Joint Routing and Scheduling in Mobile Aeronautical Ad Hoc Networks

Felix Hoffmann Daniel Medina Adam Wolisz

Abstract—**In this paper, we formulate the joint Internet gateway allocation, routing, and scheduling problem in wireless ad hoc networks with the goal of minimizing the average packet delay in a Spatial Time Division Multiple Access (STDMA) network. We first propose a mathematical programming approach consisting of two steps: a weighted hop count minimization subject to scheduling constraints, followed by average delay minimization for the previously computed routes. Since the computational complexity of this approach is prohibitive for larger networks, we also formulate a Genetic Algorithm that can be applied to larger networks as well as mobile networks. We analyze the performance of both approaches by means of simulations and compare the solution that they provide to a simple hop count based routing and gateway selection solution. It is shown that the performance of the Genetic Algorithm is comparable to the mathematical programming approach in terms of delay and packet delivery ratio at lower complexity, and is significantly better than the hop count based solution.**

I. Introduction

HIGH speed wireless Internet access has become more and more commonplace over the last years, whether at home, at work, or in public WiFi hot spots. However, inflight Internet access, especially on intercontinental flights, has not been able to keep up with this trend. Currently, satellites are the only means of providing Internet access to passengers flying in remote or oceanic regions of the world. Geostationary satellites are commonly used to offer such services, since a small number of geostationary satellites is already sufficient to provide nearly global coverage, and their large coverage allows for long lived connections without the need for handovers between satellites. However, usage of satellites is typically expensive, and geostationary satellites in particular suffer from the high propagation delay of ca. 250 ms that it takes for a signal to travel from the ground station to the satellite and back down to the subscriber. Recently, ad hoc networks have been proposed as an alternative means of providing Internet access to passengers in remote regions by extending the coverage of ground stations in a multi-hop manner [1], [2]. A depiction of such an envisaged aeronautical ad hoc network over the North Atlantic, based on current airline schedules and typical flight routes, is shown in Fig. 1.

Felix Hoffmann is with the Institute of Communications and Navigation at the German Aerospace Center (DLR) in Wessling, Germany. E-mail: felix.hoffmann@dlr.de

Daniel Medina is with the Bernstein Center for Computational Neuroscience in Planegg-Martinsried, Germany.

Adam Wolisz is with the Telecommunication Networks Group at TU Berlin and the Berkeley Wireless Research Center at UC Berkeley. E-mail: awo@ieee.org

The gateways that connect the ad hoc network to the Internet can be either located at the ground stations, or they can themselves be aircraft that make their satellite link available to other aircraft. This satellite link may serve as a backup in case there is no connectivity to a terrestrial base station, or the traffic load is too high for the base stations to handle alone.

Since such networks are intended to provide Internet access, practically all traffic would flow through the Internet gateways, potentially leading to congestion at or around these nodes. Therefore, the allocation of traffic flows to Internet gateways has a significant effect on the overall performance of the network, e.g. in terms of packet loss or delay. By allocating flows efficiently, traffic can be balanced between the gateways in order to avoid such congestion. This problem is strongly coupled with the routing of packets in the network, since the path between an aircraft and a gateway greatly affects the service that that gateway can provide to an aircraft. Finally, in a wireless network, all nodes must share the wireless channel. Therefore, the problem of scheduling how the nodes access this channel also has a great impact on the network performance.

In this paper, we consider the joint problem of Internet gateway allocation, routing, and scheduling with the goal of minimizing the average packet delay in the network. Since this is a nonconvex problem, we present a suboptimal approach splitting the problem up into two steps: a linear mixed integer program minimizing a weighted hop count in the network, subject to constraints on the feasibility of the scheduling, and a convex scheduling problem minimizing the packet delay. Still, this problem is quite complex, and the scheduling problem alone is known to be NP-complete [3]. Therefore, we introduce a Genetic Algorithm approach to efficiently find an approximate solution to the delay minimization problem. This Genetic Algorithm can be applied both to static as well as mobile networks, in contrast to common optimization techniques that are only applicable to static topologies. We compare the performance of these two approaches with each other and with a hop count based gateway selection and routing and distributed scheduling approach. The goal of this work is to optimize the network performance in a centralized manner, assuming that the relevant parameters such as the network topology and traffic demands are known at a central entity. The optimization framework developed here can serve to assess the performance of distributed algorithms in future work.

This paper is an extension of our previous work in [4], where Genetic Algorithms were applied to the optimization of static networks. Here, we significantly extend this work by the formulation of the mathematical programming approach and its two step decomposition, and compare the Genetic Al-

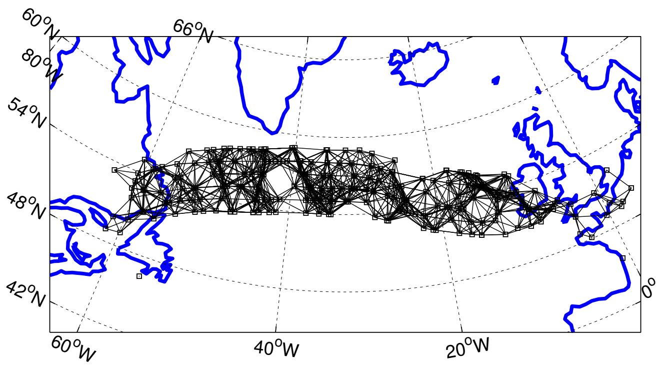


Figure 1. Example topology of the envisaged aeronautical ad hoc network over the North Atlantic. A range of 150 nmi is assumed for links between aircraft.

gorithm to the mathematical programming approach in terms of complexity and network performance. Also, we extend the previous work on Genetic Algorithms by considering their application to mobile networks as well.

The rest of this paper is structured as follows: We briefly review related work in Section II. In Section III, we present our network model and formulate our optimization problem in Section IV. In Section V, we give a short introduction to the concept of Genetic Algorithms and related work applying Genetic Algorithms to wireless network optimization. In Section VI, we define a Genetic Algorithm to solve the joint routing and scheduling problem and define extensions for the tracking of mobile networks in Section VII. In Sections VIII and IX, we assess the performance of this algorithm by means of simulations in small scale test topologies and more realistic large scale topologies, respectively. Finally, Section X provides conclusions and directions for further work.

II. Related Work

An efficient approach to solving the joint scheduling, routing, power control and rate adaptation problem in Wireless Mesh Networks based on a column generation technique is presented by Luo *et al.* in [5]. The optimization goal in [5] is to maximize the minimum throughput among all flows. The scheduling problem is modeled as the scheduling of Independent Sets (ISs), i.e. sets of links that can be active at the same. Each IS is allocated a fraction of the total time. This is in contrast to our assumption of a TDMA schedule with slots of fixed duration.

Another approach to solving the joint optimization of routing and scheduling in mesh networks using directional antennas has been proposed in [6] by Capone *et al.*, with the goal of minimizing the total number of time slots needed to satisfy the given demand. This is analogous to maximizing the throughput, as the shorter schedule can be repeated more frequently in a given time. Packet delay, or the allocation of nodes to gateways was not considered in this work.

The joint routing, gateway allocation, and scheduling problem has recently been addressed by Livingstone *et al.* in [7]. This work divides the problem up into several optimization problems that are solved sequentially. First, a routing solution minimizing the maximum link load is found, then scheduling is performed based on these routes which are now considered fixed, and free capacity is allocated to flows in a final step. Again, packet delay is not considered as a criterion.

Optimization techniques specifically for aeronautical ad hoc networks have not been previously considered. A major difference between these kinds of networks and Wireless Mesh Networks, which are more commonly treated in the literature, is the heterogeneous connectivity to the Internet via gateways with very different characteristics. One of the main motivations of our work is to balance traffic between terrestrial and satellite gateways. An analysis of the network topology of a potential ad hoc network over the North Atlantic has been conducted in [8]. Here, we will also focus on the North Atlantic scenario.

III. Network Model

In this section, we introduce the terminology required to formulate the optimization problem in the following section. The network is represented as a directed graph $\mathcal{G}(\mathcal{U}, \mathcal{V})$, where $\mathcal{U}$ is the set of all nodes in the network and $\mathcal{V}$ is the set of all directional links connecting these nodes. The subset of aircraft nodes is denoted $\mathcal{U}_{\text{AC}}$, the subset of nodes acting as Internet gateways is $\mathcal{U}_{\text{GW}}$. Note that these two subsets are not disjoint, as an aircraft with a satellite link may act as Internet gateway. In our model, the ground network is represented by a single node, referred to as node 0. Since all flows either begin or end at node 0, the problem of allocating a gateway to a flow is implicitly contained in the routing problem. The gateway allocated to an aircraft is that node on its path that has a direct connection to node 0. We are given a set $\mathcal{F}$ of traffic flows (p, q) from a source node p to a destination node q, each with a known traffic demand $R_{(p,q)}$, which is given in bits per second.

