Design and Analysis of a 3-D Gauss-Markov Mobility Model for Highly Dynamic Airborne Networks

Dan Broyles, Abdul Jabbar, and James P.G. Sterbenz
Department of Electrical Engineering & Computer Science
The University of Kansas
Lawrence, KS 66045
{dbroyl01,jabbar,jpgs}@ittc.ku.edu

ABSTRACT

Accurate mobility models are needed to simulate the physical movement of nodes in a highly-dynamic aeronautical network. The fundamental problem with many synthetic mobility models is their random, memoryless behavior. Airborne ad hoc networks require a flexible memory-based 3-dimensional mobility model. We present a new 3-dimensional implementation of the Gauss-Markov mobility model for airborne telemetry network simulations, and compare its behavior to memoryless models such as random waypoint and random walk using the ns-3 simulator.

I. INTRODUCTION AND MOTIVATION

Airborne communication presents a challenging environment for mobile ad hoc networking. High mobility, limited bandwidth and transmission range, and unreliable noisy channels, create a harsh environment for communications [1]. The problems of congestion, collisions, and transmission delays are only made worse in an ad hoc multi-hop environment [2]. Additionally, we cannot assume that any two nodes will be within transmission range of each other for very long. Two highly mobile nodes moving in opposite directions at hypersonic relative velocity might only expect to have a few seconds of opportunity to discover, setup, and transfer data, or to make a successful handoff.

New protocols are emerging to address these difficulties [3] and there is a need to evaluate their performance, initially, through realistic simulations. The ns-3 simulator is an emerging network performance simulation environment with several mobility models already built in, including random direction 2D, random walk 2D, random waypoint, constant velocity, constant acceleration, and constant position [4].

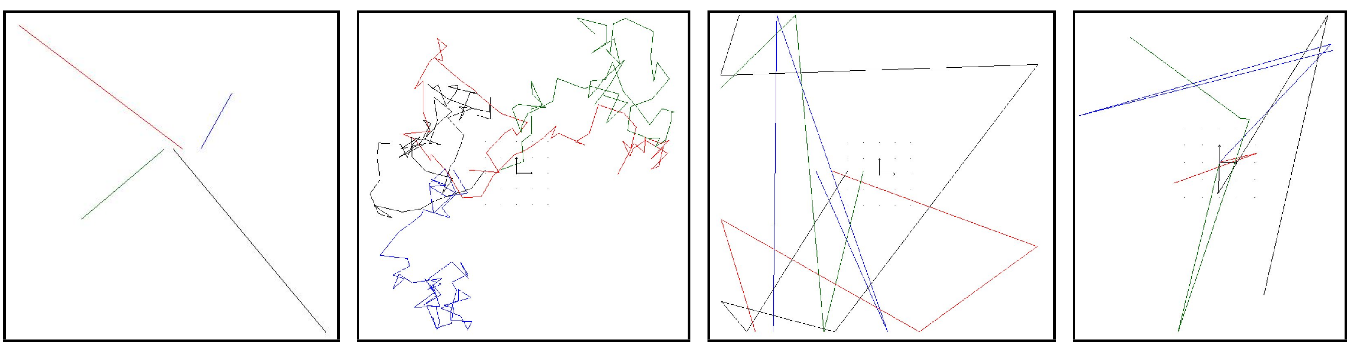
The fundamental problem with many synthetic mobility models is their random, memoryless behavior. Simulations using these mobility models exhibit unnatural movements with abrupt and often extreme changes in velocity and direction, uncharacteristic of highly-mobile airborne nodes. These mobility models are insufficient in simulating a highly-dynamic airborne ad hoc network. They also lack support for 3-dimensional position allocation, relative velocity between nodes in 3-D space, and realistic flight be-

havior. In this paper, we present a 3-dimensional implementation of the Gauss-Markov mobility model suitable for use in multi-tier, highly-dynamic MANET simulations. We present an analysis of the proposed model and its characteristics based on ns-3 simulations.

II. CURRENT NS-3 MOBILITY MODELS

As of ns-3.7, the built-in mobility models include constant velocity, random walk 2D, random direction 2D, and random waypoint. We now present a brief summary of these models.

