Design Improvement and Implementation of
3D Gauss-Markov Mobility Model

Mohammed J.F. Alenazi, Cenk Sahin, and James P.G. Sterbenz
Department of Electrical Engineering & Computer Science
Information & Telecommunication Technology Center
The University of Kansas
Lawrence, KS 66045
{malenazi,cenksahin,jpgs}@ittc.ku.edu

ABSTRACT

The current ns-3 implementation of the 3D Gauss-Markov mobility model (3D-GMMM) allows mobile nodes to reach and bounce off the simulation boundaries. This causes sudden and unnatural movement of the nodes in the vicinity of the simulation boundaries. In this paper, we present a modification to the current ns-3 implementation of the 3D-GMMM. We follow an approach in which mobile nodes are directed toward the center of the simulation region at a random angle if they are within a certain distance from the simulation boundaries. As the simulation results show, the improved ns-3 implementation of 3D-GMMM prevents mobile nodes from reaching the simulation boundaries while resulting in smooth movement.

I. INTRODUCTION AND MOTIVATION

Mobility models are used to emulate realistic motion of mobile wireless nodes during network simulations. Their accuracy is an important factor in determining the validity and the reliability of simulation results. The Gauss-Markov mobility model (GMMM) is a widely used mobility model known for having memory that results in smooth movement of nodes [1, 2]. In GMMM, the speed and the direction at a given time step depends probabilistically on the speed and the direction in the previous time step [3]. In addition, mobile nodes are forced away from the simulation boundaries to prevent unnatural movement such as bouncing off the simulation boundaries. This is done with the help of a *buffer zone* defined as the region between the simulation boundaries and the *inner zone* [1].

The current ns-3 [4] implementation of GMMM extends the model to three dimensions (3D) [5]. However, this implementation 3D GMMM (3D-GMMM) lacks a buffer zone, which allows mobile nodes to reach and bounce off the simulation boundaries. We add a buffer zone to the current ns-3 implementation to eliminate this problem and to make the implementation consistent with the original description of GMMM in which two new parameters are used by the user to specify the buffer zone [1]. However, we follow a different approach to force mobile nodes away from the simulation boundaries. In the original description of GMMM the buffer zone is divided into *sectors*, and in order to force a node away from the simulation boundaries the sector in which the node resides needs to be determined [1]. In our novel approach the buffer zone is not sectorized that results in reduction in the implementation complexity. We provide node position traces of simulations to compare the current and our improved ns-3 implementation of the 3D-GMMM.

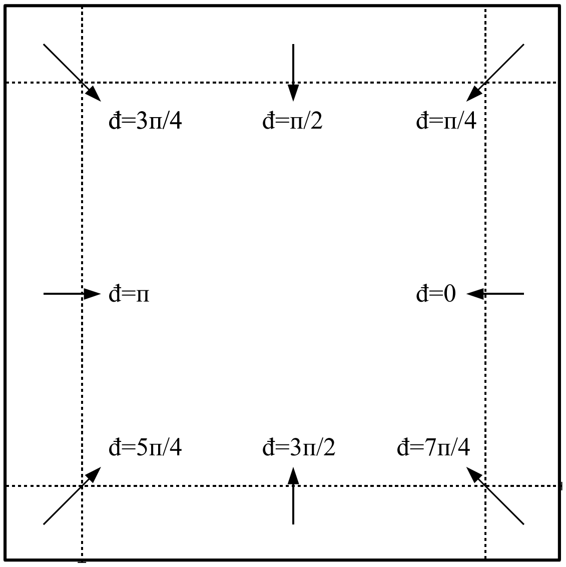


Figure 1: The sectorized buffer zone of Gauss-Markov mobility model (GMMM) [3].

The remainder of this paper is organized as follows. In Section II the descriptions of GMMM and the current ns-3 implementation of 3D-GMMM are given. In Section III the new 3D-GMMM design and implementation are presented. In Section IV simulation results are presented. The conclusions and the future work are given in Section V.

II. BACKGROUND

In this section we describe GMMM, we also present the current ns-3 implementation of 3D-GMMM.

