be obtained for providing a flexible selection of in-flight connections by solving the multi-objective routing problem formulated in terms of the delay and the throughput both for the simulated data and for our historical flight data. A promising extension of this work is to conceived a multi-task learning algorithm by exploiting the parallel nature of the NSGA-II for generating a better approximation of the Pareto set. Adopting more sophisticated performance metrics such as HV and IGD+, are capable of reflecting more comprehensive aspects of the solutions obtained for MOOPs, which constitutes another promising future research direction. Moreover, for expediting the search process of generating the Pareto set, quantum inspired algorithms having inherent parallelism also constitute a promising future research direction.

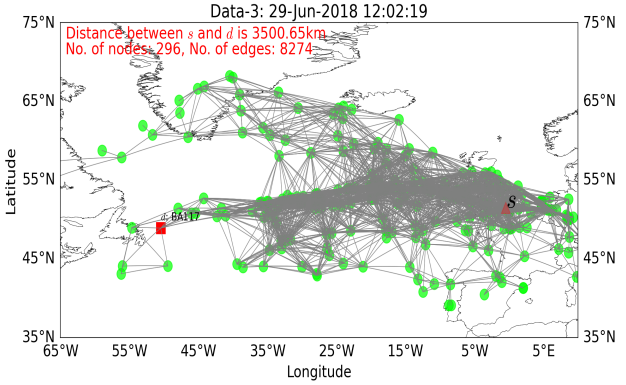
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJVT.2021.3095467, IEEE Open Journal of Vehicular Technology



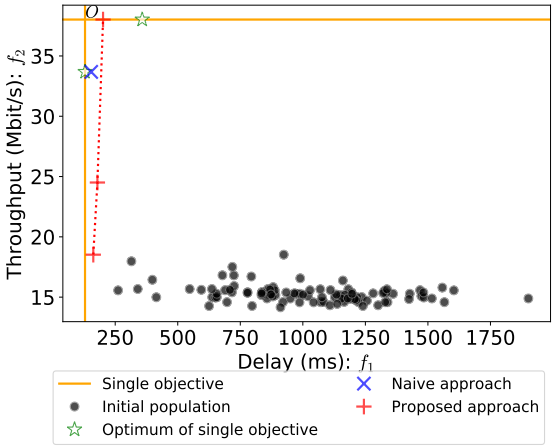
(a) Network topology of Data-2 at Sample-1.



(b) Nondominated front obtained at Data-2.



(c) Network topology of Data-3 at Sample-1.



(d) Nondominated front obtained at Data-3.

Figure 10: The network's topology and its throughput vs. delay solutions at Sample-1 of Fig. 9 in Data-2 and Data-3.

Appendix A: Procedures of Nondominated Sorting and Selection

Fast nondominated sorting and selection constitute a pair of pivotal operators in NSGA-II [16], which are designed by **Algorithm A.1** and **Algorithm A.2**, respectively.

Appendix B: Proof of Proposition 1

Let $\mathcal{F}^{PF} = \mathcal{Y}_2$ and $\tilde{\mathcal{F}}^* = \mathcal{Y}_1$, respectively. From **Definition 3**, we have

$$\mathcal{C}(\tilde{\mathcal{F}}^*, \mathcal{F}^{PF}) = \frac{|\{\mathbf{y}_2 \in \mathcal{F}^{PF} \mid \exists\, \mathbf{y}_1 \in \tilde{\mathcal{F}}^* : \mathbf{y}_1 \preceq \mathbf{y}_2\}|}{|\mathcal{F}^{PF}|}. \quad \text{(B.1)}$$

As $\mathcal{F}^{PF}$ is the Pareto front, any point $\mathbf{y}_1 \in \tilde{\mathcal{F}}^*$ weakly dominates the point $\mathbf{y}_2 \in \mathcal{F}^{PF}$ if and only if we have $\mathbf{y}_1 = \mathbf{y}_2$. Therefore, the nominator of $\mathcal{C}(\tilde{\mathcal{F}}^*, \mathcal{F}^{PF})$ becomes the number of all points that belong to both the sets $\mathcal{F}^{PF}$ and $\tilde{\mathcal{F}}^*$. This corresponds to the number of points in the intersection of $\mathcal{F}^{PF}$ and $\tilde{\mathcal{F}}^*$, i.e. $|\tilde{\mathcal{F}}^* \cap \mathcal{F}^{PF}|$. Correspondingly, (B.1) can be equivalently expressed as

$$\mathcal{C}(\tilde{\mathcal{F}}^*, \mathcal{F}^{PF}) = \frac{|\tilde{\mathcal{F}}^* \cap \mathcal{F}^{PF}|}{|\mathcal{F}^{PF}|}. \quad \text{(B.2)}$$

