

Path planning for UAVs under communication constraints using SPLAT! and MILP

Esten Ingar Grøtli and Tor Arne Johansen

Department of Engineering Cybernetics

O.S. Bragstads plass 2D

N-7491 Trondheim, Norway

Email: esten.ingar.grotli@itk.ntnu.no, tor.arne.johansen@itk.ntnu.no

Abstract

We will in this paper address the problem of offline path planning for Unmanned Aerial Vehicles (UAVs). Our goal is to find paths that meet mission objectives, are safe with respect to collision and grounding, fuel efficient and satisfy criteria for communication. Due to the many nonconvex constraints of the problem, Mixed Integer Linear Programming (MILP) will be used in finding the path. Approximate communication constraints and terrain avoidance constraints are used in the MILP formulation. To achieve more accurate prediction of the ability to communicate, the path is then analyzed in the radio propagation toolbox SPLAT!, and if the UAVs are not able to communicate according to design criteria for bandwidth, constraints are modified in the optimization problem in an iterative manner. The approach is exemplified with the following setup: The path of two UAVs are planned so they can serve as relay nodes between a target without line of sight to the base station.

I. INTRODUCTION

A. Background and motivation

Applications of UAVs proposed in literature include localization of radars [1], wildfire management, polar climatology, agricultural monitoring, border surveillance [2], reconnaissance, geo-

The work is supported by Strategic University Program on Control, Information and Communication Systems for Environmental and Safety Critical Systems

physical survey, environmental and meteorological monitoring, aerial photography, and search-and-rescue tasks [3]. In general, autonomous vehicles are chosen for tasks that are either *dirty, dull or dangerous*, or to missions where there is a cost reduction potential. Our research is mainly motivated by emerging applications such as Arctic offshore oil exploration, oil spill recovery, and complex long range multi-vehicle missions in areas without a permanent communication infrastructure. The newly signed treaty of the delimitation in the Barents sea between Norway and Russia, and recent technological improvements will make new areas available for oil explorations. With the polar ice melting, the same area could potentially be the favored traffic route between eastern Asia and Europe. The harsh weather of the Barents sea require enhanced monitoring and preparedness and reduced mobilization time, to ensure safety of personnel and material in case of accidents, and to avoid environmental catastrophes in one of the world richest marine ecosystems. Satellite monitoring and communication is currently disfavored due to poor coverage of the Arctic regions and lack of immediate presence due to their orbiting nature. The low bandwidth available for civilian users, typically 2-3 kbits/s and 3-4 seconds latency using a civilian system such as Iridium, is another reason why satellite communication alone is currently inadequate for missions that require extensive exchange of high resolution, real time data. Manned aircraft, on the other hand, are too costly and require too much of human resources for extensive operations in large areas.

B. Terminology

For later reference we will present the following terminology taken from [2]: *Meshing*: a networking architecture where each node can act as a relay to forward data; *Mobile ad-hoc network (MANET)*: mesh architecture comprised of a self-configuring network of mobile routers which are free to move randomly throughout the environment; *Direct communication*: Occurs when two nodes transmit data directly to one another; *Relaying*: Occurs when additional nodes are used to receive a transmission from a source and retransmit it to a destination; *Ferrying*: Occurs when a mobile node physically stores and carries data from one location to another.

If the nodes are separated by a distance or obstacles that have major influence on the communication bandwidth and real-time availability is not an issue, ferrying may be preferred to relaying and direct communication. Two different ferry path models are treated in [4], namely the *chain-relay model* and the *conveyor-belt model*.

C. Previous work and available technology

There are several simulators available for MANET community, see for instance [5]. Both the most popular open source network simulator, NS-2 and the most popular commercial network simulator, OPNET, has several radio propagation models available, where the latter has been evaluated for an UAV network in [6]. The impact of radio propagation model on MANET simulations, can be found in [7], [8] and [9]. For military applications there are several toolboxes developed for planning with respect to communication conditions, see for instance [10], [11] or the commercially available HTZ Warfare by ATDI. Similar problems to the one addressed in this paper, has been treated in [12], where relay chains are generated solving the *all hops optimal path* (AHOP) graph search problem. Two algorithms are presented that solves the AHOP problem; a dual ascent algorithm and a modification of the Bellman-Ford algorithm. Location and movement of UAVs are optimized in [13], in order to improve the connectivity of a wireless network, when quantifying different types of network connectivity. Optimizing communication by controlled mobility of network nodes is also addressed in [14], and in [15] the optimal placement of fixed relay nodes is treated. In [16] a decentralized extremum-seeking control algorithm for nonholonomic vehicles to form a communication chain is presented. In [17] path planning and path finding algorithms for multiple UAVs are studied. The performance of the algorithms, Dijkstra's algorithm, Bellman Ford's algorithm, Floyd Warshall's algorithm and the A* algorithm, are compared.

In this paper, MILP is used for path planning. In this context the following papers are particularly relevant: For a general introduction to mixed-integer programming for control, [18], [19] and [20] are recommended. In [21] MILP is used for UAV path planning, while constraining the probability of detection. In [22] the coordination and control of multiple UAVs are solved using MILP. Online connectivity-constrained trajectory planning for autonomous helicopters through cluttered environment are treated in [23]. In [24] MILP is used for optimizing the task allocation problem for a fleet of UAVs with tightly couples tasks and rigid relative timing constraints. In [25] the differential flatness property of the planar kinematic model of an UAV is used to derive linear constraints on the rotational velocity. MILP formulations for real-time mission scheduling of UAVs in hostile environment is treated in [26]. In [27] a optimal control problem of multiple UAVs under terrain- and communication constraints is transformed into

a MILP problem. Although the number of binary variables is often a poor indicator of the complexity of integer programming problems [18], it *might* be useful to reduce this number as the number of nodes in the solution tree grows exponentially with the number of binary variables. Some heuristic solutions related to this for cooperative vehicles are presented in [28]. See also [29] for an iterative MILP algorithm that use fewer binary variables than standard MILP methods, and require less computational effort. A different strategy to deal with the computational complexity of path planning by MILP is treated in [30], where a tunnel of convex polytopes based on a shortest path to the goal is used as constraints in the MILP formulation.

D. Contribution

In this paper we deal with some of the assumptions commonly made about wireless networks, [31]: “The world is flat”; “A radio’s transmission area is circular”; “All radios have equal range”; “If I can hear you, you can hear me”; “If I can hear you at all, I can hear you perfectly”; and “Signal strength is a simple function of distance”. MILP is used for path planning for UAVs where a simple function for the communication range is considered. We then use SPLAT! [32] to give an estimate of the path loss between each vehicle at every time sample. Based on the calculations by SPLAT! new constraints are added to the MILP problem to restrict the distance between vehicles at time samples where the path loss is assumed to be too high to maintain communication at the desired rate. The process is then repeated until paths are found in which communication can be maintained by a predefined criterion. The Windows version of SPLAT! we use were provided by [33]. SPLAT! uses digital elevation data, for instance from the Space Shuttle Radar Topography Mission (SRTM), to calculate field strength and path loss based on the Longley-Rice Irregular Terrain Model [34]. To easily toggle between different solvers within the MATLAB environment, the modeling language YALMIP is used [35]. The solvers used were the Gurobi Optimizer 4.0¹ and IBM ILOG CPLEX Optimization Studio 12.2². We will like to emphasize that we in this paper do not intend to perform a comparison of different solvers, but to some extent seek to test if different strategies are beneficial independent of the solver used. For an overview of available solvers for MILP problems, the interested reader is referred to [36].

¹<http://www.gurobi.com>

²<http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/>

The interface for Gurobi were provided by [37]. We also would like to make the reader aware of that the solution to the MILP optimization problem is not necessarily unique, and that since integer programming in general is NP hard, so is MILP.

II. COORDINATE SYSTEMS

We will make use of the following coordinate systems:

ECEF: The Earth-Centered Earth-Fixed frame, is as the name indicates, a coordinate system which origin is at the center of Earth, and with axes rotating with the angular velocity of Earth.

ENU: The East-North-Up coordinate frame is a local geodetic coordinate system whose tangent plane is fitted to the geodetic reference ellipse at some convenient point for local measurements. The x axis points towards east, the y axis points towards true north and the z axis completes the right handed orthogonal frame by pointing away from the Earth perpendicular to the reference ellipsoid.

The World Geodetic System (WGS) is a standard which defines a reference ellipsoid used for navigation, geodesy and cartography. The SRTM digital elevation maps contains data on different formats, but in this paper we have used data whose elevation is with respect to the reference ellipsoid defined in WGS84. The terrestrial presentation is in terms of the ellipsoidal parameters longitude l , (geodetic) latitude μ and ellipsoidal height h . The transformation between these parameters and the coordinates of the ENU frame is a two-step procedure. First longitude, latitude and height are transformed into the coordinates of the ECEF frame. For this, and the reverse transformation, we use the toolbox accompanying [39]. Then the ECEF coordinates are transformed into coordinates of the ENU frame by using the rotation matrix

$$R_{ECEF}^{ENU} = \begin{bmatrix} -\sin l & \cos l & 0 \\ -\sin \mu \cos l & -\sin \mu \sin l & \cos \mu \\ \cos \mu \cos l & \cos \mu \sin l & \sin \mu \end{bmatrix} \quad (1)$$

which is found by two principal rotations; a rotation $\mu - \pi/2$ about the x axis, and then a rotation $-l - \pi/2$ about the z axis.

The origin of the ENU frame is in this paper chosen to be at longitude, latitude and height, specified by the vector $\mathbf{X}^{\text{WGS}} = (63.400, 10.32, 0)$.

III. MILP FORMULATION

At the planning problem considered in this paper we assume that the p^{th} UAV is simply described by the discrete time model

$$\mathbf{p}_{p(i+1)} = \mathbf{p}_{pi} + \Delta t \mathbf{v}_{pi}, \quad (2)$$

$\forall p \in \{1, \dots, n_p\}$, where n_p is the number of UAVs, Δt is the sample time, and $\mathbf{p}_{pi} := (x_{pi}, y_{pi}, z_{pi})^\top$, and $\mathbf{v}_{pi} := (v_{1pi}, v_{2pi}, v_{3pi})^\top$, with x_p, y_p, z_p and v_{1p}, v_{2p}, v_{3p} being the positions and velocities, respectively, along the orthogonal axes of a local ENU coordinate reference frame.