A. Gauss-Markov Mobility Model

Having memory in GMMM reduces the possibility of abrupt movements of mobile nodes as observed with other mobility models such as random waypoint, random walk, and random direction mobility models [3, 4]. Another advantage of the GMMM is having only one tuning parameter. Initially each mobile node is assigned a position, a speed, and an angle of direction. Both the speed and the direction are updated at every time step by the following equations

$$s_n = \alpha s_{n-1} + (1-\alpha)\bar{s} + \sqrt{(1-\alpha^2) s_{x_{n-1}}} \tag{1}$$

$$d_n = \alpha d_{n-1} + (1-\alpha)\bar{d} + \sqrt{(1-\alpha^2) d_{x_{n-1}}} \tag{2}$$

where s_n is the speed and d_n is the direction at time step n. The memory of the model is controlled by the parameter α which varies between 0 and 1; $\bar{s}$ and $\bar{d}$ are the mean values of the speed and the direction while $s_{x_{n-1}}$ and $d_{x_{n-1}}$ are Gaussian random variables with zero mean. It is important to note that this is a 2-dimensional mobility model in which the positions of mobile nodes are updated by the equations

$$x_n = x_{n-1} + s_{n-1}\cos(d_{n-1}) \tag{3}$$

$$y_n = y_{n-1} + s_{n-1}\sin(d_{n-1}) \tag{4}$$

where $(x_n,\, y_n)$ is the position on the xy-plane. When a mobile node is in the vicinity of an edge it is pushed away from the edge by modifying the mean direction $\bar{d}$ according to the buffer zone sectors shown in Figure 1 (Figure 9 in [3]). There are 8 distinct sectors in the buffer zone. When a node enters the buffer zone, the sector in which the node resides must be determined in order to push the node away from the boundaries. It is very important to note that the variance of the random variables $s_{x_{n-1}}$ and $d_{x_{n-1}}$ need to be chosen very carefully to adjust the randomness against the memory in the model. For the same α value, different variance values result in different types of node movements [5]. When the variance of $d_{x_{n-1}}$ is small, the mean direction $\bar{d}$ dominates node movement.

B. Implementation of 3D Gauss-Markov Mobility Model in ns-3

In the current ns-3 implementation of 3D-GMMM nodes move inside a 3-dimensional *simulation box* [5]. The motion in the z direction is controlled by the pitch angle p_n given by

$$p_n = \alpha p_{n-1} + (1-\alpha)\bar{p} + \sqrt{(1-\alpha^2)p_{x_{n-1}}} \tag{5}$$

where $p_n \in [-\pi/2, +\pi/2]$. The current ns-3 GMMM model does not use a buffer zone. When the next position for a node is outside the simulation box, the node is forced to bounce back from the edge of its incoming angle. This is achieved by setting both the current direction d_n and the mean direction $\bar{d}$, and both the current pitch p_n and the mean pitch $\bar{p}$ to fixed values computed from the incoming angle.

III. DESIGN AND IMPLEMENTATION

In this section we discuss the design of the improved ns-3 implementation of 3D-GMMM and the ns-3 implementation details.

C. Design of Improved 3D Gauss-Markov Mobility Model

As we mentioned previously, GMMM as originally described in uses a buffer zone [3]. If a node enters the buffer zone it is forced away from the simulation boundary by modifying the mean direction $\bar{d}$ according to which sector the node is in. The current ns-3 implementation lacks such region; nodes are pushed away from the edge only when they try to get out of the simulation box at the angle of incidence. Both the mean direction $\bar{d}$ and the direction d_n are set to this fixed value.

There are two modifications to improve the current ns-3 implementation. The first one is to include a buffer zone whose parameters can be specified by the user. Once the buffer zone is introduced we need to decide how nodes are pushed back to the inner zone. As shown in Figure 1, the original GMMM divides the buffer zone into 8 sectors. Associated with each sector is a mean direction. Sectorizing the buffer zone in 3D-GMMM would result in 26 sectors because of the third dimension. Consequently, when a node is in the buffer zone we have to determine in which one of the 26 sectors the node resides. We propose an alternative approach that eliminates the need to determine the sector in which the node resides and results in a lower complexity ns-3 implementation. In our approach, nodes in the buffer zone are pushed toward the center of the simulation box. The new mean direction and the mean pitch are computed from the position, (x_n, y_n, z_n), of the node using inverse trigonometric functions. A much simpler implementation would be to add π to the mean direction and negate the mean pitch. The simpler approach works well when the variances of the direction and the pitch random variables are small; consequently, the movement is dominated by the mean direction $\bar{d}$ and the mean pitch $\bar{p}$.

