An Automated Network Generation Procedure for Routing of Unmanned Aerial Vehicles (UAVs) in a GIS Environment

**Irene Casas · Amit Malik · Eric M. Delmelle ·
Mark H. Karwan · Rajan Batta**

Abstract In this paper, the problem of creating a network to facilitate planning of an air reconnaissance mission is investigated. A comprehensive review of the literature shows that limited work has been done on the design of such networks in a hostile environment and that little research has accounted for terrain information. Ideally, in any routing problem a sample network is used as reference to determine paths to achieve a specified goal. The objective of this paper is to propose an automated generalized procedure developed within a Geographical Information System environment, to identify the nodes and links of a network based on the topographic information of the enemy terrain. This information is made available in the form of a Triangulated Irregular Network (TIN). Inputs to the network generation process include (I) a user-specified threshold altitude value to avoid radar detection and terrain collisions, (II) a specification of the node density in the study area and (III) the minimum/maximum lengths for the links in the desired network. The generated networks are evaluated based on traditional network analysis measures and on the area covered based on a visibility analysis. The methodology is particularly useful for air mission of Unmanned Aerial Vehicles (UAVs) and robot motion planning. As an example, a case study using elevation data from the state of Colorado is presented.

I. Casas (✉)
Department of Geography, University at Buffalo, SUNY and National Center for Geographic Information and Analysis (NCGIA), 105 Wilkeson Quad, Buffalo, NY 14261, USA
e-mail: icasas@buffalo.edu

A. Malik · M. H. Karwan
Department of Industrial and Systems Engineering, University at Buffalo, SUNY, Buffalo, NY 14261, USA

E. M. Delmelle
Department of Geography, University of Idaho, Moscow, ID 83844, USA

R. Batta
Department of Industrial and Systems Engineering, University at Buffalo,
SUNY and National Center for Geographic Information and Analysis (NCGIA), Buffalo, NY 14261, USA

1 Introduction

Complimentary to the problem of finding optimal paths for a given set of origin – destination
(OD) pairs, is the problem of developing the corresponding underlying network where such
process is carried (see i.e., Holle et al. 1995; Latombe 2001). In the case of a reconnaissance
mission, where typically there is a need for multiple aircraft, a network is needed for the
routing operation to take place. Under this scenario, the operation is threefold (Thyagarajan
et al. 2005). The first problem consists of routing aircraft from home base(s) to the origin
point(s) of the mission. The second problem is concerned with mission execution; i.e.,
developing methods to conduct mission activities. Finally, these aircraft have to be routed
from the end points of the mission back to the home bases.

Generally routing formulations require an underlying network. In most cases a
connected graph $G = (V, E)$, where V is the set of nodes and E the set of arcs, is used to
represent the network. With the graph as a base given a specific routing objective (i.e.,
shortest path; traveling salesman problem), a set of optimal paths within the available
network can be found. Reconnaissance mission planning is however different from this set
of ideal conditions. One of the major concerns in such a case is the amount of information
available of the enemy terrain. Satellite images showing the geography of the region often
act as the only initial source of terrain information which in turn can be complimented by
enemy presence information provided by the sensors deployed in the region. Therefore,
geography and the size of the terrain play an important role in this type of application.
Capitalizing on this characteristic, the task of creating a network for routing purposes for a
reconnaissance mission can be simplified by making use of Geographic Information
Systems (GIS). GIS can reduce the complexity of the process by bringing out subtle
geographic patterns and relationships that can form the basis of good decisions.

The purpose of this paper is to develop and analyze a procedure for the automated
development of a network to facilitate the aforementioned issues. The objective is to
provide a fast and flexible method to generate a network within a hostile region considering
the complexity of the topography. Once this network is available, optimal paths can be
determined for different routing objectives. Some of the advantages of having an underlying
network to plan the missions as opposed to path planning are: (1) terrain obstacles are better
circumvented by following a predefined network; (2) multiple routes can be generated and
evaluated before the actual execution of the mission; and (3) it facilitates mission planning for
a fleet of vehicles.

In Section 1, literature in network design and route planning is presented from different
research perspectives leading to the motivation of the new proposed procedure. In Section 2,
a detailed description of the network generation procedure in a GIS framework is provided,
implementation issues are addressed, measures of connectivity to characterize the quality of
the network are defined, and a small example is provided to gain insight in the
methodology. Section 3 illustrates the network design approach to a large case study in
the State of Colorado. In the final section, a summary and directions for future research are
discussed based on limitations and issues discovered in the process.

2 Review of the literature and problem statement

The objective of a network generation procedure, as understood in this paper, is to determine where to locate a set of nodes and the links connecting them based on a set of requirements, with a graph as a mathematical representation. This is not a problem commonly explored, since generally a base network is used with the objective of identifying a subset of vertices and links that meet a set of constraints within a given application domain. This subtle but clear difference allows the literature to be reviewed from at least two perspectives. First, from the operations research perspective (OR) where the term "network design" is used to refer to the problem of "identifying an optimal subgraph of a graph subject to side constraints." (Feremans et al. 2002). Second, from a motion/route planning perspective where there is no underlying network/graph and a path is planned to avoid obstacles, common in the area of robotics.

The literature is extensive and diverse in the area of network design, therefore this section will not provide a comprehensive review of the literature, instead it will touch upon some of the generalities of the problem. Network design problems (NDP) are an important class of optimization problems applicable in the fields of telecommunications, transportation, biology, and production systems, to name a few. A summary of a categorization of network design problems from an implementation perspective is provided by Noltemeier and Wirth (1999) with a detailed description of generalized network design problems presented by Feremans et al. (2002). Feremans et al. (2002) list three well known NDPs: the minimum spanning tree problem (Magnanti and Wolsey 1995), the traveling salesman problem (Lawler et al. 1985), and the shortest path problem (Ahuja et al. 1993). Magnanti and Raghavan (2005) propose a different type of NDP subject to connectivity requirements (NDC). From an applied perspective, in the area of transportation Magnanti and Wong (1984) present a summary of applied traditional network design models and algorithms, while Poorzahedy and Abulghasemi (2005) present an application of an ant system to a transportation NDP. Examples of the NDP to telecommunication networks can be found in Mihail et al. (1996) and Gupta et al. (2001). In terms of NDP algorithm implementations Goemans et al. (1994) discuss a set of good examples.

The above NDP formulations and implementations traditionally use the graph abstraction of the network and disregard the other characteristics of the geographic space, for example, topography. Cao et al. (1999), however, incorporate such characteristics by proposing a fiber deployment plan system for telecommunication purposes which combines a GIS and combinatorial optimization. They propose a loosely coupled system that makes use of the GIS as a user interface for data input and display. Levinson and Karamalaputi (2003) propose a method to predict the growth of new highway links based on the existing network, physical attributes of the network, construction and expansion history, and daily traffic using a logit model. Yamins et al. (2003) propose a road growing procedure where no underlying network is required. Their procedure involves two steps. The first step consists of determining "appropriate locations to be connected by ... links," which are based on potential functions. The second step is to build the roads and consists of a cost-based algorithm. The procedure uses a lattice as base to represent land classes in the study area. The general procedure has some similarities with the one proposed in this paper; in particular, in separating the procedure into two distinct steps, node and link creation and using raster representation as underlying data. The notion of artificial life could also be applied to network design problems since, for example, cellular automata are well suited to model complex dynamic systems "composed of large numbers of individual elements

linked by nonlinear couplings" (Openshaw and Openshaw 1997). Xie and Levinson (2006) present an example of how networks can change by modeling links as autonomous agents and Nikolos et al. (2003) utilize an evolutionary algorithm based framework to design a path planner for UAVs autonomous navigation. Other methods have been suggested for location theory, path determination, and planning. Smith (2004) proposes distance transform methods traditionally used in image processing to be included in GIS software packages. These methods provide an alternative to conventional optimization techniques.

In the area of motion/route planning in particular in the field of robotics, there is also a large body of literature related to network design. This area shares several characteristics with the routing of UAVs. First, both robots and UAVs are remotely operated and controlled. Second, they move through continuous space. Third, they need to avoid obstacles and finally, their objective is typically to achieve a task while going from an origin to a destination point. Therefore, it is important to discuss the literature in this field. Latombe (2001), for example, divides the entire problem of robot motion planning into sub problems and explains each one thoroughly. He introduces cell decomposition methods that can be used to generate a network when the environment contains a countable number of obstacles. A path between an initial point q_{orig} and a destination point q_{dest} can be determined by subdividing the motion space into smaller areas, called cells. A path can easily be found by following the adjacent, decomposed cells through free space. The exact cell decomposition is guaranteed to be complete, meaning that if a free path exists, the method will find it. Yet the trade-off of this method is that it is a complex mathematical process. The approximate method is less involved but can yield similar results.