A. Fuel penalty

For an aircraft in transit, the fuel consumption will be proportional to the air resistance, which again is assumed proportional to the square of the velocity. For small deviations about the typical cruise speed V_p^{cruise} of vehicle p , we have the following approximation of the square of the velocity at time step i using the first terms of the Taylor series: $\|\mathbf{v}_{pi}\|^2 = V_{pi}^2 \approx (V_p^{\text{cruise}})^2 + 2V_p^{\text{cruise}}(V_{pi} - V_p^{\text{cruise}})$. To penalize the fuel consumption we propose the cost function term

$$J^{\text{drag}} = \sum_{p=1}^{n_p} \sum_{i=0}^{N-1} t_p ((V_p^{\text{cruise}})^2 + 2V_p^{\text{cruise}}(V_{pi}^{\text{approx}} - V_p^{\text{cruise}})) , \quad (3)$$

where $t_p \in \mathbb{R}_{\geq 0}$ is a weighting factor, $N := T/\Delta t$, with T is the horizon over which we optimize, and V_{pi}^{approx} is an approximation of the true magnitude of the velocity vector, V_{pi} . The optimization variable V_{pi}^{approx} is found in a similar manner as in [21], here extended to the three dimensional case, by introducing the constraints:

$$\mathbf{v}_{pi}^\top \boldsymbol{\xi}_{kl} \leq V_{pi}^{\text{approx}}, \quad (4)$$

$$\alpha^{\text{vel}} \mathbf{v}_{pi}^\top \boldsymbol{\xi}_{kl} \geq V_{pi}^{\text{approx}} - M_{pkl}^{\text{vel}}(1 - b_{pikl}^{\text{vel}}), \quad (5)$$

$\forall p \in \{1, \dots, n_p\}$, $i \in \{0, \dots, N-1\}$, $k \in \{1, \dots, D^{\text{vel}}/2\}$, $l \in \{1, \dots, D^{\text{vel}}\}$, where $b_{pikl}^{\text{vel}} \in \{0, 1\}$ are optimization variables, D^{vel} is some constant even integer greater or equal to four, and

$$\boldsymbol{\xi}_{kl} := \begin{bmatrix} \cos(\theta_k) \sin(\phi_l) \\ \sin(\theta_k) \sin(\phi_l) \\ \cos(\phi_l) \end{bmatrix}, \quad (6)$$

with $\theta_k := 2\pi k/D^{\text{vel}}$ and $\phi_l := 2\pi l/D^{\text{vel}}$. The accuracy of the approximation depends on α^{vel} which is a constant slightly greater than one. The closer to one α^{vel} is, the better is the approximation, however, taking it too close may have a negative impact on the computation time of the MILP problem. Furthermore, we require that

$$\sum_{k=1}^{D^{\text{vel}}} \sum_{l=1}^{D^{\text{vel}}/2} b_{pikl}^{\text{vel}} = 1, \quad (7)$$

$\forall p \in \{1, \dots, n_p\}, i \in \{0, \dots, N-1\}$. In theory, the constant M_{pkl}^{vel} could be taken arbitrarily large, but for computational efficiency this is not recommended [18, Page 196]. We can for instance pick

$$M_{pkl}^{\text{vel}} := \max_{\substack{v_{1pi}, v_{2pi}, v_{3pi} \in [-\bar{V}_p, \bar{V}_p] \\ V_{pi}^{\text{approx}} \in [\underline{V}_p, \bar{V}_p]}} \{ \alpha^{\text{vel}} \mathbf{v}_{pi}^\top \boldsymbol{\xi}_{kl} - V_{pi}^{\text{approx}} \} \quad (8)$$

where \underline{V}_p and \bar{V}_p are the minimum and maximum velocity, respectively, of vehicle p . In YALMIP, logic implications can be expressed instead of big-M formulations such as (5), and YALMIP will automatically derive big-M coefficients by analyzing the expression. As pointed out in [40], provided that the objective function favors minimum time solutions, the two-norm approximation of a two dimensional vector requires D constraints at each time step, where as extending this approach to three dimensions typically requires $D^2/2$ constraints. In our case, minimum time solutions are not necessarily favored in the objective function, and we also need the additional constraints in (5), meaning that the approximation requires a total of D^2 constraints. If computation time is of major importance, it is in that respect better to use the approximation in [41] which only requires $3D$ constraints, or strategically allocate the constraints as in [42] to improve the two norm approximation in a direction of interest.

Beyond cruising, fuel consumption is assumed to be proportional to the acceleration, but this is only of secondary importance in our setup so we will pursue a simpler approach than for penalizing the velocity magnitude. The following cost function term is proposed in [43]:

$$J^{\text{acc}} = \sum_{p=1}^{n_p} \sum_{i=0}^{N-2} \mathbf{r}_p^\top \mathbf{w}_{pi}^{\text{acc}}, \quad (9)$$

with the additional constraints

$$(v_{jpk} - v_{jpi}) \leq w_{jpi}^{\text{acc}}, \quad (10)$$

$$-(v_{jpk} - v_{jpi}) \leq w_{jpi}^{\text{acc}}, \quad (11)$$

$\forall p \in \{1, \dots, n_p\}$, $k = i + 1$, $i \in \{0, \dots, N - 2\}$, $j \in \{1, 2, 3\}$ where $\mathbf{w}_{pi}^{\text{acc}} := (w_{1pi}^{\text{acc}}, w_{2pi}^{\text{acc}}, w_{3pi}^{\text{acc}})^\top$ and $\mathbf{r}_p \in \mathbb{R}_{\geq 0}^3$ is a nonnegative weighting vector. The motivation behind (9) is to penalize the absolute value of acceleration in each direction of the ENU frame, and to avoid a piecewise linear cost function, we have introduced slack variables w_{jpi}^{acc} .

At last we will include a term to account for the gravitational effects, namely,

$$J^{\text{gravity}} = \sum_{p=1}^{n_p} g_p (z_{p0} - z_{p(N-1)}), \quad (12)$$

where $g_p \in \mathbb{R}_{\geq 0}$ is a weighting factor of vehicle p , and z_{p0} and $z_{p(N-1)}$ are the initial and final position of vehicle p in the up-direction. Of simplicity we will use that the achieved potential energy, is approximately the energy spent by climbing.

B. Position and speed constraints

There are typically restrictions on where UAVs are allowed to fly. This may for instance be air space used for commercial air traffic, air space over a certain altitude, or air space over densely populated areas. In addition the operator might want to avoid flying into regions with bad weather, outside the area where the operator is able to communicate with UAVs or in case of military applications: areas with enemies and enemy radars. If the region the UAVs are required to stay within is convex (e.g. a rectangular box), the constraints may simply be written

$$\underline{x} \leq x_{pi} \leq \bar{x}, \quad (13)$$

$$\underline{y} \leq y_{pi} \leq \bar{y}, \quad (14)$$

$$\underline{z} \leq z_{pi} \leq \bar{z}, \quad (15)$$

$\forall p \in \{1, \dots, n_p\}$, $i \in \{1, \dots, N\}$, where $\underline{x}, \underline{y}, \underline{z}$ and $\bar{x}, \bar{y}, \bar{z}$ are the lower and upper bounds, respectively, on the state vector in the east, north and up directions. $\underline{x}, \underline{y}, \underline{z}$ and $\bar{x}, \bar{y}, \bar{z}$ are constants provided to the optimizer at start-up.

More complex (nonconvex) regions can be defined as unions of rectangles by introducing integer auxiliary variables. An obstacle, denoted by o , which the UAVs are required to avoid, may be approximated by the outside of a rectangular box, which leads to the following

constraints

$$\bar{x}_o - M_{o1}^{\text{obs}} b_{pio1}^{\text{obs}} \leq x_{pi} \leq \underline{x}_o + M_{o2}^{\text{obs}} b_{pio2}^{\text{obs}}, \quad (16)$$

$$\bar{y}_o - M_{o3}^{\text{obs}} b_{pio3}^{\text{obs}} \leq y_{pi} \leq \underline{y}_o + M_{o4}^{\text{obs}} b_{pio4}^{\text{obs}}, \quad (17)$$

$$\bar{z}_o - M_{o5}^{\text{obs}} b_{pio5}^{\text{obs}} \leq z_{pi} \leq \underline{z}_o + M_{o6}^{\text{obs}} b_{pio6}^{\text{obs}}, \quad (18)$$

$$\sum_{l=1}^6 b_{piol}^{\text{obs}} \leq 5, \quad (19)$$

$\forall p \in \{1, \dots, n_p\}$, $i \in \{1, \dots, N\}$, $o \in \{1, \dots, n_o\}$, where n_o is the number of obstacles, $b_{pil}^{\text{obs}} \in \{0, 1\}$ are variables of the optimization problem, $\underline{x}_o, \underline{y}_o, \underline{z}_o$ and $\bar{x}_o, \bar{y}_o, \bar{z}_o$ are the lower and upper bounds, respectively, on obstacle o in the east, north and up directions. (See for instance [44] for avoidance of more general obstacles.) We pick $M_{o1}^{\text{obs}} := \max_{x_{pi} \in [\underline{x}, \bar{x}]} \{\bar{x}_o - x_{pi}\} = \bar{x}_o - \underline{x}$ and similar for M_{o2}^{obs} to M_{o6}^{obs} .

In [45] and [46], among others, the (two dimensional) velocity vector is required to be within a regular polygon which over-approximates the circle of radius equal to the maximum speed. A polygon that under-approximates this circle is used in [21]; see also [25] for a different approach. In [3], the velocity constraints are extended to three dimensions, by requiring that three dimensional velocity vector is within a polyhedron whose radius approximates the maximum speed. As the speed of the vehicles are approximated by (4), (5) and (7), we simply use that

$$\underline{V}_p \leq V_{pi}^{\text{approx}} \leq \bar{V}_p, \quad (20)$$

$\forall p \in \{1, \dots, n_p\}$, $i \in \{0, \dots, N-1\}$, where \underline{V}_p and \bar{V}_p are the minimum and maximum velocity, respectively, of vehicle p . We have chosen to ignore the physical properties of the UAVs of being able to descend faster than they can climb.