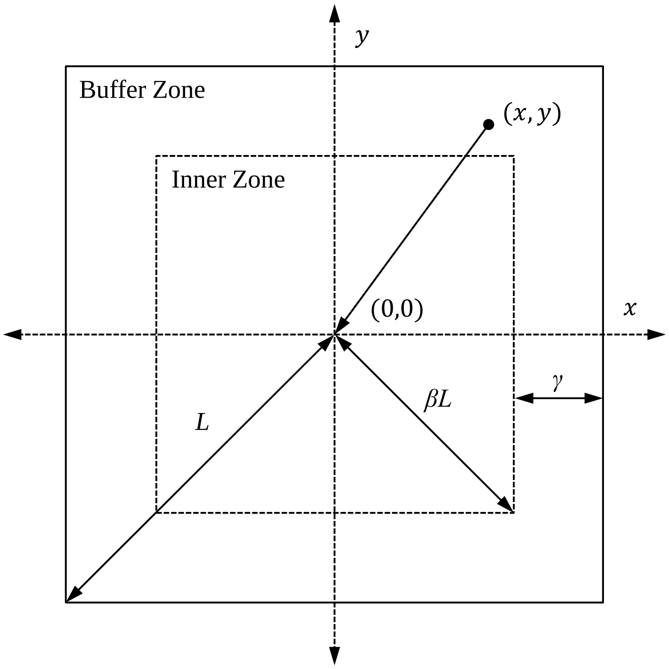


Figure 2: The unsectorized buffer zone of Gauss-Markov mobility model (GMMM).

We introduce some randomness to the way the nodes are pushed into the inner zone, with the goal of more natural movement. We also investigate the effects of forcing away from the edges by altering only the mean direction $\bar{d}$ and the mean pitch $\bar{p}$, only the direction d_n and the pitch p_n or both pairs. It is easy to show that the mean direction and the mean pitch introduce a bias to the movement that lasts much longer. In other words, the mean values are much stronger measures to control the movement of the nodes.

D. Implementation of the Improved 3D Gauss-Markov Mobility Model in ns-3

We designed the new implementation to give the user three options to specify the buffer zone. First, the user can specify the buffer zone by providing the bound points. The other two options are based on two parameters β and γ, which allows the user to specify the buffer zone as shown in Figure 2. The parameter γ is *the buffer thickness* that is the distance from the inner zone boundaries to the the simulation boundaries, with a nominal value in meters. The parameter β is *the buffer ratio*, which is the proportion of the inner simulation zone to the simulation boundary given that the centers of both boxes overlap, with a value between 0 and 1. If the user desires to specify a buffer thickness, β is set to 1 and the thickness of the buffer zone is specified by γ. If the user desires the inner zone to have the same shape as the simulation boundary, γ is set to 0 and β takes a value smaller than 1. The default values of γ and β are set to 0 and 1 respectively ,which means that the inner zone is the same as the simulation boundary by default.

When a node's next position is in the buffer zone it is directed to the center of the simulation box as shown in Figure 2. Note that Figure 2 is two dimensional, but it is implied that the nodes are forced to the center of the 3D simulation box. In this case we set the mean direction and the mean pitch values to the values that direct the node to the center of the simulation box as shown in the equations below where (x, y, z) is the current position of the node.

$$\bar{d} = \tan^{-1}(-y, -x) \tag{6}$$

$$\bar{p} = \tan^{-1}(\sqrt{x^2 + y^2}, -z) \tag{7}$$

If a node's next position is outside the simulation box in addition to setting the mean pitch and the mean direction values to the values shown above, we set the direction and pitch values to their corresponding mean values:

$$\bar{d} = \tan^{-1}(-y, -x) \tag{8}$$

$$d_n = \bar{d} \tag{9}$$

$$\bar{p} = \tan^{-1}(\sqrt{x^2 + y^2}, -z) \tag{10}$$

$$p_n = \bar{p} \tag{11}$$

We also introduce randomness to the way nodes are forced to the center by adding random values from a normal distribution to the mean direction and pitch values. The new mean direction and mean pitch values are given by equations

