GENERATION AND ANALYSIS OF REALISTIC MOBILITY MODELS FOR MOBILE AD HOC NETWORKS

by

Hassan Tahır

B Sc , Unıversıty of South Asıa, Lahore,
Pakıstan, 2005

THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE
IN
MATHEMATICAL, COMPUTER AND PHYSICAL SCIENCES

THE UNIVERSITY OF NORTHERN BRITISH COLUMBIA

August 2010

Library and Archives
Canada

Published Heritage
Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et
Archives Canada

Direction du
Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN 978-0-494-75110-7
Our file Notre référence
ISBN 978-0-494-75110-7

Dedicated to my parents

and

my loving wife

Abstract

Simulation modeling is an integral part of conducting research in communication networks and distributed systems In systems involving mobile nodes, accurate modeling of mobility has primary importance Mobility has a fundamental influence on the behavior and performance of the system However, only few mobility models have been used in nearly all simulations in the past These models are simple and highly random As a result, the simulation studies based on these random mobility models have been heavily criticized for their credibility We feel that availability of a software tool with the following capability, at least in part, would alleviate this crisis The software must facilitate researchers to (i) model a wide range of mobility with varying degrees of realism, (ii) analyze the modeled mobility visually and statistically, and (iii) transport the mobility trace in a format that can be used in most widely used simulators The development of a software tool with the above mentioned capabilities is the main contribution of this thesis

In this thesis, after presenting a comprehensive survey on realistic mobility models, we present a *realistic mobility generator software* called **RLMobiGen** that can be used to specify, generate, analyze, and then export the mobility trace The mobility trace can then be used in the simulation studies of mobile ad hoc networks RLMobiGen is a comprehensive, highly interactive, and user friendly software

Contents

List of Tables

List of Figures

Publications From This Thesis

1 A Aravınd, and H Tahır Towards Modelıng Realıstıc Mobılıty for Performance Evaluatıons ın MANET *Proc of the 9th Internatıonal Conference onAd Hoc Networks and Wıreless (ADHOC-NOW), Lecture Notes ın Computer Scıence (LNCS) , Sprınger-Verlag*, 109-122, 2010

2 A Aravınd, H Tahır, and Baldeep RealMobıSım realıstıc mobılıty sımulator and analyzer *Proc of the 3rd ACM Workshop on Performance Monıtorıng and Measurement of Heterogeneous Wıreless and Wıred Networks*, 186-189, 2008

Acknowledgments

Throughout my time as a graduate student at UNBC, I came across many people who have encouraged me in one way or another, accompanied me through the tough times and helped me I can not possibly name them all here, but I would like to single out special thanks to a few

First of all, I would like to express my sincere gratitude to my supervisor, Dr Alex Aravind for his continued guidance, constant encouragement and support Without his unconditional help and erudite supervision this research would not have been possible I am also very thankful to Dr Waqar Haque and Dr Balbinder Deo for serving on my graduate committee and endowing me with their encouraging words, advice and valuable suggestions My thanks are also due to the external examiner for his/her time to read this thesis Here, I would like to take this opportunity to thank Dr Ian Hartley, Dean of Graduate Studies at UNBC for generously supporting my research through a Graduate Research Assistantship and a travel grant

A special word of gratitude to all my colleagues and friends that I have worked and spent memorable time with over the past few years in Canada I must thank my fellow graduate students at UNBC, Mr Bharath Govinda Reddy, Ms Baldeep and Mr Viswanathan Manickam for engaging me in useful research discussion They each helped make my time more fun and interesting I would also like to thank Mrs Alex Aravind for her dinner invitations on various occasions

I will always be thankful to my parents who have influenced my life in so many ways I must acknowledge my wife, Sidra, without her love, encouragement and patiencc, I would not havc finished this thesis I also thank God for bestowing upon me such a loving and supportive family I doubt that I will ever be able to convey my appreciation fully, but I owe everyone my eternal gratitude

Hassan Tahir

Chapter 1

Introduction

1.1 Background

Networks are infrastructures created and used throughout human civilization primarily for sharing resources The resources could be physical (such as food, water, electricity, etc) or logical (such as message, data, voice, services, etc) Among these networks, computing and communication networks have revolutionized the world in the last several decades and also played a fundamental role in creating other types of networks This thesis is concerned with a special type of networks called mobile ad hoc networks (MANETs)

Telecommunication networks (telephone and telegraph networks), were created for people to talk to each other or send and receive messages In telephone networks, telephones were initially connected directly together in pairs, which later connected through telephone exchanges These exchanges are connected to each other, forming a network called public switched telephone network (PSTN) Telephone networks connect telephones to exchange voice and computer networks connect computers to exchange digital data They are similar in the sense that they both exchange information but in different form