Other studies in robot planning are geared more towards specific environments and constraints than Latombe's method. Fox et al. (2000) present a path planning algorithm for autonomous vehicles that works in two stages. First, the path planner generates a path composed of line segments using simple algebraic equations. Second—and this is an advantage of this technique—collision detection and avoidance are performed on these individual line segments to ensure the safety of the path. According to Nelson (1998), the path planner is a component of a larger general-purpose control architecture for autonomous vehicles called Strategic Tactical Execution Software Control Architecture (STESCA).

Stentz (1994) describes an algorithm aimed at generating optimal paths for a robot operating with a sensor and a map of a partially known environment. It is formulated as an optimal find-path problem within a directed graph, where the arcs are labeled with cost values. The operational aspect of the algorithm is presented, along with informal proofs of its soundness, optimality and completeness. The algorithm works on a directed graph with weighted arcs. The author points out that one of the drawbacks of the implementation is the use of a quad tree, which has limited use where cost values range over a continuum. He proposes a solution where quad trees will work under such environment as part of his future work.

Sleumer and Tschichold-Gürman (1999) present an algorithm for the automatic generation of a map that describes the operation environment of an indoor mobile service robot. They use a CAD description of a building consisting of line segments, representing the walls as an input. The algorithm is based on the exact cell decomposition approach and generates a connectivity graph based on cell indices. However, for successful implementation of exact cell decomposition, it is required that the geometry of each cell be simple. Moreover, it is important to test the adjacency of two cells to find a path crossing the portion of the boundary shared by the two cells. Both exact cell decomposition and approximate cell decomposition methods are accurate, yet may be computationally expensive

when used in a terrain environment since numerous obstacles at varying elevations could be found. They are most effective when the environment contains a countable number of obstacles.

In case of scenarios where the information of the threats is uncertain, construction of a safe path by solving an optimization problem is not possible. For such scenarios Jun and D'Andrea (2003) suggest a path planning approach based on a map of probability of threats, which can be built from a priori surveillance data to complete a mission (to arrive at a given target within a pre-specified time) while maximizing the safety of the unmanned aerial vehicles (UAVs). In cases where optimality is desired but not required, the autonomous routes for a vehicle subject to terrain constraints, threat constraints, navigational constraints, and vehicle constraints can be generated using genetic algorithms. Once the network is generated, based on the possible routes the path planning and re-planning is done using Dijkstra's algorithm with the objective of finding minimum cost paths through a graph from a specified start and goal node. The graph may not be fully known from the start and might be discovered as the algorithm proceeds, which can be viewed as partial knowledge of a network during routing.

The relevant literature shows that most of the research has been confined to the domain of robot and motion planning. Very little effort has been devoted to network design outside of these applications. Ravi et al. (2004) suggest an evolutionary flight path planning algorithm which is one of the alternatives to classical optimization methods that are capable of mapping paths for UAVs functioning under several aerodynamic constraints. A feasible path is typically found rapidly, in 100 generations or less, while an increase in generations insures a strike very near to the target center. Nikolos et al. (2003) also suggest a planner for UAVs based on an evolutionary algorithm under 3D rough terrain environments. A breeder genetic algorithm (BGA) is used to determine the locations that can be part of the path. As in the case of Ravi et al. (2004), the algorithm produces feasible paths after a small number of generations (less than 10, much less than in Ravi et al. 2004). Sasiadeck and Duleba (2000) suggest a 3D heuristic-based planner for local trajectory planning in a partially known environment with moving obstacles and a predefined global path. The route planners discussed make use of continuous space.

In a recent paper, Thyagarajan et al.(2005) make an effort to generate the network in a GIS environment using some of the work presented by Latombe (2001). Routing is carried out on this network and a case study is presented. The procedure, carried out manually, is time consuming and has a major limitation that is it is terrain specific.

While the discussed network design problems lack the inclusion of terrain information and the presence of obstacles (enemy threats), the robot route-planners typically focus on only one "vehicle" instead of on a fleet. Therefore, there is a need for a generalized procedure for the network design problem that is both fast and flexible and that would take into account such considerations. The main concern remains to generate the nodes in the network and to connect them in such a way that maximum network connectivity, maximum area of coverage, and minimum number of nodes are guaranteed. These are conflicting objectives, since network connectivity increases as the number of nodes increases; therefore equilibrium has to be achieved. For instance, increasing the number of nodes will make the problem more complex to solve, while minimizing them will harm the network connectivity and area of coverage. Besides these conflicting objectives, additional constraints may be required, tailoring the network to specific needs.

In this paper, we present a GIS-embedded strategy to design a network aimed at facilitating the routing of UAVs over an enemy terrain. The terrain information is made

available in the form of a Triangulated Irregular Network (TIN) created from a Digital Elevation Model (DEM). The network is designed with the objective of minimizing probability of detection of the UAVs by including specific routing constraints as part of the network design, such as the flight detection altitude, and minimum and maximum segment lengths.

3 Network design in a GIS framework

There are two challenges while creating a network, first determining the nodes and second determining the links that will connect these nodes. Questions that arise when facing these challenges include: How many nodes to locate? Where to locate them? Which nodes are relevant and which nodes should be discarded? How is the usability of a node defined? How should nodes be connected? Should nodes far apart be linked together or only nodes close to each other? In the case of the algorithm proposed here the main constraint faced to answer these questions is the topography of the terrain.

In this section a GIS-embedded algorithm that facilitates the creation of a network, its implementation, and a simple example of how it works are presented. The network generation procedure has been divided into two phases: the *generation of nodes* across the study area and the *creation of links* connecting them. Each phase is presented following the form of a script.

3.1 Generation of nodes

The input to this phase of the methodology is a Triangulated Irregular Network (TIN) of the region of interest. A TIN represents a surface as a set of contiguous triangular facets of different shapes and sizes that are non-overlapping (see Lee 1991 for TIN extraction methods from DEMs). The TIN is used within a GIS-platform as a base to create the nodes and links that will result in a network. The procedure starts by requesting from the user the area in which the network would be created and the density at which the nodes that will constitute the vertices of the network are to be generated latitudinally and longitudinally. The output of this step is a set of nodes arranged in a systematic pattern. The specific steps of the procedure follow.

Step 1   Select an area over the TIN where the network should be created. This corresponds to the area where the UAV's mission is to be carried out and results in the boundary of a grid or definition of the extent that would be used to generate the nodes that would be part of the network.

Step 2   Define the number of rows and columns that will determine the density of the node grid. As a result, a systematic pattern of cells is generated. The importance of such a pattern has often been stressed as it ensures to capture equally likely the characteristics of the terrain complexity. This pattern could be modified with knowledge of pre-existing enemy information. In the latter case, density of nodes may be reduced as it presents a real danger. On the other hand, the number of nodes could be increased where information becomes more necessary.

Step 3   Nodes are placed at the centroids of the cells. X, Y and Z values for each of these centroids are calculated. The Z or altitude value is obtained from the underlying TIN.

Step 4   The detection altitude or maximum elevation desired is determined by the user
         which assumes knowledge of the enemy terrain. Nodes below the detection
         altitude are extracted and used as base in the generation of links. The nodes above
         the detection altitude are discarded, given that at those elevations the UAVs could
         be detected (the detection altitude is the universal altitude, the true elevations).

3.2 Generation of links

The output of Step 4 from the node generation procedure is a grid with nodes placed at the
centroids of the grid cells which are below the detection altitude. These centroids form a set
of potential nodes from which the actual nodes of the final network are extracted.

   Two important parameters are identified to generate a network: (I) the *link length* and
(II) the *maximum elevation* on the link. These parameters are specified by the user and are
problem-specific. Appropriate values for these parameters are based on the user's
knowledge of: the terrain; the operation of the UAVs; the purpose, logistics, and safety of
the mission; and the mechanics of the network generation procedure. This means that the
user's input is a determinant in the quality of the resulting network. A range is defined for
the link length, since long links reduce the flexibility of the network to provide alternative
sets of routes if needed and they can be approximated by shorter links. However, links
should not be too short either, since when routing this will result in routes where the UAV
will have to change direction frequently. A desirable maximum and minimum length need
to be specified (i.e., MIN ≤ link length ≤ MAX) for the links. It is important to note that
these two values are measured in number of links, therefore are tied to the density of the
grid. For example, the minimum and maximum lengths could be 1 and 3 links respectively.
If 30 rows and 30 columns were specified in a squared terrain of 300 m by 300 m, 1 would
correspond to 10 m and 3 to 30 m. These values would vary based on the terrain and the
number of rows and columns specified. The maximum elevation of the links is considered a
constraint ensuring that at any given time the link connecting two nodes does not pass
across an area with an elevation greater than the threshold altitude value. Overlooking this
constraint could cause the aircraft to fly above the threshold altitude value even though the
two nodes lie below the maximum allowable altitude. This would unnecessarily increase
the risk of being detected.