Which nodes are connected by wireless links is restricted by two effects. First, the curvature of the Earth limits the radio horizon. If both nodes are aircraft flying at a typical cruise altitude of 10,000 m, the maximum distance at which the nodes can "see" each other is $d_{\text{horizon}} = 824$ km. For air to ground links between an aircraft and a ground station, the radio horizon is half of this distance. Second, for communication to be feasible between any two nodes, we require that the link fulfill a minimum Signal to Interference and Noise Ratio (SINR) γ_0. For a maximum transmit power and free space signal propagation, this results in the maximum distance at which communication is possible in the absence of interference, denoted d_{max}. Thus a link (i, j) from node i to node j exists if the distance $d_{i,j}$ between the nodes satisfies

$$d_{i,j} < \min(d_{\text{horizon}}, d_{\text{max}}), \tag{1}$$

and thus the set of wireless links in the ad hoc network is given by

$$\mathcal{V}_{\text{air}} = \{(i, j) | d_{i,j} < \min(d_{\text{horizon}}, d_{\text{max}}),\ i, j \in \mathcal{U},\ i \neq j\}\,. \tag{2}$$

The additional links connecting the ground node with the gateways are contained in the set $\mathcal{V}_{\text{static}}$. Together, $\mathcal{V}_{\text{air}}$ and $\mathcal{V}_{\text{static}}$ define the set $\mathcal{V}$ of all links in the network graph.

The neighbors $\mathcal{N}_i$ of node i are those nodes to which i has a communications link:

$$\mathcal{N}_i = \{j \in \mathcal{U} | (i,j) \in \mathcal{V}\}. \quad (3)$$

The set of links (p, q) that can cause interference to a wireless link (i, j) are those links whose transmitter p is within the radio horizon of node j. This set of links interfering with receiver j is denoted $\mathcal{I}_j$:

$$\mathcal{I}_j = \{(p,q) \in \mathcal{V}_{\text{air}} | d_{p,j} \leq d_{\text{horizon}}\}. \quad (4)$$

We assume that access to the wireless channel is controlled by a distributed Spatial Time Division Multiple Access (STDMA) protocol [9]. STDMA has been previously considered for aeronautical ad hoc networks e.g. in [10] and [11]. Time is divided up into periodically repeating frames, which are in turn divided up into W time slots. These slots are allocated to links based on their traffic demand. Such a traffic sensitive STDMA system is described in detail e.g. in [12]. In a static scenario, the allocation of slots to links is the same in every frame, i.e. if link 1 is activated in slot 1 of frame n, it will be activated in slot 1 of all subsequent frames as well. Changes in the TDMA schedule may be caused by movement of the nodes, resulting in different interference conditions or links being created or deleted, or by changes in the traffic demand. One packet can be transmitted in each time slot, and each transmission must begin at the beginning of a time slot. This is certainly a simplification; a more realistic model would take the different Signal to Interference and Noise Ratio (SINR) values of different links into account and adjust each link's rate of transmission accordingly. A link that has been allocated a single time slot in a frame is able to transmit data at the basic rate of R_0 bits per second. To function properly, TDMA requires that the nodes are synchronized. In our scenario, it is reasonable to assume that the commercial aircraft are equipped with GPS receivers and can use this information as a common timing reference. We introduce the set of binary variables $u_{i,j}[n] \in \{0,1\}$ to indicate whether link (i,j) is active in slot n (1) or not (0). With this notation, the SINR $\gamma_{i,j}[n]$ of link (i,j) in slot n is

$$\gamma_{i,j}[n] = \frac{u_{i,j}[n] G_{i,j}[n] P_i d_{i,j}^{-2}}{n_j + \sum_{(p,q)\in\mathcal{I}_j} u_{p,q}[n] G_{p,j}[n] P_p d_{p,j}^{-2}}, \quad (5)$$

where n_j is the noise at the receiver[1] and P_i is the transmit power of node i. $G_{i,j}[n]$ denotes the combined transmit antenna gain of transmitter i in the direction of receiver j and receive antenna gain of j in the direction of i in slot n. This model allows for arbitrary antenna patterns. In order for a link to be used for transmitting data, we require that its SINR according to Eq. 5 be greater than some threshold γ_0.

The average delay of a packet from source to destination is given by the sum of the average delay of each of the links that make up the path. As long as a link is not overloaded, the average delay of a packet on that link can be well approximated by the time that it must wait until the next TDMA slot in which this link becomes available, also referred to as offset delay, plus the time that is required for the actual transmission of the packet. This service time is assumed to be constant at one slot duration. The offset delay accounts for the fact that slots used by the incoming and outgoing links of the same session might be "shifted in phase", i.e. while the links are running synchronously, the beginning of the incoming slot can be shifted in time relative to the beginning of the proper outgoing slot. In the best case (perfect alignment), this shift is equal to zero and the packet is subject to immediate transmission. In the worst case, a packet may need to wait almost an entire frame before it can be transmitted. For a given fixed schedule, all packets of a session will experience the same offset delay at a node. If the slots assigned to a link are distributed in the frame at regular intervals, the average delay $\delta_{i,j}$ on link (i,j) over all sessions can be expressed as

$$\delta_{i,j} = T_s \left(1 + \frac{W}{2\sum_{n=1}^{W} u_{i,j}[n]}\right), \quad (6)$$

where T_s is the duration of a time slot, and W is the number of slots per frame. The '1' in the parentheses accounts for the service time, whereas the fraction inside the parentheses accounts for the offset delay.

Note that Eq. 6 does not consider queueing delay, i.e. the delay caused by other packets waiting in the transmit queue in front of the packet being considered. However, link layer simulations of the STDMA system indicate that the delay remains almost constant as the traffic load increases, as long as the link is not overloaded. Our definition in Eq. 6 differs slightly from the delay model for multi-hop wireless TDMA systems by Djukic *et al.* in [13], since we do not assume any kind of coordination between the slots assigned to incoming and outgoing links of a node, and [13] does not include the transmission time in the delay.

The delay of satellite links is assumed to be constant at 240ms, and the delay of ground links connecting a gateway to the backbone node is neglected, regardless of the traffic load on the link. Then, the average delay of all traffic flows in the network is given by

$$\hat{\delta} = \frac{1}{\sum_{(p,q)\in\mathcal{F}} R_{(p,q)}} \left(\sum_{(p,q)\in\mathcal{F}} R_{(p,q)} \sum_{(i,j)\in\mathcal{P}(p,q)} \delta_{i,j} \right), \quad (7)$$

where $\mathcal{P}(p,q)$ is the set of all links that make up the path used by flow $(p,q) \in \mathcal{F}$. Note that if the scheduling variables $\{u_{i,j}\}$ were continuous, this would be a convex function. Due to the restriction to binary variables though, the link delay is no longer convex.

IV. Joint Routing and Scheduling Problem

In this paper, we are interested in minimizing the average delay of all traffic flows in the network. We assume that the traffic demand of the users is given and must be handled as efficiently as possible. As noted above, the ad hoc network can be connected to the Internet both by terrestrial air/ground gateways and by satellite gateways. Since satellite links are typically quite expensive and incur a high delay, we wish to

[1] Actually, n_j is the thermal noise scaled by the factor $\frac{\lambda^2}{(4\pi)^2}$, where λ is the wavelength of the carrier.

avoid these gateways and send traffic primarily over terrestrial gateways as long as possible. By considering the high delay of the satellite link in the objective function, these links will not be used at low traffic load. Only when the increased traffic load leads to higher delay of the terrestrial gateways will the satellite gateways begin to take over a part of the traffic. Other objective functions, such as a maximization of the throughput, are certainly also possible. However, this would lead to a full utilization of all gateways, which is contrary to our goal of using the satellite gateways only when necessary. Finally, for a given traffic load, the delay is directly related to the Quality of Service perceived by the users. In the expression given for the average flow delay in Eq. 7, it was assumed that the routing of packets through the network is known. However, if we desire to minimize the average delay in the network, we must consider the allocation of traffic to the links as well, since we only sum over those links that are actually used, and a link's contribution to the average packet delay depends on the amount of traffic flowing over that link. We define the variables $\ell_{(i,j)(p,q)} \in [0,1]$ to denote the fraction of traffic demand of the flow from node p to node q that is sent over link (i,j). To force the solution to use only single-path routing, these variables can be restricted to be binary: $\ell_{(i,j)(p,q)} \in \{0,1\}$. However, it is also possible to consider the more general case of continuous variables and multipath routing. Unfortunately, this optimization over both $\{\ell_{(i,j)(p,q)}\}$ and $\{u_{i,j}\}$ simultaneously results in a nonconvex objective function, so that the minimization of the average delay of packets in the network cannot be used as a reasonable objective for classical optimization techniques. Instead, we divide the problem up into two subproblems, similar to the approach in [7]. In [7], the first step of the optimization is to determine paths for each flow such that the maximum link utilization in the network is minimized. This step is done independently of any scheduling or interference constraints. Subsequently, in a second step, the length of the schedule is minimized, under the condition that each link receives enough slots to satisfy its demand. We make several observations regarding this approach. First, ignoring the scheduling constraints in step 1 may yield a routing solution for which a feasible TDMA schedule does not exist. Second, minimizing the maximum link utilization without considering link scheduling will tend to spread traffic out over as many links as possible. In the subsequent scheduling step, each link will receive fewer slots, thereby increasing the queueing delay. Therefore, we adopt the following approach: first, minimize the average hop count of all flows in the network, weighted with the flows' traffic load, and subject to scheduling constraints which guarantee that each link will be able to handle its assigned traffic load. Second, using the fixed link demands resulting from step 1, minimize the average flow delay by solving the scheduling subproblem. Step 1, the minimization of the Weighted Hop Count (mWHC) is defined in detail as follows:

$$\min_{\{u_{i,j}\},\{\ell_{p,q}\}} \sum_{(i,j)\in\mathcal{V}} w_{(i,j)} \sum_{(p,q)\in\mathcal{F}} R_{(p,q)}\ell_{(i,j)(p,q)}, \tag{8}$$

$$\text{s.t.} \sum_{j\in\mathcal{N}_p} \ell_{(p,j)(p,q)} = 1 \quad \forall (p,q)\in\mathcal{F} \tag{9}$$

$$\sum_{i\in\mathcal{N}_q} \ell_{(i,q)(p,q)} = 1 \quad \forall (p,q)\in\mathcal{F} \tag{10}$$

$$\sum_{i\in\mathcal{N}_k} \ell_{(i,k)(p,q)} = \sum_{j\in\mathcal{N}_k} \ell_{(k,j)(p,q)}$$

$$\forall k\in\mathcal{U}\backslash\{0\}, (p,q)\in\mathcal{F},\ k\neq p,q \tag{11}$$

$$\sum_{j\in\mathcal{N}_i} (u_{i,j}[n]+u_{j,i}[n]) \leq 1$$

$$\forall i\in\mathcal{U}\backslash\{0\}, 1\leq n\leq W, \tag{12}$$

$$R_0 \sum_{n=1}^{W} u_{i,j}[n] \geq$$

$$\sum_{(p,q)\in\mathcal{F}} \ell_{(i,j)(p,q)} R_{p,q} \quad \forall (i,j)\in\mathcal{V}_{\text{air}}, \tag{13}$$

$$\gamma_{i,j}[n] \geq \gamma_0 u_{i,j}[n] \quad \forall (i,j)\in\mathcal{V}_{\text{air}},\ 1\leq n\leq W. \tag{14}$$

The factors $w_{i,j}$ in the objective function are link weights, allowing e.g. satellite links to be assigned a higher cost than air/ground links, thereby helping to avoid the costly satellites. Constraints 9 and 10 assure that flows begin at their source and are terminated at their destination, respectively. Flow conservation at relay nodes is guaranteed by 11. Due to physical limitations, a node's wireless interface cannot transmit and receive at the same time, nor can it transmit to or receive from two different nodes simultaneously. These duplex constraints are covered by Constraint 12. Constraint 13 ensures that the total capacity allocated to a link is at least equal to the total traffic demand on that link. Finally, for a link to be usable in a certain time slot, it must achieve a minimum SINR γ_0. This requirement is addressed by Constraint 14. Unfortunately, this last constraint is nonlinear. However, it has been shown in [14] that it is possible to convert this into an equivalent *linear* constraint by taking advantage of the fact that the link activation variables $\{u_{i,j}\}$ are binary. This linearization does not work if the variables are continuous, as could be the case if power control is also added to the scope of the problem. However, it is possible to include power control by allowing multiple links between the same pair of nodes, each assigned a distinct power level and capacity. This comes at the expense of a larger number of optimization variables. Step 2 of the optimization problem, the minimization of the Average Flow Delay (mAFD), is defined as

$$\min_{\{u_{i,j}\}} \sum_{(p,q)\in\mathcal{F}} \sum_{(i,j)\in\mathcal{V}} \ell_{(i,j),(p,q)}\delta_{i,j}, \tag{15}$$

$$\text{s.t.} \sum_{j\in\mathcal{N}_i} (u_{i,j}[n]+u_{j,i}[n]) \leq 1 \quad \forall i\in\mathcal{U}\backslash\{0\}, 1\leq n\leq W, \tag{16}$$

$$\gamma_{i,j}[n] \geq \gamma_0 u_{i,j}[n] \quad \forall (i,j)\in\mathcal{V}_{\text{air}},\ 1\leq n\leq W. \tag{17}$$

The link delay of wireless links is calculated according to Eq. 6. This problem assumes that the link demands are given, and minimizes the average flow delay in the network, while respecting the duplex constraints 16 and minimum SINR constraints 17. The output of these two problems are the variables $\{u_{i,j}\}$ and $\{\ell_{p,q}\}$, telling us the exact route used for the traffic of each flow, as well as the STDMA schedule, telling us which links are activated in which time slot.

The first subproblem is a purely linear (mixed) integer program. The objective function of the second problem is nonlinear, but deals with fewer integer variables, since the link loads are already known. In practice, finding the optimum schedule in step 2 requires much more computational effort than step 1. Indeed, the scheduling problem alone is NP-complete, limiting the applicability of the optimization problem to small-sized networks. A summary of the notation used above is given in Table I.

$\mathcal{G}$	connectivity graph
$\mathcal{U}$	set of all network nodes
$\mathcal{U}_{\text{GW}}$	set of gateway nodes
$\mathcal{U}_{\text{AC}}$	set of aircraft nodes
$\mathcal{V}$	set of all links
$\mathcal{V}_{\text{air}}$	set of all wireless links within the ad hoc network
$\mathcal{V}_{\text{static}}$	set of all links between ground node and a gateway
$\mathcal{N}_i$	set of all neighbor nodes of node i
$\mathcal{I}_i$	set of all links potentially causing interference at node i
$\ell_{(i,j)(k,l)}$	variable indicating whether link (i,j) is used for flow (k,l)
$u_{i,j}[n]$	variable indicating whether link (i,j) is activated in slot n
$\gamma_{i,j}[n]$	SINR of link (i,j) in slot n
η_j	thermal noise at node j
P_i	transmit power of node i
$d_{i,j}$	distance between nodes i and j
$G_{i,j}[n]$	combined antenna gains of transmitter i towards receiver j and of j towards i in slot n
W	number of time slots per frame
T_s	time slot duration
n	time slot index
$\delta_{i,j}$	mean delay of packets on link (i,j)
$h_{i,j}$	number of slots allocated to link (i,j) per frame
$w_{i,j}$	weight assigned to link (i,j)

Table I
SUMMARY OF NOTATION USED FOR THE JOINT ROUTING, SCHEDULING, AND GATEWAY SELECTION PROBLEM.

V. GENETIC ALGORITHM APPROACH TO WIRELESS NETWORK OPTIMIZATION

Genetic Algorithms (GAs) have become increasingly popular in the last years as a simple but effective means of solving optimization problems that are difficult or practically impossible to solve using more traditional optimization methods such as integer or nonlinear programming. A good introduction to GAs can be found in [15]. The idea behind GAs is to consider a large number, or population, of potential solutions to the problem in parallel. The manner in which the problem solution is encoded in each individual is referred to as the genome. For example, the genome for a scheduling problem could be a binary vector of zeroes and ones, where each entry specifies whether a certain link is active in a certain slot or not.

In each iteration, or generation, of the algorithm, the performance of each individual is ranked according to a fitness function. Based on their fitness, a subset of individuals is selected to remain in the population in the next generation, and some less fit individuals are discarded. To replace these individuals, the genomes of some individuals of the population, termed parents, are combined with each other, thereby creating new genomes, or children. This operation is called recombination or crossover.

This whole process, consisting of the steps recombination between genomes, as well as the introduction of random mutations in the genome, together with a selection of individuals based on a fitness function, mimics the natural process of evolution. The offspring may have either a higher or lower fitness than their parents. However, the selection of individuals based on their fitness assures that those offspring with lower fitness will slowly disappear from the population, whereas the fitter individuals are retained, thereby improving the overall fitness of the whole population.