References

[1] M. Vondra, E. Dinc, M. Prytz, M. Frodigh, D. Schupke, M. Nilson, S. Hofmann, and C. Cavdar, "Performance study on seamless DA2GC for aircraft passengers toward 5G," *IEEE Commun. Mag.*, vol. 55, no. 11, pp. 194–201, Nov. 2017.

[2] M. Vondra, M. Ozger, D. Schupke, and C. Cavdar, "Integration of satellite and aerial communications for heterogeneous flying vehicles," *IEEE Network*, vol. 32, no. 5, pp. 62–69, Sep. 2018.

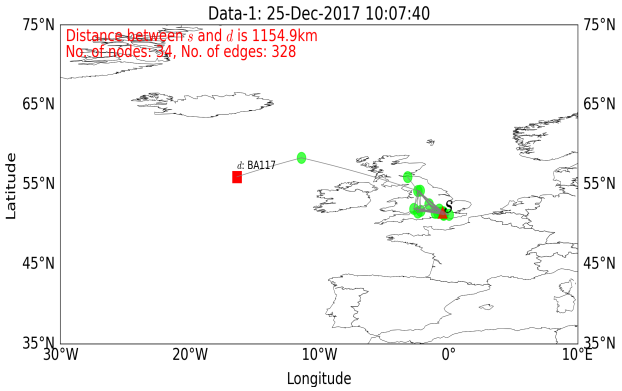
[3] J. Zhang, T. Chen, S. Zhong, J. Wang, W. Zhang, X. Zuo, R. G. Maunder, and L. Hanzo, "Aeronautical *ad hoc* networking for the Internet-above-the-clouds," *Proc. IEEE*, vol. 107, no. 5, pp. 868–911, May 2019.

[4] D. Medina, F. Hoffmann, F. Rossetto, and C. Rokitansky, "A geographic routing strategy for North Atlantic in-flight Internet access via airborne mesh networking," *IEEE/ACM Trans. Netw.*, vol. 20, no. 4, pp. 1231–1244, Aug. 2012.

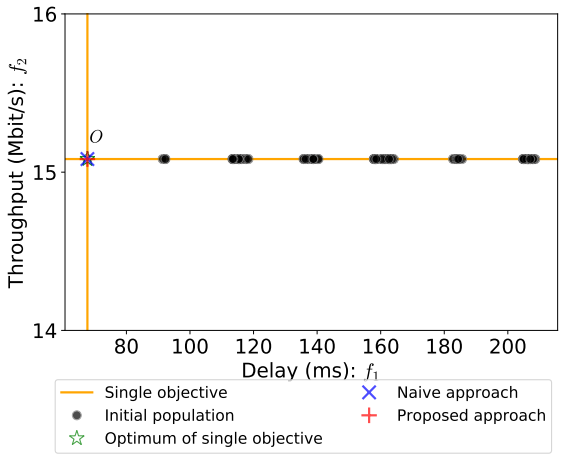
[5] B. Cheng, A. Coyle, S. McGarry, I. Pedan, L. Veytser, and J. Wheeler, "Characterizing routing with radio-to-router information in a heterogeneous airborne network," *IEEE Trans. Wireless Commun.*, vol. 12, no. 8, pp. 4183–4195, Aug. 2013.

[6] Q. Vey, S. Puechmorel, A. Pirovano, and J. Radzik, "Routing in aeronautical *ad-hoc* networks," in *IEEE/AIAA Digital Avionics Systems Conference (DASC)*, Sep. 2016, pp. 1–10.