In constant velocity, nodes proceed along their initial velocity vector for the duration of the simulation as shown in Figure 1a. There are no geographic boundaries in this model. In random walk 2D, each node is given a random trajectory (speed and direction) and travels on that trajectory for a fixed period of time or a fixed distance as shown in Figure 1b. When nodes reach the limits of the 2-dimensional boundary, they bounce off in a new direction mirroring the previous direction and velocity. Figure 1c shows an example of the random direction model, in which nodes travel on a random trajectory until they reach the 2D boundary, at which time they pause for a random period of time and head off in a new random direction and speed. In the random waypoint model, as illustrated in Figure 1d, each node travels to a random waypoint (x,y coordinate), pauses for a period of time, and then heads off to another waypoint. The waypoints, node speeds, and pause times are modeled as uniformly distributed random variables.



(a) Constant Velocity (b) Random Walk (c) Random Direction (d) Random Waypoint

Figure 1: ns-3 Memoryless Mobility Models

These simple synthetic mobility models do not mimic the motion of airborne nodes very well. The nodes undergo sudden changes in speed and direction at random, which is uncharacteristic of real objects (e.g. aircraft).

III. AIRBORNE MOBILITY MODEL REQUIREMENTS

Some of the challenges faced by airborne networks include high mobility, limited bandwidth, limited transmission range, and intermittent connectivity [1]. Airborne nodes are highly dynamic and require 3-dimensional models. The current ns-3 mobility models are designed for 2-dimensional movement. The random waypoint model, the basic design of which is not limited to 2 dimensions, can only select from waypoints generated from the ns-3 position allocation class. And at this time, there are only three position allocation models identified in the position allocation class: grid position, random rectangle, and random disc, all of which are 2-dimensional allocation schemes.

Simulated airborne nodes represent the motion of physical objects flying through the air, for which natural laws must be obeyed. This implies that the path of an airborne node will not be completely random, but its position at any point in time will be dictated largely by its previous position and velocity vector. Therefore, the mobility model must have memory. The mobility models mentioned previously are all memoryless. One characteristic of a memoryless mobility model is the existence of very sharp and sudden changes in direction and speed. One measure by which a good airborne mobility model should be judged is the fluidity of movement between consecutive positions.

A multi-tier MANET topology is required to support various quality of service needs while also attempting to provide seemless airborne connectivity [5, 3]. Multi-tier networks introduce coverage asymmetry, since higher altitude nodes can potentially have far more neighbors than ground-based or low-altitude nodes. An airborne MANET will have nodes at different altitudes performing different functions and with different communication and mobility characteristics. Thus our model also supports multi-tier networks.

IV. GAUSS-MARKOV MOBILITY ALGORITHM

In this section, we first present the basic Gauss-Markov algorithm for modeling two-dimensional mobility. We then extend the model to three dimensions as well as introduce new parameters to accurately model the mobility of airborne nodes.

A. The Basic 2D Gauss-Markov Algorithm

The Gauss-Markov mobility model is a relatively simple memory-based model with a single tuning parameter, alpha α, which determines the amount of memory and variability in node movement. In this paper, we describe other tuning parameters that have a significant impact on the dynamics and characteristics of the Gauss-Markov mobility model as well as the selection of alpha.

In the traditional 2-dimensional implementation of the Gauss-Markov model, each mobile node is assigned an initial speed and direction, as well as an average speed and direction. At set intervals of time,

a new speed and direction are calculated for each node, which follow the new course until the next time step. This cycle repeats through the duration of the simulation. The new speed and direction parameters are calculated as follows [6]:

$$
\begin{aligned}
s_n &= \alpha s_{n-1} + (1-\alpha)\bar{s} + \sqrt{(1-\alpha^2)}s_{x_{n-1}} \\
d_n &= \alpha d_{n-1} + (1-\alpha)\bar{d} + \sqrt{(1-\alpha^2)}d_{x_{n-1}}
\end{aligned} \tag{1}
$$

where α is the tuning parameter, $\bar{s}$ and $\bar{d}$ are the mean speed and direction parameters, respectively, and $s_{x_{n-1}}$ and $d_{x_{n-1}}$ are random variables from a Gaussian (normal) distribution that give some randomness to the new speed and direction parameters.