$$\bar{d} = \tan^{-1}(-y, -x) + N(0, {\sigma_d}^2) \tag{12}$$

$$\bar{p} = \tan^{-1}(\sqrt{x^2 + y^2}, -z) + N(0, {\sigma_p}^2) \tag{13}$$

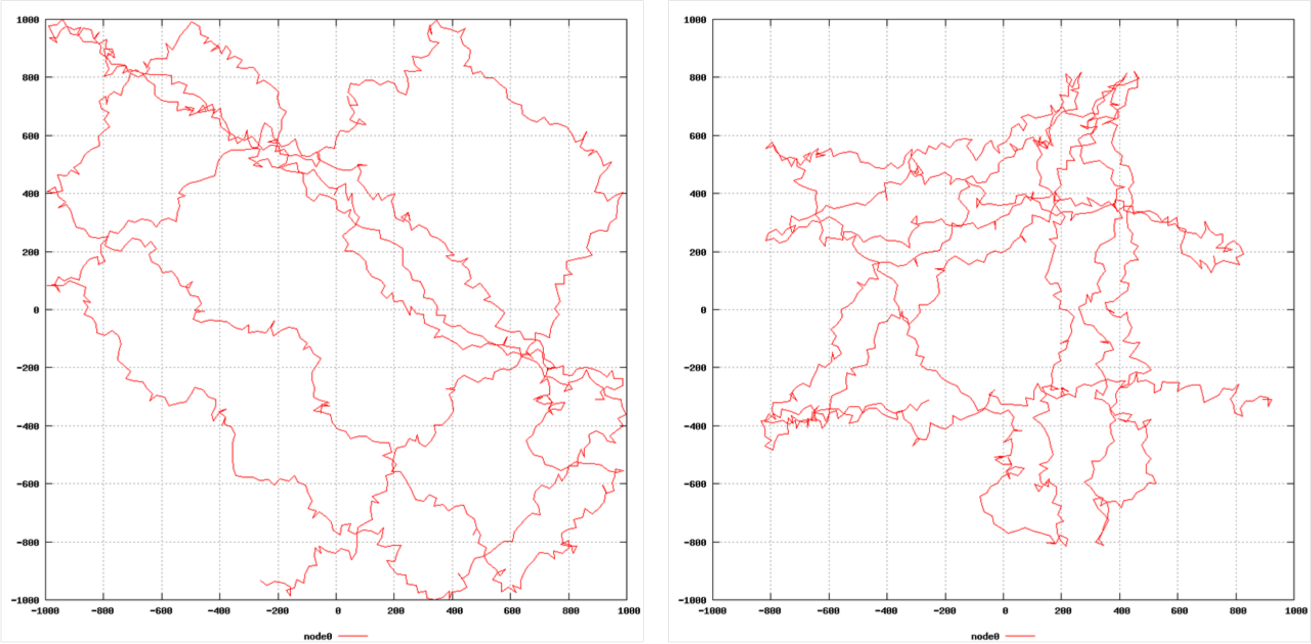
where ${\sigma_d}^2$ is a variance of a normal random direction variable with a mean of 0 and ${\sigma_p}^2$ is a variance of a normal random pitch variable with a mean of 0. With the randomness introduced above it is more likely for nodes to move along the inner zone boundaries. The variance values are calculated ${\sigma_d}^2 = 0.2304$ and ${\sigma_p}^2 = 0.0576$ to guarantee that the mean direction will be within $\pi/2$ of the direction value pointing to the center and the mean pitch value will be within $\pi/4$ of the pitch value pointing to the center 99.9 % of the time.

IV. ANALYSIS AND RESULTS

In this section we compare the current and the improved ns-3 implementations of 3D-GMMM. We present how the the α value (memory) impacts the movement of a given node. In addition, we rely on node trace plots that clearly show the addition of the buffer zone. As discussed previously, a three dimensional buffer zone is implemented. In order to verify that both the buffer zone and the handling of the node movement in the buffer zone work properly we need to generate node trace plots on the xy, yz, and xz plane. However, the simulation results presented in this paper are limited to the node trace plots on the xy plane.