Genetic Algorithms have been proposed for use in the optimization of wireless networks in a number of papers. Recently, Lee *et al.* [16] have presented a GA approach to the allocation of wireless users to access points. However, their scenario is limited to a single-hop environment, and does not address radio resource scheduling. GAs have been previously applied to the problem of joint routing and scheduling in Wireless Mesh Networks by Badia *et al.* [14] and Pries *et al.* [17]. Badia *et al.* demonstrate the suitability of GAs for solving this type of problems. However, the authors do not apply an objective function related to the network performance, and focus mainly on finding a feasible solution. Pries *et al.* also address the joint routing and channel assignment problem in WMNs. Each node is assigned a radio channel on which it is allowed to transmit. Although the allocation of channels is similar to the allocation of TDMA slots, our problem formulation is more comprehensive, in that each link may be allocated an arbitrary number of TDMA slots. Therefore, the resource scheduling can be better adapted to the actual traffic load in the network. In [18], Lin *et al.* propose a Genetic Algorithm for the planning of Wireless Mesh Networks. The channel assignment subproblem is solved by means of a Genetic Algorithm, where the value of the objective function is determined by optimizing the routing subproblem using a linear programming approach. This is in contrast to our work, where both the routing and scheduling are optimized by the GA.

In this paper, we extend our previous work in [4], where we consider GAs for the purpose of delay minimization in static networks. Here, we compare the performance of the GA to the performance of the optimization problem presented in Section IV, and we study the applicability of GAs for the tracking of mobile networks.

VI. PROPOSED GENETIC ALGORITHM

Although Genetic Algorithms are well suited to a wide range of problems, many choices can be made in the detailed design of the algorithm, depending on the unique characteristics of the problem at hand. In this section, we define the different components of the Genetic Algorithm that we use for our network optimization problem. For the moment, the algorithm is assumed to operate on a static topology. Extensions for the case of mobile networks are given in Section VII.

A. Cost Function

The goal of our optimization is to minimize the average packet delay in the network. Therefore, the cost function according to which an individual g of the population is ranked must reflect the packet delay. To penalize infeasible solutions, we also add a penalty term that is proportional to the amount of unfulfilled demand in the network:

$$penalty(g) = \sum_{(i,j)\in\mathcal{V}} \max\left(load(i,j) - \sum_{n=1}^{W} u_{i,j}[n], 0\right), \tag{18}$$

where g is a genome of the population and $load_{(i,j)}(g)$ is the total traffic load of link (i,j) in genome g, given in bits per second, as it results from the routing of flows in the network. Using the notation of Section IV, we can write $load_{(i,j)}(g) = \sum_{(p,q)\in\mathcal{F}} \ell_{(i,j)(p,q)} R_{(p,q)}$.

The $\max$ operator ensures that a link with overprovisioned capacity does not affect the penalty function. Thus, the cost function is written as

$$c(g) = delay(g) + penalty(g), \tag{19}$$

where the average delay of the genome, $delay(g)$, is calculated according to Eq. 7. In GAs, the term *fitness* is more commonly used for the ranking of individuals. However, our problem is formulated in such a manner that we wish to minimize the cost instead of maximizing the fitness of the population.

It is possible that no feasible solution is found by the algorithm. In this case, the individual with the lowest cost is the one that is best able to fulfill the traffic demand in the network. Note that the optimization problem formulated in Section IV did not attempt to minimize the average delay directly, due to the nonconvex nature of this problem. However, GAs can be applied to nonconvex problems quite successfully. This is due in part to the random initialization of the population across the search space, and partly to the random modifications caused by mutations and crossover, which allow individuals to overcome local optima.

B. Genome Encoding

A solution to the routing and scheduling problem must be encoded within each genome of the population. For each flow, a path is encoded as a list of nodes in the order that they are visited by a packet from a source to its destination. During operation of the GA, nodes may be inserted, removed, or replaced in this list. The TDMA schedule of each genome is encoded in a separate table containing the set of active links in each time slot. A slot that is allocated to a link is not tied to any dedicated flow. During the operation of the GA, links may be added to or removed from the schedule of a genome.

C. Initialization of Population

When a GA is applied to a problem, an initial population must be generated before the process of recombination, mutation, and selection can begin. A common technique is to initialize the genomes randomly. In principle, this is also possible in our case. However, our joint routing and scheduling problem possesses a large number of constraints that separate feasible solutions from infeasible solutions, and almost all randomly chosen initial solutions will be infeasible. Although the penalty term in the cost function would direct the algorithm towards feasible solutions, we dedicate some additional effort to initializing the population with solutions that are likely to be close to feasibility, in order to accelerate the convergence of the algorithm. More precisely, the genomes are initialized with a valid path to a gateway, and all subsequent operations on the genome ensure that all paths remain valid. A valid path is defined as a list of nodes beginning with the flow's source and ending with the flow's destination in which each node is within transmission range of its preceding and following nodes, and each node is allowed to appear at most once.

To this end, we calculate the shortest path between each node and each gateway. When a new individual is created, a path to a random gateway is selected for each flow and time slots are scheduled for each flow. A link is allowed to be scheduled in a certain slot if the SINR of all links already active in that slot as well as the SINR of the new link remain above the required SINR threshold after the new link is added. It is possible that some links may not receive a sufficient number of slots initially, since links cannot be scheduled due to the interference constraints. However, this can be fixed by the following evolutionary process, or the genome will eventually be removed from the population because the overloaded links contribute to the penalty term in the cost function. Due to the selection of a random path for each flow, the genetic diversity of the population is still quite high. Subsequent mutations will lead to deviations from these precomputed paths that may use the wireless channel more efficiently by allowing for more spatial reuse in the network.

D. Selection

Several different mechanisms have been proposed for selecting the subset of a population that will generate offspring for the next generation. The most simple form of selection is based on *ranking* all members of the population according to their fitness. The fittest members are allowed to recombine in order to create new members for the next generation, whereas the least fit individuals are removed from the population.

A second approach, which is used here, is referred to as *tournament selection*. Here, a pool of individuals is randomly chosen as potential parents for the next generation. From this pool, two individuals are selected at random and the fitter of the two is kept as a parent, whereas the less fit individual is dropped from the population. The major advantage of this approach compared to ranking selection is that less fit individuals also have a chance of creating offspring. They must only be selected for a tournament in which the second participant is less fit. This aspect is quite important, since modifications of a genome, either due to recombination or mutation, may not immediately result in a higher fitness. For example, if a node on a path is replaced by another one, the fitness may initially be worse, since only few time slots have been reserved for the new links. However, over the course of several generations, additional time slots may be allocated to these new links, until the fitness of the genome has become higher than it was before the switching of the node. Thus, a selection mechanism that does not immediately remove individuals with lower fitness from the population is well suited to our problem.

E. Crossover Operator

A crossover is the creation of new individuals from a set of parents. To ensure that the paths remain valid, only complete paths are exchanged between two genomes. When two genomes are recombined, each of their paths for a flow may be exchanged with a certain probability. However, this exchange may create conflicts in the schedule. Therefore, it is first checked, whether the links of the new path can be scheduled in the same time slots as they were assigned in the parent. If this is not possible, the algorithm attempts to schedule the link in a different time slot. If it is not possible to schedule one or more of the links, this will again be reflected by the penalty term in the cost function.

F. Mutation Operators

The purpose of mutation operators is to introduce small random variations in the genomes of some individuals, in the hope that these will improve the fitness of the individual. In fact, most mutations will degrade the fitness, and may lead to the individual being removed from the population. However, the new information that is brought into the population by a mutation may occasionally lead to an improvement that would not have been achievable by means of recombination alone.

The problem that we are faced with when choosing appropriate mutation operators is that a random change in the routing or the scheduling will in most cases lead to an infeasible solution. Therefore, instead of randomly modifying parts of the genome, we propose the following operations that consider the constraints of the network optimization problem. Mutations operate at three distinct levels of granularity. The first three mutation operators modify only the allocation of slots to links. The next three operators act on the routing, by modifying the path that is taken between any two nodes. However, the path remains connected and free of loops. Finally, the last mutation operator acts at the flow level, replacing a flow's entire path with a path ending at another gateway.

- *Slot insertion*: This operation attempts to schedule an additional slot for a link that is already active in some other slot, thereby reducing the delay of packets on this link.
- *Slot removal*: This operation attempts to remove a time slot that has been allocated to a certain link from the schedule. However, it only does so if the time slots that remain allocated to the link are still sufficient to carry its allocated load. The advantage of this operator is that it frees up resources the schedule, which may later be consumed by other links.
- *Slot exchange*: This operation attempts to remove a time slot that has been allocated to a certain link from the schedule and replace it with another time slot. Thus, the link's capacity, and the genome's cost function, are not affected. However, the resulting schedule may allow links that previously were not able to receive further slots to do so now.
- *Node insertion*: This operation attempts to insert an additional node k into a path by breaking up an existing link between nodes i and j into two links (i, k) and (k, j).
- *Node removal*: This operation attempts to remove a node j from a path by merging the incoming and outgoing links (i, j) and (j, k) into the single link (i, k).
- *Node exchange*: This operation attempts to reroute traffic by replacing a node j by another node ℓ. The incoming and outgoing links (i, j) and (j, k) must accordingly be replaced by the links (i, ℓ) and (ℓ, k), respectively.
- *Path exchange*: This operation replaces an flow's entire path, similar to the manner in which the crossover operator works. Here however, paths are not exchanged between two genomes. Rather, a random path from the initial set of shortest paths is selected to replace an existing path of the genome. This can bring new information into the population and significantly alter the interference situation. Note that this operator has a unique function in that it is the only mutation operator that allows a node to change its Internet gateway.