   The output of this stage is a set of links connecting a subset of the nodes resulting from
the node generation process, as long as the link does not violate the pre-specified detection
altitude. Figures 1 and 2 illustrate the link generation procedure with minimum and
maximum length of 2 and 3 for the network links which correspond to 0.5 and 0.8 km
respectively. Dark grey areas denote regions above the detection altitude and the elevation
of the nodes in the center of each grid cell have values below the detection altitude. The
procedure is explained thoroughly in the following steps.

Step 0   Specify a starting point (node) for network creation and label it "Node Being
         Explored" (NBE). Note that the starting point has an effect on the coverage and
         connectivity of the resulting network. If the starting point is selected in an area
         surrounded by high elevation, for example, given the link lengths chosen, it might
         not be possible to connect this point to any other node in the terrain.
Step 1   Specify the minimum and maximum lengths for the links in the network.
Step 2   Two buffers, of size *a* and *b* are drawn around the starting point (NBE) which corre-
         spond to the minimum and maximum lengths respectively, specified in step 1 (Fig. 1(a)).

Step 2.1   All the nodes within the buffer of radius a (inner circle) are discarded (Fig. 1(b)).

Step 2.2   A donut buffer is calculated using buffers of size a and b, which includes the possible nodes within the desired distances for possible link creation (Fig. 1(b)).

Step 3   The buffers are split into four sections, called quarter grids (Fig. 2(a)). The idea behind splitting the buffer into four is for extending the creation of links in four cardinal directions, providing more flexibility for connectivity (this could be generalized to any number of sections).

Step 4   Determine the centroid for each of the four sections and find the closest node to it.

Step 5   Connect this closest node to the starting point. If maximum elevation is violated while trying to connect these two nodes, the algorithm will try to connect to the second closest node to that centroid (Fig. 2(b)). Connections can be done to a maximum of four nodes, given that each quarter is assigned at the most to one node. Once quadrant nodes (children of the parent starting node) have been selected, repeat the steps with each of the children being a donut center, or an NBE. For the subsequent children nodes, a maximum of three directions will be checked as one has already been explored when the parent node was explored. This means that four directions are checked only for the starting node; subsequently the direction of the parent node is not checked since that is where the children nodes came from. At this step the parent node is marked as "explored" and the children nodes are marked as "to be explored."

Step 6   Repeat the steps for each node in the "to be explored" list. Note that in Fig. 2(b) some nodes fall exactly on the quarter grid separating line. These nodes belong to adjacent quarter grids delimited by this line. This creates duplicate links to the central node. However, the script accounts for this problem by making use of the "explored" list. If a node falls within the donut buffer of a subsequent node, it would not be included as part of that node's analysis. This prevents duplicate links.

Step 7   If the node is on the boundary, the algorithm stops exploring in that direction. Steps 1 through 6 are repeated for all the children nodes.

The resulting links have elevation associated with them given that the starting and ending nodes are created from the TIN and have an altitude value. Even though, the network generated is not truly in 3D, it is in 2 1/2 D and the links are not all in one plane.
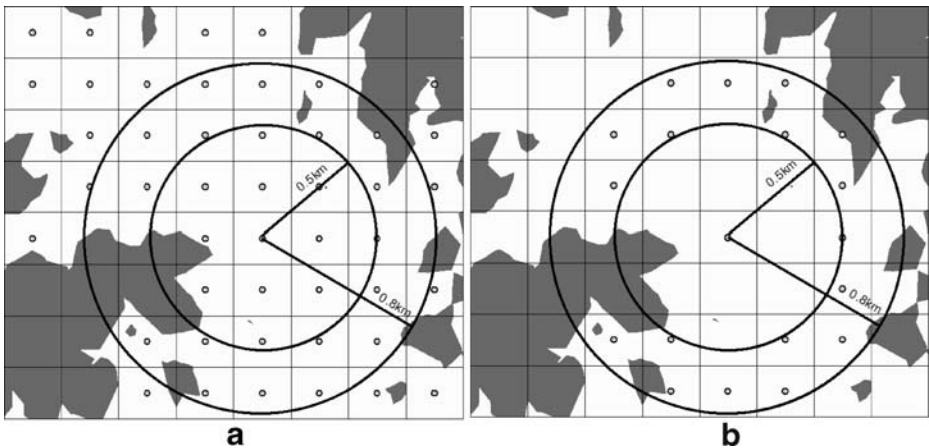

Fig. 1 Buffer generation procedure. (a) Creation of a minimum (0.5 km) and a maximum (0.8 km) buffer. (b) Nodes outside of the donut are discarded
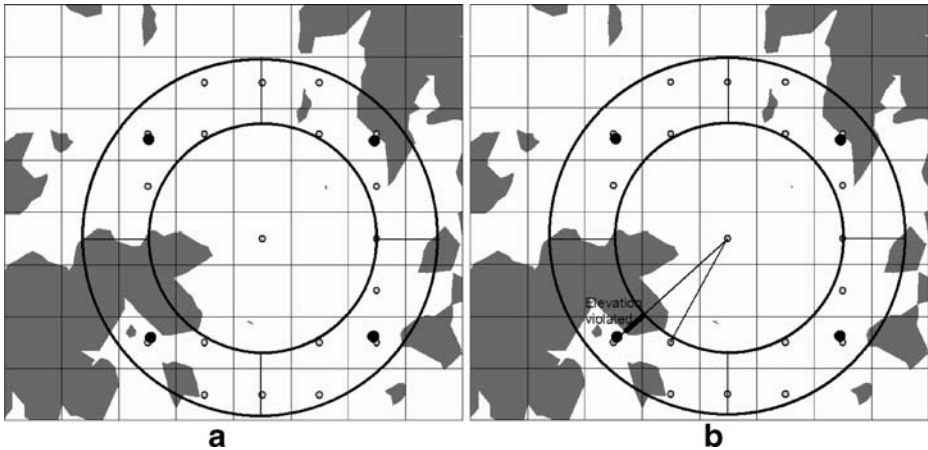
Fig. 2 Buffer partitioning. (a) Buffer partitioned in four quarters with centroid. (b) Closest node to centroid of quarter determined and linked

3.3 Implementation

The network generation procedure is implemented in Avenue, ArcView GIS (ArcView, ESRI Redlands, CA) scripting language. Avenue is a pseudo-object oriented programming language, which allows the manipulation but not the creation of new objects in the GIS environment. Scripts are incorporated as buttons/tools/menus within the GIS and allow the user to interact with all the other components available in the system. Two separate scripts are developed, one for each phase of the methodology. The execution of the first script results in an output (the nodes), that is then used as input to the second script. The interaction between the two scripts is done by means of the GIS environment. Both scripts have complexity $O(n^2)$ where n is the number of nodes.

3.4 Characterizing the network

The network generation procedure proposed introduces a level of flexibility by allowing changes in the input parameters like the node density and the network starting point. These changes typically yield different networks which then need to be evaluated based on their properties and final application domain. To do so the network structure is simplified by using concepts from graph theory (see Taaffe et al. 1996), in particular a matrix representation of the graph or what is known as a connectivity matrix (**C**-matrix). Based on this representation a set of well-known standard measures can be used to characterize the network. These include the minimum and maximum number of edges, the network diameter, and three connectivity measures, the degree of the nodes, the **T**-matrix and the **D**-matrix (for a detailed description and how to calculate the measures refer to Taaffe et al. (1996).

The minimum and maximum number of edges are used as a characteristic of the network, where neither one of the two extremes are desired. A minimally connected network would limit the routing process, while the maximally connected will complicate it. Therefore, in the examples presented as a result of the algorithm proposed, these values and the number of edges in the network generated are reported but there is no expected threshold.

The diameter K represents the number of linkages needed to connect the two most remote nodes in the network. As the connectivity among nodes increases, the diameter decreases. The degree of a node corresponds to the number of links connected to it and represents the nodes' accessibility. The accessibility of a node is measured by taking the sum of each row in the **C**-matrix. The **C**-matrix has a 1 in the cell entry if a link exists between the origin and destination nodes and a 0 otherwise. For a node i, the row Sum(i) represents the total number of other nodes in the network that are connected to i by a direct (or one-step) connection. The degree has serious limitations as a measure of accessibility as it does not account for indirect connections.

The total accessibility or **T**-matrix is an improvement on the **C**-matrix since it includes direct and indirect connections among pairs of nodes. To obtain the **T**-matrix, the **C**-matrix is powered to 1, and then to 2, up to the value of the diameter K, allowing the two most remote points to be linked to each other. To obtain the **T**-matrix, the powered **C**-matrices are added together. The **T**-matrix provides the number of paths between every pair of vertices; the number of ways in which a vertex can be reached from every other vertex.

The Shimbel distance or **D**-matrix addresses the problem of linkage redundancy (Shimbel 1953). In the **D**-matrix, the interest is in the shortest path between two nodes. The **D**-matrix is free from the redundancies of the **T**-matrix and does account for attenuation (the difference between direct and indirect links) by reducing the importance of multi linkages connections. The row sum of this matrix provides the minimum number of steps required to connect a particular node i to all other nodes in the network.

These measures provide a general indication on the quality and suitability of the network for the routing of UAVs.

