Decentralized Collaborative Decision-Making for Topology Building in Mobile Ad-Hoc Networks

Philipp Helle, Sergio Feo-Arenis, Kevin Shortt, and Carsten Strobel
Airbus Central R&T, Germany
email: {philipp.helle, sergio.feo-arenis, kevin.shortt, carsten.strobel}@airbus.com

***Abstract*—In ad-hoc networks with a large number of mobile participants, i.e. nodes, centralized decision-making for building the network topology becomes impracticable. At Airbus, an architecture for an ad-hoc network amongst flying platforms (e.g. commercial aircraft, satellites, high altitude platforms) incorporating free-space optical communication links has been developed. This paper presents an algorithm for a decentralized collaborative decision-making process for building the topology of this mobile ad-hoc network. The design, the implementation in an agent-based modelling tool, and the evaluation of this so-called ConOps Agent algorithm using real flight data are described.**

***Index Terms*—FANET, MANET, VANET, ConOps, FSO, ABM**

I. Introduction

Mobile Ad-hoc Networks (MANETs) [3], [6] are wireless mobile nodes that cooperatively form a network without infrastructure. Vehicular Ad-hoc Networks (VANETs) [1] are a specific subclass where moving vehicles act as either a node, or a router to exchange messages between vehicles, or an access point.

At Airbus Central R&T (CRT), a novel architecture for a Flying Ad Hoc Network (FANET), a special kind of VANET, has been developed that effectively extends the reach of traditional, high capacity optical fiber networks. A FANET is classically defined as one airborne networking domain characterized by a wireless ad-hoc network of flying platforms connected via wireless communication links [4]. While FANETs are typically associated with groups of UAVs, the CRT FANET encompasses networks of a variety of different flying elements such as commercial aircraft, satellites, high-altitude pseudo-satellites (HAPS) and others. Furthermore, in order to realize the full capabilities of fiber networks beyond the contrains of the Earth's surface, the architecture described here incorporates the use of point-to-point links, specifically, Free-space Optical Communication (FSO) links.

To enable the CRT FANET, a Concept of Operations (ConOps) was first defined that describes how the overall network operates within a telecommunications network context. By defining the ConOps in this fashion, the FANET operations could then be seamlessly integrated into existing telecommunication operations. For this integration to be as transparent as possible, an algorithm was developed that establishes the overall network topology, i.e. how the network manifests itself amongst the multitude of moving nodes. This algorithm resides in each node in the network, constantly monitoring for changes in the topology and reconfiguring the topology as necessary. It is this ConOps Agent - more specifically, its design, implementation and evaluation using Agent-based Modeling (ABM) - that is the focus of this paper.

This paper is structured as follows: Section II provides the motivation for the presented work in more detail. Section III explains the ConOps for the architecture and the developed decentralized ConOps Agent. Section IV provides a description of how the ConOps Agent was evaluated using ABM together with the evaluation results before Section V provides a final conclusion.

II. Motivation

Today, FSO as a commercial technology is used primarily in static configurations for high bandwidth, wireless communication links [16]. In research there are, however, investigations to use FSO in mobile applications in the form of hybrid FSO/Radio Frequency (RF) MANETs [7], where FSO- and RF-based communication technologies complement each other and mitigate each other's weaknesses.

The most significant advantage of FSO over RF based technology is extremely high throughput over long link distances, i.e. several Gigabit per second (Gbps) in currently fielded systems with link distances of a kilometer or more. The main disadvantages are the requirement for optical links to maintain line-of-sight (LOS) and the fact that the achievable FSO link distance and throughput are a function of atmospheric conditions. These disadvantages are less significant in the operating environment of airborne platforms than on ground [9].

Similar technology was developed by Google in the project Loon [15], which was recently shut down "after failing to find a sustainable business model" [14]. The main difference concerning deployment between the Loon approach and the CRT FANET architecture is that Loon was based on launching and maintaining new flying platforms, whereas the work presented here aims at using existing platforms, e.g. passenger aircraft. Another contrast is the envisioned end-users, where Loon targeted end-users on the ground while we target airline passengers, operational stakeholders, and a variety of backhaul applications. Finally, a major difference in the control strategy is that Loon aimed to develop a centralized control algorithm for establishing the network topology, whereas in FANETs, typically each network node makes decentralized decisions.