The evolution of computer networks has been driven essentially by two goals, sharing local resources and connecting to remote computers These two goals led to two types of computer networks to emerge and they are called local area network (LAN) and wide area network (WAN) Initially, in the early 1960s, LAN started with connecting terminals to mainframes to share the computing power (a precious resource at that time) This network of terminals, typically placed in a room, and connected to the mainframe, allowed multiple users to share the system in parallel As processors became cheaper, desktop computers emerged and replaced the terminals This allowed the users to have their own computers in their offices connected to other systems within an organization Typically, LAN connects two or more computers over a relatively short distance normally within a house or an office building

On the WAN side, in 1969, the first network connected two computers from distant locations (University of California, Loss Angeles and Stanford Research Institute) Soon, two more nodes, one from the University of Utah and the other from the University of California, Santa Barbara, were added forming the first 4-node WAN called Advanced Research Project Agency Network (ARPANET) Until late 1980s, the distinction between LAN and WAN, (e g , in data transfer rates and methods, in length and quality of communication links, and in services) was evident and then started to fade slowly Now, in most cases, WANs are typically used to connect LANs Today, Internet, the most widely known network, the largest WAN spanning the entire world, is a collection of LANs and small WANs After the establishment of LANs and WANs, again based on distance, more classifications of computer networks emerged For example, the networks that are larger than LAN and smaller than WAN are called metropolitan area network (MAN), and the networks that are smaller than LAN are called personal area network (PAN) A PAN connects the computer and different devices close to one person All these networks have three main components nodes, links, and protocols Nodes and links form the topology and protocols and algorithms dictate network behaviour Nodes could be computers, telephones, switches, and other devices used to produce, consume, and/or forward resources (message, data, voice, etc) Links are communication medium to transport resources, and the protocols are to regulate the formation and operation of the network

Irrespective of the network, communication between two directly connected nodes is simple and straightforward This is not the case when several intermediate nodes are involved in establishing communication between two end user nodes Setting up a "path" between two communicating nodes across the network and maintaining it throughout the communication period involves a number of tasks, particularly when several asynchronous communication are initiated Earlier telephone networks used a technique called circuit switching, in which the communication path is decided upon before the data transmission starts Then, for the entire session, the path stays as dedicated and exclusive Computer networks took a different innovative technique called packet switching In packet-switching, messages are broken into small pieces called packets, and the packets are sent towards the destination irrespective of each other This technique works very similar to how postal systems work There is no predetermined path and each packet has to find its own path to reach the destination Basically, each intermediate node decides, based on several factors including the destination address, congestion at that time, priority of the packet, etc , where to forward the packet next The process is repeated until the packet reaches the destination

For quite some time, these networks used physical wires as the communication medium The introduction of wireless communication created a whole new dimension for these networks to evolve further Apart from eliminating the complex mess of physical wires, wireless networks allow the nodes to move freely As a result, wireless LANs and PANs have become very common and a new type of telephone network called cellular network started to emerge A cellular network has a number of cells (a transmission region), and each cell is served by a fixed wireless node called base station When joined together these base stations provide the coverage over a wider geographic area With this setup, people or vehicles with mobile devices such as mobile phones, pagers, etc , could communicate with each other without losing communication when they move from one location to another The network of base stations takes care of the "hand-off" communication between adjacent cells during the mobility of the end user node In this setup, devices such as computers, routers, and switches play key roles in implementing the communication protocols and network management More importantly, computer and telephone networks started to adopt and use devices and technologies from each other and the distinction between these two networks is more blurred than before Most modern networks are hybrid in nature than either of the pure networks described above

The networks discussed so far typically have a centralized control and are supported by fixed infrastructure also known as the *backbone* of the system These infrastructures involve several components such as wires, bridges, routers, base stations, etc Setting up a fixed infrastructure involves several resources such as time, money, permit, etc However, such infrastructure is not viable or even feasible in several situations involving applications such as wildlife tracking, military exercises in battlefield to maneuver war fighters, environment monitoring, forest fire monitoring, flood monitoring, traffic monitoring, mine site operations, mobile patient monitoring, remote health, conferences and business meetings, etc This application pull and technology push discussed above paved the way for a new type of networks called *Mobile Ad Hoc Network* (MANET), which enables people setup and connect their computing and communication devices to networks whenever they need and wherever they go In a nutshell, MANET is a wireless communication technology designed to build a network with the following characteristics

- *Spontaneous and Ad hoc* - the network must be setup in a shorter period of time, typically within hours and days instead of months and years, typically for a specific application

- *Self Organized* - the nodes in the network must reorganize themselves with no or least assistance from human to establish the network Each node can perform the roles of both hosts and routers

- *No Fixed Infrastructure* - the network should not depend on a fixed infrastructure, because such an infrastructure may not be feasible or possible