3.5 Example

To illustrate the procedure proposed, a study area in the Central region of the state of Colorado is selected (see Fig. 3). The study area is 81 km long and 95 km wide and characterized by variation in elevation. The region has a non-rectangular shape which is
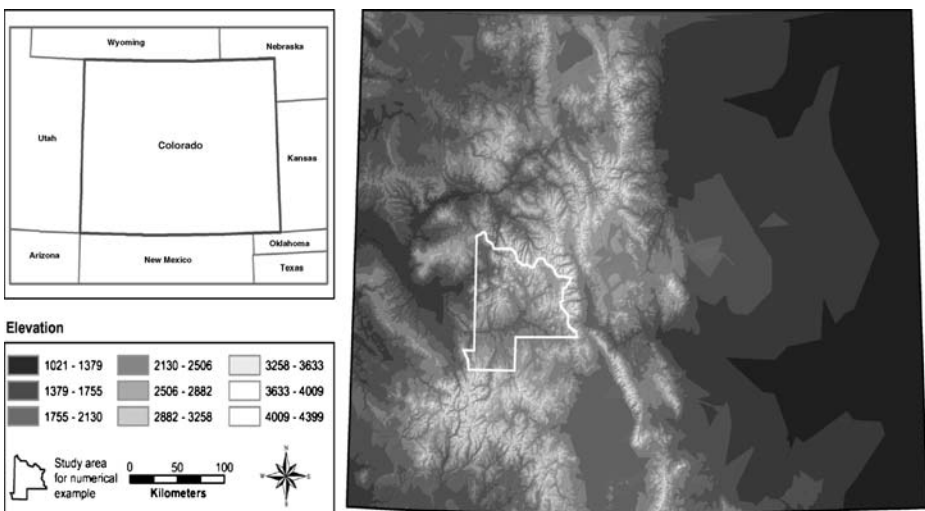

Fig. 3 Gunisson County, CO. The altitude values for the TIN range from 1,000 to 4,000 m

typical of counties in Colorado. Therefore, the provided measures correspond to the area within the county used in the network generation procedure. The TIN used as input to the network generation procedure is created in the GIS using DEM files available from the United States Geological Survey (USGS).

3.5.1 Network generation procedure

The network generation procedure is demonstrated by varying the algorithm parameters in particular the node density. The study area is discretized in two ways, following a sparse and a dense node density. In the first case the area has been divided into 10 rows and 10 columns, yielding a limited number of nodes. In the second case, the study area has been divided into 30 rows and 30 columns. Only the nodes below the detection altitude have been retained in each case (Fig. 4). The detection altitude is set at 3,500 m. This elevation is chosen based on the elevation changes in the terrain. There are a total number of 100 potential nodes for the sparser example and 900 potential nodes for the denser case. The process continues by selecting the starting point of the network generation process, represented by a star in Fig. 5, followed by the creation of the links. For the sparsely populated network, a minimum and maximum length for the links of 1 and 3 is chosen. Since the generated nodes in the dense network are too close to each other, and in order to prevent redundant links, a greater minimum and maximum length of 3 and 7 is selected, respectively. The resulting network on the right (30 rows * 30 columns) in Fig. 5 reaches more locations in the study area than the network on the left (10 rows * 10 columns). One of the reasons for this is the node density, which by being higher in number increases the probability of links being drawn in particular areas where elevations above the threshold are
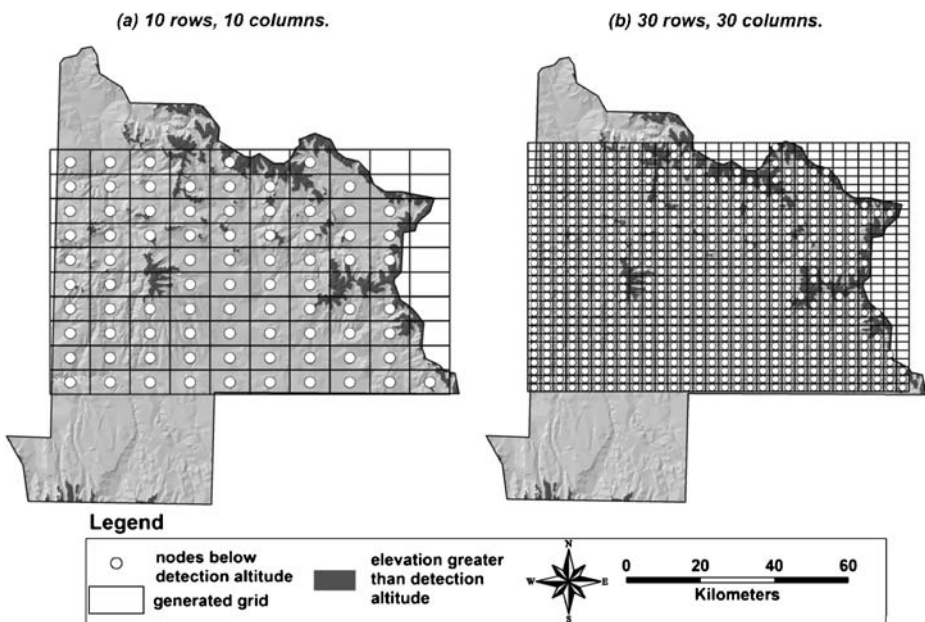

(a) 10 rows, 10 columns.   **(b) 30 rows, 30 columns.**

Legend

○ nodes below detection altitude

□ generated grid

■ elevation greater than detection altitude

N
W–E
S

0   20   40   60
Kilometers

Fig. 4 Systematic generation of nodes for Gunisson County for a grid of 10 by 10 in (a) and a 30 by 30 grid in (b)

common. This is the case in the upper left corner of the area and around high elevation areas (i.e., areas above the threshold elevation). The latter seems to be strongly developed in the northern part of the area (low elevations represented in grey in Fig. 5). In both cases, the algorithm generates a highly connected network in low altitude regions.

3.5.2 Network connectivity

The connectivity values for each node have been calculated for both networks (two different node densities), using the measures previously described. These measures offer the analyst evaluating the network a set of characteristics to go by to decide if it is suitable for the application domain; in this case routing of UAVs. The results are obtained by combining the capabilities of Transcad GIS (Caliper-Corp. 2002) and Matlab v. 6.5 (MathWorks 2004). In the following figures, the graph on the left side represents the sparse network; the one on the right depicts the denser network. The degree for each node is given in Fig. 6. Nodes characterized by a high degree are located in the valleys, low elevation areas. Given the flat areas that characterize valleys, links can generally be drawn in all directions without violating the elevation threshold. Figure 7 represents the D value or how many linkages are needed to connect a particular node to all other nodes. Since the network on the right is the largest in terms of links (largest defined as more links), it is not surprising to encounter greater D values, especially for remote nodes. The results for the T value are similar to the degree of each node, where nodes located in the middle and central regions exhibit a higher value, since this measure incorporates direct and indirect linkages to the other nodes. The nodes on the denser graph have greater T values, as a result of a greater total number of linkages (the T value maps are not included).

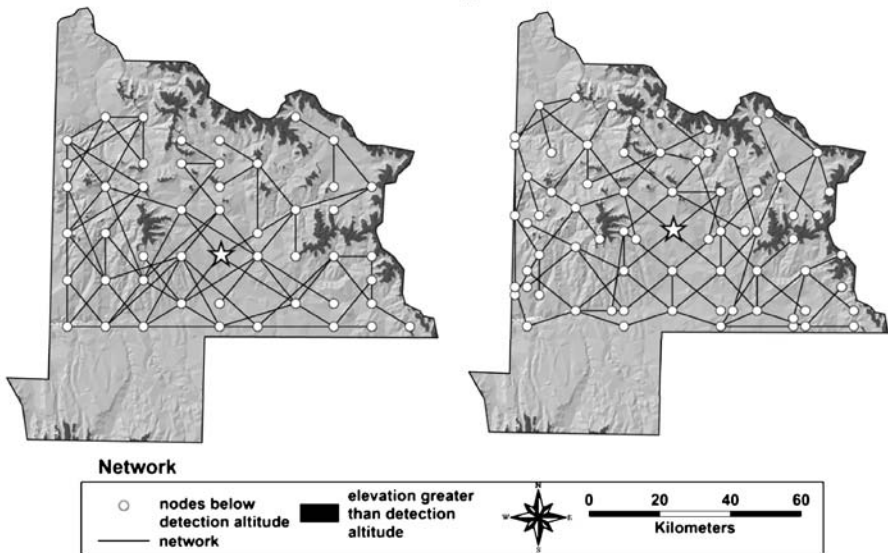(a) 10 rows, 10 columns. Max - min links: 1 - 3 intervals (b) 30 rows, 30 columns. Max - min links: 3 - 7 intervals

Network

○ nodes below detection altitude

— network

■ elevation greater than detection altitude

0   20   40   60
Kilometers

Fig. 5 Generated networks for Gunisson County with minimum number of links of 1 and maximum of 3 in (a). In (b) with minimum number of links of 3 and maximum of 7