VII. Tracking of Mobile Networks With Genetic Algorithm

In this section, we discuss how the Genetic Algorithm presented in the previous section can be extended to allow for the tracking of mobile networks.

In general, GAs suffer from slow convergence, so that adapting to a mobile network on the fly may not be feasible in many situations. However, the specific characteristics of the aeronautical networks considered here allow the GA to be applied

Although the aircraft are moving at high velocities, their relative movement is very low. In the North Atlantic scenario considered here, aircraft fly on parallel tracks for hours, keeping the same neighbors for most of the flight. Although the connections to the static terrestrial gateways are much more short-lived, their lifetimes are still on the order of tens of minutes. All aircraft are flying at cruise altitudes, so that all links are line of sight, without significant multipath components. Therefore, the overall network environment is very stable, giving the GA the opportunity to converge between major changes in the network topology, such as link up or link down events.

Changes in the network topology, such as the creation or deletion of links or nodes, can easily be incorporated into the population of the GA on the fly. In case link (i, j) breaks, the algorithm checks whether any genome was using this link. If this is the case, it tries to repair the path by finding an intermediate node k such that the links (i,k) and (k,j) can be activated instead. Likewise, if an aircraft node is deleted, the algorithm checks all genomes if this node was used to relay traffic for any other node. If so, it again tries to reroute the flow around the deleted node. If it is not possible to reroute the flow in either case, the algorithm tries to replace the entire path with a path to a different gateway. If a new link is created, no special actions are taken. However, subsequent mutations may introduce this link into the existing genomes. If a new aircraft node is created, the corresponding up- and downstream flows are introduced into every genome. In addition, the movement of the nodes continuously changes the interference levels at each receiver. Therefore, each time slot of the TDMA schedule must be checked in each generation in order to ensure that the set of simultaneously activated links are still compatible, i.e. each receiver's SINR is above the threshold γ_0. In case of an SINR violation, one or more of the links need to be rescheduled or deallocated in order to return to a valid schedule.

After these changes have been made to all individuals in the population, the algorithm continues to run with the modified population. Obviously, this is much more efficient than beginning a new optimization run with every change in the topology. One danger in this approach is that the population converges during the operation of the GA. When a change in the network topology occurs, it may become difficult for the GA to react, if the population has become too homogeneous. This effect can be prevented by running the algorithm with a relatively high mutation rate. To prevent these frequent mutations from degrading the performance of the best individual in the population, which will be used to derive the routing and scheduling information for the nodes in the network, it is useful to introduce the concept of elitism. In Genetic Algorithms, this means that the fittest individuals of the population, referred to as the *elite set*, are automatically copied into the next generation.

In principle, the algorithm can be run with a constant number of generations per simulated second. However, it is more effective to increase the algorithm's update rate whenever significant changes have occurred in the topology and slow the algorithm down when the topology is more static. This improves the algorithm's ability to react to changes quickly and prevents unnecessary effort when the algorithm has already converged to a solution.

VIII. Performance Evaluation With Small-Scale Scenarios

Below, we present simulation results to assess the performance of the proposed mWHC/mAFD optimization and Genetic Algorithm approach to solving the joint gateway allocation, routing, and scheduling problem in small scale scenarios. The performance of the GA in larger network topologies is analyzed in Section IX. The optimization problem has been solved with the Lingo optimization software [19]. The Genetic Algorithm implementation as well as the simulations to determine the network performance have been performed using the OMNeT++ network simulator [20]. We consider a simulation topology with aircraft flying from left to right between two air/ground gateways, as shown in Fig. 2. At first, only three aircraft are located near the left gateway. Shortly before these aircraft lose their direct links to the gateway, three new aircraft are generated at the left. This process repeats until the first three aircraft have come within range of the right gateway. At this moment, the network consists of a total of 15 aircraft, as shown in Fig. 2(c). Only one of these aircraft is equipped with a satellite link and can also act as Internet gateway for the other aircraft.

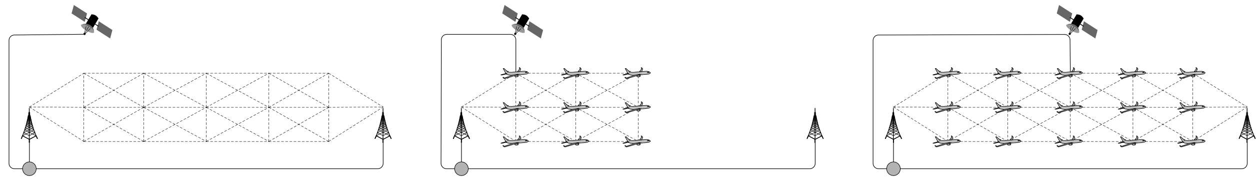
Parameter	Value, static case	Value, mobile case
Population size	600	600
Selection mechanism	Tournament selection	Tournament selection
Pool size	260	260
Elite set size	0	14
update rate	N/A	1 s^{-1}
p(slot insertion)	0.02	0.02
p(slot removal)	0.02	0.02
p(slot exchange)	0.02	0.03
p(node insertion)	0.01	0.02
p(node removal)	0.02	0.02
p(node exchange)	0.02	0.04
p(path exchange)	0.02	0.04
p(crossover)	0.1	0.2

Table II
SUMMARY OF GA PARAMETERS USED FOR SIMULATIONS

However, the mWHC/mAFD optimization can only be performed for static topologies, so that we also consider six discrete snapshots of this mobile network. In the first step, only three aircraft are present, occupying the three leftmost vertices. In each subsequent step, all aircraft move one position to the right, and three additional aircraft are introduced at the left side, until in step 5, all 15 vertices of the grid are occupied by aircraft, but the rightmost aircraft do not yet have connectivity to the air/ground gateway on the right. These three air/ground links are introduced in step 6, without any further movement by the aircraft.

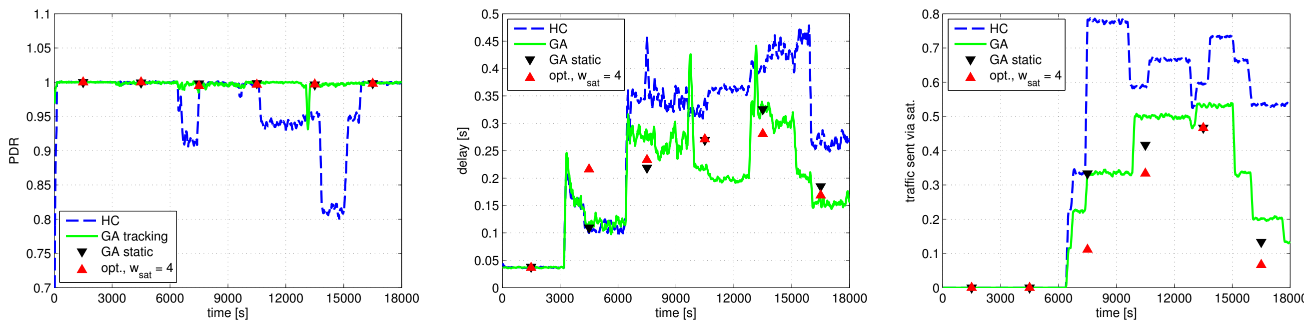
Both the aircraft and ground stations are equipped with Uniform Circular Arrays consisting of 16 antenna elements, whose beam patterns are calculated so as to maximize the gain along the direction of the link. The frame length is set to eight slots, with one time slot lasting 0.01 s. A minimum SINR of 10dB is required for a link to be usable for communication. Each link has a transmit queue which is able to hold up to 20 packets. Packets arriving at a full queue are discarded. For simplicity, only downstream traffic is considered in this scenario. Parameters specific to the Genetic Algorithm are shown in Table II. With Genetic Algorithms, a good deal of experimentation is typically required until the set of parameters providing the best results has been found, and the best parameters may vary according to the network parameters. Here, we have used one set of parameters providing reasonably good performance for all static GA runs and another set for all mobile GA runs.