The following applications of the CRT FANET can be considered:

978-1-6654-8550-0/22/$31.00 ©2022 IEEE

- *Improving passenger connectivity:* The FANET enables the aggregation of traffic generated by hundreds of thousands of passengers with broadband connectivity per user. This capability allows for new revenue streams for the airline and for more efficient airline operations by increasing the amount of data shared between aircraft.
- *Backhaul:* The FANET can support backhaul services for mobile operators and Internet Service Providers (ISPs), providing supplemental capacity to terrestrial fiber networks, in addition to extending their reach. While satellites can act as an alternative for backhaul, the inclusion of other platforms in the FANET could facilitate a more staggered deployment strategy with lower front-end CapEx and earlier Return on Investment while also affording the opportunity to further mature the technology through shorter development cycles.
- *Improving aircraft connectivity:* The FANET can provide sufficient bandwidth to an aircraft to download data generated by any onboard sensors and telemetry data in real time. It can also be an enabler for disruptive technologies such as Single Pilot Operation (SPO) and even remote-controlled operation, which place higher requirements on communication systems for throughput and latency.

III. CONOPS AGENT

Critical to any system development is how that system will ultimately be used. This aspect of the system is captured in what is called the Concept of Operations and, within the context of the FANET, encompasses how everything interacts, from the human resources to the network management to the physical hardware. That being said, the ConOps can be seen to drive the requirements of the other the systems involved since, after all, if the way the system is used is not defined, then the sub-system requirements cannot be defined.

The ConOps is a key area for innovation as no such set of processes have been defined and developed for a heterogeneous wireless network composed of RF and optical links between an arbitrary set of moving nodes. To date, the literature on MANETs, typically begins with the assumption that the links, and hence the network, have already been established and focuses on the efficient routing of packets through that network. But what are the decision processes, either through human intervention or software-based algorithms, that lead to the initial establishment of the network in the first place? It is precisely this question that the ConOps sets out to answer.

For traditional wired networks, the network topology is determined by the physical cables connecting network nodes and is generally mostly static. If nodes are mobile, the topology is dynamic and is implicitly determined by the environment, the transmission ranges affected by the transmission power, and the node movement. In directional networks, i.e., networks that utilize steerable directional communication links, such as FSO links, a topology must be explicitly determined and managed as nodes move. Nodes have a finite number of terminals for building network links and, therefore, a finite number of directly connected neighbors. Each directional FSO terminal must be commanded to point at its intended neighbor, who also must point one of its terminals back. More importantly, each node must determine with which of its potentially many neighbors it should directly connect. This decision, with which neighboring node to form a connection, is at the heart of the ConOps Agent.

A. Design

The ConOps Agent is the logical component that hosts the ConOps logic. It receives the information needed and applies some decision rules to command actions directed at fulfilling the ConOps objectives. The ConOps Agent will be implemented by software on a controller hardware to command some actuators (e.g. FSO terminals) and provide network management information. In this section, we briefly discuss several alternatives for designing a ConOps Agent.

a) Pre-computed action plan: An initial thought is to utilize the information available *a priori* about the movement of the network nodes in order to precalculate a sequence of actions. When evaluating this alternative, we observe that in aviation, plans are mostly informative, and mainly exist for the purpose of traffic control. Both the execution times and the actual flight routes may vary greatly in order to adapt to uncertainties during operation such as unforeseen delays, changing weather conditions and emergencies. It becomes evident, that a *reactive* control strategy is necessary.

b) Centralized reactive decision-making: In order to cope with the multiple uncertainties encountered during flight operations, a design goal is to create a controller that observes the evolving situation of the network nodes and issues commands in order to change the topology such that the network goals are fulfilled. We observe three main difficulties. First, transporting information regarding the state of the network through omnidirectional and directional links appears costly. The bandwidth necessary to relay state and control information towards a controller increases with the number of nodes. At large scales, the network would be flooded with state and control traffic, leaving a diminishing portion available for actual data payloads. Second, assuming that the network state is observable, an algorithm to determine a semi-optimal configuration has likely a high computational complexity. The envisioned system comprises numerous nodes, thus making a centralized algorithm infeasible since very long execution times would be expected. An instance of this strategy was tried in the Loon [10] project, where performance bottlenecks were encountered already at network sizes on the order of hundreds of nodes. Finally, having a dedicated component for controlling the network topology poses a reliability problem. The correct functioning of the network is predicated upon the central controller being operational and reachable. Ensuring reliability of such a component would result in additional costs, both in bandwidth and redundant hardware requirements.