Special Case: $\alpha = 0$
When α is zero, the model becomes memoryless; the new speed and direction are based completely upon the average speed and direction variables and the Gaussian random variables.

$$
\begin{aligned}
s_n &= \bar{s} + s_{x_{n-1}} \\
d_n &= \bar{d} + d_{x_{n-1}}
\end{aligned} \tag{2}
$$

Special Case: $\alpha = 1$
When α is 1, movement becomes predictable, losing all randomness. The new direction and speed values are identical to the previous direction and speed values. In short, the node continues in a straight line.

$$
\begin{aligned}
s_n &= s_{n-1} \\
d_n &= d_{n-1}
\end{aligned} \tag{3}
$$

Setting α between zero and one allows for varying degrees of randomness and memory. In addition to α, the dynamics of the Gauss-Markov mobility model are greatly influenced by other variables like the time step, the selection of the average speed and direction, and the mean and standard deviation chosen for the Gaussian random variables. For example, choosing a standard deviation on the Gaussian distribution governing the direction that is much larger than the average direction generates a very different movement pattern than if the standard deviation and average direction values are similar.

B. Extending the Model to Three Dimensions

In this section, we discuss several methods to extend the basic Gauss-Markov 2-dimensional model to three dimensions. The first approach is to apply the Markov process to the x, y, and z axis of a 3-dimensional velocity vector. The velocity vector is computed as:

$$
\begin{aligned}
x_n &= \alpha x_{n-1} + (1-\alpha)\bar{x} + \sqrt{(1-\alpha^2)}x_{x_{n-1}} \\
y_n &= \alpha y_{n-1} + (1-\alpha)\bar{y} + \sqrt{(1-\alpha^2)}y_{x_{n-1}} \\
z_n &= \alpha z_{n-1} + (1-\alpha)\bar{z} + \sqrt{(1-\alpha^2)}z_{x_{n-1}}
\end{aligned} \tag{4}
$$

The advantage of this approach is that at each time step, the new velocity vector values can be applied directly to the constant velocity helper class in ns-3 that will calculate the position of the mobile nodes automatically. This eliminates the need for direction variables entirely, along with the trigonometric calculations that would be required to determine the node velocity vector.

The problem with this method is that it is uncharacteristic of aircraft in flight; it is not easy to model airplane flight based upon the plane's velocity in the x-direction, its velocity in the y-direction, and its velocity in the z-direction. Aircraft flight can be more accurately modeled using a velocity variable combined with variables for both direction and pitch. In the second approach, we start with the speed and direction variables found in the 2-dimensional Gauss-Markov model, and add a third variable to track the vertical pitch of the mobile node with respect to the horizon as follows:

$$\begin{aligned} s_n &= \alpha s_{n-1} + (1-\alpha)\bar{s} + \sqrt{(1-\alpha^2)}s_{x_{n-1}} \\ d_n &= \alpha d_{n-1} + (1-\alpha)\bar{d} + \sqrt{(1-\alpha^2)}d_{x_{n-1}} \\ p_n &= \alpha p_{n-1} + (1-\alpha)\bar{p} + \sqrt{(1-\alpha^2)}p_{x_{n-1}} \end{aligned} \tag{5}$$

Note that these formulæ represent basic 3-dimensional node movement. The objective of this model is to accurately represent the 3-dimensional node movement while limiting the complexity. It is not necessary to model the various flight controls, like the rudder, flaps, ailerons, angle of bank, etc. It is sufficient to model the aircraft movement itself using the Gauss-Markov algorithm, for which we assume the direction and pitch variables represent the actual angles at which the aircraft is moving.