- *No Centralized Control* - the network must operate by the cooperation of all or most of the nodes in the network The responsibility of the network is typically shared among the nodes in the system due to several constraints such as energy, security, stability, availability, etc

1.1.1 Role of Mobility in MANET

In MANETs, mobility plays a fundamental role Mobility influences the structure and behavior of the network and facilitates services across the network It impacts connectivity, coverage, routing and communication, and capacity of the network When mobility becomes inseparable from MANETs, selection of suitable mobility pattern and modeling it accurately are not only helpful but also essential for the understanding and interpretation of the performance of the protocols and services under study A good mobility model should capture an acceptable level of realistic mobility patterns of the expected scenarios

1 1 2 Role of Simulation in MANET

Simulation is a technique of emulating or reproducing a real-life situation, process or a hypothetical circumstance In general, simulation is heavily used to study the behavior of complex dynamic systems MANETs encompass a wide range of dynamic systems which are typically complex Due to their versatility, dynamic nature, and inherent complexity, it is difficult to characterize them analytically or conduct experiment using mobile nodes and wireless networks in realistic scenarios Real implementation of MANETs entails high cost and intensive toil Simulation is a useful and attractive alternative prior to actual implementation Therefore, nearly all published research works in this field are heavily based on simulations[5, 13, 20, 23]

1.2 Motivation and Observations

Although most published works in MANET are heavily based on simulations, they have been heavily criticized, particularly in recent times, on the lack of rigor in simulation studies and questioned the credibility of the published claims[5, 13, 20, 23, 32, 34] Specifically, these studies indicate that the credibility of the simulation results in the field has decreased while the use of simulation has steadily increased However, there is no consensus on how to address the concern constructively and systematically This thesis makes a step towards addressing this important issue in relation to mobility models

Most simulations in MANET uses random mobility models, particularly random waypoint model and its variations A specific type of MANET called vehicular ad hoc network (VANET) uses specifically designed mobility models incorporating roads and vehicular aspects

Simulation with a very simple model and simulation with high realism are two extreme cases Between simple and sophisticated modeling, there is a spectrum of simulation models in which several useful models can be chosen with varying levels of details We feel that the purpose and contribution of the research should dictate the level of details in simulation Not all these mobility models are easy to implement from scratch Some, particularly those involving high level of realism, are very challenging to implement In such cases, how to ease this burden? What are the bases for realism in most common applications? Addressing these issues is the main motivation of this thesis

1.3 Contributions

This thesis focuses on addressing the issues outlined in the motivation section by design and implementation of a mobility generation software tool called RLMobiGen [1] This tool can be used to create and analyze various mobility models suitable for a wide range of MANET applications The main contributions of this thesis are

1 A comprehensive survey of realistic mobility models and survey of mobility generation softwares

2 Characterization of a unified framework for important geometric objects in the simulation region that has the potential to influence the mobility of the nodes in the system

3 Design and Implementation of RLMobiGen

RLMobiGen has four main components

1 Creation of geometric elements

2 Visualization and analysis of movement scenarios

3 Calculation, analysis, and visualization of statistical metrics

4 Exporting mobility traces in 6 different formats (NS-2, GloMoSim/Qualnet, NAM, XML, JPG, and PDF)

Creation of geometric elements involves both random creation and extraction from GIS databases The latter involves extraction of realistic road topology, actual road speed limits, point of interests, traffic signals and stop signs information, etc , from Open Street Map GIS database RLMobiGen provides the following performance metrics

1 Movement metrics number of legs, leg distance, leg speed, leg duration, etc

2 Connectivity metrics number of connections, connection duration, connection changes, connection availability, etc

3 Coverage metrics node distribution, coverage, etc

By using our software tool, users can generate, analyze, and adjust various scenarios of the specified mobility model By suitably controlling the parameters, various mobility models with different level of realism can be generated and visualized

1.4 Thesis Organization

A comprehensive survey of realistic mobility models and survey of mobility generation softwares are presented in Chapter 2 Next, in Chapter 3, we discuss the unified framework of realistic mobility models Chapter 4 presents the design of the RLMobiGen, a simulation software that we built and implemented to generate and analyze several realistic mobility models, which can be used in performance studies Experiments performed using RLMobiGen are discussed in Chapter 5 Finally, in Chapter 6, we conclude the thesis while outlining some future directions to extend the work carried out in this thesis

Chapter 2

Literature Review

This chapter reviews the literature on mobility models and mobility generation softwares There exists a variety of mobility models that have been proposed in the literature[16, 18, 28, 36]

2.1 Survey of Mobility Models

Mobility models are classified in the literature as purely random or realistic This section reviews the mobility models specifically proposed for mobile ad-hoc networks