(a) 10 rows, 10 columns. Max - min links: 1 - 3 intervals   (b) 30 rows, 30 columns. Max - min links: 3 - 7 intervals
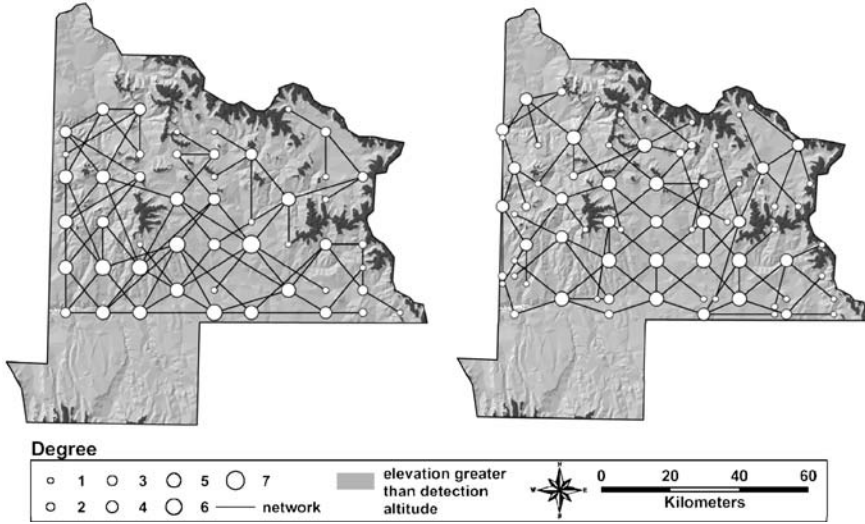

Fig. 6 Degree of each node for Gunisson County generated networks. In (a) for a grid of 10 by 10 and in (b) for a 30 by 30 grid

Table 1, summarizes the connectivity results for the two networks. Recall that the set of potential nodes for the dense network was approximately ten times greater than the sparse network. Yet the number of remaining nodes in the denser network is less than twice greater than the sparse network. The number of links is almost the same, 81 vs. 88, which results in similar diameters of the two networks. This result shows that more nodes do not necessarily

(a) 10 rows, 10 columns. Max - min links: 1 - 3 intervals   (b) 30 rows, 30 columns. Max - min links: 3 - 7 intervals
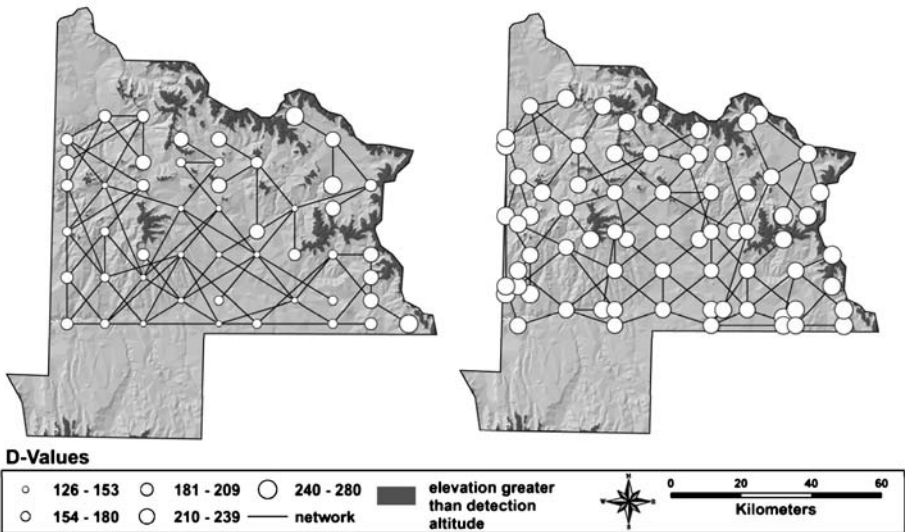

Fig. 7 D value for each node for Gunisson County generated network. In (a) for a grid of 10 by 10 and in (b) for a 30 by 30 grid

imply more links. The maximum D value for the denser network is higher, given the density of nodes in the network. Greater T values are obtained for the denser network, stressing its superior connectivity provided the higher node density.

This example provides an overview of the method as well as how traditional network analysis measures are used to characterize the networks generated. Measures help compare one network to another and can provide the user with tools to decide on what network it would be more advantageous to perform a routing operation on. The Colorado area used in this example is relatively small and changes in elevation are not distributed homogeneously over the region, therefore in the next section the complete state is used to test the algorithm.

4 Larger problems

The previous example illustrated the methodology and demonstrated the nature of the problem. However, it is more realistic to be concerned about larger problems exhibiting greater altitude variation. Increasing the size of the study area helps in testing the stability of the algorithm. The case study presented in this section embraces the entire area of the State of Colorado (Fig. 3), with an area of 269,595 sq. km, and a mean elevation of 2,073 m. Its highest point culminates at 4,399 m (Mount Elbert) and its lowest point at 1,021 m.

4.1 Network generation procedure

Similarly to the previous example, in this case the network generation procedure is tested under two different scenarios. The study area is discretized following a sparser (50 rows, 60 columns) and a denser (69 rows, 94 columns) pattern. With a detection altitude of 3500 m, the sparser grid results in a set of 2,900 nodes and the denser in a set of 6,275 potential nodes. A starting point in the southern part of Colorado is selected, represented by a star in Fig. 8. Since there are fewer points in the sparse network, a greater minimum and maximum length for the links of 8 and 13 is used. A total of 137 points remain and 293 links are created (Table 2).

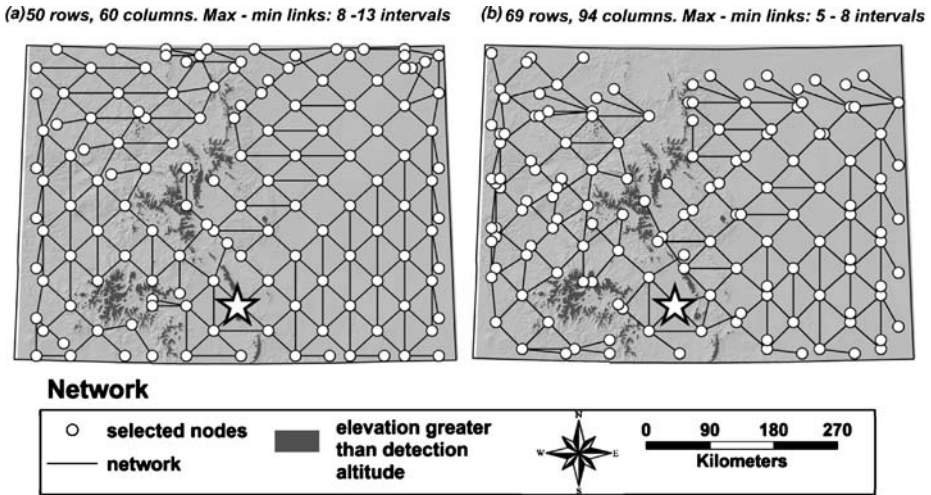| Table 1 The connectivity matrix		Sparse network 10*10	Dense network 30*30
	V	49	72
	e_{min}	48	71
	e_{max}	1,176	2,556
	e	81	88
	K	8	9
	D-matrix		
	Sum$_{min}$	126	220
	Sum$_{max}$	280	473
	Total	9,304	25,272
	T-matrix		
	Sum$_{min}$	2,256	5,644
	Sum$_{max}$	507,186	365,071
	Total	8,503,196	651,548

Fig. 8 Generated networks for the State of Colorado with minimum number of links of 8 and maximum of 13 in (a). In (b) with minimum number of links of 5 and maximum of 8

For the denser network, a minimum and maximum length for the links of 5 and 8 is chosen, respectively. A total of 146 nodes are kept and 311 links are generated. Central Colorado is characterized by a range of elevation values, increasing the difficulty of connecting the western to the eastern part. The network generated from densely populated nodes on the right has a single link connecting the eastern and western part of the study region. This is due to the location of the starting point, the link lengths, and the number of rows and columns defined. In general, both networks cover the study area well and particularly in the densely populated network, several nodes are very close to each other, causing redundancy. It is important to note that the starting point and the link lengths for the network generation procedure are chosen based on knowledge about the area. Longer links in the sparse density might result in a network covering less area. These are examples of the

Table 2 Connectivity summary		Dense network 50*60	Sparse network 69*94
	V	137	146
	e_{min}	136	145
	e_{max}	9,316	10,585
	e	293	311
	K	21	19
	D-matrix		
	Sum$_{min}$	936	869
	Sum$_{max}$	1,859	1,685
	Total	203,428	168,034
	T-matrix		
	Sum$_{min}$	5.40E + 08	4.70E + 11
	Sum$_{max}$	1.80E + 12	1.40E + 15
	Total	2.80E + 13	3.20E + 16

kind of networks that can be generated by the procedure. Other networks could be generated based on the user's needs and by changing the parameters.