E. *Improved vs Current*

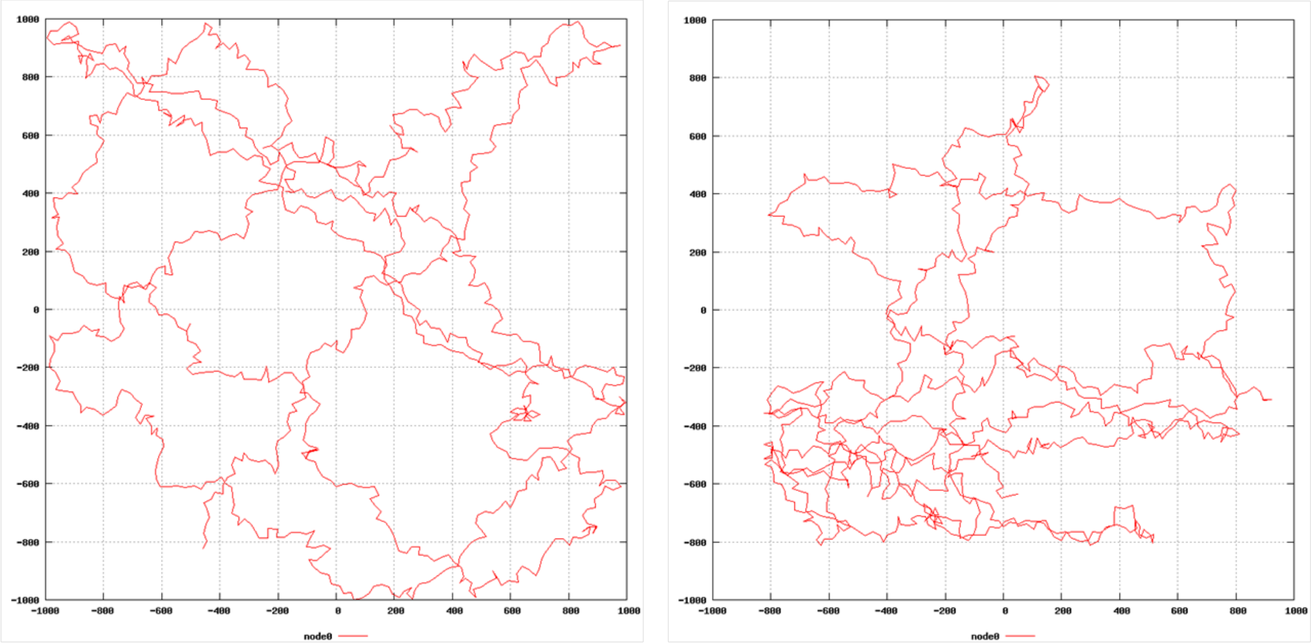
We use a buffer ratio of 0.8 in our simulations. The simulation box is a 2000×2000 square specified by the box bounds $([-1000, 1000], [-1000, 1000], [0, 0])$. The time step is equal to 1, the initial mean direction is chosen uniformly from $[-\pi, +\pi]$ and the mean speed is uniformly chosen from $[10, 30]$. The direction random variable variance is set to 1 while the speed random variable variance is set to 5. The