After calculating these variables, the algorithm must determine a new velocity vector and send that information to the ns-3 constant velocity helper, in which the new node location is calculated. Assuming the direction and pitch variables are given in radians, the velocity vector $\bar{v}$ is calculated as:

$$\begin{aligned} v_x &= s_n \cos(d_n)\cos(p_n) \\ v_y &= s_n \sin(d_n)\cos(p_n) \\ v_z &= s_n \sin(p_n) \end{aligned} \tag{6}$$

V. SIMULATIONS

We implemented the proposed 3D Gauss-Markov model in ns-3 simulator. Table 1 summarizes the various parameters of our implementation, the attributes they represent, and the default values. In order to examine the results of the mobility model in 3D, a Windows application was developed that parses the ns-3 trace file data and displays the mobile node paths on the computer screen. This viewer allows one to zoom in and out and rotate the traces in three dimensions about the origin. As various parameters of the Gauss-Markov mobility model are adjusted, the changes in node mobility characteristics can be displayed and analyzed.

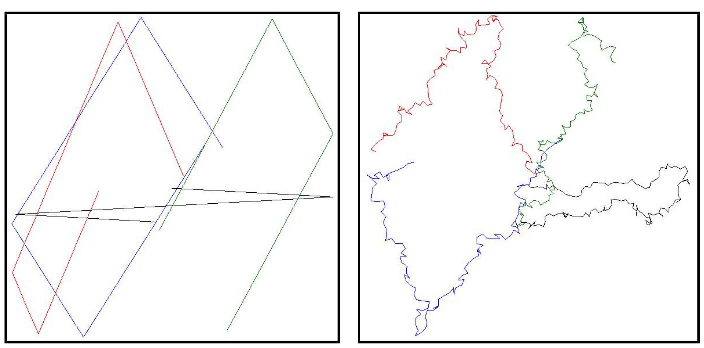
We conducted ns-3 simulations to evaluate various characteristics of the mobility model. Simulations included 4 mobile nodes with a simulation boundary of 300 km^2. The simulations were run for a duration

Table 1: Parameter description for 3D Gauss-Markov implementation in ns-3

Simulation parameter	Attribute description	Default value
`Bounds`	3D boundary of the cruising area	X: $[-100, 100]$, Y: $[-100, 100]$, Z: $[0, 100]$
`TimeStep`	parameter recalculation interval	1 second
`Alpha`	tunable parameter α	1.0
`MeanVelocity`	average velocity of nodes	Uniform random variable [0 1]
`MeanDirection`	average direction of nodes (radians)	Uniform random variable [0 2π]
`MeanPitch`	average node pitch above horizon	Constant Random Variable = 0
`NormalVelocity`	individual node velocities	Normal distribution mean = 0, std. dev. =1.0
`NormalDirection`	individual node directions	Normal distribution mean = 0, std. dev. =1.0
`NormalPitch`	individual node pitch	Normal distribution mean = 0, std. dev. =1.0

of 200 seconds and the `TimeStep` was set to 5.0 seconds. All plots were generated using the Windows-based Trace Viewer application.

C. *Verifying Special Cases ($\alpha = 0, 1$)*



(a) $\alpha = 1$ (b) $\alpha = 0$

Figure 2: Gauss-Markov Model with $\alpha = 1$ and $\alpha = 0$

The node movements observed for cases of $\alpha = 0$ and 1 are shown in Figure 2. As expected from the Gauss-Markov calculation, when $\alpha = 1$ the nodes move in a straight line (unless they run into the simulation boundary). On the other hand, when $\alpha = 0$ the nodes movements are based completely on the average velocity and direction offset by the Gaussian random variables. Since there is no memory component to the calculation, this results in more abrupt movements and sharper angles as shown in