The GA tracking algorithm and the hop count based routing and gateway allocation methods were run in parallel to the movement of the aircraft. In addition, for each of the static snapshots, the weighted hop count and delay minimization



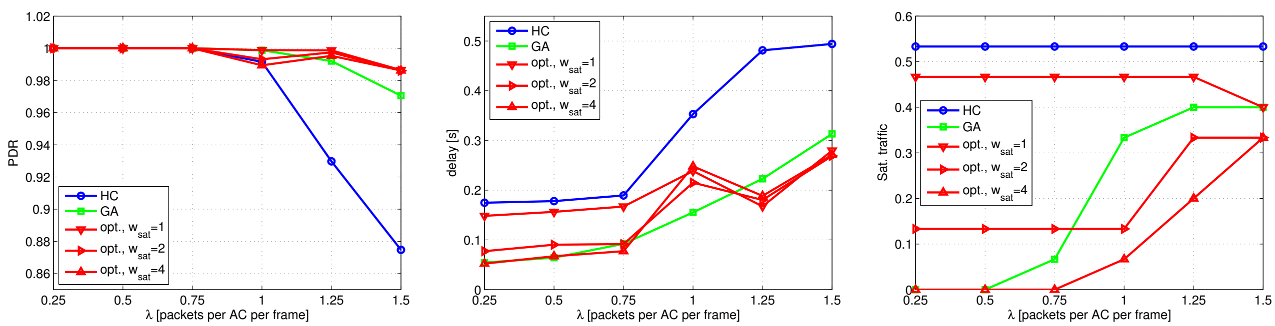
(a) Internet gateways and vertices where aircraft will be placed (b) Step 3. The aircraft with the satellite link has just been created. (c) Step 6. In this final step, the cloud of aircraft reaches the Internet gateway on the right.

Figure 2. Topology for small-scale simulations.



(a) (b) (c)

Figure 3. PDR, average delay, and percentage of traffic sent over satellite as a function of time for the small scale test topology.



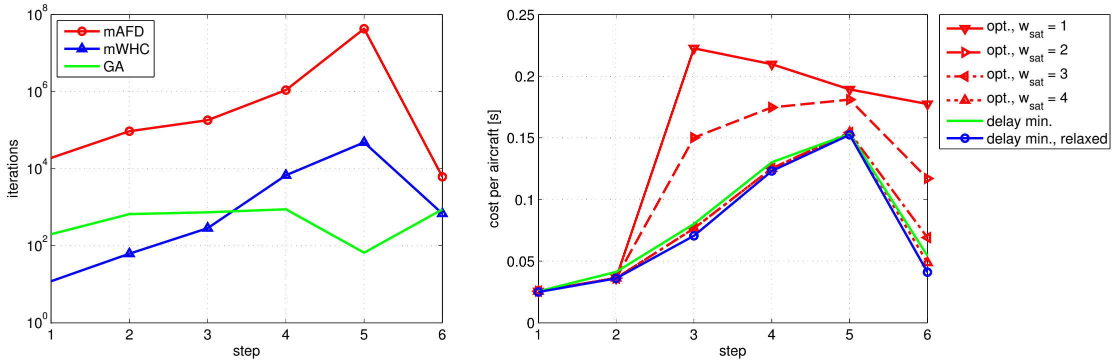
(a) (b) (c)

Figure 4. PDR, average delay, and percentage of traffic sent over satellite for increasing values of packet generation rate λ in Step 6 of the small scale sample topology.

problem according to Section IV and the static GA according to VI have been executed. The GA tracking should not result in worse performance than the static GA algorithm, as this would be a sign that the population is not able to adapt properly to the changes in the network topology. The GA advances by one generation per simulated second.

The routing and scheduling information resulting from the GA and mWHC/mAFD optimization was then used as input for network simulations in order to determine the network performance. The hop count based routing and gateway selection scheme relied on the traffic sensitive STDMA link scheduling algorithm proposed in [12]. The resulting Packet Delivery Ratio (PDR) and average packet delay achieved by each of these solutions are shown in Fig. 3 with the ground node generating packets for the aircraft at a mean rate of 12.5 packets per second per aircraft, corresponding to one packet per aircraft per frame. The fraction of the aircraft that are using the satellite gateway instead of the terrestrial gateways is also given. The mWHC/mAFD optimization was done with a weight $w = 1$ for all ad hoc and terrestrial links, and $w = 4$ for the satellite links.

It can be seen that the hop count solution in all steps sends more traffic over the satellite than the other solutions. This is due to the prominent position of the aircraft offering the satellite link in the middle of the ad hoc network. In step 3, when the satellite gateway has just been created, it practically "eclipses" the terrestrial gateway at the left side, since only one aircraft is closer to the terrestrial gateway than to the satellite gateway. The GA performs nearly as well as the mathematical programming solution. The hop count based solution loses almost 20% of the packets just before the cloud of aircraft establishes a link to the gateway on the right side, whereas the GA and mWHC/mAFD solutions deliver practically all packets over the entire simulation. The average packet delay of the tracking GA is comparable to the static GA and mWHC/mAFD solutions, and significantly lower than



(a) Number of solver iterations (mWHC/mAFD) and GA generations required for each of the six small scale topology snapshots.

(b) mWHC/mAFD cost for different satellite links weights, and cost when attempting to minimize the delay directly.

Figure 5. Behavior of mWHC/mAFD and GA algorithms.

the delay of the hop count solution. It is important to note in Fig. 3(c) that, when a connection is made to the right hand gateway, the GA solution succeeds in switching most traffic away from the satellite link. This shows that the population is still able to react efficiently to significant changes in the topology.

In Fig. 4, a closer look is taken at the network performance in a static case. Here, we consider the network of 15 aircraft just before a connection is made to the right hand terrestrial gateway, for varying values of the packet generation rate. Again, the PDR, average packet delay, and the fraction of traffic sent over the satellite gateway, are considered. The mWHC/mAFD optimization is performed with several different values for the weight of the satellite link. In general, higher values of w will lead the algorithm to avoid the satellite link longer. In our small network, a value of $w = 4$ would be sufficient to force all traffic to go over the terrestrial gateway. However, this is prevented by the interference and capacity constraints. Thus, $w = 4$ shows us the minimum amount of traffic that must be sent over the satellite in order for the solution to remain feasible. It is interesting to note that until $\lambda = 1$ packet per aircraft per frame, the routing solution found by mWHC/mAFD is not affected by the interference and capacity constraints. However, at $\lambda = 1$, the network has become quite congested, packet delay is increasing, and some packets are being lost due to random fluctuations in the queue lengths. In the next step, the mWHC/mAFD solution has adapted its routing to this congestion, actually leading to less packet delay and packet loss, although the amount of traffic has increased. This behavior is caused by the two-step nature of the approach, where the first step only tries to minimize the weighted hop count, restricted only in that its solution must be feasible. Here, the GA has an advantage, since it attempts to minimize the packet delay directly, and gracefully shifts traffic from the terrestrial gateway to the satellite gateway as the load increases. In our network, the best performance is achieved by the mWHC/mAFD with $w = 4$. However, it would be difficult to predict the optimum value for this parameter for arbitrary topologies. In both Fig. 3(b) and Fig. 4(b), the average delay can increase above 250 ms. In these cases, it would be better (from a delay minimization perspective) to transmit packets directly over a satellite link. However, only one aircraft in the simulation is equipped with a satellite link, so packets must be sent through the congested ad hoc network.

We see these results as a confirmation that the GA approach in static networks, as well as the tracking of mobile wireless networks with a GA on the fly, are useful tools in network optimization, since the resulting performance is very similar to what can be achieved by classical mathematical programming methods.

A comparison of the computational complexity of the mWHC/mAFD approach and the GA is shown in Fig. 5(a). The number of solver iterations required by Lingo to solve the mWHC and mAFD problems, as well as the number of generations of the static GA until convergence are shown for each of the six snapshots of the small scale topology. It can be seen that the complexity of the mathematical programming approach increases tremendously with the problem dimensions. However, the number of iterations also depends on the ”hardness” of the problem. In step 6, the problem is larger than in step 5, but do to the additional connectivity of the second terrestrial gateway, the problem has become easier to solve. Note that the mAFD scheduling problem is much more difficult to solve than the mWHC routing problem, which contents itself with finding a feasible schedule. On the other hand, the number of generations required by the GA increase much more moderately. However, one weakness of the GA is that it converges prematurely in step 5, hence the low number of generations, because the problem has become too difficult. The resulting degradation in performance with respect to the mWHC/mAFD solution can be seen e.g. in 3(b). The performance of the GA could be increased here by a larger population size or by a stricter convergence criterion. Here, we have opted to use one set of parameters for the GA in all six steps. It is likely that performance can be improved by adapting these parameters to the network conditions.

Fig. 5(b) shows the value of the cost function of the mAFD step after solving the mWHC step with different link weights for the link from the satellite gateway to the ground network. The weights of all other links are kept at one. It can be seen that the cost, corresponding to the average flow delay, decreases as the satellite link weight increases and traffic is shifted away from the satellite as. Setting w_{sat} to five does not result in a further reduction of the cost.