2.1 1 Random Mobility Models

Guerin's mobility model [46] has become the basis for a number of mobility models In Guerin's Mobility Model, as the simulation starts, each mobile node randomly chooses a direction θ from the range $(0 \quad 2\pi)$, randomly selects the speed from a user-defined distribution of speeds and then each node moves in its selected direction at its selected speed for randomly chosen period of time and the process repeats

Various variations of this model have been proposed over the time One variance of this model is referred as *Brownian Motion* [41] The implementation of this mobility model is as follows as the simulation starts, each mobile node chooses a direction θ from the range $(0 \quad 2\pi)$ using uniform distribution, selects the speed using normal distribution and then the node travels in the chosen direction with selected speed for randomly chosen time period and the process repeats. It is renamed as *Random Direction model* in [36]. In a different variation of this model [44] when a node hits the simulation boundary on the vertical edge it is reflected back into the coverage area with an angle of -θ and with an angle of (π-θ) in case if it hits the simulation boundary on horizontal edge. Another simplified variation of this model is introduced in [33]. Unlike the previous model the velocity of the node is held constant until it hits the boundary. When it hits the boundary, it bounces back into the simulation region without pausing at the boundary. In a different variation of this model [35] when a node hits the simulation boundary, it pauses for the fixed time and then randomly chooses a direction θ from the range $(0 \quad \pi)$.

A variation in random direction mobility model is introduced in [36] and referred as *Smooth Random Mobility Model*. In this model, speed and direction of mobile node changes incrementally, and smoothly and the direction change correlates with the speed change (i.e. node slows down while changing the direction).

In *Random Walk Model* (RW) as described in [28], each mobile node randomly chooses a direction θ from the range $(0 \quad 2\pi)$, selects the speed between 0 and 10m/s and then travels for a fixed time period of 60 seconds and the process repeats. In another variation of this model as also described in [28], instead of traveling for a fixed time period (60s), each mobile node travels a fixed number of steps (10 steps) in a direction θ randomly chosen from the range $(0 \quad 2\pi)$ at the speed between 0 and 10m/s.

A variation of random walk model is implemented in Global Mobile Information System Simulator (GloMoSim) [47] and referred as *Random Drunken Mobility Model*. In this model, each node is randomly placed within a field. The node then moves in a direction randomly chosen from the list of possible directions and ensures that it remains in the field boundaries. If a node is at location (x,y) then it can move to any of the (x, y-1), (x, y+1), (x-1, y), or (x+1, y) locations as long as the new location is within the coverage region.

The *Gauss-Markov Mobility Model* is introduced in [42] In this model, the current speed and direction determines the next speed and direction, for every fixed interval of time Therefore, it is a temporally dependent mobility model and the memory level parameter determines the degree of dependency

One frequently used mobility model is *Random Waypoint Model* (RWM) It was introduced in [43] It became a benchmark mobility model to evaluate the MANET protocols, because of its simplicity and wide availability The implementation of this mobility model is as follows as the simulation starts, each mobile node randomly selects one location in the simulation region, then the node pauses at its current location for a fixed period, which is called pause time, chosen uniformly from the range of minimum and maximum pause time, and then randomly chooses a new location as the destination, it then travels towards the destination with constant velocity chosen randomly from the range of minimum and maximum velocity by using uniform distribution and the process repeats

Another model is introduced in [45] and referred as *Boundless Simulation Area Mobility* model It is proposed to eradicate the *border* effect in simulation area by introducing *wrap around* effect on a simulation boundary In this model, when a node hits a boundary of the simulation area it reappears from the opposite side of the simulation boundary with the same speed and direction This technique creates a torus-shaped simulation area allowing mobile nodes to travel unobstructed But one of the inadequacies of this model is that a stationary node and a mobile node moving in same direction become neighbors over and over again [28]

The *Reference Point Group Mobility* model was introduced in [39] In this model, several mobile nodes move together, forming a group as well as each individual mobile node moves randomly within the group The mobility of the logical center for each group, and the mobility of each individual mobile node within the group, are both implemented using the Random Waypoint mobility model

The *Pursue Mobility* model as mentioned in [28] is based on reference point group mobility model This model simulates scenarios where several mobile nodes attempt to capture single mobile node ahead For example, this model could represent police officers (i e , seeker nodes) attempting to catch an escaped criminal (i e , target node) The target node moves freely using Random Waypoint model and seeker nodes try to capture the target node by directing themselves towards the position of the targeted node

A generalized model is introduced in [30] and referred as *Graph-based Mobility* model In this model, the graph represents the locations that the user might visit and the edges model the connection between these locations, for example, streets or train connections Initially, each mobile node is placed at a random vertex in the graph and selects another vertex randomly as its destination and then moves towards it using the shortest possible path After reaching the destination, the node pauses for some randomly selected period and then selects another destination and the process repeats