C. Connectivity constraints

The approach taken in this section is motivated by the approach of [27]. We will in the following refer to base stations and vehicles involved in the radio communication as *nodes*. Of simplicity, we will assume that the ability of node p to successfully transmit data at a specified rate to node q , at some time instance i , depends on whether the relative distance between the two nodes are below a certain threshold, R_{pqi} . This threshold would typically depend on the antenna gains of the receiver and transmitter node, surrounding terrain, data rate, etc. We stress that R_{pqi}

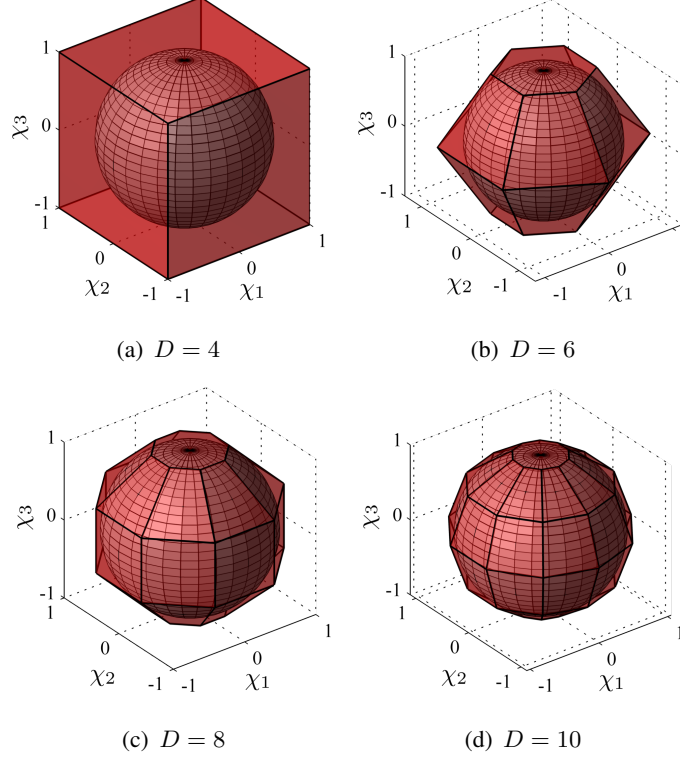


Fig. 1. The constraints $\chi^\top \xi_{kl} \leq 1, \forall k \in \{1, \dots, D/2\}, l \in \{1, \dots, D\}$ with discretization level: (a) $D = 4$, (b) $D = 6$, (c) $D = 8$ and (d) $D = 10$.

is not necessarily equal to R_{ppi} , that is, the threshold depends on the direction of communication. Instead of requiring that node q are within a sphere of radius R_{pqi} of node p , we require that node q is within a polygon that approximates the sphere. The approximation is formed by taking the inner product of the vector $\chi_{pqi} := (x_{pi} - x_{qi}, y_{pi} - y_{qi}, z_{pi} - z_{qi})^\top$ and the unit vectors ξ_{kl} defined by (6) where $\theta_k := 2\pi k/D^{\text{con}}$ and $\phi_l := 2\pi l/D^{\text{con}}$ and $k \in \{1, \dots, D^{\text{con}}/2\}$, $l \in \{1, \dots, D^{\text{con}}\}$ and the discretization level D^{con} is some constant even integer greater or equal to 4. The polygon over-approximates the sphere as seen in Figure 1, where the radius is simply taken as unity. We also introduce binary indicator variables $\tilde{b}_{pqi}^{\text{con}} \in \{0, 1\}$ such that

$$\tilde{b}_{pqi}^{\text{con}} = 1 \iff \chi_{pqi}^\top \xi_{kl} - R_{pqi} \leq 0, \quad (21)$$

$\forall p \in \{0, \dots, n_p+1\}$, $q \in \{p+1 | p \neq n_p+1\} \cup \{p-1 | p \neq 0\}$, $i \in \{1, \dots, N\}$, $k \in \{1, \dots, D^{\text{con}}/2\}$, $l \in \{1, \dots, D^{\text{con}}\}$ that is, the indicator variable $\tilde{b}_{pqi}^{\text{con}}$ is *true* if and only if, node p can directly transmit to vehicle q , where $p = 0$ denote the base station and $p = n_p + 1$, denote the target. Relating this to graph theory, the matrix A_i , with elements $[A_i]_{pq} = \tilde{b}_{pqi}^{\text{con}}$, is the adjacency matrix

at time step i . The logical statement in (21) can be achieved by the introduction of a number of additional optimization variables b_{pqikl}^{con} , see Proposition 1. In the example of Section V, we require an ordered communication chain, and the indicator variables will have to satisfy the constraints

$$\mathbf{x}_{pqi}^\top \boldsymbol{\xi}_{kl} - R_{pqi}^{\text{con}} \leq M_{pqkl}^{\text{con}}(1 - b_{pqikl}^{\text{con}}), \quad (22)$$

$$\mathbf{x}_{pqi}^\top \boldsymbol{\xi}_{kl} - R_{pqi}^{\text{con}} \geq \varepsilon + (m_{pqkl}^{\text{con}} - \varepsilon)b_{pqikl}^{\text{con}}, \quad (23)$$

$$\tilde{b}_{pqi}^{\text{con}} \leq b_{pqikl}^{\text{con}}, \quad (24)$$

and

$$\sum_{k=1}^{D^{\text{con}}} \sum_{l=1}^{D^{\text{con}}/2} b_{pqikl}^{\text{con}} - \tilde{b}_{pqi}^{\text{con}} \leq \frac{(D^{\text{con}})^2}{2} - 1, \quad (25)$$

$\forall p \in \{0, \dots, n_p + 1\}$, $q \in \{p + 1 | p \neq n_p + 1\} \cup \{p - 1 | p \neq 0\}$, $\forall i \in \{1, \dots, N\}$, $k \in \{1, \dots, D^{\text{con}}/2\}$, $l \in \{1, \dots, D^{\text{con}}\}$, where ε is a small positive scalar, $m_{pqkl}^{\text{con}} := \min\{\mathbf{x}_{pqi}^\top \boldsymbol{\xi}_{kl} - R_{pqi}^{\text{con}}\}$ and $M_{pqkl}^{\text{con}} := \max\{\mathbf{x}_{pqi}^\top \boldsymbol{\xi}_{kl} - R_{pqi}^{\text{con}}\}$, with the minimum and maximum taken over all admissible \mathbf{x}_{pqi} . We now introduce the additional optimization variable $b_i^{\text{chain}} \in \{0, 1\}$ at each time step i , and the constraints

$$\sum_{p=0}^{n_p} \tilde{b}_{pqi}^{\text{con}} - b_i^{\text{chain}} \leq (n_p + 1) - 1, \quad (26)$$

$\forall i \in \{1, \dots, N\}$ and $q = p + 1$

$$\sum_{p=1}^{n_p+1} \tilde{b}_{pqi}^{\text{con}} - b_i^{\text{chain}} \leq (n_p + 1) - 1, \quad (27)$$

$\forall i \in \{1, \dots, N\}$ and $q = p - 1$

$$b_i^{\text{chain}} \leq \tilde{b}_{pqi}^{\text{con}}, \quad (28)$$

$\forall i \in \{1, \dots, N\}$, $p \in \{0, \dots, n_p\}$, and $q = p + 1$, and

$$b_i^{\text{chain}} \leq \tilde{b}_{pqi}^{\text{con}}, \quad (29)$$

$\forall i \in \{1, \dots, N\}$, $p \in \{1, \dots, n_p + 1\}$, and $q = p - 1$. These constraints enforces the binary optimization variable b_i^{chain} to be true if and only if there is a two-way communication chain between the base station and the target. We also add the cost function term

$$J^{\text{chain}} = \gamma^{\text{chain}} \sum_{i=1}^N (1 - b_i^{\text{chain}}), \quad (30)$$

where γ^{chain} is a weighting factor to our objective function, which emphasizes the importance for being within each others communication range. Notice that our approach differs from the approach of [27], in that we implement communication as a soft constraint. This may result in slightly more unreliable communication being favored, but allows the nodes to be outside of each others communication range at start-up.

It is common to assume, see for instance [48], that the communication connectivity constraints only depends on the relative distance between the vehicles. We emphasize that this assumption, although used in the above constraints, will be relaxed later in this paper.

D. Anti-collision constraints

To avoid collision between vehicles we will implement the method of [43]. Let that the position of vehicle p and vehicle q at the i^{th} time-step be given by (x_{pi}, y_{pi}, z_{pi}) and (x_{qi}, y_{qi}, z_{qi}) , respectively. The constraints are given as

$$d_x - M_{pq1}^{\text{col}} b_{pqi1}^{\text{col}} \leq x_{pi} - x_{qi} \leq M_{pq2}^{\text{col}} b_{pqi2}^{\text{col}} - d_x, \quad (31)$$

$$d_y - M_{pq2}^{\text{col}} b_{pqi3}^{\text{col}} \leq y_{pi} - y_{qi} \leq M_{pq4}^{\text{col}} b_{pqi4}^{\text{col}} - d_y, \quad (32)$$

$$d_z - M_{pq3}^{\text{col}} b_{pqi5}^{\text{col}} \leq z_{pi} - z_{qi} \leq M_{pq6}^{\text{col}} b_{pqi6}^{\text{col}} - d_z, \quad (33)$$

$$\sum_{l=1}^6 b_{pqi l}^{\text{col}} \leq 5 \quad (34)$$

where $b_{pqi l}^{\text{col}} \in \{0, 1\}$ are binary variables of the optimization algorithm, $\forall p \in \{1, \dots, n_p - 1\}, q \in \{p + 1, \dots, n_p\}, i \in \{1, \dots, N\}$, where d_x, d_y and d_z are the safety distances in the east, north and up directions, respectively. The minimum separation distance while still maintaining the possibility for avoidance maneuvers can for instance be calculated by reachability analysis, see [49]. The constant M_{pq1}^{col} can for instance be chosen as $M_{pq1}^{\text{col}} := \bar{x} - \underline{x} + d_x$ with \bar{x}, \underline{x} as in (13), and correspondingly for M_{pq2}^{col} to M_{pq6}^{col} .