c) Decentralized collaborative decision-making: In order to overcome the difficulties presented by the previous alternatives, we arrive at a design where decision-making is distributed to each of the network nodes. That is, each node

Term	Definition/Description
Node	A participant in the network, e.g. an aircraft or a base station.
Terminal	A device that can establish a directed link with another terminal; a terminal is attached to and controlled by a node. Terminals are used by the nodes to establish links between nodes. A terminal is suitable for establishing a connection with another if it is possible to steer the terminal's emitter and receiver so that they point in the direction of the other node's terminal.
Link negotiation	Two nodes negotiate with each other if they can establish a directed link between them using their respective terminals.
(Network) Link	A logical connection between two nodes through terminals based on a directed link.
(Network) Topology	The network topology is the arrangement of communication links between nodes in a network.
Path-to-ground	A node is considered to have a path-to-ground if the current network topology contains a path from that node to a base station. Base stations have a permanent path-to-ground.

TABLE I: ConOps terms

computes and executes its own actions without relying on a central controller. The decentralized nature of this alternative fundamentally solves the reliability issue.

We also target a design where complete network observability is no longer a prerequisite for computing which actions to execute. We thus assume only *local* observability, where a network node receives information about the state of its neighboring nodes (those inside a defined perimeter, e.g., within a geographical radius). In this manner, the requirements on bandwidth for state and control data remain limited to the defined neighborhood perimeter. Additionally, the computational complexity of a decentralized control strategy does not depend on the overall number of nodes in the network, but on the size of a node's neighborhood. Assuming that neighborhood sizes are more or less homogeneous, the computational complexity then remains constant as the number of network nodes increases.

B. Terms and Assumptions

Table I provides a definition of terms that are used throughout the paper.

One basic assumption for the ConOps Agent design is that, as an initial step, nodes can gather data about neighboring nodes within a defined sphere of influence through a secondary, more easily accessible channel, such as an omnidirectional RF data link.

This process is illustrated by Figure 1, where node A can detect nodes B, C, D and F. It can potentially form FSO links only with nodes B and F. Node E, on the other hand, is not within node A's sphere of influence and is thus basically invisible to node A. That is, it cannot be considered by node A for its decision-making process. Note, that Figures 1 and 2 show simplified representations in two dimensions (2D), while the real position of the nodes is in three-dimensional (3D) space.

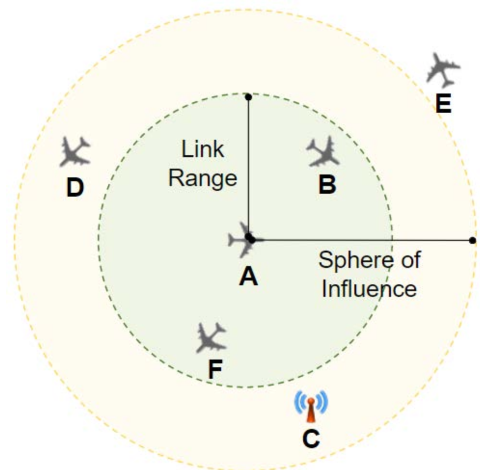

Fig. 1: Example nodes

Figure 2 shows the link range for all nodes of the previous example. Given the overlap of these link ranges, a network topology depicted by the blue lines between nodes could be formed, if every node agreed to this. Note that in this resulting network, the aircraft nodes A, B, E, and F have a communication path towards a base station, thus they are said to have a *path-to-ground*, whereas node D would not. Having a path-to-ground is one of the data items included in the node state information used for decision-making.