4.2 Network connectivity

For both networks created, the connectivity measures for each node have been calculated. In Figs. 9 and 10 the picture on the left side represents the sparse network; the one on the right displays the densely populated. Figure 9 displays the degree of each node. A maximum of 6 is obtained in both cases, but the left graph shows a greater number of high degree nodes. The average degree value for the left graph is 3.35 and 2.51 for the one on right. Note that the network on the right shows that several low-degree nodes are in the direct vicinity of high-degree nodes. Potentially these could be merged together to increase the degree values. This in turn will result in a more connected network. The D value is on average greater for the left graph (1,484.8) than for the one on the right (1,150.9). This indicates that the sparsely populated network has a greater connectivity overall. Figure 10 illustrates, for each node, how many steps are needed to be connected to all other nodes in the network. Nodes on the edge of the network have a greater value given their location and therefore the greater number of links required to link them. The centrally-located nodes have a lower value, also as a result of their location. The sparsely populated network on the left has a lower T value on average (2.05E11) than the denser network (2.2E14). In this case given the increased number of edges and vertices the T values are large in magnitude. Table 2 summarizes the connectivity results for the two networks.

4.3 Visibility analysis

An issue which remains is to determine the amount of area that can be covered by traveling the created network. Viewshed analysis has been used, for example by Lee and Stucky (1998) to determine least-cost paths on DEMs based on different criteria. The procedure identifies the areas that can be traversed given a set of constraints. In the case of a reconnaissance mission over a network, the purpose is to collect the maximum amount of
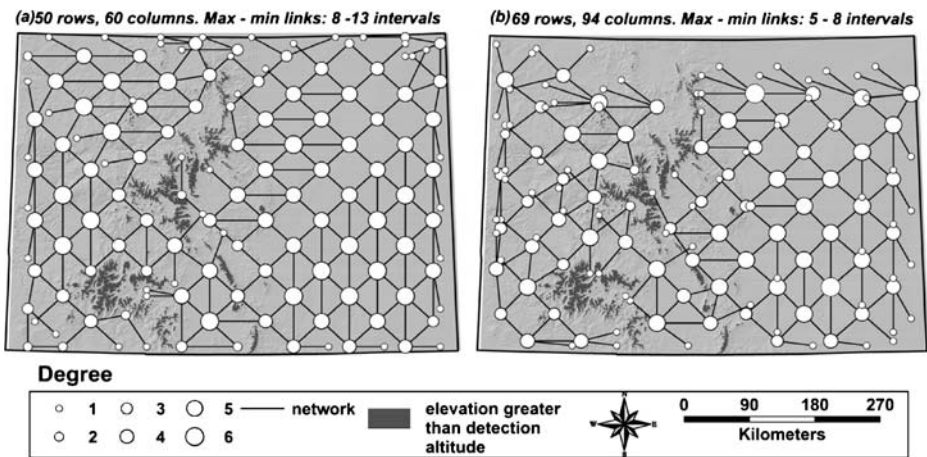

Fig. 9 Degree of each node for the State of Colorado generated networks. In (a) for a grid of 50 by 60 and in (b) for a 69 by 94 grid
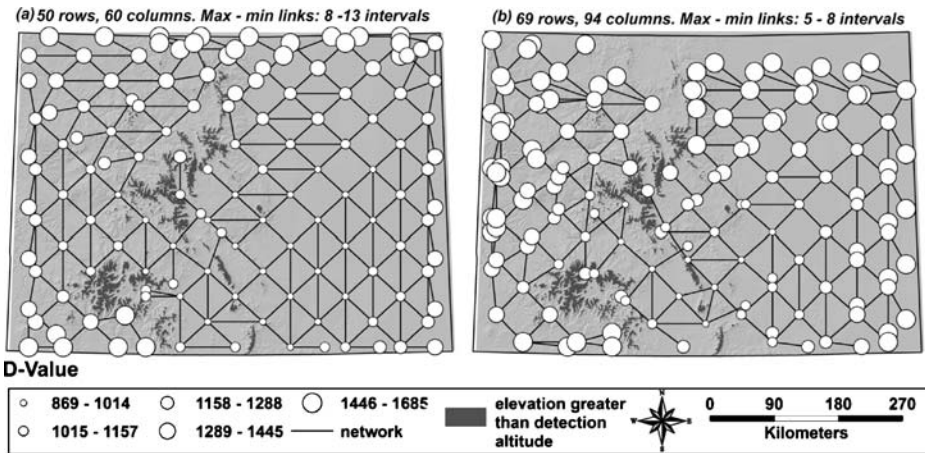
Fig. 10 *D* value for each node for the State of Colorado generated network. In (a) for a grid of 50 by 60 and in (b) for a 69 by 94 grid

information about the enemy terrain captured using imagery sensors carried by the UAV. Therefore, the success of the mission can be determined by the amount of terrain visible from the network. In a GIS, it is possible to determine the visible terrain by performing a visibility analysis from a linear feature. The analysis consists of determining the visible cells within an area from the begin-node and end-node of that feature. However, when long segments are present this approach is not realistic given that the areas that can be seen from locations along the segment are not accounted for. This issue could be addressed by computing visibility at a very fine interval along the line segments. However, it would be a very time-consuming process to collect such information. An appropriate interval for this kind of analysis should be determined based on the terrain extent, the topography, and the minimum and maximum lengths chosen to create the links in the network. In the case of Colorado an interval of 5 km is identified as suitable.

To generate the potential visibility sample points on the network, two scripts are written in Avenue. The first script reports the length of each segment on the network. The second script allocates points on these segments, 5 km apart from each other. This continues until the distance separating the last generated point from the end-node is less than 5 km. When a segment is smaller than 1 km, the algorithm stops after the first step; only one point will be generated, corresponding to the end-node of the segment. A table is created as part of the script's output. Each record in the table corresponds to a new sample point and has three fields: the new sample point ID, the corresponding road segment ID, and the distance separating that point from the end-node of that segment. The sample points generated are displayed using standard GIS commands.

Capitalizing on the locations of the generated points and altitude information for the state of Colorado, it is possible to determine whether any region in the study area can be viewed from the path of a UAV, given a visibility distance and the height of the vehicle. The visibility analysis accounts for obstructions along the path (see Clarke 1995). This analysis is performed using ArcInfo GIS (ESRI) by means of macros written in AML, the ArcInfo macro language. Visibility is computed for an infinite coverage radius and a flying height of 3500 m (corresponding to the detection altitude used in the Colorado area). The result of the visibility analysis is a binary surface with values of 1 when a location is visible

from a point along the flying route and 0 otherwise. Figure 11 portrays the results from the visibility analysis for the two networks in Colorado. The small points denote the discretized network. For each of those points, a visibility analysis has been performed.

Variation in the visibility appears as a result of the topography in the southeast direction. For the sparse network on the left, 27.59% or 1,228,933 cells are non visible. For the dense network, this number raises to 30.78% or 1,371,151 cells. The surface on the right shows a greater amount of non-visible cells, especially in the northern and central regions, given the absence of links generated in those areas. This illustrates the fact that a greater number of initial begin- and end-nodes does not systematically guarantee better visibility results. The surface on the left is characterized by less non-visible cells, because there are many more links connecting eastern to western regions. This visibility analysis is simple in regards to the input parameters used; however, there is room for improvement.

The visibility parameters are terrain/network specific. Threshold elevations vary from one terrain to another; therefore they need to be modified as the case grants it. If the extent of the terrain is small—a small area is being used—the distance of 5 km between nodes might be too long and would need to be adjusted. However, there are no major modifications required in terms of the algorithms that calculate the visibility.

5 Advantages of the network generation procedure

Advantages of the proposed network generation procedure are discussed in comparison to the NDPs and route planning methods discussed in the literature review section. As mentioned, the NDPs have as an objective to identify a subgraph within a graph based on a set of constraints. The procedure proposed departs from this approach by designing a network which corresponds to the input graph of the network design problems. This allows the use of traditional route planning and path determination algorithms to be used where no network is originally available (like in the UAVs case). The procedure goes beyond the use of the simple 2D space by incorporating an altitudinal dimension in the creation of the network, which results in links at different elevations based on the relief. This is a characteristic that the previously discussed approaches do not have and is possible by using
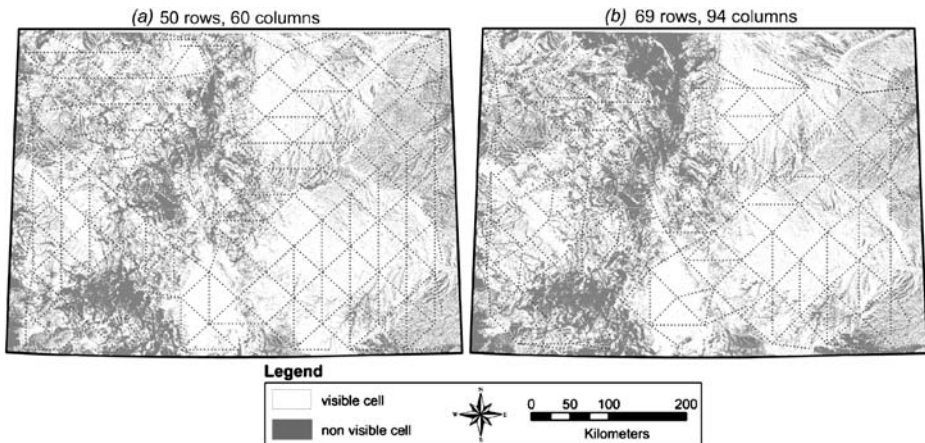

Fig. 11 Visibility surface from the sparse network on the left and the denser network on the right

GIS as a development environment. Even though in the literature references can be found in the use of GIS and OR to solve NDPs (i.e., Cao et al. 1999), in such cases GIS has only been used for data input and display. In the network generation procedure presented here the GIS is used as part of the implementation method and its inherent functions are fully exploited.