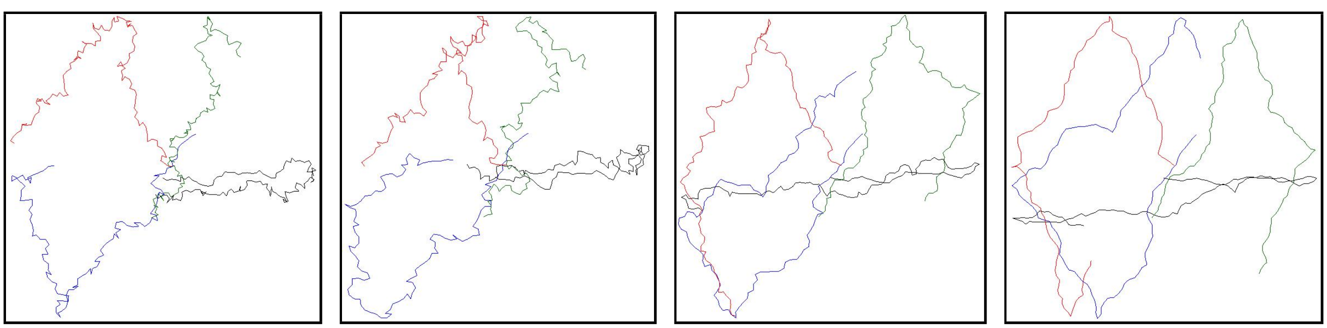
Figure 2b. The larger the standard deviation (with respect to average values), the more random is the node movement. Notice that when $\alpha = 0$, the movements are not completely random; the nodes move in a general direction over time. This is because Equation 2 includes the average speed and direction as well as the Gaussian random variable. Removing the average from the calculation would allow the nodes to wander more freely.

D. Variations in α

Setting α between zero and one allows us to tune the Gauss-Markov model with degrees of memory and variation. In order to analyze the impact of α on the mobility, we conducted simulations using the parameters given in Table 2. Figure 3 shows variation in node movement with varying values of α. We observe that as α increases, the node paths become less random and more predictable.

Table 2: Simulation setup while studying the impact of α

Simulation parameter	Value
`MeanVelocity`	[800 1200] m/s
`MeanDirection`	[0 2π] radians
`MeanPitch`	[-0.05 0.05] radians
`NormalVelocity`	$N(0, 80)$
`NormalDirection`	$N(0, 1.4)$
`NormalPitch`	$N(0, 0.2)$



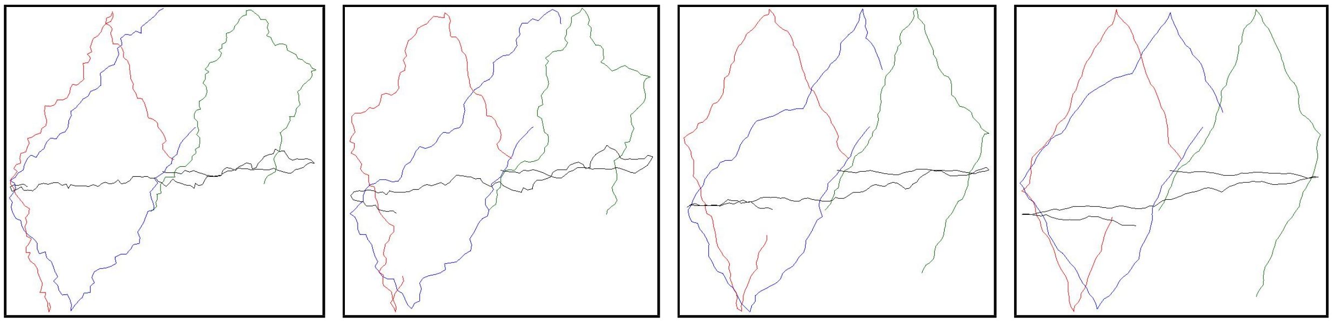
(a) $\alpha = 0.25$ (b) $\alpha = 0.5$ (c) $\alpha = 0.85$ (d) $\alpha = 0.95$

Figure 3: Gauss-Markov model with $\alpha = 0.25$, 0.5, 0.85, and 0.95

E. Change in Standard Deviation

In addition to α, a significant parameter affecting node movement is the standard deviation of the Gaussian random variables. By only changing the standard deviation of the normal direction variable from

its previous value of 1.4 radians to only 0.4 radians (about 23 degrees), the node movements become much less random as shown in Figure 4. As expected, setting the standard deviation of the Normal (Gaussian) distribution much smaller than the mean value provides relatively straight paths. Setting the standard deviation on par with the mean value results in a more random path. Setting the standard deviation much larger than the mean results in a much more random path.