E. Anti grounding constraints

As in [44] we will represent the terrain as triangulated irregular network (TIN). Terrain avoidance constraints in MILP form are given in [50], and will be used here. T^{TIN} non-overlapping triangles with m^{TIN} vertices $P_l(x_l^{\text{TIN}}, y_l^{\text{TIN}}, h_l^{\text{TIN}})$ are used to represent the piecewise affine terrain

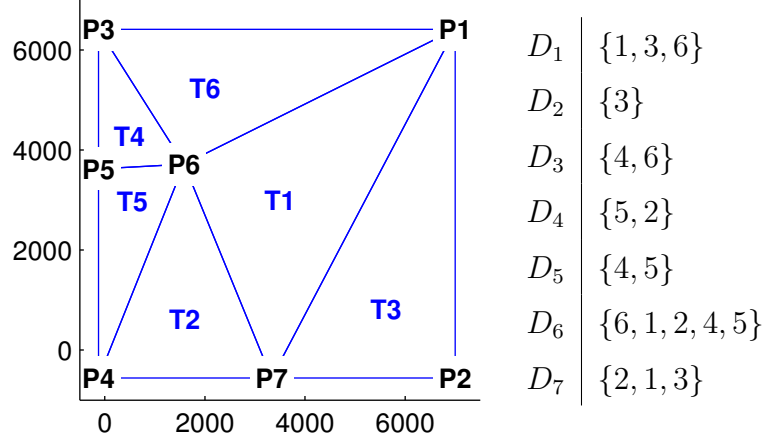


Fig. 2. Top view of a TIN with vertices P_l , $l \in \{1, \dots, 7\}$, and to the right the corresponding sets D_l of triangles which have P_l as common vertex.

surface. The point strictly below vehicle p at time step i is given by (x_{pi}, y_{pi}, h_{pi}) , and satisfy

$$x_{pi} = \sum_{l=1}^{m^{\text{TIN}}} \lambda_{pil}^{\text{TIN}} x_l^{\text{TIN}}, \quad (35)$$

$$y_{pi} = \sum_{l=1}^{m^{\text{TIN}}} \lambda_{pil}^{\text{TIN}} y_l^{\text{TIN}}, \quad (36)$$

$$h_{pi} = \sum_{l=1}^{m^{\text{TIN}}} \lambda_{pil}^{\text{TIN}} h_l^{\text{TIN}}, \quad (37)$$

$$\sum_{l=1}^{m^{\text{TIN}}} \lambda_{pil}^{\text{TIN}} = 1, \quad (38)$$

$$\lambda_{pil}^{\text{TIN}} \leq \sum_{t \in \mathcal{D}_l} b_{pit}^{\text{TIN}}, \quad \forall l = \{1, \dots, m^{\text{TIN}}\}, \quad (39)$$

$$\sum_{t=1}^{T^{\text{TIN}}} b_{pit}^{\text{TIN}} = 1, \quad (40)$$

and $\lambda_{pil}^{\text{TIN}} \in [0, 1]$, $\forall p \in \{1, \dots, n_p\}$, $i \in \{1, \dots, N\}$, where the binary variables $b_{pit}^{\text{TIN}} \in \{0, 1\}$ corresponds to triangle t , and \mathcal{D}_l is the set of the serial-numbers of triangles that have common vertex P_l . An example of this is shown in Figure 2. Finally, the terrain avoidance constraint can be expressed as

$$z_{pi} \geq h_{pi} + d^{\text{TIN}}, \quad (41)$$

where d^{TIN} is the minimum vertical distance from the UAVs to the ground. We emphasize that x_{pi} , y_{pi} , z_{pi} are the position of the vehicle p at time step i as defined in the beginning of Section III and are variables of the optimization problem. Furthermore, in the terrain anti-grounding problem, we have introduced the following variables of the optimization problem: h_{pi} , $\lambda_{pil}^{\text{TIN}}$ and b_{pit}^{TIN} . The constant d^{TIN} representing the safety distance between the UAVs and the ground, and the constants x_l^{TIN} , y_l^{TIN} , h_l^{TIN} which represents the coordinates of the triangle's vertices, are known *a priori*. The TIN is generated by Delaunay triangulation using the elevation data. To reduce the complexity of the problem, only a subset of the data available are used in the MILP formulation. Based on incremental Delaunay triangulation, implemented in [51], points from the

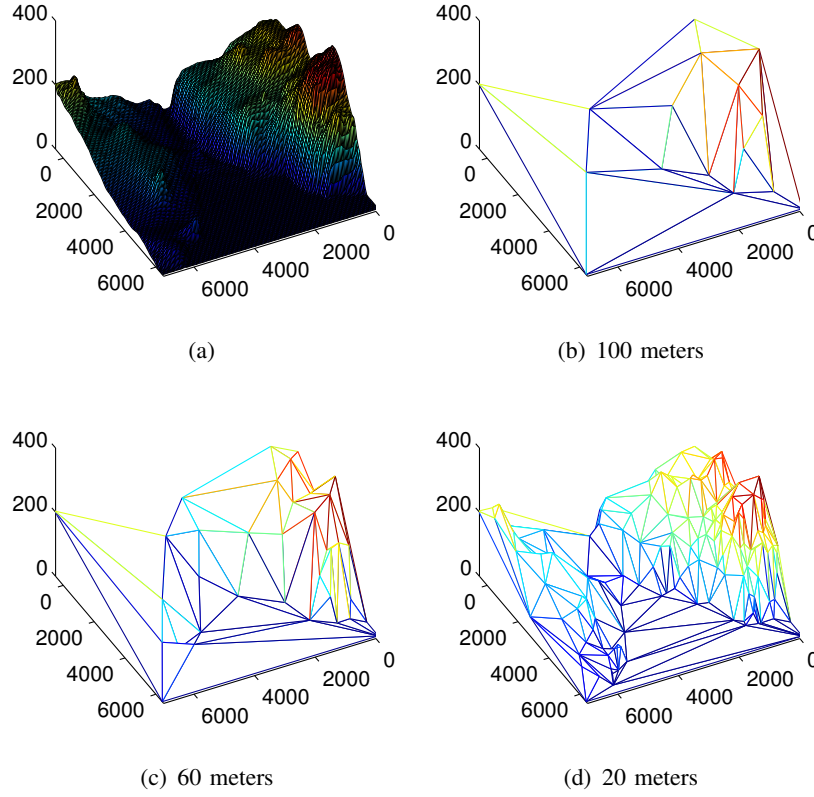


Fig. 3. Elevation data is shown in (a). The other figures show the generated TIN with labels denoting maximum relative vertical error to the original data at its sample points. This requires (b) $m^{\text{TIN}} = 16$ vertices and $T^{\text{TIN}} = 24$ triangles, (c) $m^{\text{TIN}} = 37$ vertices and $T^{\text{TIN}} = 62$ triangles and (d) $m^{\text{TIN}} = 169$ vertices and $T^{\text{TIN}} = 318$ triangles.

elevation data is added until the desired vertical accuracy between the data and the triangulation is achieved. This process is shown in Figure 3. Special care must be taken by low altitude flight,

because although (41) is satisfied at each time step i , a linear interpolation between two time steps may intersect the ground, if the ground is locally concave down. In this paper we will not address this issue, but simply choose the vertical distance from the UAVs to the ground sufficiently large, and the timesteps sufficiently small.

F. Simplification of the MILP problem

In this section we will investigate different strategies for reducing the computation time of the MILP problem. We will investigate the effect of taking into consideration the initial position and the dynamics of the vehicles, which allows us to decide the values of some of the binary variables.

First we will look at the binary variables related to communication. As mentioned in Section I-D, the path planning problem will be solved in an iterative manner. Of simplicity we assume that the communication radius at the first iteration is independent of time, that is $R_{pqi}^{\text{con}} =: R_{pq}^{\text{con}}$. Then, we define d^{pre} as the difference between the initial distance between vehicle p and q and R_{pq}^{con} , that is

$$d_{pq}^{\text{pre}} := \|\mathbf{p}_p(0) - \mathbf{p}_q(0)\| - R_{pq}^{\text{con}}. \quad (42)$$

If node q is outside a radius R_{pq}^{con} from node p initially, such that $d_{pq}^{\text{pre}} > 0$, it will stay outside at least for the time

$$t^{\text{pre}} = \text{floor} \left\{ \frac{|d^{\text{pre}}|}{(\bar{V}_p + \bar{V}_q)} \right\}, \quad (43)$$

where \bar{V}_p and \bar{V}_q is the maximum velocity of node p and q , respectively, and is zero for the static nodes. We can then be sure that $\tilde{b}_{pqi}^{\text{con}} = 1$ for all $i \leq t^{\text{pre}}/\Delta t$. If, on the other hand, $d_{pq}^{\text{pre}} \leq 0$, then we will have that $\tilde{b}_{pqi}^{\text{con}} = 0$ and $b_i^{\text{chain}} = 0$ for all $i \leq t^{\text{pre}}$. The stage is summarized in Algorithm 1 in the Appendix.

Furthermore, we can also determine some of the binary variables b_{pit}^{TIN} , by considering the initial position and the maximum velocity of each vehicle, and the time horizon over which we optimize. More precisely, we assign $b_{pit}^{\text{TIN}} = 0$ if the horizontal distance from the initial position of vehicle p , given by \mathbf{p}_{p0} to triangle t is greater than the distance the vehicle can cover flying at maximum velocity, \bar{V}_p in i time steps. The horizontal distance between the initial point of vehicle p and the triangle t is calculated using the function $\text{pointTriangleDist}(\text{Tri}_t, \mathbf{p}_{p0})$, see for

instance [52], where

$$\text{Tri}_t := \begin{bmatrix} x_\alpha^{\text{TIN}} & y_\alpha^{\text{TIN}} & z_{p0} \\ x_\beta^{\text{TIN}} & y_\beta^{\text{TIN}} & z_{p0} \\ x_\gamma^{\text{TIN}} & y_\gamma^{\text{TIN}} & z_{p0} \end{bmatrix}, \quad (44)$$

and where $x_\alpha^{\text{TIN}}, x_\beta^{\text{TIN}}, x_\gamma^{\text{TIN}}$ and $y_\alpha^{\text{TIN}}, y_\beta^{\text{TIN}}, y_\gamma^{\text{TIN}}$, are the east and north components, respectively, of the vertices of triangle t . This is summarized in Algorithm 2 in the Appendix. Notice that we have only considered horizontal flight, that is, we have set the up coordinate of each vertex to be the the initial altitude vehicle p , z_{p0} . By also taking into account the terrain height and initial altitude of the UAVs, better approximations of the time to reach over a specific triangle can be calculated, and further reduced computation time can be expected. If the triangles, above which the UAVs are initially, are sufficiently large, it may also be beneficial to calculate the least time they will stay over these triangles. This would allow us to assign $b_{pit}^{\text{TIN}} = 1$ for some triplets of p, i and t .