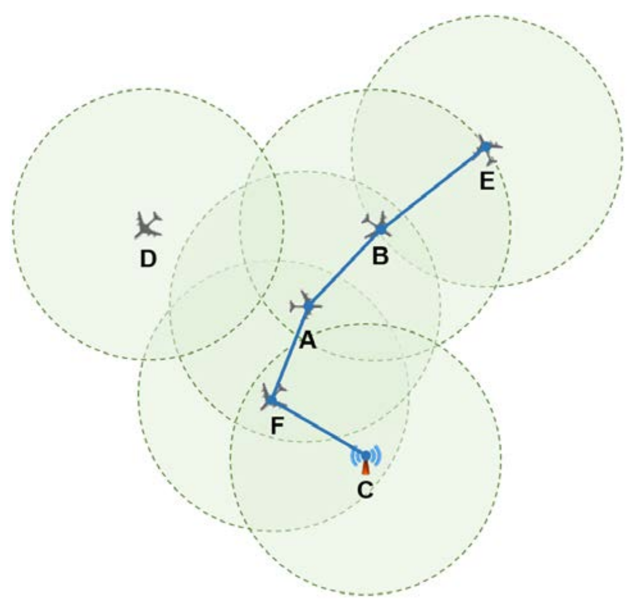

Fig. 2: Example topology

C. Algorithm

The primary function of the ConOps Agent algorithm is the management of directed links. When a link is established, the two terminals at either end of the link are pointed towards each other so that the beam between the terminals can be used to transfer data. The ConOps Agent algorithm consists of two parallel processes: The node detection process and the link management process.

Node Detection Process: Communication with nearby nodes happens continuously using the secondary communication channel, e.g., RF-communication. This is done to gather relevant information such as position, trajectory, number of free optical terminals and path-to-ground state from these neighboring nodes.

Link Management Process: Figure 3 depicts the flow for the link management process of the ConOps Agent algorithm.

- **Step 1.** Based on the information obtained by the node detection process, secondary metrics are calculated in order to prioritize the neighbors.
- **Step 2a.** Based on the information from **Step 1**, all the nodes within range are ranked, i.e., the list becomes a sorted list, according to the currently active link selection strategy.
- **Step 2b.** The highest ranked node in the list is selected.
- **Step 3.** The node checks each of its optical communication terminals to determine if the terminal is suitable for establishing an optical link with the selected node. This includes checking for range, pointing angle, and occlusion constraints.
- **After Step 3.**
  - If a terminal that was found suitable for connecting to the selected node is currently available, then the algorithm proceeds to **Step 4**.
  - If no suitable terminal was found even though free terminals are available, then the selected node is removed from the list and the algorithm proceeds with **Step 2b**.
  - If all terminals are currently in use but one of those terminals is deemed suitable, and the current node does not have a path-to-ground, the algorithm proceeds with **Step 6**.
  - If no suitable terminal was found and all terminals are currently in use, and the current node does have a path-to-ground, then the algorithm stops.
- **Step 4.** The current node sends a link request to the selected node and starts the link negotiation. The link request consists of information regarding the node, i.e., position and trajectory, and information regarding the selected terminal, i.e., elevation and azimuth constraints.
- **Step 5.** After the link negotiation is started, the selected node is removed from the list and the selected terminal is reserved for linking with that node.
- **After Step 5.** If the node still has more free terminals, the algorithm proceeds with **Step 2b**. If the node does not have more free terminals, then the algorithm stops.
- **Step 6.** Since all terminals are currently in use but one those terminals has been deemed suitable, and the current node does not have a path-to-ground, the suitable terminal that is currently in use is forced to terminate its link and the algorithm proceeds to **Step 4**.

IV. Evaluation using ABM

To ensure, that the ConOps Agent achieves its goal to build a stable network in the sky, ABM was used to model and evaluate the system concept.

A. Agent-based modelling

In ABM, a system is modelled as a collection of autonomous decision-making entities called agents. Each agent individually assesses its situation based on inputs it receives, makes decisions on the basis of a set of rules, and acts according to that decision.

Franklin and Graesser collected a variety of definitions for the term *agent* from different sources and distilled them into the following essential definition: "An autonomous agent is a system situated within and a part of an environment that senses that environment and acts on it, over time, in pursuit of its own agenda and so as to effect what it senses in the future." [8]

Bonabeau [2] lists the benefits of ABM over other modelling techniques as follows:

1) ABM captures emergent behavior. The term "emergent" is used when an entity is observed to have properties that its parts do not have on their own, i.e., properties or behaviors that emerge only when the parts interact in a wider whole.
2) ABM is flexible, e.g. it is easy to add new agents to an environment or adapt the behavior of a single agent without the need to modify the rest of the model.
3) ABM provides a natural description of a system, especially when describing and simulating a system composed of "behavioral" entities, where it is easier to describe the behavior of a single agent rather than the behavior of the whole system.

A wide range of ABM tools exist. North et al. provide an overview in [12]. The choice for the ABM tool in the presented work was Repast Simphony [5], which is an "agent-based modelling toolkit and cross-platform Java-based modelling system" [13]. It is important to note that in contrast to more traditional network simulations [11], the ConOps Agent modelling and simulation activities described here are not on the network level, i.e., focused on the protocol, routing, data rates and such but rather on the overall FANET system level as it is described in Section III, in particular, on the establishment of the physical topology of the network.