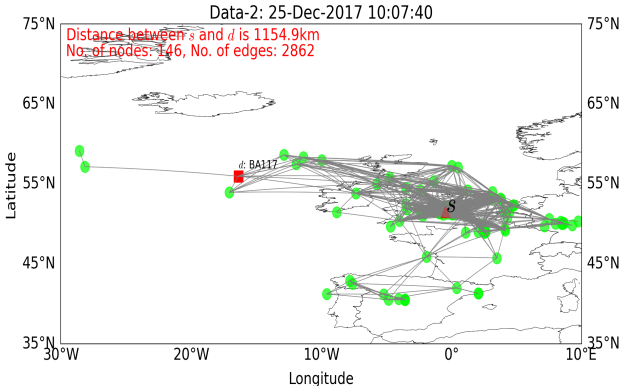
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJVT.2021.3095467, IEEE Open Journal of Vehicular Technology



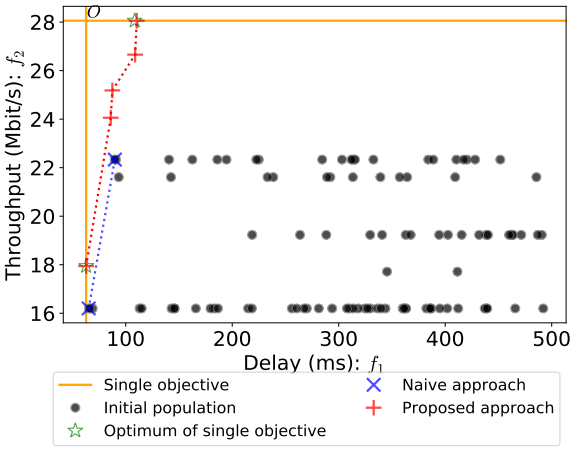
(a) Network of Data-1



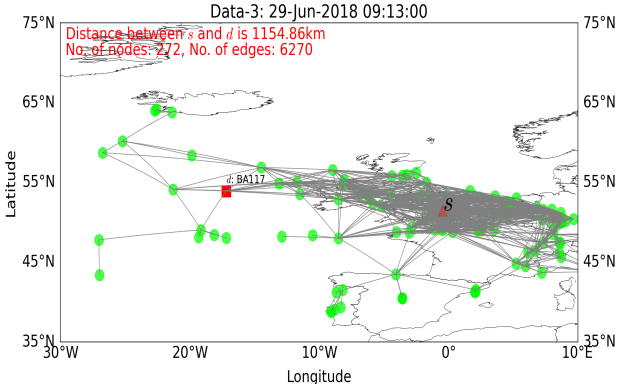
(b) Obtained front in Data-1



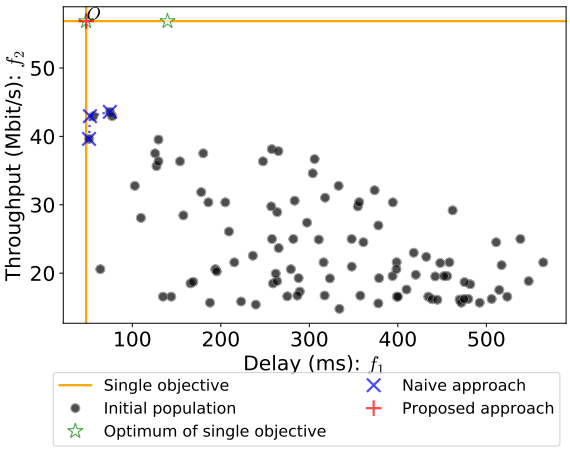
(c) Network of Data-2



(d) Obtained front in Data-2



(e) Network of Data-3



(f) Obtained front in Data-3

Figure 11: The network's topology and its throughput vs. delay solutions at Sample-2 of Fig. 9 in Data-1, Data-2 and Data-3.

[7] M. Sbeiti, N. Goddemeier, D. Behnke, and C. Wietfeld, "PASER: Secure and efficient routing approach for airborne mesh networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 3, pp. 1950–1964, March 2016.

[8] Q. Luo and J. Wang, "Multiple QoS parameters-based routing for civil aeronautical *ad hoc* networks," *IEEE Internet Things J.*, vol. 4, no. 3, pp. 804–814, 2017.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJVT.2021.3095467, IEEE Open Journal of Vehicular Technology