2 1 2 Realistic Mobility Models

Several researchers have proposed mobility models discussed below, which are designed to be more realistic than the models discussed earlier

A variation of random waypoint model is proposed in [11] and referred as *Realistic Mobility Model* As the simulation starts, both initial location and the destination are randomly chosen from uniform distribution, speed of the node is determined by speed zone which has minimum and maximum speed limits, and initial direction is chosen from δ directions that are separated by segments with equal angles $(\frac{2\pi}{\delta})$ The direction towards the destination is chosen with probability d_0, while a direction that is i segments away from the leading direction is chosen with probability of d_i

Street Map-based model is introduced in [25] to extract the roads from Tiger database [58] In this model, a node randomly selects a destination point on the road graph and moves towards it using Dijkstra's shortest path algorithm

An activity based mobility model is introduced in [40] In this model, simulation region is divided into multiple cells and each cell represents a unique location Node selects an activity (or trip purpose) Each activity has an associated time of day, duration and location Given the current activity, the current time period and the node type, the next activity is randomly selected from the corresponding entries in activity transition matrix Once the activity is selected, its duration is chosen from the activity duration matrix Node then travels to a new location, determined from the type of the activity After reaching at the destination cell, node stays there for the duration of the activity and the process repeats

Another simple realistic mobility model is introduced using random waypoint model in [29] and referred as *Restricted Random Waypoint* model also called as *Localized Random Waypoint* model To bring realism to random waypoint model, towns and highways are introduced Towns are sub geographic regions that are connected with highways inside a simulation region Node moves with the random waypoint mobility model inside towns for majority of the time However, after a certain number of movements in the same town, a node moves to another town over a highway

To simulate the geometry of city streets, *City Section Mobility* model [37], is proposed It models a section of the city having a downtown area with freeway Initially, each mobile node is placed randomly on some point in street, and then randomly chooses a new location on some other street as the destination It then travels towards the destination through shortest path between two points keeping the safe driving characteristics such as speed limit and safe distance between two nodes Upon reaching the destination, the node pauses at its current location for a fixed period and repeats the process until the simulation ends

To synthesize the real-world mobility, couple of models are introduced in [26] and referred as *Freeway Mobility* model and *Manhattan Mobility* model The movement of node is restricted to pathways in the simulation field In the freeway mobility model, the map consists of several freeways and each freeway has lanes in both directions The velocity of mobile node is temporally dependent on its previous velocity Also, the velocity of the following node cannot exceed the velocity of preceding node - also known as spatial dependency In Manhattan mobility model, the map consists of horizontal and vertical lines representing downtown area The movement of node is restricted to horizontal and vertical pathways in the simulation field At an intersection, the mobile node can turn left, right or go straight with a probability of 0 25, 0 25, and 0 5 respectively Unlike freeway mobility model where node cannot change the lane, Manhattan mobility model gives a node some freedom to change its direction Except the above difference, the Manhattan model is the same as Freeway model because the Manhattan mobility model also has high spatial and temporal dependence A variation of Manhattan mobility model is introduced in [6] and referred as *Simple Model* which models vertical and horizontal mobility patterns without direction changes

Another mobility model is introduced in [27] and referred as *Obstacle Mobility* model A realistic movement model is created through the incorporation of *obstacles* and pathways using *Voronoi diagram* [38] of obstacle vertices in the simulation field These obstacles are used to restrict node movement as well as obstruct wireless transmissions In this model each node randomly selects a destination point and then moves towards the selected point through the shortest route from its current location After reaching the destination, the node pauses for some time period The process repeats The obstacles also impact the way radio propagates However, since the location of obstacles and destination of each motion phase is randomly chosen, a certain level of randomness still exists for this model

Several mobility modeling tools for vehicular ad hoc network (VANET) extracts necessary geometry from the local GIS databases such as Tiger database [58] or Geobase [51] The models rely on GIS maps also called GIS Models VANETs are a subclass of MANETs in which mobility patterns are more constrained as they involve specific aspects such as traffic rules, traffic flow, road capacity, vehicle characteristics, etc

As one of the important characteristics of vehicular mobility is the behavior of mobile nodes at traffic intersections (stop signs, traffic light) A model was introduced by Potnis and Mahajan [14] known as *Stop Sign/Traffic Sign* Model that simulates the stop sign and traffic lights In the Stop Sign Model, each vehicle stops at every intersection and waits for a fixed time period In the Traffic Sign Model, vehicles stop with probability of 0 5 for a random time in front of a traffic light

Another mobility model called *Car-Following Model* (GIS-F) [4], is designed to simulate the microscopic behavior of vehicles In this model, the current speed of a vehicle depends on it's previous speed and the desired speed, also maintaining the minimum safety distance to the front vehicle (if any) Another variation of this model is called GIS-F-T (car-following & traffic lights) [4] It is the same as GIS-F however traffic lights are incorporated at intersections