In addition, the cost resulting from trying to solve the original delay minimization problem directly, i.e. in a single step, is also shown. Since this is a nonconvex problem, it is not possible to determine the true global optimum. However, the problem has been solved by Lingo using a nonlinear solver with multiple starting points. Using multiple starting points significantly increases the computational cost, but reduces the risk of converging to a local optimum. Although Lingo also provides a global solver for nonlinear problems, the computational cost is prohibitive even for the smallest problem, step 1, consisting of only three aircraft and one gateway. Here, we show the cost for the direct delay minimization in case all variables have been relaxed to continuous values between zero and one (blue line), and in case all variables are constrained to be binary (green line). As could be expected, the cost of the relaxed problem is less than the binary variable problem. Surprisingly, though, the cost for the mAFD problem with $w_{\text{sat}} = 4$ is actually slightly less than the cost when attempting to solve the delay minimization problem directly. This can be attributed to convergence of the direct delay minimization problem to a non-global optimum. Clearly, the mWHC/mAFD approach can also provide an advantage in terms of performance, but only if the satellite weight is chosen correctly. In larger networks, setting w_{sat} to unnecessarily high values will lead to very long paths being used, potentially resulting in a larger delay than if a satellite gateway at a smaller distance were used instead.

IX. Performance of Genetic Algorithm in Large-Scale Scenarios

In this section, we analyze the performance of the GA in larger mobile topologies that can no longer handled by the direct mWHC/mAFD approach due to the computational complexity. The performance of the GA in large static topologies has been previously analyzed in [4].

A. Simulation Setup

The performance of the GA approach is assessed by means of simulations intended to reflect the aeronautical environment in the North Atlantic that we are considering. We consider a rectangular area 3000 by 440 km in size. Terrestrial Internet gateways are placed in each of the four corners of this field. A fifth terrestrial gateway is placed at the middle of the Northern edge of this rectangle, corresponding to a gateway in Greenland or Iceland. Again, we consider a dynamic topology, with aircraft being created at the left side of the field and being deleted when they have reached the right side. The generation rate has been modeled according to the actual rate of aircraft entering the North Atlantic corridor in a twelve hour time period. Information about the actual air traffic characteristics has been extracted from a flight database of scheduled airline flights [21]. At the beginning and end of the cloud, the generation rate is lower than during the peak

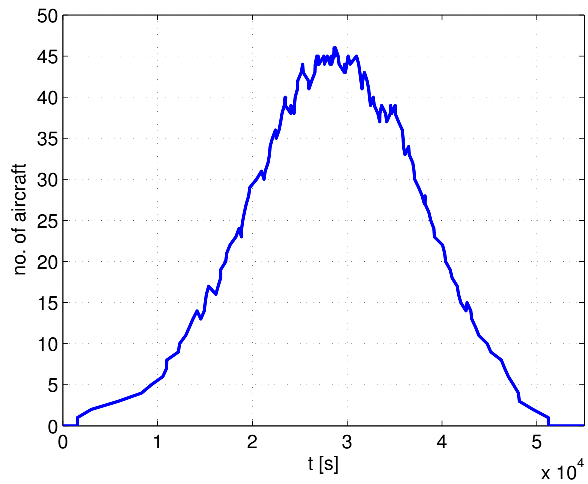


Figure 6. Number of aircraft in the network over time.

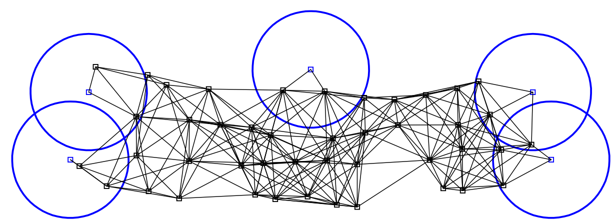


Figure 7. Example topology at time t=30000s.

period. The generation rate in the simulations was set to one fourth of the true rate derived from the schedule, since not all aircraft or airlines may participate in such an ad hoc network. The variation over time of the number of aircraft in our simulated network resulting from this model is shown in Fig. 6. Whenever an aircraft is created, it is chosen with probability 0.5 to be equipped with a satellite link and be able to act as Internet gateway for other aircraft.

In the North Atlantic, aircraft typically fly on predefined tracks, the so called North Atlantic Tracks [22], that ensure separation between aircraft in the uncontrolled airspace. To model these tracks, aircraft in our simulation are placed on a system of four parallel tracks, each separated by one degree of latitude and with a minimum spacing of 145 km between two aircraft on the same track. Again, these values have been chosen to fit the North Atlantic scenario. A screenshot of a typical network topology is shown in Fig. 7. The transmission ranges of the terrestrial gateways are indicated by blue circles.

A summary of the key network parameters used for the simulations is given in Table III, a summary of the parameters used for the GA, especially the probabilities with which each of the GA operators is invoked, is given in Table IV.

B. Network Performance

As in the previous section, the hop count based gateway selection and routing again serves as a reference. Simulations have been performed with a high traffic load (4 downstream packets per aircraft per frame) and a low traffic load (2

Parameter	Value
Playground size	3000 × 440 km
No. terrestrial gateways	5
Probabability of acting as sat. GW	0.5
Max. range d_{horizon}	824 km
Min. required SINR γ_0	10 dB
TDMA slot duration	10 ms
TDMA frame length	80 slots
Traffic asymmetry (DS:US)	4:1
Traffic load (high)	4 packets per frame per node DS 1 packet per frame per node US
Traffic load (low)	2 packets per frame per node DS 0.5 packets per frame per node US

Table III
SUMMARY OF NETWORK PARAMETERS USED FOR LARGE SCALE SCENARIO.

Parameter	Value
Population size	300
Selection mechanism	Tournament selection
Pool size	140
update rate	1 s^{-1}
p(slot insertion)	0.02
p(slot removal)	0.006
p(slot exchange)	0.1
p(node insertion)	0.06
p(node removal)	0.1
p(node exchange)	0.1
p(path exchange)	0.2
p(crossover)	0.5

Table IV
SUMMARY OF GA PARAMETERS USED FOR LARGE SCALE SCENARIO.

downstream packets per aircraft per frame). Simulation results of the average packet delay and the fraction of the overall traffic that is sent via a satellite link rather than over an air/ground link are shown in Fig. 8 and 9. The packet loss rate in this scenario is negligible due to the high number of satellite gateways. It can be seen that neither the traffic load nor the gateway selection and routing scheme has a large effect on the average delay. However, the true value of the GA approach here is to increase the utilization of the terrestrial gateways, as seen in 9. Both the GA and the hop count approach rely less on satellite gateways at the beginning and end of the simulation, when the cloud of aircraft is close to one side of the simulated area, but the GA is able to reduce the amount of traffic sent over satellites considerable over the whole duration of the simulation. This is true especially for low traffic load, when there is more room in the network to send traffic over longer paths to one of the terrestrial gateways.

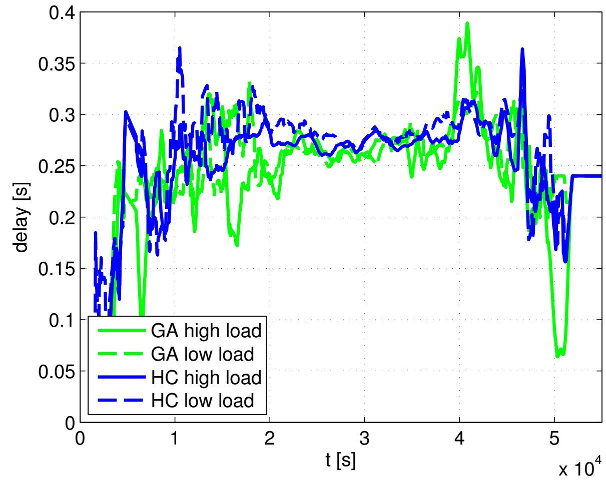


Figure 8. Average packet delay of hop count and GA based solutions.

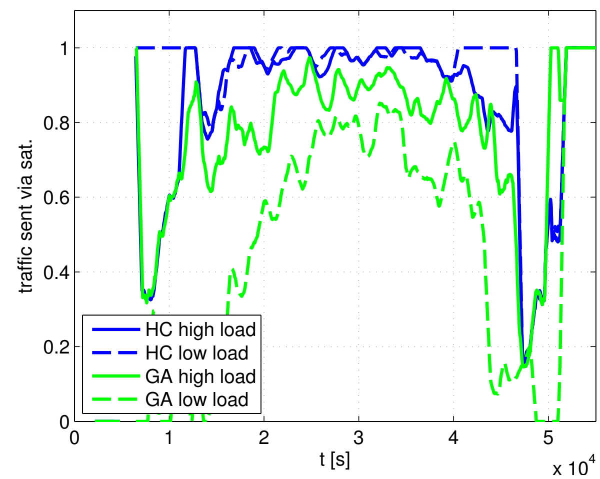


Figure 9. Fraction of traffic sent via satellite.