Algorithm A.1: Fast nondominated sorting algorithm

Function `Fast-nondominated-sort(`$\mathcal{C}$`)`**:**

$\mathcal{F}_1 = \emptyset, \mathbf{n} = \mathbf{0}_{|\mathcal{C}|}$;
for $p \in \mathcal{C}$ **do**
  $\mathcal{S}_p = \emptyset$;
  for $q \in P$ **do**
    if $p \prec q$ **then** /* p dominates q */
      $\mathcal{S}_p = \mathcal{S}_p \cup \{q\}$; /* Add q to the set of solutions dominated by p */
    else if $q \prec p$ **then**
      $\mathbf{n}[p] = \mathbf{n}[p] + 1$; /* Increase the nomination counter of p */
  if $\mathbf{n}[p] = 0$ **then** /* p belongs to the first front */
    $\mathcal{F}_1 = \mathcal{F}_1 \cup \{p\}$;
$i = 1$; $\mathcal{F} = \emptyset$;
while $\mathcal{F}_i \neq \emptyset$ **do**
  $\mathcal{Q} = \emptyset$; /* $\mathcal{Q}$ is used to store the members of the next front */ **for** $p \in \mathcal{F}_i$ **do**
    for $q \in \mathcal{S}_p$ **do**
      $\mathbf{n}[q] = \mathbf{n}[q] - 1$;
      if $\mathbf{n}[q] = 0$ **then**
        $\mathcal{Q} = \mathcal{Q} \cup \{q\}$;
  $i = i + 1$;
  $\mathcal{F}_i = \mathcal{Q}$;
return *The union of the sorted fronts* $\mathcal{F} = \{\mathcal{F}_1, \mathcal{F}_2, \cdots\}$

Algorithm A.2: Selection

Function `Selection(`$\mathcal{F}$`)`**:**

$\mathcal{P}_{t+1} = \emptyset$;
repeat
  for $i = 1$ **to** $|\mathcal{F}_i|$ **do**
    if $|\mathcal{P}_{t+1}| + |\mathcal{F}| < P$ **then**
      $\mathcal{P}_{t+1} = \mathcal{P}_{t+1} \cup \mathcal{F}_i$
    else
      $\mathcal{D}_{cd}$ = crowding_distance($\mathcal{F}_i$);
      Sort $\mathcal{F}_i$ based on $\mathcal{D}_{cd}$ in descending order;
      $\mathcal{P}_{t+1} = \mathcal{P}_{t+1} \cup \mathcal{F}_i[0 : P - |\mathcal{P}_{t+1}|]$;
until $|\mathcal{P}_{t+1}| + |\mathcal{F}_t| \geq P$;
return $\mathcal{P}_{t+1}$

[9] P. K. Biswas, S. J. Mackey, D. H. Cansever, M. P. Patel, and F. B. Panettieri, "Context-aware smallworld routing for wireless *ad-hoc* networks," *IEEE Trans. Commun.*, vol. 66, no. 9, pp. 3943–3958, Sep. 2018.

[10] A. Al-Saadi, R. Setchi, Y. Hicks, and S. M. Allen, "Routing protocol for heterogeneous wireless mesh networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 12, pp. 9773–9786, Dec 2016.

[11] P. Zhou, J. Xu, W. Wang, Y. Hu, D. O. Wu, and S. Ji, "Toward optimal adaptive online shortest path routing with acceleration under jamming attack," *IEEE/ACM Trans. Netw.*, vol. 27, no. 5, pp. 1815–1829, Oct 2019.

[12] E. Zitzler, M. Laumanns, and S. Bleuler, "A tutorial on evolutionary multiobjective optimization," in *Metaheuristics for Multiobjective Optimisation*, X. Gandibleux, M. Sevaux, K. Sörensen, and V. T'kindt, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 3–37.

[13] C. A. C. Coello, G. B. Lamont, D. A. Van Veldhuizen *et al.*, *Evolutionary algorithms for solving multi-objective problems*. Springer, 2007, vol. 5.

[14] K. Vijayalakshmi and P. Anandan, "A multi objective tabu particle swarm optimization for effective cluster head selection in wsn," *Cluster Computing*, vol. 22, no. 5, pp. 12 275–12 282, 2019.

[15] S. Bandyopadhyay, S. Saha, U. Maulik, and K. Deb, "A simulated annealing-based multiobjective optimization algorithm: Amosa," *IEEE Trans. Evol. Comput.*, vol. 12, no. 3, pp. 269–283, 2008.

[16] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.

[17] H. Yetgin, K. T. K. Cheung, and L. Hanzo, "Multi-objective routing optimization using evolutionary algorithms," in *IEEE Proc. of Wireless Commun. and Net. Conf. (WCNC)*, 2012, pp. 3030–3034.