2.2 Survey of Mobility Generators

In this section, we present the comparison of existing mobility generation tools Several software tools are available that can be used for mobile computing simulations Some of them are proprietary tools that are not freely available, such as QualNet [55], OPNET [53], or VISSIM [59] However, there are freeware and open source simulation software which are widely used by the research community Simulation softwares can be divided into three main categories (a) mobility generators (b) network simulators, and (c) combination of both mobility generators and network simulator

Several mobility generation tools exists which are as follows

- **NS-2** The network simulator is a discrete event simulator, developed by VINT project research group and supported by DARPA [52] NS-2 provides substantial support for simulation of network communication protocols over both wired and wireless (local and satellite) networks NS-2 was later extended to include node mobility feature by Monarch research group and it supports random waypoint mobility model

- **GloMoSim** It is a parallel discrete-event simulator, developed by the parallel computing laboratory at UCLA [50] It exhibits a scalable simulation environment for both wireless and wired networks It supports random waypoint, random drunken and trace based mobility models

- **QualNet** It is a commercial version of GloMoSim which provides ultra high-fidelity network evaluation [55]

- **STRAW** *Street Random Waypoint* [56] is specifically designed for VANET, uses a vehicular mobility model on US road topology, which constraints the node movement to streets of real US cities It is limited in providing the whole world map data and also the generated mobility trace can only be exported for SWAN [57]

- **SUMO** *Simulation of Urban Mobility* [31] is also based on vehicular mobility model and extracts the real world road topology Its main features include multi-lane streets with lane changing capability, collision detection and intersection based rules Though it is a powerful tool to generate traffic network simulation, it does not have an option to export generated traces for available network simulators, which makes it very limited to study network protocols

- **MOVE** *Mobility Model Generator for Vehicular Networks* [10] is built on top of SUMO, which generates mobility models for vehicular network simulations The mobility trace can be exported for network simulation tool such as NS-2 or GloMoSim

- **VanetMobiSim** *Vehicular Ad Hoc Network Mobility Simulator* [15] is an extension of another tool called CanuMobiSim [48] which generates mobility trace from a user specified XML configuration of a mobility pattern VanetMobiSim can import maps from the TIGER [58] database as well as generate random maps using Voronoi tessellation Its main features include vehicle acceleration, deceleration, multi-lane streets with lane changing functionality, but does not provide a support of multiple vehicle types However, the generated mobility trace can be exported for different mobile networks tools including NS-2, GloMoSim, and QualNet

- **GrooveSim** It is a mobility and communication simulator [19] GrooveSim imports map files from the TIGER database to generate realistic topologies It was designed to simulate the motion constraints of vehicles related to the speed limit of roads However, it does not model vehicles micro-mobility and does not provide facility to export generated traces for available network simulators

- **MobiREAL** It is proposed with the intent of modeling a real world environment, specifically designed to study urban pedestrian mobility [2] It is a network simulator that can simulate realistic mobility of nodes using probabilistic destination selection User creates street graph and routes though GUI using mouse clicks which creates an extreme time overhead It was initially designed to simulate MANETs and was later extended to include VANETs by incorporating the traffic simulator NETSTREAM [17] that was developed by TOYOTA Since NETSTREAM is a proprietary software, user can not access and modify this part of the simulator which limits its wide usage

- **CityMob** It generates the mobility based on three models, Simple Model, Manhattan Model, and Downtown model [6] It generates the random road network using Manhattan grid model and does not provide the facility to create user defined roads or road extraction through GIS data All streets are two-way, with lanes in both directions and node moves with random speed, within an user-defined range of values Generated mobility trace can be exported in NS-2 format

- **FreeSim** *Free Way Simulator* [49] is specifically designed for VANET that allows for multiple freeway systems to be easily represented and loaded into the simulator as a graph data structure The traffic data used by the simulator can be user generated or be converted from real-time data gathered by a transportation organization However, FreeSim does not support multiple node types and its generated trace can not be used by the available network simulators, which is a serious shortcoming

- **IMPORTANT** The IMPORTANT framework [26] contains metrics to capture mobility characteristics and evaluate the impact of the performance of routing protocols but does not offer visualization of animation

- **GMSF** The *Generic Mobility Simulator Framework* [9] is a tool to simulate and analyze node mobility in vehicular ad hoc networks The road topology is extracted from official Swiss national map (Landeskarte) which constrains the node movement to Swiss streets The mobility trace can be generated using Random Waypoint model, Manhattan model, GIS model, and MMTS model The MMTS mobility model uses the vehicular traces to generate the movement of mobile node Only roads which are accessible by vehicles are imported into the road topology The mobility trace can be exported in multiple formats