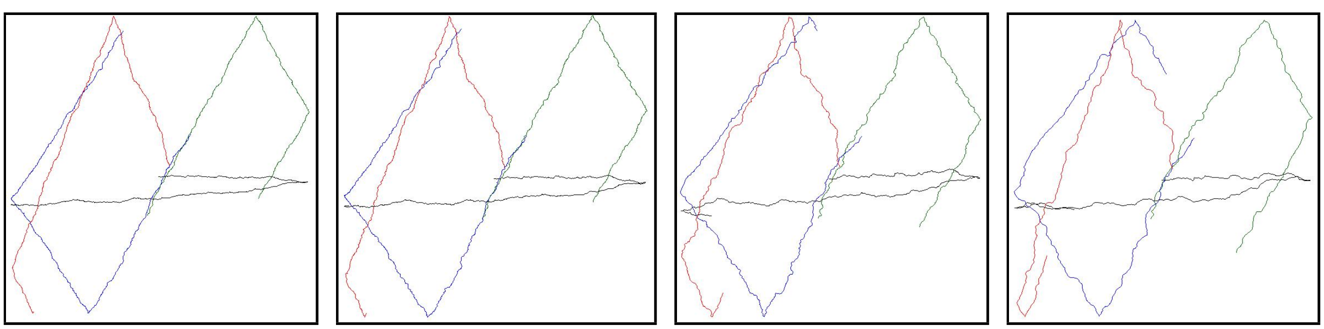


(a) $\alpha = 0.25$ (b) $\alpha = 0.5$ (c) $\alpha = 0.85$ (d) $\alpha = 0.95$

Figure 4: Gauss-Markov model with low standard deviation of 0.4 radians

F. Change in Timestep

Another significant parameter affecting node movement is the `TimeStep`, or how often a new set of values are calculated for each node. Since the nodes velocity and direction are fixed until the next timestep, setting a large timestep will result in long periods of straight movement. A short timestep (say 0.25 seconds), will result in a path that is almost continuously changing as shown in Figure 5. Nevertheless, the zero mean on the Gaussian random variables should result in a relatively straight path, even when α is small.



(a) $\alpha = 0.25$ (b) $\alpha = 0.5$ (c) $\alpha = 0.85$ (d) $\alpha = 0.95$

Figure 5: Gauss-Markov model with small time step of 0.25 seconds

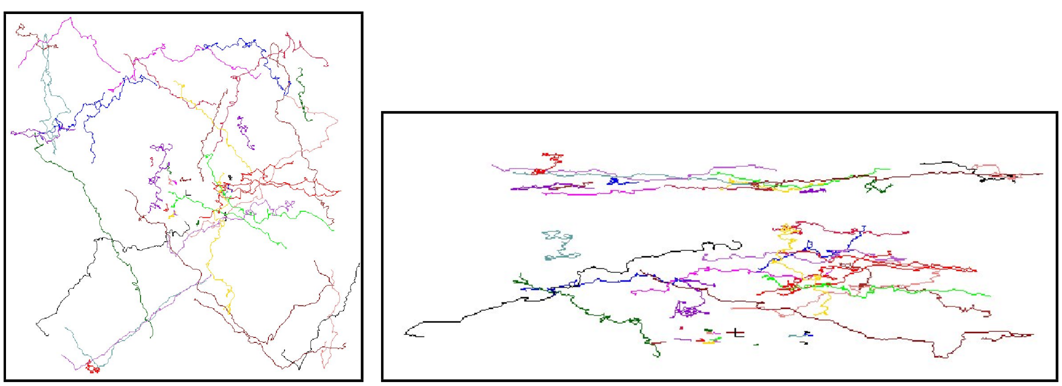
When the time step is relatively large (e.g. 5 seconds), we observed in that the node movements become more linear when increasing α from 0 to 1 (Figures 3 and 4). However, this is not the case when the time step parameter is small. Figure 5 shows that when using `TimeStep` = 0.25 seconds, the trace is

fairly linear at $\alpha = 0.25$, but it becomes more random as we approach $\alpha = 0.95$. This behavior is exactly opposite when compared to the results with time step of 5 seconds. This is because of the increasing memory effect with increasing values of α. The node's new direction is based more upon it's previous direction than when α is small, therefore, small changes from the Gaussian random variable have the opportunity to accumulate and alter the look of the path.