(a) Current implementation

(b) Proposed implementation

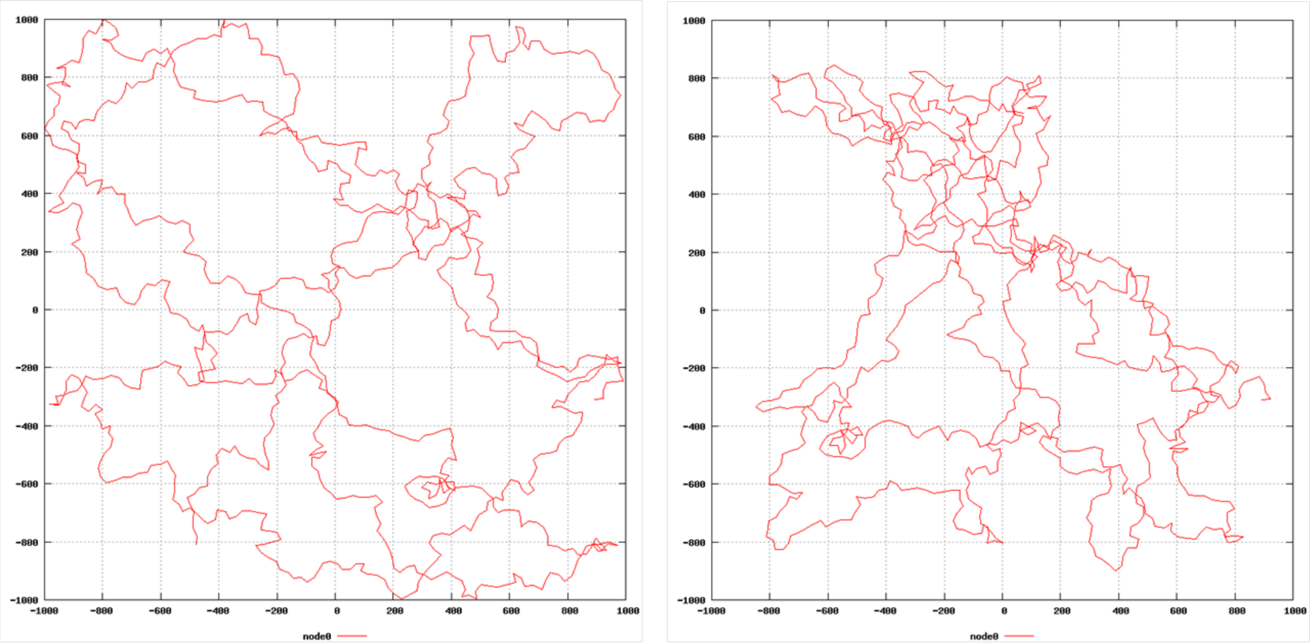
Figure 3: $\alpha = 0.0$



(a) Current implementation

(b) Proposed implementation

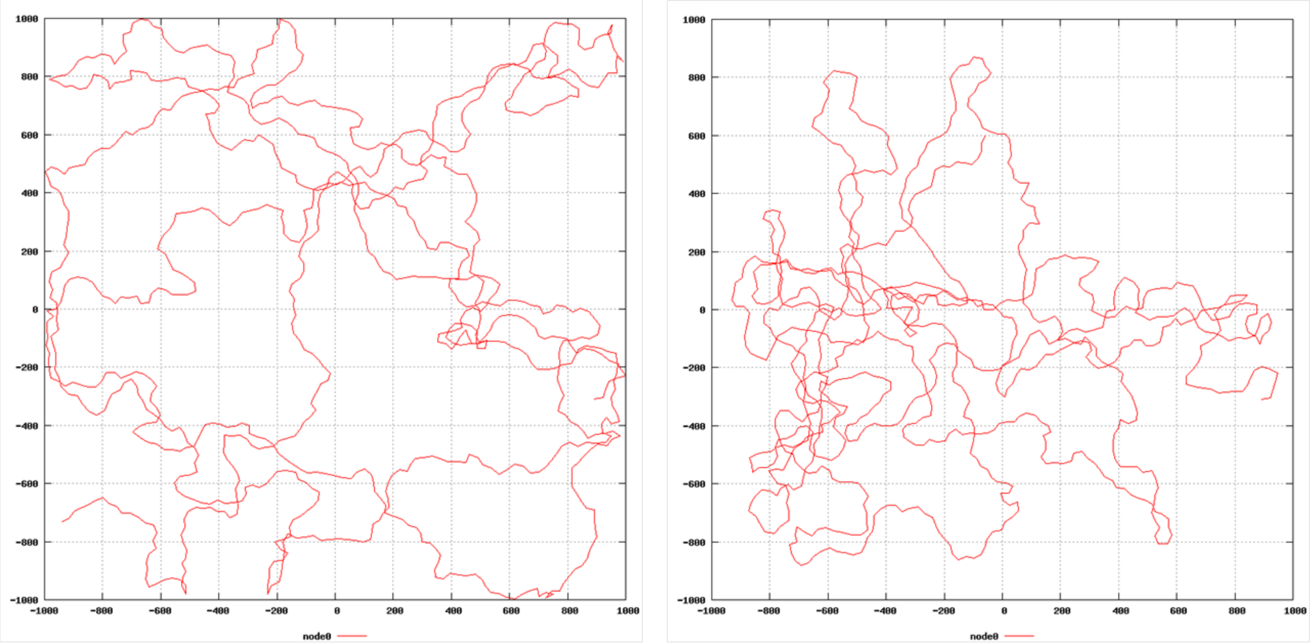
Figure 4: $\alpha = 0.25$



(a) Current implementation

(b) Proposed implementation

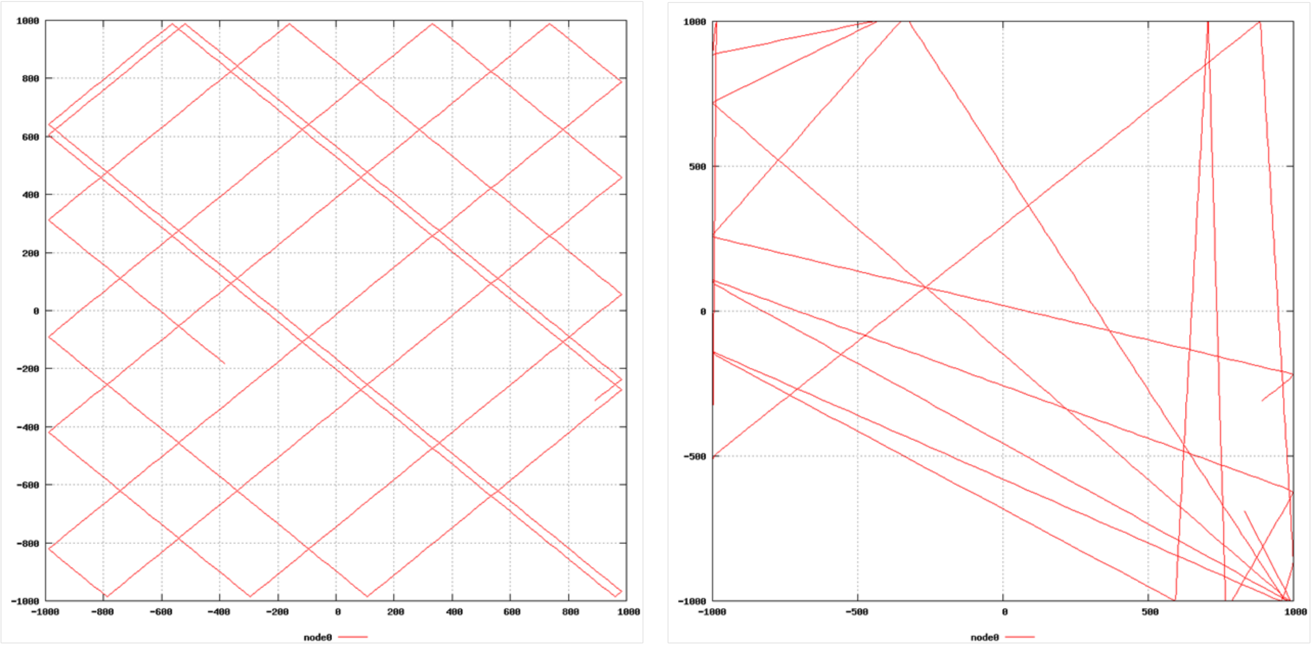
Figure 5: $\alpha = 0.5$



(a) Current implementation

(b) Proposed implementation

Figure 6: $\alpha = 0.75$



(a) Current implementation

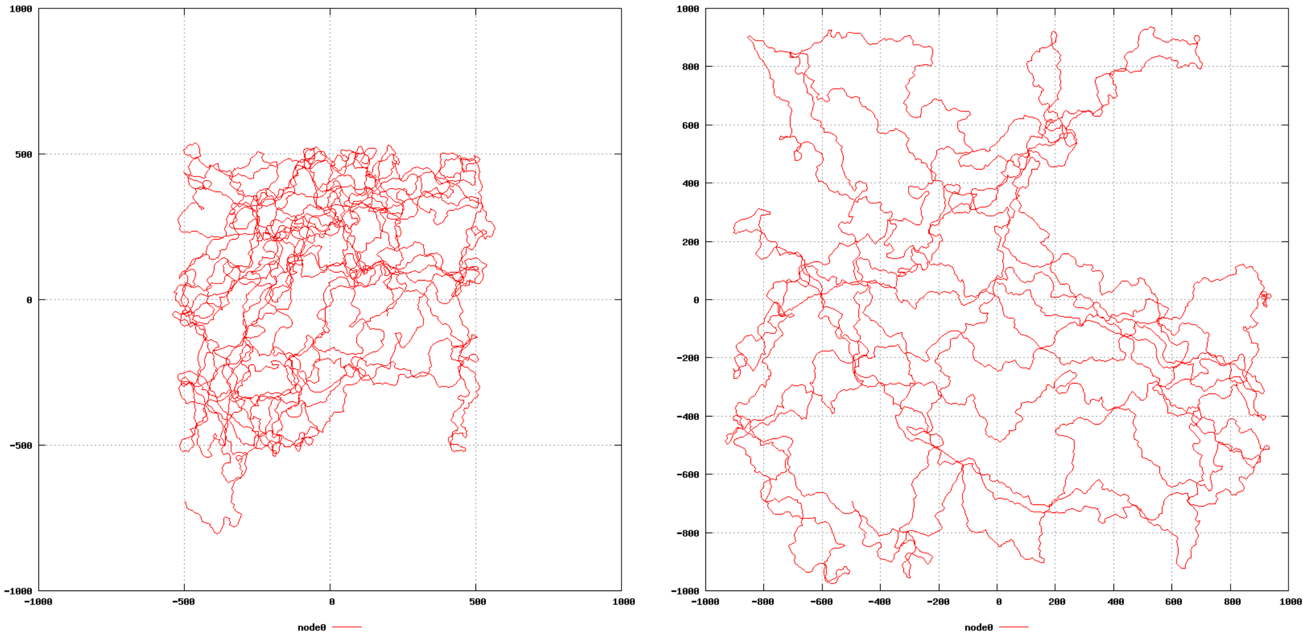
(b) Proposed implementation

Figure 7: α = 1.00

simulation time is 1000 seconds. Node trace plots are generated for both the current ns-3 implementation and the improved ns-3 implementation for tuning parameter values $\alpha = 0, 0.25, 0.5, 0.75$, and 1. Figures 3 through 7 show that nodes move smoothly around the inner zone boundaries and don't approach the simulation boundaries due to the presence of the buffer zone. It is important to note that when alpha is set to 1 the mean direction has no effect on the node movement and nodes are not forced away from the buffer zone as seen in Figure 7. Also seen in Figure 7 is the randomness in the way nodes are pushed away from the simulation boundaries. In the proposed model, nodes bounce from the simulation boundaries with a different angle every time, while on the current model, nodes bounce from the simulation boundaries with the incoming angle.