IV. STRUCTURE OF ALGORITHM

The algorithm consists of several different stages, see the flowchart in Figure 4, and is summarized in Algorithm 3 in the Appendix.

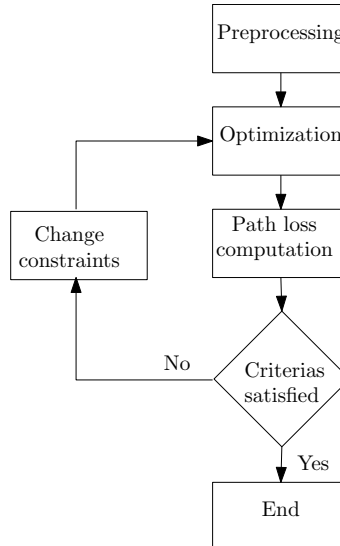


Fig. 4. Flow chart of Algorithm 3

A. Preprocessing of MILP problem

At this stage, we will incorporate the procedures of Section III-F with respect to reducing computation time.

B. Optimization

At this stage the cost is given by the sum of (3), (9), (12) and (30), i.e.

$$J = J^{\text{drag}} + J^{\text{acc}} + J^{\text{gravity}} + J^{\text{chain}} \quad (45)$$

is minimized, subject to (2), (4), (5), (7), (10), (11), (13)-(20), (22)-(29) and (31)-(41). In addition, when solving the MILP repeatedly, we will from the second iteration onwards add the constraint

$$J^{\text{chain}} \leq J^{\text{chain,prev},*} + \gamma^{\text{chain}} \epsilon^{\text{chain}} \quad (46)$$

where $J^{\text{chain,prev},*}$ is the cost corresponding to the optimal solution at the previous iteration, and ϵ^{chain} is a positive integer. The motivation for introducing a bound on J^{chain} , is that it is beneficial from a computation time perspective, as will be shown in Section V-B. Finding a tight over-approximation based on the initial positions of the nodes and the maximum velocities of the vehicles is also possible, but is difficult because terrain might occlude the line of sight. However, once a solution is obtained we assume that the cost J^{chain} will not change substantially by a new call to the solver, even if R_{pqi}^{con} in (22) and (23) is reduced by a distance r^{con} as will be motivated in Section IV-D. The integer ϵ^{chain} should be chosen large enough to account for a possible increase in the cost, but also small enough such that the constraint (46) has an effect on the computation time. Picking ϵ^{chain} too small may turn the optimization problem infeasible.

C. Path loss computation

At this stage the optimal path found in the previous step is investigated using SPLAT!; see also Section VI for additional elements that could be included at this stage. Similar as in [53], we will set up a data-link budget. More precisely, we will define the maximum allowed path loss for node p to successfully transmit to node q at a capacity C_i , by

$$L_{pqi}^{\text{max}} := T_p + G_p^T - N_q^J + G_q^R - N_q^F - \text{SNR}(C_i) \quad (47)$$

with the dependency on the time step i reflecting that the capacity demand may be time varying. Here, T_p is the power delivered to transmitting antenna in dBm, G_p^T is the transmitting antenna

gain in the direction of the receiver in dBi, N_q^J is the Johnson noise in dBm, G_q^R is the receiver antenna gain in dBi, N_q^F is the noise figure in dB of the receiver, and $SNR(C_i)$ is the required signal-to-noise ratio (and fade margin) to achieve a link capacity of C_i . A theoretical maximum of the channel capacity based on the signal-to-noise ratio can be found using the Shannon-Hartley theorem, although the actual usable capacity is better approximated with empirical models as reported in [54].

D. Change constraints

After the analysis of the path in SPLAT!, some of the communication constraints of the MILP problem are changed as follows: If the path loss $L_{pqi} \leq L_{pqi}^{\max}$, then the communication radius $R_{pqi}^{\text{con}} = R_{pqi}^{\text{con}} - r^{\text{con}}$ where r^{con} is a predefined distance. Evidently if the nodes are not initially within each others communication range, tightening the bounds R_{pqi}^{con} for small i does not improve the ability to form a communication chain due to the finite velocity of the vehicles. However, since there are no hard constraints on the communication, we allow the algorithm to iterate while $L_{pqi} \leq L_{pqi}^{\max}$, or until $R_{pqi}^{\text{con}} = 0$ for some triplet of p , q and i .

V. SIMULATIONS

In this section we will use two UAVs to form a communication chain from a base station at $(6000, 1000, 150)^\top$ to a target station at $(2000, 6000, 10)^\top$. A no-fly zone $(\underline{x}_o, \overline{x}_o, \underline{y}_o, \overline{y}_o, \underline{z}_o, \overline{z}_o)$ is specified at $(3500, 5000, 3500, 5000, 0, 500)$. The initial placement of the UAVs are chosen to be at $(6000, 5500, 250)^\top$ and $(6000, 5000, 250)^\top$. Table I shows most of the parameters used in the MILP problem. The digital elevation data provided at [55] were used, with the accuracy shown in Figure 3(b). The default solver parameters of Gurobi and CPLEX were used, although there exists algorithms for automatic tuning of MILP solvers, see [56]. The algorithms were run on a HP EliteBook 8540w, with Intel Core i7 CPU Q720 @1.6 GHz, 16 GB RAM and a Windows 7, 64-bit operating system. Furthermore, we used MATLAB version R2010a and YALMIP version 20101208.

A. Effect of preprocessing

First we will look at the effect of applying Algorithm 1, that is, the effect on the computation time by calculating the values of binary variables based on initial positions and maximum

TABLE I
MILP PARAMETERS

| Parameter | Value | Parameter | Value |
|--|---------------------------|------------------------------------|-------|
| t_p | 0.1 | d_x, d_y, d_z | 150 |
| γ^{chain} | 60 | r^{con} | 50 |
| d^{TIN} | 100 | \bar{x} | 7000 |
| D^{vel} | 8 | \underline{x} | 0 |
| D^{con} | 8 | \bar{y} | 7000 |
| V_1^{cruise} | 25 | \underline{y} | 0 |
| V_2^{cruise} | 21 | \bar{z} | 500 |
| \bar{V}_1 | 33 | \underline{z} | 0 |
| \bar{V}_2 | 25 | $\underline{V}_1, \underline{V}_2$ | 6 |
| $\mathbf{r}_1^{\text{acc}}, \mathbf{r}_2^{\text{acc}}$ | $(0.01, 0.01, 0.01)^\top$ | g_1, g_2 | 0.01 |

TABLE II
SOLVER-TIME WITH AND WITHOUT USING ALGORITHM 1.

| Case | Optimal J | Solver-time |
|--------------------------|-------------|-------------|
| Gurobi, no preprocessing | 2120.0 | 991.8 |
| Gurobi, preprocessing | 2120.0 | 993.0 |
| CPLEX, no preprocessing | 2120.7 | 9799.8 |
| CPLEX, preprocessing | 2120.0 | 10027 |

velocity. When found, these values were added as equality constraints to the MILP problem described in IV-B, and the results are summarized in Table II. We used $R_{pqi}^{\text{con}} = 3500$ $\forall p \in \{0, \dots, 3\}, q \in \{p+1 | p \neq 3\} \cup \{p-1 | p \neq 0\}, i \in \{1, \dots, N\}$. The horizon is 300 s, and the discretization step is of 3 s. The results show that it is not beneficial to apply Algorithm 1 in our test case, as in addition to the solver-time, some time will be required to calculate the values of the binary variables in question.

Table III shows the computation time of the same setup, without and with the use of Algorithm 2. The results are definitively more promising in terms of decreased computation time. Before conclusions can be made, solver times should be averaged over multiple runs and different initial conditions, a topic which will not be further investigated in this paper.

TABLE III
SOLVER-TIME WITH AND WITHOUT USING ALGORITHM 2.

| Case | Optimal J | Solver-time |
|--------------------------|-------------|-------------|
| Gurobi, no preprocessing | 2120.0 | 925.3 |
| Gurobi, preprocessing | 2119.7 | 350.8 |
| CPLEX, no preprocessing | 2120.7 | 9384.3 |
| CPLEX, preprocessing | 2119.3 | 1631.7 |

TABLE IV
COMPARISON OF SOLVER-TIME IN GUROBI WITH AND WITHOUT USING THE BOUND ON J^{con} (46).

| Case | Optimal J | Optimal J^{chain} | Solver-time |
|---|-------------|----------------------------|-------------|
| Without bound | 2870.9 | 6000 | 2101.8 |
| With bound, $\epsilon^{\text{chain}} = 1$ | 2871.6 | 6000 | 344.5 |
| With bound, $\epsilon^{\text{chain}} = 2$ | 2872.1 | 6000 | 1390.5 |
| With bound, $\epsilon^{\text{chain}} = 3$ | 2872.3 | 6000 | 1193.6 |
| With bound, $\epsilon^{\text{chain}} = 4$ | 2870.9 | 6000 | 958.3 |
| With bound, $\epsilon^{\text{chain}} = 5$ | 2870.9 | 6000 | 2235.8 |

B. Effect of bounding J^{chain} and of warm-start in consecutive iterations

We will now look at what effect bounding J^{chain} has on the computation time. First the MILP problem of Section IV-B is solved with $R_{pqi}^{\text{con}} = 3500 \forall p \in \{0, \dots, 3\}, q \in \{p+1 | p \neq 3\} \cup \{p-1 | p \neq 0\}, i \in \{1, \dots, N\}$, but without any bound on J^{con} . Table IV shows the computation time of solving the same problem, but with $R_{pqi}^{\text{con}} = 3400 \forall p \in \{0, \dots, 3\}, q \in \{p+1 | p \neq 3\} \cup \{p-1 | p \neq 0\}, i \in \{1, \dots, N\}$, both without and with the additional constraint (46). The horizon is 300 s, and the discretization step is of 3 s. The results are promising for low ϵ^{chain} , but come at the expense of the possibility of making the optimization problem infeasible.