X. CONCLUSION

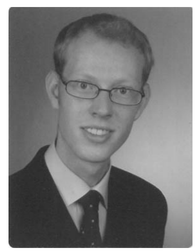
We have formulated the joint problem of gateway allocation, routing, and TDMA resource scheduling in a wireless ad hoc network with the goal of minimizing the average packet delay in the network as a mixed integer program. Due to the mathematical intractability of this problem, we have separated it into two subproblems: a weighted hop count minimization (mWHC) subject to the scheduling and interference constraints in order to optimize the routing, followed by a minimization of the average flow delay (mAFD). In addition, we have proposed a Genetic Algorithm approach to solve the delay minimization problem directly. We have assessed the performance of these approached in small test topologies and found that the performance of the GA is close to the performance of the mWHC/mAFD approach, and its complexity scales much more favorably. In more realistic large scale topologies, we showed that the GA approach succeeds in sending traffic over terrestrial air/ground gateways as long as possible and routing traffic over satellite gateways only when it becomes necessary. Although the work here was presented in the context of aeronautical ad hoc networks, the proposed GA is also applicable to other similar domains where ad hoc networks are connected to a ground network by multiple heterogeneous gateways, such as automotive networks or disaster response scenarios, where the infrastructure of cellular networks may be partially destroyed. In this case, the link model would likely need to be adapted to the more challenging channel conditions that would be encountered in a terrestrial network. The optimization performed here is of interest primarily to determine lower bounds to the achievable performance. Still, the availability of flight schedules in advance, as well as the predictable nature of the air to air radio channel could allow network optimizations of the kind discussed here to be performed offline for an operational transatlantic aeronautical ad hoc network. However, such precomputed routing and scheduling tables would likely be overly sensitive to inconsistencies between the assumptions made during the optimization process and the actual routes flown by the aircraft later on. Therefore, future work will focus on finding distributed algorithms that achieve similar performance without the requirement for global knowledge about the network in a central processor.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers, whose comments helped to greatly improve the quality of the paper.

REFERENCES

[1] E. Sakhaee and A. Jamalipour, "The Global In-Flight Internet," *Selected Areas in Communications, IEEE Journal on*, vol. 24, no. 9, pp. 1748–1757, September 2006.
[2] K. Karras, T. Kyritsis, M. Amirfeiz, and S. Baiotti, "Aeronautical Mobile Ad Hoc Networks," in *Proc. European Wireless 2008*, Prague, Czech Republic, June 2008.
[3] O. Goussevskaia, Y. Oswald, and R. Wattenhofer, "Complexity in Geometric SINR," in *MobiHoc '07: Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing*, Montreal, Canada, September 2007.
[4] F. Hoffmann, D. Medina, and A. Wolisz, "Optimization of Routing and Gateway Allocation in Aeronautical Ad Hoc Networks Using Genetic Algorithms," in *Proc. IWCMC 2011*, Istanbul, July 2011.
[5] J. Luo, C. Rosenberg, and A. Girard, "Engineering Wireless Mesh Networks: Joint Scheduling, Routing, Power Control, and Rate Adaptation," *Networking, IEEE/ACM Transactions on*, vol. 18, no. 5, pp. 1387–1400, October 2010.
[6] A. Capone, I. Filippini, and F. Martignon, "Joint Routing and Scheduling Optimization in Wireless Mesh Networks with Directional Antennas," in *ICC '08: Proceedings of the IEEE International Conference on Communications*, Beijing, China, May 2008.
[7] H. Livingstone, H. Nakayama, T. Matsuda, X. Shen, and N. Kato, "Gateway Selection in Multi-Hop Wireless Networks Using Route and Link Optimization," in *Proc. Globecom 2010*, Miami, December 2010.
[8] D. Medina, S. Ayaz, and F. Hoffmann, "Feasibility of an Aeronautical Mobile Ad Hoc Network Over the North Atlantic Corridor," in *Proc. SECON 2008*, San Francisco, USA, June 2008.
[9] R. Nelson and L. Kleinrock, "Spatial TDMA: A collision free multihop channel access protocol," *IEEE Transactions on Communications*, vol. 33, no. 9, pp. 934–944, September 1985.
[10] D. Medina, F. Hoffmann, F. Rossetto, and C.-H. Rokitansky, "A Geographic Routing Strategy for North Atlantic In Flight Internet Access Via Airborne Mesh Networking," *IEEE/ACM Transactions on Networking*, vol. 20, no. 4, pp. 1231–1244, August 2012.
[11] J. B. Cain, T. Billhartz, L. Foore, E. Althouse, and J. Schlorff, "A Link Scheduling and Ad Hoc Networking Approach Using Directional Antennas," in *Proc. Milcom*, October 2003, Monterey, USA.
[12] J. Grönkvist, "Interference-based scheduling in spatial reuse tdma," Ph.D. dissertation, KTH Stockholm, 2005.
[13] P. Djukic and S. Valaee, "Delay Aware Link Scheduling for Multi-Hop TDMA Wireless Networks," *IEEE/ACM Transactions on Networking*, vol. 17, no. 3, pp. 870–883, June 2009.
[14] L. Badia, A. Botta, and L. Lenzini, "A Genetic Approach to Joint Routing and Link Scheduling for Wireless Mesh Networks," *Ad Hoc Networks*, vol. 7, no. 4, pp. 654–664, June 2009.
[15] X. Yu and M. Gen, *Introduction to Evolutionary Algorithms*. Springer Verlag, 2010.
[16] J.-H. Lee, B.-J. Han, H.-J. Lim, Y.-D. Kim, N. Saxena, and T.-M. Chung, "Optimizing of Access Point Allocation Using Genetic Algorithmic Approach for Smart Home Environments," *The Computer Journal*, vol. 52, no. 8, pp. 938–949, 2009.
[17] R. Pries, D. Staehle, B. Staehle, and P. Tran-Gia, "On Optimization of Wireless Mesh Networks using Genetic Algorithms," *International Journal of Advances in Internet Technology*, vol. 3, no. 1, pp. 13–28, July 2010.
[18] T.-Y. Lin, K.-C. Hsieh, and H.-C.-Huang, "Applying Genetic Algorithms for Multiradio Wireless Mesh Network Planning," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 5, pp. 2256–2270, June 2012.
[19] *Lingo User's Manual*, Lindo Systems, Chicago.
[20] OMNeT++ Discrete Event Simulation System. Online. Available online at http://www.omnetpp.org.
[21] Innovata LLC. Online. http://www.innovatallc.com.
[22] *North Atlantic MNPS Airspace Operations Manual*, ICAO European and North Atlantic Office, August 2008.



Felix Hoffmann received his Diploma in electrical engineering from the Technische Universiät München in 2006. Since then, he has been working at the Institute of Communications and Navigation at the German Aerospace Center (DLR) in Oberpfaffenhofen near Munich, Germany. In parallel, he is a PhD student at the Technische Universität Berlin. His research interests are in the field of aeronautical communications in general, and the applicability of ad hoc networks in the field of civil aviation in particular.

Daniel Medina received his master's degree in Telecommunications Engineering (2005) from the Technical University of Valencia, Spain, and his PhD in Computer Science (2011) from the University of Salzburg, Austria. During his PhD, he worked for the German Aerospace Center (DLR) as a Research Engineer at the Institute of Communications and Navigation in Oberpfaffenhofen near Munich, Germany. He is currently a postdoctoral researcher at the Bernstein Center for Computational Neuroscience in the Ludwig-Maximilians-University of Munich. His current research focus is on plasticity in recurrent neural networks and the mechanisms underlying semantic learning and memory in the hippocampus.

Adam Wolisz Adam Wolisz has completed his Diploma, Doctorate and Habilitation at the Silesian University of Technology, Gliwice, Poland. In the period 1972-1989 he has been with the Polish Academy of Sciences, Institute of Foundations of Informatics, in the period 1990-2013 with GMD Fokus Berlin. Since 1993 he is chaired professor of Electrical Engineering and Computer Science at the Technische Universität Berlin, where he has founded and is leading the Telecommunication Networks Group (TKN). In the years 1993-2005 he has been in parallel also head of the department and alter member of the Directorate of the FhG Fokus. Currently he is executive director of the Institute for Telecommunication Systems, TU Berlin, grouping the activities in Communications, Networking and Distributed Systems. Since 2005 he is also adjunct professor Department of EE&CS, University of California, Berkeley (BWRC). His research interests are in architectures and protocols of communication networks as well as in protocol engineering with impact on performance (including energy efficiency) and QoS aspects. He is author of over 250 papers, and has supervised over 30 PhD dissertations. He is Senior Member of IEEE.