B. Evaluation Method

In order to evaluate the performance of the ConOps Agent algorithm, the ABM simulation environment was instrumented to output several performance indicators. The evaluation presented here is focused on two metrics:

- *Ground Connectedness:* The number of aircraft with a path-to-ground relative to the total number of aircraft flying.

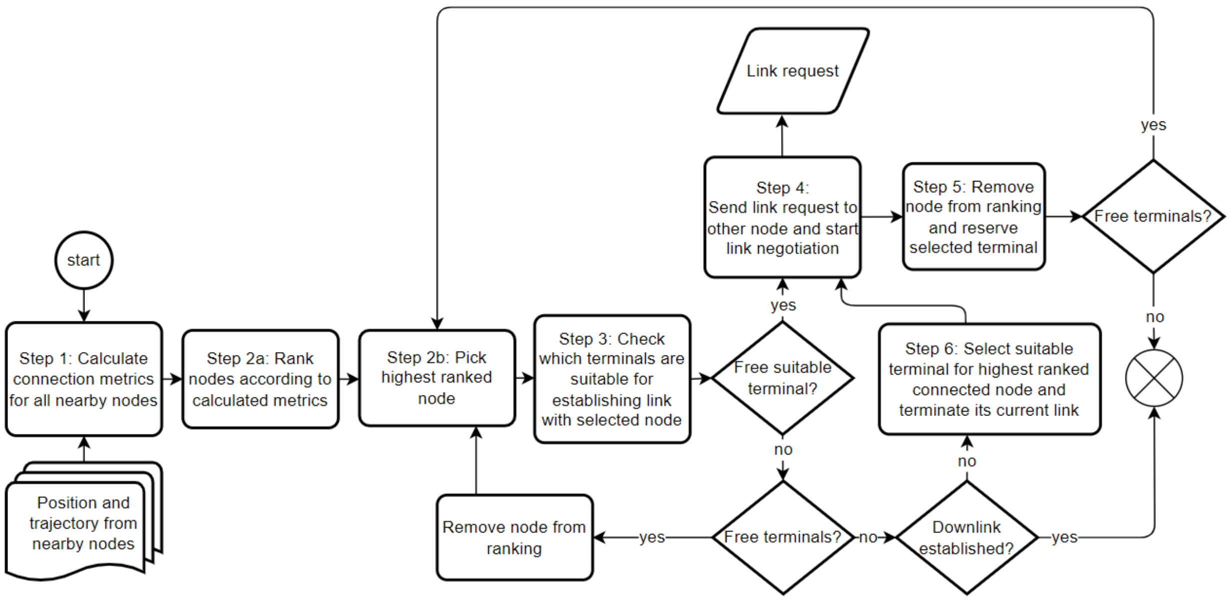

Fig. 3: Link management process flow

- *Link Duration:* The time elapsed between the establishment and interruption of a link.

The presented evaluation is based on a configuration with three terminals per aircraft, of which two are installed *on the underside* of the aircraft, i.e., field of regard restricted to 360°ein azimuth and between 1°and -90°in elevation. To evaluate the efficiency of the ConOps Agent in a real world context, the simulation in this work uses previously collected flight path data to model the movements of aircraft in the network. The data used for the evaluation is based on Automatic Dependent Surveillance - Broadcast (ADS-B) data recorded over a 24-hour period starting from midnight on Tuesday, January 14th 2020 showing aircraft movements on a typical day over Europe. This time period includes in total 9295 individual flights that are sufficiently representative for any extended period of flight operation data sets. In order to illustrate the effectiveness of the ConOps Agent to select neighboring nodes for connections as described in Section III-C, we compare it to a link selection algorithm that randomly selects a neighboring node.

C. Results

1) Ground Connectedness: In order to compare the scenarios we evaluate the amount of time over the simulated period in which connectedness exceeds ratios of 50%, 80% and 90%.

A plot of the collected data is provided in Figure 4. 50% connectedness is attained only after around 4:00 until shortly before midnight. An 80% connectedness level is achieved between 5:30 and 21:30 approximately, and a 90% connectedness only between 11:00 and 14:45. The random algorithm fails to achieve 50% ground connectedness through most of the simulated time period. As expected, the connectedness is strongly correlated with the number of aircraft flying. Low connectedness at night when the sky is less densely populated is generally not an issue when considering the overall goals of the FANET. Backhauling support, as well as increased passenger connectivity, are mostly required during the daytime.