Using the same setup tests were carried out to look at the benefits of applying the solution of the previous iteration for warm-start. It turned out that assigning the complete optimal solution (i.e. the optimal value of all variables) from the previous iteration, would not lead to decreased computation time when the communication radius were changed between iterations. However, it

is reasonable to assume that the results would have been better if less constraints were changed from one iteration to the next, or if only a subset of the optimal values from the previous call were provided. This assumption has not been further investigated in this paper. We remark that both CPLEX and Gurobi support assigning values to a subset of the variables.

C. Path planning results

We will assume that for video communication, a bit rate of $C = 8$ Mbits/s over a channel with bandwidth $B = 20$ MHz, will require a signal-to-noise ratio SNR of 25 dB. The rest of the communication parameters can be found in Table V. We initially set $R_{pqi}^{\text{con}} = 4500 \forall p \in \{0, \dots, 3\}, q \in \{p+1|p \neq 3\} \cup \{p-1|p \neq 0\}, i \in \{1, \dots, N\}$ and use $r^{\text{con}} = 100$. The horizon is 150 s, and the discretization step is of 3 s. Figure 5 shows a top view of how the planned path changes as the communication radius is decreased due to the update of R_{pqi}^{con} by calling SPLAT!. The black diamond in the lower right corner is the base station, and the one in the upper left corner is the target station. Furthermore, the black square is the no-fly zone and the triangles corresponds to the TIN. The initial positions of the UAVs are denoted with red diamonds; vehicle $p = 1$ being the northernmost. In Figure 6(a), the velocity and the velocity bounds of the UAVs are depicted. We see that the two UAVs fly at almost full speed until a communication chain is created. From then on the UAVs fly at low speed to save fuel. The values of the binary variables associated with connectivity between each node $\tilde{b}_{pqi}^{\text{con}}$, and the communication chain b_i^{chain} are shown in Figure 6(b). The red dotted and dashed lines in Figure 6(c), show how the communication radius R_{pqi}^{con} is changed from initially being 4500 $\forall p \in \{0, \dots, 3\}, q \in \{p+1|p \neq 3\} \cup \{p-1|p \neq 0\}, i \in \{1, \dots, N\}$. The blue lines show the relative distances between the nodes. At last, Figure 6(d) depicts the altitude constraints in red and the altitude of the UAVs in blue. We remark that the functions being plotted appear continuous due to the interconnection of straight lines between sample points.

VI. FUTURE WORK

The computation time can possibly be improved by using the regional analysis function in SPLAT!, which would give an estimate of the area in which the base stations/vehicles are able to communicate, and therefore the radius of communication in the first call to the MILP solver. Furthermore, more extensive analysis of the computed path at the *Path loss computation* stage

TABLE V
COMMUNICATION PARAMETERS

| Parameter | Value | Parameter | Value |
|------------------------------|----------|------------------------------|-----------|
| B | 20 MHz | C | 8 Mbits/s |
| G_1^T, G_2^T | 3 dBi | G_1^R, G_2^R | 3 dBi |
| G_0^T, G_3^T | 6 dBi | G_0^R, G_3^R | 6 dBi |
| $N_0^J, N_1^J, N_2^J, N_3^J$ | -101 dBm | $N_0^F, N_1^F, N_2^F, N_3^F$ | 8 dB |
| T_0, T_3 | 37 dBm | T_0, T_3 | 27 dBm |

of the algorithm, see Section IV-C, should be performed. The vehicle-to-ground distance can be computed in a high accuracy map, compared with a predefined safety criterion, and if the criterion fails to be satisfied the optimization routine will be called again with additional constraints that addresses this safety issue. The stage could also include a simulator that captures the kinematics of the vehicle. Again, if the path resulting from the optimization problem is infeasible taking the kinematics into account, suitable constraints will be added the MILP problem.

One can also imagine taking other important design criteria of the communication problem into account, such as bandwidth, transmit power and latency of each node, and the required bandwidth and accepted latency between end nodes. Traffic flow constraints, traffic flow continuity and link capacity constraints in a MILP setting although for static nodes, have been introduced in [57]. Taking this into account in our problem may give answers to the number of nodes necessary for a specified mission, and whether ferrying and relaying is the preferred form of communication. In this paper we have not considered the direction of the antennas, which is a reasonable assumption if the antennas are omni-directional or can be directed independent of the heading of the vehicles. As shown in [58], it is also possible to incorporate beam-forming antennas into a MILP formulation.

VII. CONCLUSION

Due to the complexity of the path planning problem, a hierarchical decomposition of the problem seems attractive. In this paper we have considered what can be thought of as an exemplification of the middle level abstraction layer. Typically the high level would consist of a mission plan, where one of the mission objectives could be to create a communication chain

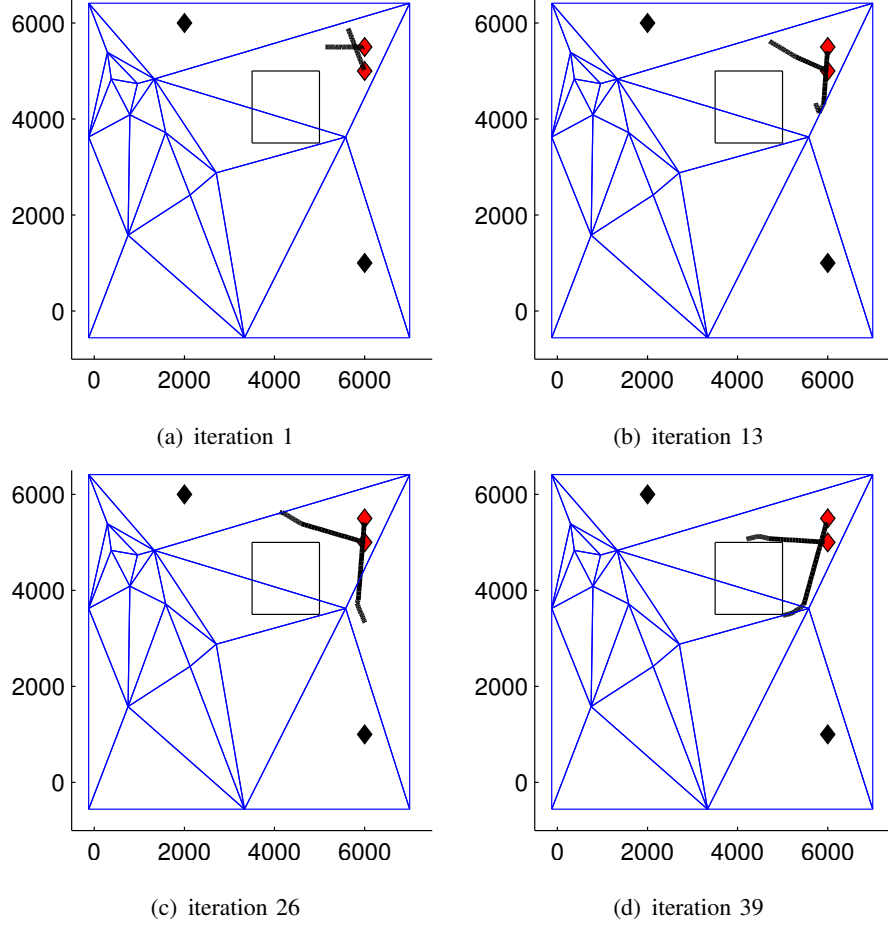


Fig. 5. Top view of the paths as R_{pqi}^{con} are decreased to correspond to more realistic path loss for the desired communication bandwidth. The base station and target station are denoted with a black diamond in the lower right and upper left corner, respectively, whereas the black square is the no-fly zone. The starting point of vehicles are denoted with a red diamond. Figure 5(a) corresponds to the case with initial values of R_{pqi}^{con} . Figure 5(b), 5(c) and 5(d), are corresponding plots after 13, 26 and 39 iterations, respectively.

as presented in this paper. At the lower level the path would be further improved by taking into account high accuracy models of UAVs and environment. At the middle level abstraction layer, MILP is our preferred choice, among other reasons, because of its ability to give a global solution, and to take into account non-convex constraints. We have used MILP to calculate paths of UAVs to form a communication chain between the start and end nodes. We have successfully demonstrated the ability to plan paths that achieve the desired communication topology, while minimizing fuel consumption, avoiding collision and no-fly zones, and satisfy altitude constraints with respect to the terrain. Furthermore, we have in an iterative manner improved the accuracy