F. Impact of changing the buffer ratio

The buffer ratio is the ratio of the inner zone to the simulation box. Choosing the buffer ratio value depends on the scenario needed. Here, we select two buffer ratio values to show how it affects the node's movement. In Figure 8, the node trace plots of the proposed implementation for buffer ratio values of 0.5 and 0.9 are shown. All other simulation parameters are identical to those of the previous plots. For the buffer ratio value of 0.5, a mobile node is initially placed outside the buffer zone as seen in Figure 8 (a). As a result, the node is forced inside the inner zone since the mean direction value changes whenever it enters the buffer zone. For the buffer ratio value of 0.9, the mobile node gets very close to the edge of



(a) 0.5 buffer

(b) 0.9 buffer

Figure 8: The impact of changing the buffer ratio.

the simulation boundary. For both buffer ratio values, as seen in Figure 8 (b), the direction of the node is changed toward the center with some randomness depending on the value of alpha whenever it attempts to get outside the simulation box.

V. CONCLUSIONS AND FUTURE WORK

We improved the ns-3 implementation of 3D Gauss-Markov mobility model. The improved implementation which includes a buffer zone allows mobile nodes to move smoothly around the simulation boundaries without bounces. The simulation results show that the improved implementation works properly on a two dimensional simulation region.

ACKNOWLEDGEMENTS

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). The Executing Agent and Program Manager work out of the AFFTC. This work was also funded in part by the International Foundation for Telemetering (IFT). We would like to thank Kip Temple and the membership of the iNET working group for discussions that led to this work.

REFERENCES

[1] V. Tolety and V. Tolety, "Load Reduction in Ad Hoc Networks Using Mobile Servers," Master's thesis, Colorado School of Mines, CO, USA, 1999.

[2] J. P. Rohrer, E. K. Çetinkaya, H. Narra, D. Broyles, K. Peters, and J. P. G. Sterbenz, "AeroRP Performance in Highly-Dynamic Airborne Networks using 3D Gauss-Markov Mobility Model," in *Proceedings of the IEEE Military Communications Conference (MILCOM)*, (Baltimore, MD), pp. 834–841, November 2011.

[3] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," *Wireless Communications and Mobile Computing*, vol. 2, no. 5, pp. 483–502, 2002.

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

[5] D. Broyles, A. Jabbar, and J. P. G. Sterbenz, "Design and analysis of a 3–D gauss-markov mobility model for highly-dynamic airborne networks," in *Proceedings of the International Telemetering Conference (ITC)*, (San Diego, CA), October 2010.