2) Link duration: In order to estimate the stability of the network topology, we compare the distribution of link durations. As Figure 5 depicts, the ConOps Agent achieves significantly higher link durations than the random algorithm. Median link duration for the random algorithm is 420 seconds compared to 590 seconds for the ConOps Agent algorithm.

Overall, the ConOps Agent algorithm proves to be superior to the random link selection algorithm in all evaluated aspects.

V. Conclusion

This paper described the operational concept of a FANET, an airborne networking domain characterized by a wireless ad-hoc network of commercial aircraft, connected via point-to-point free space optical links, that has been developed within Airbus CRT. More specifically, the ConOps, which encompasses how every part of the system interacts and how the network is operated and managed, and its implementation in the ConOps Agent algorithm have been described.

To ensure scalability of the network with a high number of nodes, a decentralized approach was pursued, where each node in the network decides independently with which other nodes it wants to form network links based on information that is readily available to it. The ConOps Agent algorithm specifies this decision-making for each node in the network.

To ensure that the ConOps Agent algorithm achieves its goal to build a stable network in the sky, ABM was used to model and evaluate the system concept. Using Repast Simphony, an open-source Java-based agent modelling and simulation framework in which realistic flight data obtained from real

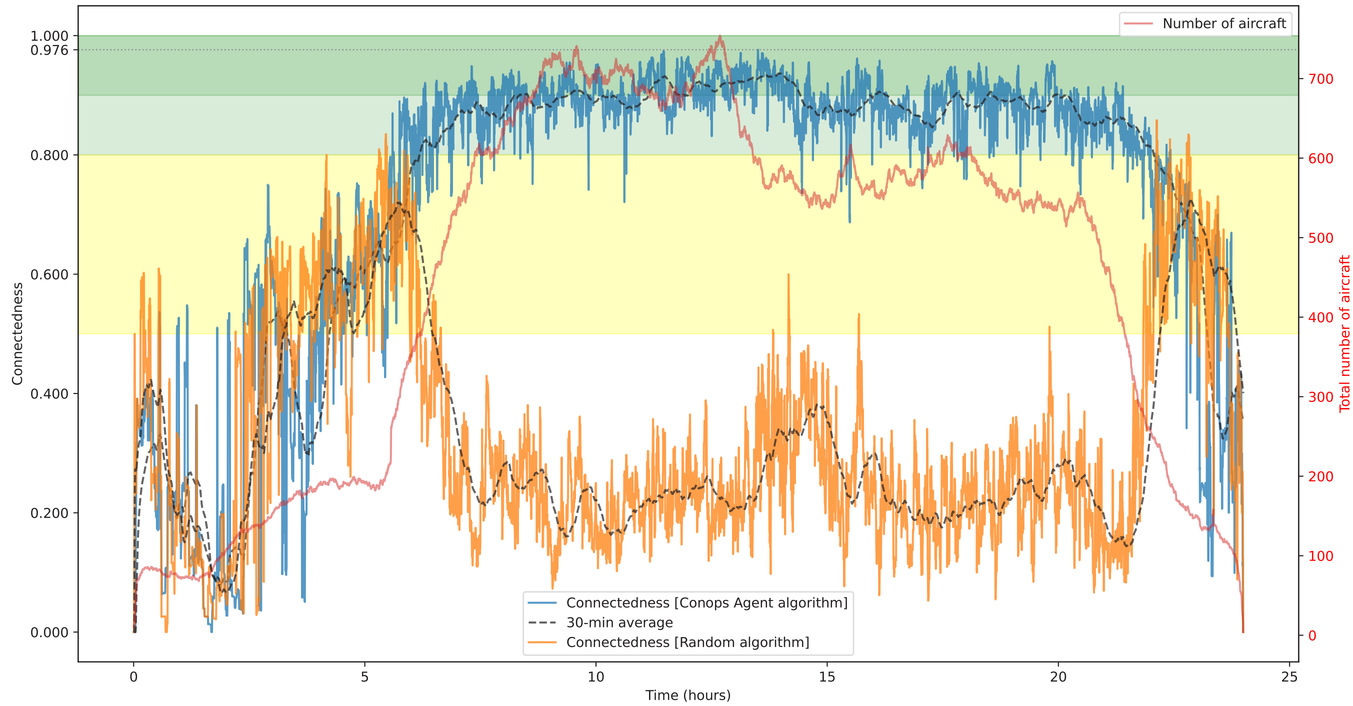

Fig. 4: Comparison of connectivity

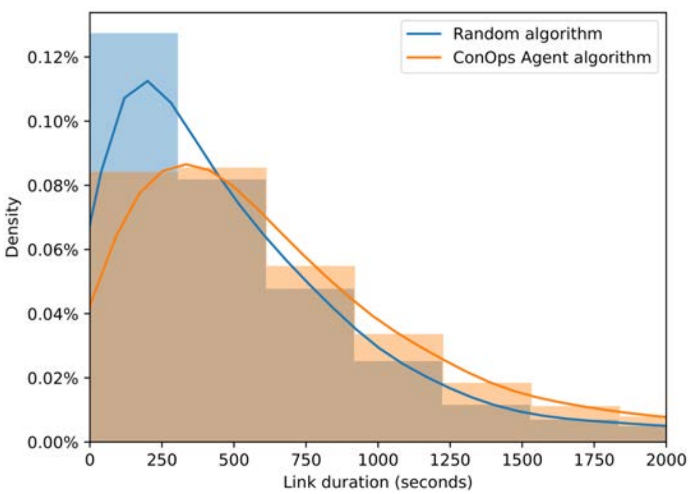

Fig. 5: Comparison of link duration

flight records was used, it has been shown that a significant number of aircraft in European airspace can be connected with the proposed FANET architecture.

References

[1] Anwer, M.S., Guy, C.: A survey of VANET technologies. Journal of Emerging Trends in Computing and Information Sciences **5**(9), 661–671 (2014)

[2] Bonabeau, E.: Agent-based modeling: Methods and techniques for simulating human systems. Proceedings of the national academy of sciences **99**(suppl 3), 7280–7287 (2002)