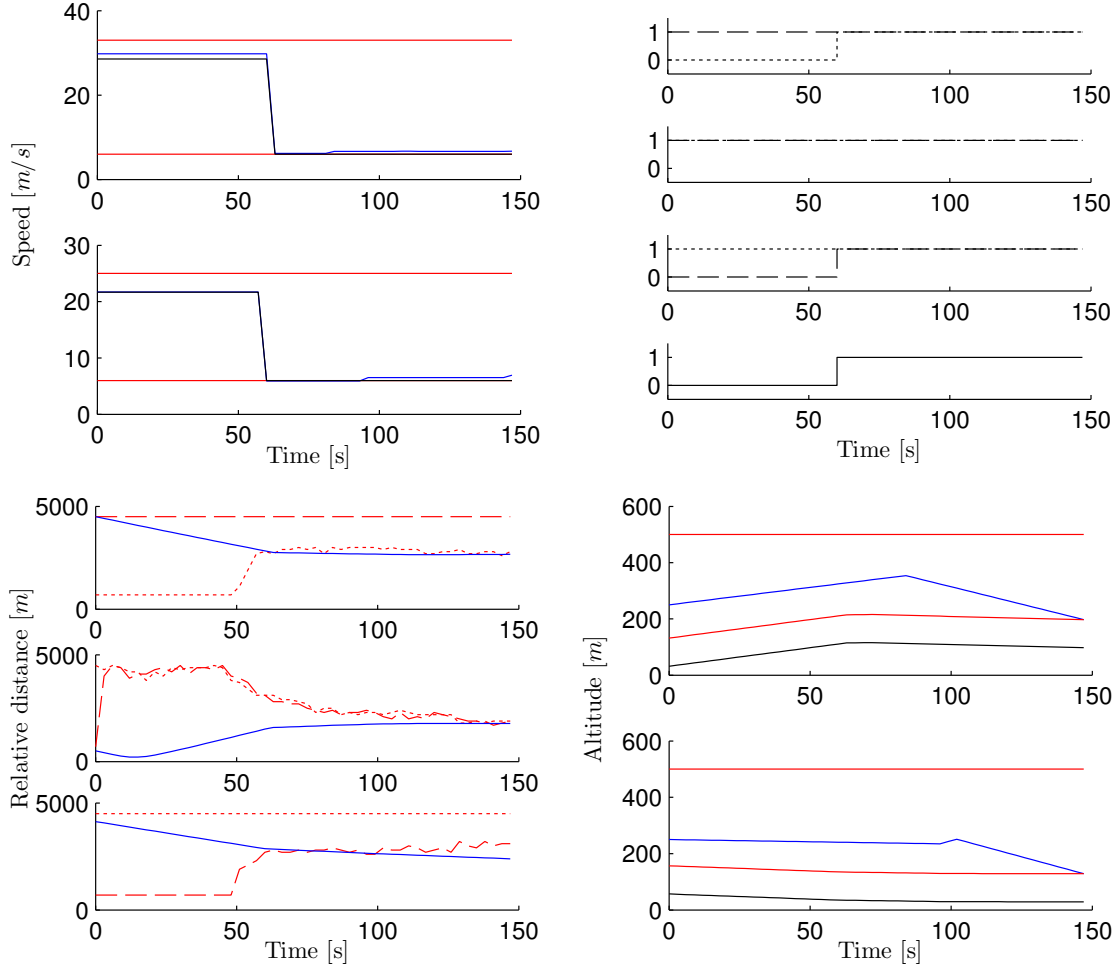


Fig. 6. (a) In black V_{pi}^{approx} , in blue: $\|v_{pi}\|$ and in red the maximum and minimum velocity \underline{V}_p and \overline{V}_p for $p = 1$ (top) and $p = 2$ (bottom). (b) In dashed line the connectivity $\tilde{b}_{p(p-1)}^{\text{con}}$ and in dotted line $\tilde{b}_{(p-1)p}^{\text{con}}$ for $p = 1$ (top), $p = 2$ (middle top) and $p = 3$ (middle bottom). In bottom plot is the value of the binary variable b^{chain} . (c) In blue solid line the relative distance $\|p_p - p_{p-1}\|$, in red dotted line the communication radius $R_{p(p-1)}$, and in red dashed line the communication radius $R_{(p-1)p}$ for $p = 1$ (top), $p = 2$ (middle) and $p = 3$ (bottom). (d) In black the terrain, in red the minimum and maximum altitude, and in blue the actual flight height for vehicle $p = 1$ (top) and $p = 2$ (bottom).

of the achievable communication rate by simulating the planned path in the radio propagation program SPLAT!.

VIII. APPENDIX

A. Proposition

Following [59], we let p_i be an atomic proposition or statement, which can either be true or false. Furthermore, let there for each atomic proposition, be an associated binary decision variable δ_i , such that $\delta_i = 1$ if and only if p_i is true, and $\delta_i = 0$ if and only if p_i is false.

Proposition 1:

$$\tilde{\delta} = 1 \iff \bigwedge_{i=1}^n f_i(x) \leq 0 \equiv \left\{ \begin{array}{l} f_i(x) \leq M_i(1 - \delta_i) \\ f_i(x) \geq (m_i - \varepsilon)\delta_i + \varepsilon \\ \bigwedge_{i=1}^n \tilde{\delta} \leq \delta_i \\ \sum_{i=1}^n \delta_i - \tilde{\delta} \leq n - 1 \end{array} \right.$$

where δ_i is the binary variable associated with the statement $f_i(x) \leq 0$, $M_i := \max_{x \in \mathcal{X}} f_i(x)$, $m_i := \min_{x \in \mathcal{X}} f_i(x)$, ε is a small tolerance.

Proof: From [18],[19], $\delta_i = 1 \iff f_i \leq 0$ is true if and only if

$$f_i(x) \leq M_i(1 - \delta_i),$$

$$f_i(x) \geq (m_i - \varepsilon)\delta_i + \varepsilon,$$

which in turn, means that $\tilde{\delta} = 1 \iff \bigwedge_i \delta_i = 1$. According to [59, Table 4.2], $p \iff q \equiv (p \implies q) \wedge (q \implies p)$, for some atomic propositions p and q . Therefore, $\tilde{\delta} = 1 \iff \bigwedge_{i=1}^n \delta_i = 1$ is equivalent to $(\tilde{\delta} = 1 \implies \bigwedge_{i=1}^n \delta_i = 1) \wedge (\bigwedge_{i=1}^n \delta_i = 1 \implies \tilde{\delta} = 1)$. The implication $\tilde{\delta} = 1 \implies \bigwedge_{i=1}^n \delta_i = 1$ can be expressed as $\bigwedge_{i=1}^n \tilde{\delta} \leq \delta_i$, where as the implication $\bigwedge_{i=1}^n \delta_i = 1 \implies \tilde{\delta} = 1$ can be expressed as $\sum_{i=1}^n \delta_i - \tilde{\delta} \leq n - 1$. ■

B. Algorithms

Algorithm 1 Preprocessing communication constraints

```

1: for  $p = 0 : 1 : n_p + 1$  do
2:   for  $q = p - 1 : 2 : p + 1$  do
3:     if  $q < 0$  or  $q > n_p + 1$  then
4:       continue
5:     end if
6:      $d_{pq}^{\text{pre}} = ||\mathbf{p}_p(0) - \mathbf{p}_q(0)|| - R_{pq}^{\text{con}}$ 
7:      $t^{\text{pre}} = \text{floor} \left\{ \frac{|d_{pq}^{\text{pre}}|}{(\overline{V}_p + \overline{V}_q)} \right\}$ 
8:      $i = 0$ 
9:     while  $i \leq t^{\text{pre}} / \Delta t$  do
10:      if  $d_{pq}^{\text{pre}} \leq 0$  then
11:         $\tilde{b}_{pqi}^{\text{con}} = 1$ 
12:      else
13:         $\tilde{b}_{pqi}^{\text{con}} = 0$ 
14:         $b_i^{\text{chain}} = 0$ 
15:      end if
16:       $i = i + 1$ 
17:    end while
18:  end for
19: end for

```

Algorithm 2 Preprocessing terrain constraints

```

1: for  $p = 1 : 1 : n_p$  do
2:   for  $i = 1 : 1 : N$  do
3:     for  $t = 1 : 1 : T^{\text{TIN}}$  do
4:        $d_{pt}^{\text{pre}} = \text{pointTriangleDist}(\text{Tri}_t, \mathbf{p}_{p0})$ 
5:       if  $d_{pt}^{\text{pre}} > i\Delta t \bar{V}_p$  then
6:          $b_{pit}^{\text{TIN}} = 0$ 
7:       end if
8:     end for
9:   end for
10: end for

```

Algorithm 3 Path planning

```

1: Preprocessing, see section III-F
2: Solve MILP problem
3:  $\mathcal{P} = \{0, \dots, n_p + 1\}$ 
4:  $\mathcal{Q}_p = \{p + 1 | p \neq n_p + 1\} \cup \{p - 1 | p \neq 0\}$ 
5:  $\mathcal{I} = \{0, \dots, N - 1\}$ 
6: Find  $L_{pqi}$  using SPLAT!
7: while  $\exists L_{pqi} \geq L_{pqi}^{\max}, \forall p \in \mathcal{P}, q \in \mathcal{Q}_p, i \in \mathcal{I}$  do
8:    $R_{pqi} = R_{pqi} - r$ 
9:   Solve MILP w/ new constraints
10:  Find  $L_{pqi}$  using SPLAT!
11: end while

```

REFERENCES

- [1] K. Purvis, K. Astrom, and M. Khammash, "Estimation and optimal configurations for localization using cooperative uavs," *Control Systems Technology, IEEE Transactions on*, vol. 16, no. 5, pp. 947–958, sept. 2008.
- [2] E. W. Frew and T. X. Brown, "Networking issues for small unmanned aircraft systems," *Journal of Intelligent and Robotic Systems*, vol. 54, pp. 21–37, 2008.
- [3] J. J. Ruz, O. Arévalo, G. Pajares, and J. M. de la Cruz, "UAV trajectory planning for static and dynamic environments," in *Aerial Vehicles*, T. M. Lam, Ed. InTech, 2009, pp. 581–600.