- **GEMM** This tool is proposed in [22], brings realism into random way point model by introducing attraction points, activities, roles and group behavior Given the role type, activity is chosen based on its trigger time and hence the attraction point is selected based on its popularity level and the activity triggered Nodes may move between attraction points or any other random location (similar to random waypoint)

SUMO, MOVE, STRAW, and VanetMobiSim all have good software features However, only VanetMobiSim provides excellent trace support [3] Generic simulation tools like NS-2 [52], OPNET [53], QualNet [55] support only limited mobility models such as random waypoint model and its variations Similar software to generate various random mobility models is proposed in [7]

Chapter 3

A Unified Framework for Realistic Mobility Models

3.1 Introduction

In this section we introduce some basic terminology related to mobility models, and provide a simple classification of mobility models, characterize a framework for geometric objects which influence realism in mobility, and discuss some performance metrics First we introduce some definitions which will be used later

Definition 3 1 *A* **node** *refers to the mobile devices used in MANET i e mobile phone, PDA, laptop, palmtop, etc*

Definition 3 2 *A* **node carrier** *(NC) refers to the transporter of the node This can be a car, a pedestrian or a bicycle, for example*

Definition 3 3 *A* **force point** *is known as a point of interest where node may or may not go*

Definition 3 4 *A* **road** *is known as a straight path between two points*

Definition 3 5 *An* **intersection** *is a road junction where two or more roads either meet or cross at the same level*

Definition 3 6 *A* **connecting point** *is a point on the road where a road crosses the simulation boundary region*

3.2 A Classification of Mobility Models

Though several attempts have been made in the literature to classify mobility models, the distinction between random mobility and realistic mobility is not clear and the boundary between them is often fuzzy For example, many mobility models that just avoid sharp turns and sudden speed change are referred as realistic mobility models in the literature[18, 22, 24, 28], although they are heavily influenced by randomness

We use different classifications to make the distinction better We classify the mobility models into two classes *unguided mobility* and *guided mobility* [8] as shown in figure 3 1 In unguided scheme, the mobility is mainly governed by the characteristics of the node In guided scheme, the mobility of a node is governed by both the characteristics of the node as well as the geometry of the region Primary geometry includes mainly the transportation paths such as roads, railways, ferry routes etc Many additional geometrical elements such as malls, gas stations, hotels, etc can be introduced into the region to create more realism

Although random mobility is very useful for many research studies, mobility of living creatures do not follow a complete random mobility pattern Often they are guided by many factors such as infrastructure (roads, malls, trails, playgrounds, mountains, etc), profession or role (students, bus drivers, factory workers, flight attendants, etc), activities, time, etc That is, mobile nodes in real life often follow constrained and predictable mobility patterns, and also, the mobility of a node may be bounded by streets, freeways, obstacles or buildings