The network generation procedure proposed has more similarities with the route planning methods discussed than it does with the NDPs. The method could be compared to creating all possible paths for a robot to move within an environment given a set of obstacles. The main difference is, however, that in robot motion planning the location of obstacles that need to be avoided is in 2D, while in the case of the UAVs it is in 3D (i.e., mountains, enemy locations). Although, there has been research in identifying shortest paths using viewshed analysis in raster images (Lee and Stucky 1998), like in the case of the robots, it is only one path and not multiple paths.

In summary, the procedure proposed has the following advantages. First, it incorporates elevation for the links in the network. Second, it considers 3D obstacles such as mountains and enemy threats. Third, it creates a network instead of defining a subgraph from a network based on a set of constraints, which in turn allows this network to be used for routing a fleet of vehicles instead of just one vehicle at a time. And finally it is implemented using GIS as a development environment which introduces flexibility into the process.

6 Limitations and issues

6.1 Limitations

The limitations discussed are the result of design decisions and would be incorporated in a future version of the algorithm. These include: node shift, minimum flying elevation, presence of obstacles in the terrain, merging of nodes, and dynamic design of nodes depending on terrain complexity.

Node shift The procedure proposed is not flexible in displacing the location of existing nodes. This would be of use when two points cannot be connected because of the presence of a mountain/obstruction—that is above the detection altitude—along the link connecting them. Figure 12 illustrates that it may still be possible to connect two nodes by displacing them a given distance. The motivation behind the node shift is that if the obstacle is avoidable by a small deviation, the UAV can perform this operation without any problem.

Minimum flying elevation There is a safety concern associated with the maximum flying altitude as it can be detected by enemy radars. There is also a risk of accident when the plane flies too low. The introduction of a minimum flying altitude above the existing terrain can improve upon the automated procedure presented it this paper.

Presence of obstacles in the terrain A more realistic approach consists of adding obstacles in the terrain. Based upon their locations, nodes of the network that could be in their control area can be deleted.

Merging of Nodes As mentioned, a high node density may result in the presence of nodes that are very close, yet not connected to each other. This forms a cluster of nodes. Closeby
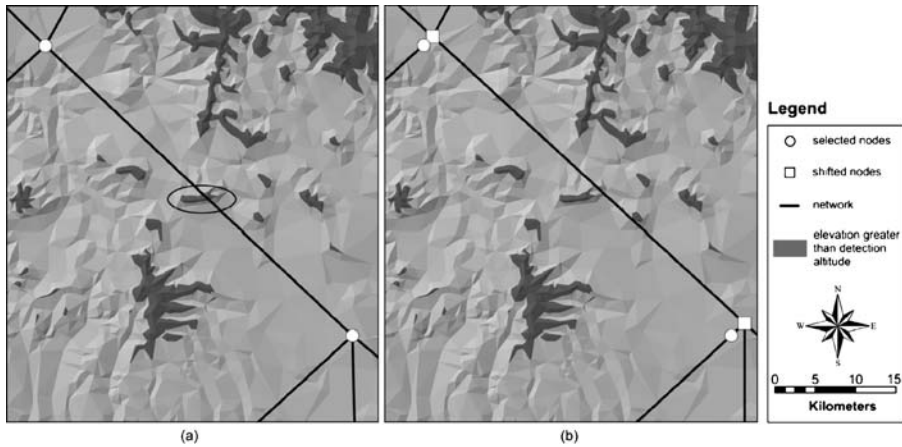
Fig. 12 Illustration of the *node shift* concept for a portion of Colorado

nodes should be reduced to a single node, increasing the total connectivity of the graph. Figure 13 shows an example.

Dynamic design of nodes depending on terrain complexity In regions where the terrain exhibits a strong variation in altitude, a greater node density can be chosen and links made smaller. In areas with lots of open space, links should be larger at a coarser resolution.

6.2 Issues

A set of issues are identified as important while testing the algorithm under different scenarios and varying input parameters. Some can be addressed by further testing of the algorithm, others require additional research. These include: the starting point; time performance of the algorithm; relationship between T value and altitude; number of buffer sections; node resolution, interval, and connectivity; relationship between terrain variation and connectivity; and potentiality for continuous space.

The starting point Changes in the starting point can and will generate completely different networks. The starting point is related to the terrain topography, the lengths desired for the links, and the node density. Therefore, the user should be familiar with the network generation process in order to obtain optimal results. The best way to do so is by trial and error.

Time performance of the algorithm The time performance of the algorithm has not been reported since it was not seen as an issue during the testing of the algorithm. However, research is needed on the time needed to generate (1) the set of nodes and (2) the resulting network, as a function of the selected density.

Relationship between T value and altitude It is important to analyze whether the altitude of a node and the relative altitude of its surrounding nodes can be a reason of high T value. When a node at low altitude is surrounded by other low-elevated nodes (such as in a valley or a plain), the connectivity increases.
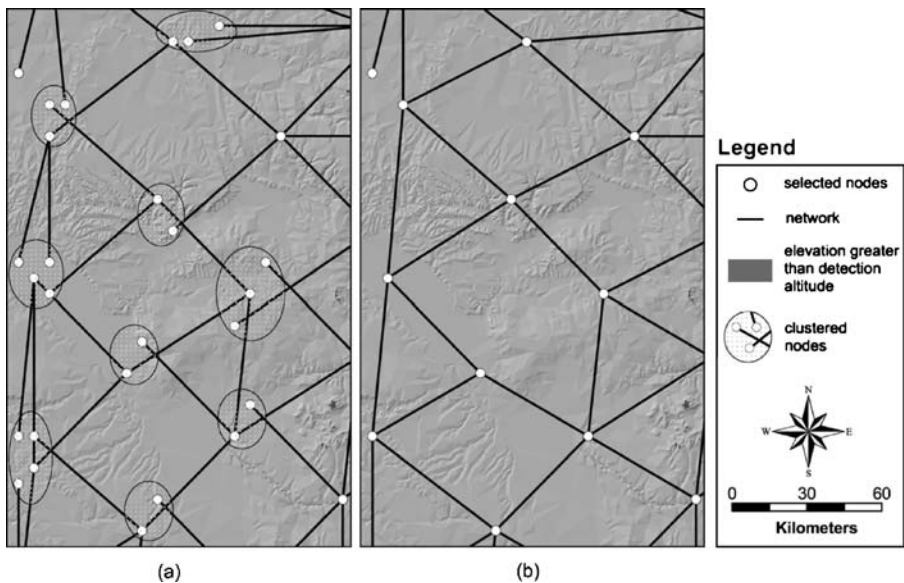
Fig. 13 Illustration of the *node declustering* concept for the southwestern corner of Colorado. Close by nodes can be aggregated to improve upon the generated network

Number of buffer sections The number of buffer sections is not modified in the examples presented given that this will require modifying the code, based on the procedure discussed. However, it can be anticipated that increasing/decreasing the number of sections will have an effect in the complexity of the network. Adding more sections, for example, results in more links starting at every node being explored. The resolution defined for the grid will also have an effect if more buffer sections are used. If a large number of columns and rows are specified, using more than four sections will not be a problem because potentially nodes to link to could be found in every direction (high node density will increase the probability of finding nodes that meet the elevation threshold in every direction, Fig. 14(a)). However, if few columns and rows are used, increasing the number of sections will increase the probability of not finding nodes to link to (low node density will decrease the probability of finding nodes that meet the elevation threshold in every direction, Fig. 14(b)).

On the other hand, if less than four sections are used the number of links leaving a node will be reduced. Based on the resolution of the grid, more or less nodes will be available to be part of the network. It is suspected that in the case of less than four sections and a sparse grid, there will still be nodes available to link to, which is not necessarily the case when the number of sections increases. The consequences of using more/less buffer sections needs to be further explored.