- [4] D. Henkel and T. X. Brown, "On controlled node mobility in delay-tolerant networks of unmanned aerial vehicles," in *Proc. of International Symposium on Advanced Radio Technologies*, 2006.
- [5] L. Hogie, P. Bouvry, and F. Guinand, "An overview of MANET simulators," in *Proc. of the First International Workshop on Methods and Tools for Coordinating, Distributed and Mobile Systems*, 2006.
- [6] C. M. Durham, T. R. Andel, K. M. Hopkinson, and S. H. Kurkowski, "Evaluation of an OPNET model for unmanned aerial vehicle (UAV) networks," in *Proc. of the Spring Simulation Multiconference*, 2009.
- [7] I. Stepanov, D. Herrscher, and K. Rothermel, "On the impact of radio propagation models on MANET simulation results," in *IFIP International Conference on Mobile and Wireless Communications Networks*, 2005.
- [8] A. Schmitz and M. Wenig, "The effect of the radio wave propagation model in mobile ad hoc networks," in *In Proc. of the 9th ACM international symposium on Modeling analysis and simulation of wireless and mobile systems*, 2006.
- [9] D. Dhoutaut, A. Régis, and F. Spies, "Impact of radio propagation models in vehicular ad hoc networks simulations," in *VANET '06: Proc. of the 3rd international workshop on Vehicular ad hoc networks*. New York, NY, USA: ACM, 2006, pp. 40–49.
- [10] O. Ukrainsky, H. Zebrowitz, C. Hein, A. Cortese, A. Rubin, C. Poon, A. Bard, and H. Reyes, "An open environment for rapid embedded planning of on-the-move communications networks using multi-level abstraction," in *Military Communications Conference*, 2005, pp. 2631 – 2636.
- [11] R. M. McGraw, G. Shao, D. Mumme, and R. MacDonald, "Design of an agent-based course of action (coa) analysis with radio effects toolbox," *International Journal of Intelligent Control and Systems*, vol. 14, pp. 104–114, 2009.
- [12] B. Burdakov, P. Doherty, K. Holmberg, J. Kvarnström, and P.-M. Olsson, "Positioning unmanned aerial vehicles as communication relays for surveillance tasks," in *Robotics Science and Systems, Online Proceedings*, 2009.
- [13] Z. Han, A. Swindlehurst, and K. Liu, "Optimization of MANET connectivity via smart deployment/movement of unmanned air vehicles," *Vehicular Technology, IEEE Transactions on*, vol. 58, no. 7, pp. 3533 –3546, sept. 2009.
- [14] D. K. Goldenberg, J. Lin, A. S. Morse, B. E. Rosen, and Y. R. Yang, "Towards mobility as a network control primitive," in *In MobiHoc 04: Proc. of the 5th ACM international symposium on Mobile ad hoc networking and computing*. ACM Press, 2004, pp. 163–174.
- [15] A. So and B. Liang, "Minimum cost configuration of relay and channel infrastructure in heterogeneous wireless mesh networks," in *Networking 2007. Ad Hoc and Sensor Networks, Wireless Networks, Next Generation Internet*. Springer Berlin / Heidelberg, 2007, pp. 275–286.
- [16] C. Dixon and E. W. Frew, "Decentralized extremum-seeking control of nonholonomic vehicles to form a communication chain," in *Advances in Cooperative Control and Optimization*. Springer Berlin / Heidelberg, 2007, pp. 311–322.
- [17] B. Moses Sathiyaraj, L. C. Jain, A. Finn, and S. Drake, "Multiple UAVs path planning algorithms: a comparative study," *Fuzzy Optimization and Decision Making*, vol. 7, pp. 257–267, 2008.
- [18] H. P. Williams, *Model Building in Mathematical Programming*, 4th ed. John Wiley and Sons, 1999, ISBN: 978-0471997887.
- [19] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica*, vol. 35, pp. 407–427, 1999.
- [20] A. Richards and J. How, "Mixed-integer programming for control," in *Proc. of the American Control Conference*, 2005.
- [21] A. Chaudhry, K. Misovec, and R. D'Andrea, "Low observability path planning for an unmanned air vehicle using mixed integer linear programming," in *Proc. of the 43rd IEEE Conference on Decision and Control*, 2004.

- [22] J. S. Bellingham, "Coordination and control of uav fleets using mixed-integer linear programming," Master's thesis, Massachusetts Institute of Technology, 2002.
- [23] T. Schouwenaars, A. Stubbs, J. Paduano, and E. Feron, "Multivehicle path planning for nonlinear-of-sight communication," *Journal of Field Robotics*, vol. 23, pp. 269–290, 2006.
- [24] M. Alighanbary, Y. Kuwata, and J. How, "Coordination and control of multiple uavs with timing constraints and loitering," in *Proc. of the American Control Conference*, vol. 6, june 2003, pp. 5311 – 5316 vol.6.
- [25] Y. Hao, A. Davari, and A. Manesh, "Differential flatness-based trajectory planning for multiple unmanned aerial vehicles using mixed-integer linear programming," in *American Control Conference, 2005. Proceedings of the 2005*, 2005, pp. 104 – 109.
- [26] Y. Kim, D.-W. Gu, and I. Postlethwaite, "Real-time optimal mission scheduling and flight path selection," *Automatic Control, IEEE Transactions on*, vol. 52, no. 6, pp. 1119 –1123, june 2007.
- [27] C. Reinl and O. von Stryk, "Optimal control of multi-vehicle-systems under communication constraints using mixed-integer linear programming," in *Proc. of the International Conference on Robot Communication and Coordination*, 2007.
- [28] C. Branca and R. Fierro, "A hierarchical optimization algorithm for cooperative vehicle networks," in *Proc. of the American Control Conference*, 2006, pp. 4225–4230.
- [29] M. G. Earl and R. D'Andrea, "Iterative MILP methods for vehicle-control problems," *IEEE Transactions on Robotics*, vol. 21, pp. 1158–1167, 2005.
- [30] M. P. Vitus, V. Pradeep, G. M. Hoffmann, S. L. Waslander, and C. J. Tomlin, "Tunnel milp: Path planning with sequential convex polytopes," in *Proc of the AIAA Guidance, Navigation, and Control Conference*, 2008.
- [31] D. Kotz, C. Newport, R. S. Gray, J. Liu, Y. Yuan, and C. Elliott, "Experimental evaluation of wireless simulation assumptions," in *Proc. of the 7th ACM Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, 2004, pp. 78–82.
- [32] J. Maglicane, "SPLAT! An RF signal propagation, loss and terrain analysis tool," 2010 (Accessed August 18, 2010). [Online]. Available: <http://www.qsl.net/kd2bd/splat.html>
- [33] J. McMellen, "RF propagation modeling with SPLAT! for windows," 2010 (Accessed August 18, 2010). [Online]. Available: <http://blog.gearz.net/2007/09/rf-propagation-modeling-with-splat-for.html>
- [34] A. G. Longley and P. L. Rice, "Prediction of tropospheric radio transmission loss over irregular terrain: A computer method," U. S. Government, Tech. Rep., 1968.
- [35] J. Löfberg, "Modeling and solving uncertain optimization problems in YALMIP," in *Proc. of IFAC World Congress*, 2008.
- [36] A. Lodi and J. T. Linderoth, "MILP software," in *Encyclopedia for Operations Research*. Wiley, 2010, To appear.
- [37] W. Yin, "Gurobi Mex: A MATLAB interface for Gurobi," 2010 (Accessed December 12, 2010). [Online]. Available: http://www.caam.rice.edu/~wy1/gurobi_mex
- [38] J. A. Farrell and M. Barth, *The Global Positioning Systems & Inertial Navigation*. McGraw-Hill, 1998, ISBN: 0-07-022045-X.
- [39] T. I. Fossen, *Marine Control Systems: Guidance, Navigation, and Control of Ships, Rigs and Underwater Vehicles*. Marine Cybernetics, 2002.
- [40] K. F. Culligan, "Online trajectory planning for uavs using mixed integer linear programming," Master's thesis, Massachusetts Institute of Technology, 2006.
- [41] K. Culligan, M. Valenti, Y. Kuwata, and J. P. How, "Three-dimensional flight experiments using on-line mixed-integer linear programming trajectory optimization," in *Proc. of the American Control Conference*, 2007.

- [42] B. Luders, "Robust trajectory planning for unmanned aerial vehicles in uncertain environments," Master's thesis, MIT, 2008.
- [43] T. Schouwenaars, B. De Moor, E. Feron, and J. How, "Mixed integer programming for multi-vehicle path planning," in *Proc. of the European Control Conference*, 2001.
- [44] C. S. Ma and R. H. Miller, "MILP optimal path planning for real-time applications," in *Proc. of the American Control Conference*, 2006.
- [45] A. Richards and J. P. How, "Aircraft trajectory planning with collision avoidance using mixed integer linear programming," in *Proc. of the American Control Conference*, 2002.
- [46] W. A. Kamal, D.-W. Gu, and I. Postlethwaite, "MILP and its application in flight path planning," in *Proc. of the 16th IFAC World Congress*, 2005.
- [47] M. Kvasnica, P. Grieder, and M. Baotić, "Multi-Parametric Toolbox (MPT)," 2004 (Accessed December 13, 2010). [Online]. Available: <http://control.ee.ethz.ch/~mpt/>
- [48] R. W. Beard and T. W. McLain, "Multiple UAV cooperative search under collision avoidance and limited range communication constraints," in *Proceedings of the 42nd IEEE Conference on Decision and Control*, 2003, pp. 25–30.
- [49] I. Mitchell, A. M. Bayen, and C. J. Tomlin, "A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games," *IEEE Transaction on Automatic Control*, vol. 50, pp. 947–957, 2005.
- [50] Z. Shengxiang and P. Hailong, "Real-time optimal trajectory planning with terrain avoidance using MILP," in *Proc. of the International Symposium on Systems and Control in Aerospace and Astronautics*, 2008.
- [51] O. Sakhi, "Image and terrain modeling using incremental delaunay triangulation," 2010 (Accessed January 22, 2011). [Online]. Available: <http://www.mathworks.com/matlabcentral/fileexchange/26615-image-and-terrain-modeling-using-incremental-delaunay-triangulation>
- [52] G. Fischer, "Distance between a point and a triangle in 3d," 2009 (Accessed January 22, 2011). [Online]. Available: <http://www.mathworks.com/matlabcentral/fileexchange/22857-distance-between-a-point-and-a-triangle-in-3d>
- [53] P. G. Fahlstrom and T. J. Gleason, *Introduction to UAV Systems*, 1998.
- [54] E. W. Frew, C. Dixon, J. Elston, and M. Stachura, "Active sensing by unmanned aircraft systems in realistic communication environments," in *Proc. of the IFAC Workshop on Networked Robotics*, 2009.
- [55] J. de Ferranti, "Digital elevation data," 2005 (Accessed January 22, 2011). [Online]. Available: <http://www.viewfinderpanoramas.org/dem3.html>
- [56] F. Hutter, H. H. Hoos, and T. Stützle, "Automatic algorithm configuration based on local search," in *Proc. of the 22nd Conference on Artificial Intelligence*, 2007.
- [57] X. Cao, "An integer linear programming approach for topology design in owc networks," in *Proc. of the IEEE GLOBECOM Workshops*, 2008, pp. 1–5.
- [58] M. Kiese, C. Hartmann, and R. Vilzmann, "Optimality bounds of the connectivity of adhoc networks with beamforming antennas," in *Proc. of GLOBECOM*, 2009.
- [59] L. Magatão, "Mixed integer linear programming and constraint logic programming: Towards a unified modeling framework," Ph.D. dissertation, The Federal Center of Technological Education of Paraná, 2005.