G. Simulating Multi-Tier Networks

Realistic airborne MANETs employ the services of different groups of nodes – each group performing a particular function [5]. The mobility of these groups may vary significantly from one another. Hence a generic mobility model cannot accurately represent the mobility of all groups. While one group may experience little or no mobility (e.g. ground nodes), the other group may be moving at Mach speeds (e.g. airborne nodes). Hence, both our model and implementation provide for simulating *multi-tier* networks, in which different groups of nodes can have different mobility parameters. In this section, we present the results of a three-tier network that was simulated in ns-3. The three tiers were modeled using three different instances of the proposed Gauss-Markov model. First, the ground nodes were modeled using low speed and velocity variance, and moderate variance for the direction and pitch. Secondly, the medium altitude nodes were modeled with high velocity and variance, and finally, the high altitude nodes were modeled with moderate velocity and variance.

Figure 6 depicts the top down view of the single multi-tier trace file. The individual tiers can be visualized by rotating the viewing angle to a nearly horizontal direction as shown in Figure 6b. In this figure, we are able to observe the node movements grouped by their altitudes. The high altitude nodes are positioned at the top of the image, the medium altitude nodes are positioned in the middle of the image, and the ground nodes are the short path lines clustered closely together in the bottom center of the image.



(a) Top Down View

(b) Horizontal View

Figure 6: Multi-Tier Nodes – Top Down and Horizontal Views

VI. CONCLUSIONS

The three dimensional Gauss-Markov mobility model presented in this paper can be used to model multi-tier airborne MANETs. Compared to the previously available ns-3 mobility models, the Gauss-Markov model provides more realistic node movement. Different types of aircraft movements can be simulated by tweaking the three basic tuning parameters of α, standard deviation, and time step. The Gauss-Markov model integrates well with the existing ns-3 mobility models, and also provides 3D support for the constant velocity helper and the position allocation classes used by the existing mobility models. Furthermore, it can be used in simulating multi-tier network environments, where different groups of nodes serve different functions and have different communication and mobility characteristics.

ACKNOWLEDGMENTS

The authors would like to thank the Test Resource Management Center (TRMC) Test and Evaluation/Science and Technology (T&E/S&T) Program for their support. This work was funded in part by the T&E/S&T Program through the Army PEO STRI Contracting Office, contract number W900KK-09-C-0019 for AeroNP and AeroTP: Aeronautical Network and Transport Protocols for iNET (ANTP). This work was also funded in part by the International Foundation for Telemetering (IFT). We would like to acknowledge the ResiliNets group member Kevin Peters for useful discussions on this work.

REFERENCES

[1] J. P. Rohrer and J. P. Sterbenz, "Performance and disruption tolerance of transport protocols for airborne telemetry networks," in *International Telemetering Conference (ITC)*, (Las Vegas, NV), October 2009.

[2] J. Prokkola, L. Leppännen, and T. Bräysy, "On the effect of traffic models to the performance of ad hoc network," in *IEEE MILCOM*, pp. 422–427, October 2003.

[3] E. P. Justin P. Rohrer, Abdul Jabbar and J. P. Sterbenz, "Cross-layer architectural framework for highly-mobile multihop airborne telemetry networks," in *IEEE MILCOM*, pp. 1–9, November 2008.

[4] "The ns-3 network simulator." http://www.nsnam.org, July 2009.

[5] B. Ryu, T. Anderson, T. Elbatt, and Y. Zhang, "Multi-tier mobile ad hoc networks: Architecture, protocols, and performance," in *IEEE MILCOM*, pp. 1280–1285, October 2003.

[6] T. Camp, J. Boleng, and V. Davies, "A survey of models for ad hoc network research," *Wireless Communication and Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends, and Applications*, vol. 2, no. 5, pp. 483–502, 2002.