Node resolution, link interval and connectivity One concern pertains to studying whether there is a relationship between the densities of the nodes versus the final connectivity of the network for a given starting point. Along those lines, it is also interesting to look at the relationship between the maximum and minimum length of a segment versus the generated network.

Relationship between terrain variation and connectivity Can the terrain roughness be a limiting factor for connectivity? Can a network be spread out better if the spatial pattern in
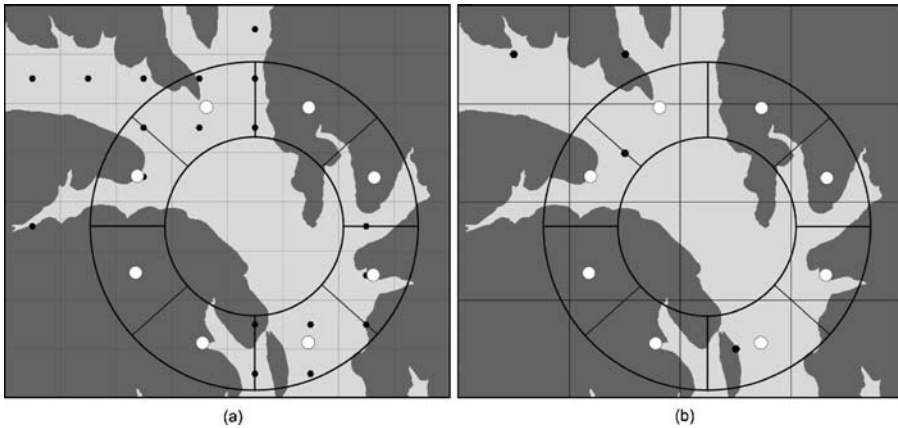
Fig. 14 Number of buffer sections effects

terrain altitude exhibits less variation than in mountainous areas for instance? And hence, how can this terrain roughness be quantified? This in turn could give some indication on where to select the starting point.

Potentiality for continuous space? If the nodes can be located anywhere, can the study area be covered similarly to a location – allocation problem, just from origin to destination?

7 Conclusions

This paper presents an automated generalized procedure to identify the nodes and links to generate a network, accounting for topographic information of the enemy terrain. This information is made available in the form of a TIN. The method proposed is fast and flexible given that it allows the user to: input a threshold altitude value to avoid radar detection and terrain collisions, specify the node density in the study area, and specify the minimum/maximum link lengths of the desired network. A Triangular Irregular Network (TIN), extracted from a DEM, is the only required input.

Measures of connectivity characterizing the different networks have been presented, thereby allowing to compare the generated networks. Such measures include—but are not limited to—the degree of a node, its T, and D values. A case study for the State of Colorado has been performed. Results indicate that a dense concentration of initial nodes does not guarantee the creation of many links. Generally, the degree of a node is greater where altitude variation is smaller. Additionally, a high density of initial nodes reduces the degree value, due to the presence of such nodes very close to each other, at distances smaller than the minimum link length. Since the ultimate goal of a reconnaissance mission is to collect terrain information, a visibility analysis is performed from a discretized network. Variation in the visibility appears as a result of the topography in the southeast direction, especially in the northern and central regions. More cells are visible for the sparse network. This illustrates the fact that a greater number of initial begin- and end-node does not systematically guarantee better visibility results. More testing should be conducted to identify the sensitivity of the parameters to the algorithm.

The information provided in this paper is significant for the planning of an air reconnaissance mission given that it provides insights on the parameters affecting the

generation of a network in a three-dimensional framework. The procedure can also be applied to the field of robot and motion planning. In this paper, we have suggested an automated methodology to design a network for the routing of unmanned aerial vehicles within a GIS framework. The procedure has limitations and is sensitive to some of the input parameters which result in the set of issues discussed, leaving room for improvements in future research.

References

Ahuja RK, Magnanti TL, Orlin JB (1993) Network flows: theory, algorithms, and applications. Prentice-Hall, Englewoods Cliffs, New Jersey

Caliper-Corp. (2002) Transcad GIS. Newton, Massachusetts

Cao B, Sun M, Macleod C (1999) Applying GIS and combinatorial optimization to fiber deployment plans. Journal of Heuristics 5:385–402

Clarke KC (1995) Analytical and computer cartography. Prentice Hall, Englewood Cliffs, New Jersey

Feremans C, Labbé M, Laporte G (2002) Generalized network design problems. Accessed on: 2005. http://citeseer.ist.psu.edu/

Fox R, Garcia A, Nelson M (2000) A generic path planning strategy for autonomous vehicles. Report No. CS-00-25.The University of Texas – Pan American

Goemans MX, Goldberg AV, Plotkin S, Shmoys DB, Tardos É, Williamson DP (1994) Improved approximation algorithms for network design problems. Proceedings of the Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, pp 223–232

Gupta A, Kleinberg J, Kumar A, Rastogi R, Yener B (2001) Provisioning a virtual private network: a network design problem for multicommodity flow. STOC, Hersonissos, Crete, Greece, pp 389–398

Holle AT, Moret ME, Saia J, Boroujerdi A (1995) Network routing models applied to aircraft routing problems. Proceedings of Winter Simulation Conference, Arlington, pp 1200–1206

Jun M, D'Andrea R (2003) Path planning for unmanned aerial vehicles in uncertain and adversarial environments. In: Butenko S, Murphey R, Pardalos P (eds) Cooperative control: models, applications and algorithms. Kluwer Academic Publishers, Massachusetts, USA

Latombe J-C (2001) Robot motion planning. Kluwer, Boston

Lawler EL, Lenstra JK, Kan AHGR, Shmoys DB (eds) (1985) The traveling salesman problem: a guided tour of combinatorial optimization. Wiley, New York

Lee J (1991) Comparison of existing methods for building triangular irregular network models of terrain from grid digital elevation models. Int J Geogr Inf Syst 5(3):267–285

Lee J, Stucky D (1998) On applying viewshed analysis for determining least-cost paths on digital elevation models. Int J Geogr Inf Sci 12(8):205–891

Levinson D, Karamalaputi R (2003) Predicting the construction of new highway links. J Transp Stat 6 (2/3):81–89

Magnanti TL, Raghavan S (2005) Strong formulations for network design problems with connectivity requirements. Networks 45(2):61–79

Magnanti TL, Wolsey LA (1995) Optimal trees. In: Magnanti TL, Monma CL, Nemhauser GL (eds) Handbooks in operations research and management science. Elsevier Science, Amsterdam

Magnanti TL, Wong RT (1984) Network design and transportation planning: models and algorithms. Transp Sci 18(1):1–55

MathWorks (2004) MatLab. Natick, Massachusetts

Mihail M, Shallscross D, Dean N, Mostrel M (1996) A commercial application of survivable network design: ITP/INPLANS CCS network topology analyzer. Proceeding of the Seventh Annual ACM-SIMA Symposium on Discrete Algorithms, Atlanta, Georgia, pp 279–287

Nelson ML (1998) A software control architecture for autonomous vehicles. 31st Hawaii International Conference on Systems Sciences HICSS-31, Hawaii, pp 226–232

Nikolos IK, Valavanis KP, Tsourveloudis NC, Kostaras AN (2003) Evolutionary algorithm based offline/online path planner for UAV navigation. IEEE Trans Syst Man Cybern Part B Cybern 33(6):898–912

Noltemeier H, Wirth H-C (1999) Network design and improvement. ACM Comput Surv 31(3):1–5

Openshaw S, Openshaw C (1997) Artificial intelligence in geography. Wiley, New York

Poorzahedy H, Abulghasemi F (2005) Application of ant system to network design problem. Transportation 32:173–251

Ravi V, Cem H, Troy P, Roger Q (2004) Evolutionary path planning for autonomous air vehicles using multiresolution path representation. Report No. Orbital Research INC

Sasiadeck JZ, Duleba I (2000) 3-D local trajectory planner for UAV. J Intell Robot Syst 29:191–210

Shimbel A (1953) Structural properties of communication networks. Bull Math Biophys 15:501–507

Sleumer NH, Tschichold-Gürman N (1999) Exact cell decomposition of arrangements used for path planning in robotics. Report No. 329. Institute of Theoretical Computer Science

Smith, MJd (2004) Distance transforms as a new tool in spatial analysis, urban planning, and GIS. Environ Plann B Plann Des 31:85–104

Stentz A (1994) Optimal and efficient path planning for unknown and dynamic environments. IEEE Int Conf Robot Automation

Taaffe EJ, Gauthier HL, O'Kelly ME (1996) Geography of transportation. Prentice Hall, Upper Saddle River, New Jersey

Thyagarajan K, Batta R, Karwan MH, Szczerba RJ (2005) Planning dissimilar routes for military units. Mil Oper Res 10:25–42

Xie F, Levinson D (2006) The decline of over-invested transportation networks. Transportation Research Board Meeting, Washington District of Columbia

Yamins D, Rasmussen S, Fogel D (2003) Growing urban roads. Netw Spat Econ 3(1):69–85