[18] D. Alanis, P. Botsinis, Z. Babar, H. V. Nguyen, D. Chandra, S. X. Ng, and L. Hanzo, "A quantum-search-aided dynamic programming framework for Pareto optimal routing in wireless multihop networks," *IEEE Trans. Commun. Technol.*, vol. 66, no. 8, pp. 3485–3500, 2018.

[19] C. Zhang, C. Li, and Y. Chen, "A Markov model for batch-based opportunistic routing in multi-hop wireless mesh networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 12, pp. 12 025–12 037, 2018.

[20] F. Hoffmann, D. Medina, and A. Wolisz, "Joint routing and scheduling in mobile aeronautical $ad hoc$ networks," *IEEE Trans. Veh. Technol.*, vol. 62, no. 6, pp. 2700–2712, Jul. 2013.

[21] L. G. Valiant, "The complexity of enumeration and reliability problems," *SIAM Journal on Computing*, vol. 8, no. 3, pp. 410–421, 1979.

[22] S. Hofmann, A. E. Garcıa, D. Schupke, H. E. Gonzalez, and F. H. Fitzek, "Connectivity in the air: Throughput analysis of air-to-ground systems," in *IEEE Proc. of International Commun. Conf. (ICC)*, May 2019, pp. 1–8.

[23] M. Sikora, J. N. Laneman, M. Haenggi, D. J. Costello, and T. E. Fuja, "Bandwidth- and power-efficient routing in linear wireless networks," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2624–2633, June 2006.

[24] T. Back, *Evolutionary algorithms in theory and practice: evolution strategies, evolutionary programming, genetic algorithms*. Oxford university press, 1996.

[25] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach," *IEEE Trans. Evol. Comput.*, vol. 3, no. 4, pp. 257–271, 1999.

[26] M. Munetomo, Y. Takai, and Y. Sato, "A migration scheme for the genetic adaptive routing algorithm," in *IEEE Int. Conf. Systems, Man, and Cybernetics (Cat. No.98CH36218)*, vol. 3, 1998, pp. 2774–2779.

[27] C. Wook and A. R. S. Ramakrishna, "A genetic algorithm for shortest path routing problem and the sizing of populations," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.

[28] C. Blum and A. Roli, "Metaheuristics in combinatorial optimization: Overview and conceptual comparison," *ACM Comput. Surv.*, vol. 35, no. 3, p. 268–308, Sep. 2003.

[29] E. Zitzler, *Evolutionary algorithms for multiobjective optimization: Methods and applications*. Citeseer, 1999, vol. 63.

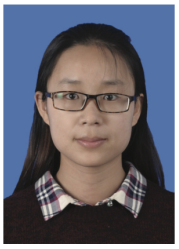
[30] R. Sedgewick, *Algorithms in C, Part 5: Graph Algorithms*. Pearson Education, 2001.

[31] B. Roberts and D. P. Kroese, "Estimating the number of $s-t$ paths in a graph." *J. Graph Algorithms Appl.*, vol. 11, no. 1, pp. 195–214, 2007.

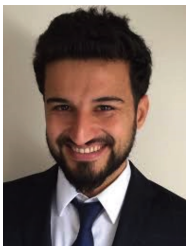
[32] M. Li and X. Yao, "Quality evaluation of solution sets in multiobjective optimisation: A survey," *ACM Comput. Surv.*, vol. 52, no. 2, Mar. 2019. [Online]. Available: https://doi.org/10.1145/3300148

[33] M. Peinhardt and V. Kaibel, "On the bottleneck shortest path problem," Technical Report ZIB-Report 06-22, Konrad-Zuse-Zentrum für Informationstechnik Berlin, Tech. Rep., 2006.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJVT.2021.3095467, IEEE Open Journal of Vehicular Technology



Jingjing Cui (M'18) received the Ph.D from Southwest Jiaotong University, Chengdu, China 2018. She is currently a research fellow with the School of Electronics and Computer Science, University of Southampton, UK. She was a research assistant with the School of Electronic Engineering and Computer Science, Queen Mary University of London, UK, from May 2018 to May 2019. Her research interests include classical & quantum optimization theory and algorithm design, machine learning for wireless networks, and quantum communications. She received the Exemplary Reviewers of the IEEE Transactions on Communications in 2019 and IEEE Communication Letters in 2018. She has also served as a TPC Member for IEEE conferences, such as ICC and IEEE GLOBECOM.