[3] Chlamtac, I., Conti, M., Liu, J.J.N.: Mobile ad hoc networking: imperatives and challenges. Ad hoc networks **1**(1), 13–64 (2003)

[4] Chriki, A., Touati, H., Snoussi, H., Kamoun, F.: FANET: Communication, mobility models and security issues. Computer Networks **163**, 106877 (2019)

[5] Collier, N.: Repast: An extensible framework for agent simulation. The University of Chicago's Social Science Research **36**, 2003 (2003)

[6] Corson, S., Macker, J.: RFC2501: Mobile ad hoc networking (MANET): Routing protocol performance issues and evaluation considerations (1999)

[7] Derenick, J., Thorne, C., Spletzer, J.: Hybrid free-space optics/radio frequency (FSO/RF) networks for mobile robot teams. In: Multi-Robot Systems. From Swarms to Intelligent Automata Volume III, pp. 263–268. Springer (2005)

[8] Franklin, S., Graesser, A.: Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents. In: International Workshop on Agent Theories, Architectures, and Languages. pp. 21–35. Springer (1996)

[9] Ghassemlooy, Z., Popoola, W.O.: Terrestrial free-space optical communications. InTech (2010)

[10] K, D.K., Ponseelan, A.J., D, D.P., V.B, D.M., S, S.: A Study on Google Project Loon- Opportunities and Challenges. IJETIE **6**(3) (March 2020), https://ssrn.com/abstract=3565713

[11] Kurkowski, S., Camp, T., Colagrosso, M.: MANET simulation studies: the incredibles. ACM SIGMOBILE Mobile Computing and Communications Review **9**(4), 50–61 (2005)

[12] North, M.J., Collier, N.T., Ozik, J., Tatara, E.R., Macal, C.M., Bragen, M., Sydelko, P.: Complex adaptive systems modeling with Repast Simphony. Complex adaptive systems modeling **1**(1), 3 (2013)

[13] Repast project: Repast simphony reference manual. https://repast.github.io/docs/RepastReference/RepastReference.html (2021), [Online; accessed 3-February-2021]

[14] Singh, M.: Alphabet shuts down Loon internet balloon company. https://techcrunch.com/2021/01/21/google-alphabet-is-shutting-down-loon-internet/, [Online; accessed 3-February-2021]

[15] Teller, E., Patrick, W.G.: Balloon clumping to provide bandwidth requested in advance (Nov 21 2017), US Patent 9,826,407

[16] Willebrand, H., Ghuman, B.S.: Free space optics: enabling optical connectivity in today's networks. SAMS publishing (2002)