Halil Yetgin received the B.Eng. degree in computer engineering from Selcuk University, Turkey, in 2008, the M.Sc. Degree in wireless communications from the University of Southampton, U.K., in 2010, and the Ph.D. degree in wireless communications from the Next Generation Wireless Research Group, University of Southampton in 2015. He is an Assistant Professor of the Department of Electrical and Electronics Engineering, Bitlis Eren University, Turkey, and a Research Fellow of the Department of Communication Systems, Jozef Stefan Institute, Ljubljana, Slovenia. His research interests include the development of intelligent communication systems, energy efficient cross-layer design, resource allocation of the future wireless communication networks and machine learning for wireless networks. He was a recipient of the Full Scholarship granted by the Republic of Turkey, Ministry of National Education. He is an Associate Editor of IEEE ACCESS. He was a TPC member for IEEE VTC-2018, VTC-2019, VTC-2020, and IEEE Globecom-2020.

Dong Liu (d.liu@soton.ac.uk) received his B. Eng. and Ph.D. degree from Beihang University (BUAA), Beijing, China in 2013 and 2019, respectively. He is currently a research fellow at the University of Southampton, UK. His research interests lie in the area of mobile AI, mobile edge caching, and space-air-ground integrated networks.

Jiankang Zhang (S'08-M'12-SM'18) is a Senior Lecturer at Bournemouth University. Prior to joining in Bournemouth University, he was a senior research fellow at University of Southampton, UK. Dr Zhang was a lecturer from 2012 to 2013 and then an associate professor from 2013 to 2014 at Zhengzhou University. His research interests are in the areas of aeronautical communications, aeronautical networks, evolutionary algorithms and edge computing. He serves as an Associate Editor for IEEE ACCESS.

Soon Xin Ng (S'99-M'03-SM'08) received the B.Eng. degree (First class) in electronic engineering and the Ph.D. degree in telecommunications from the University of Southampton, Southampton, U.K., in 1999 and 2002, respectively. From 2003 to 2006, he was a postdoctoral research fellow working on collaborative European research projects known as SCOUT, NEWCOM and PHOENIX. Since August 2006, he has been a member of academic staff in the School of Electronics and Computer Science, University of Southampton. He was involved in the OPTIMIX and CONCERTO European projects as well as the IU-ATC and UC4G projects. He was the principal investigator of an EPSRC project on "Cooperative Classical and Quantum Communications Systems". He is currently a Professor of Next Generation Communications at the University of Southampton.

His research interests include adaptive coded modulation, coded modulation, channel coding, space-time coding, joint source and channel coding, iterative detection, OFDM, MIMO, cooperative communications, distributed coding, quantum communications, quantum error correction codes, joint wireless-and-optical-fibre communications, game theory, artificial intelligence and machine learning. He has published over 260 papers and co-authored two John Wiley/IEEE Press books in this field.

He is a Senior Member of the IEEE, a Fellow of the Higher Education Academy in the UK, a Chartered Engineer and a Fellow of the IET. He acted as TPC/track/workshop chairs for various conferences. He serves as an editor of Quantum Engineering. He was a guest editor for the special issues in IEEE Journal on Selected Areas in Communication as well as editors in the IEEE Access and the KSII Transactions on Internet and Information Systems. He is one of the Founders and Officers of the IEEE Quantum Communications & Information Technology Emerging Technical Subcommittee (QCIT-ETC).

L ajos Hanzo (http:www-mobile.ecs.soton.ac.uk, https:en.wikipedia.orgwikiLajos_Hanzo) (FIEEE'04, Fellow of the Royal Academy of Engineering F(REng), of the IET and of EURASIP), received his Master degree and Doctorate in 1976 and 1983, respectively from the Technical University (TU) of Budapest. He was also awarded the Doctor of Sciences (DSc) degree by the University of Southampton (2004) and Honorary Doctorates by the TU of Budapest (2009) and by the University of Edinburgh (2015). He is a Foreign Member of the Hungarian Academy of Sciences and a former Editor-in-Chief of the IEEE Press. He has served several terms as Governor of both IEEE ComSoc and of VTS. He has published 1900+ contributions at IEEE Xplore, 19 Wiley-IEEE Press books and has helped the fast-track career of 123 PhD students. He holds the Chair of Telecommunications and directs the research of Next-Generation Wireless at the University of Southampton, UK.