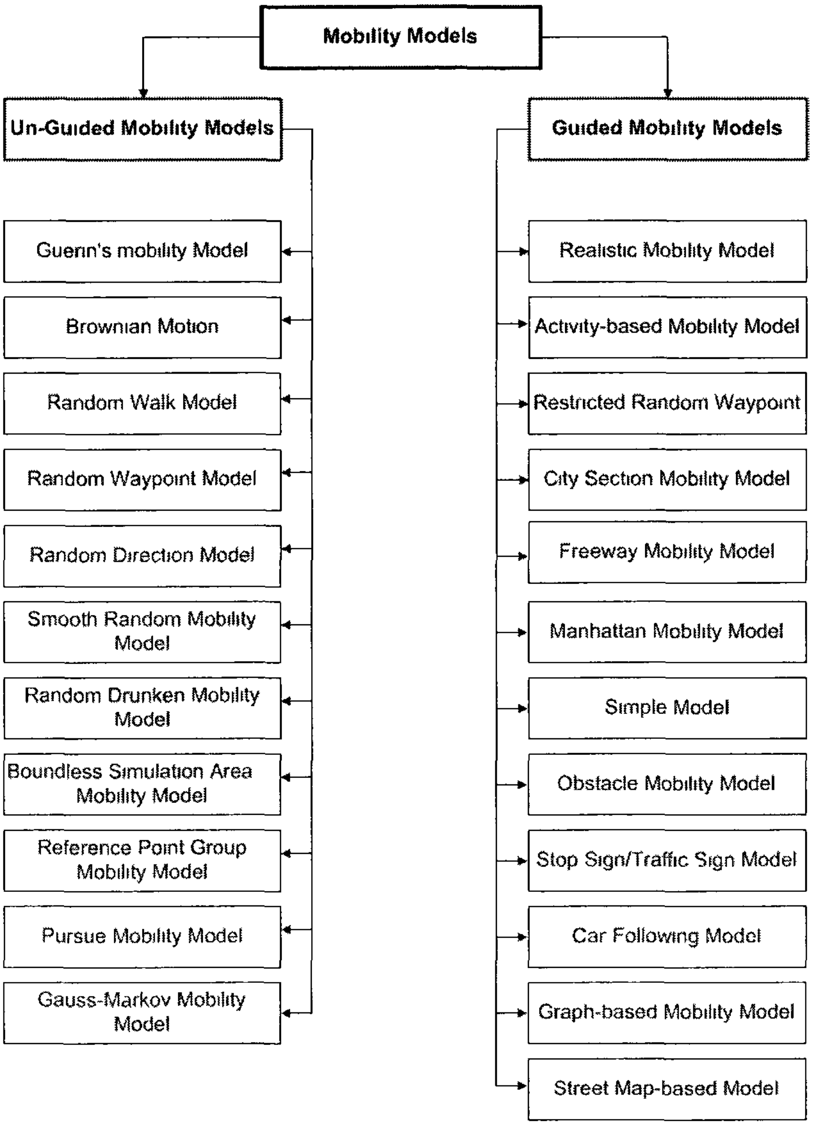

Mobility Models
Un-Guided Mobility Models
Guided Mobility Models
Guerin's mobility Model
Brownian Motion
Random Walk Model
Random Waypoint Model
Random Direction Model
Smooth Random Mobility Model
Random Drunken Mobility Model
Boundless Simulation Area Mobility Model
Reference Point Group Mobility Model
Pursue Mobility Model
Gauss-Markov Mobility Model
Realistic Mobility Model
Activity-based Mobility Model
Restricted Random Waypoint
City Section Mobility Model
Freeway Mobility Model
Manhattan Mobility Model
Simple Model
Obstacle Mobility Model
Stop Sign/Traffic Sign Model
Car Following Model
Graph-based Mobility Model
Street Map-based Model

Figure 3 1 Classification of Mobility Models

3.3 Characterization of Force Points

Generally, regions of interest have objects that could influence the mobility of the nodes in the system. These objects are broadly identified in the literature as *obstacles* that the mobile nodes cannot pass through, *attraction points* that the mobile nodes might attracted to, and *repulsion points* that the mobile nodes would want to avoid going near to. To bring all these objects into a generic framework, we refer to them collectively as **force points** and the force between an individual class of nodes and a force point can be defined using suitable force function. Let FP be the set of all force points and N be the set of different classes of mobile nodes. Then the domain of the inter-force function FF is $FP \times N$ (Cartesian product of FP and N) and the range is R (real numbers). For any class of mobile nodes, the force point is attraction point if the value of inter-force falls in the range $(-\infty, 0)$. Similarly, the force point is repulsion point if the value of inter-force falls in the range $(0, \infty)$. The force point is obstacle, when the force is 0.

These force points could be stationary or mobile. For example, ice cream trucks, ambulance, trucks with inflammable materials can be considered as mobile force points. Parks, theater, museum, etc., can be considered as stationary force points. So each force point may be attached with a speed parameter. The value of this parameter may vary with time. Now, we have three parameters related to each force point - speed, time, and inter-force. If we consider these three parameters in three orthogonal dimensions, each force point will have a unique position in this space. This unified characterization of obstacles, attraction points, and repulsion points, along with their mobile characteristics as force points in a common framework may be useful to model and study the characterization of mobile nodes in MANET.

3.4 Boundary Actions

Another interesting issue in generating mobility trace with a bounded region is what happens when a node hits the simulation boundary. This can be handled in many ways We implement three simple approaches and we refer them as exit, replace, and enter-and-exit

Approach 3 1 Exit *The node instantaneously disappears when it hits the boundary In this model, the number of nodes in the system decreases as system progresses*

Approach 3 2 Replace *When a node hits the boundary, it reappears from another random location on any of the road in the simulation region In this scheme, the number of nodes remains the same throughout the simulation*

Approach 3 3 Enter-and-Exit *The node instantaneously disappears when it hits the boundary Also, nodes appear from the boundary randomly In this model, the number of nodes in the system varies according to enter and exit rate set by the user*

In realistic mobility scenario a node may enter in and exits from a simulation region In a trip, a node may choose either a force point or a connecting point as its destination and proceeds towards it System collects all the possible connecting points based on the road traffic flow (unidirectional or bidirectional)

3.5 Mobility Generation

The main goal of RLMobiGen is to generate mobility with high realism We emphasize that environmental factors influence realism in the mobility models as illustrated in figure 3 2 The first step in generating the mobility is the identification of all the environmental factors which impacts the mobility behaviour We have divided the environmental factors into two sub classes force point scheduling and road geometry Having all the information about the force point, scheduler calculates the availability of all the force points in a given simulation time Road geometry keeps information about all the vertices in a geometry and location of the force points It also stores the location of all the road points intersecting the boundary to implement the boundary action

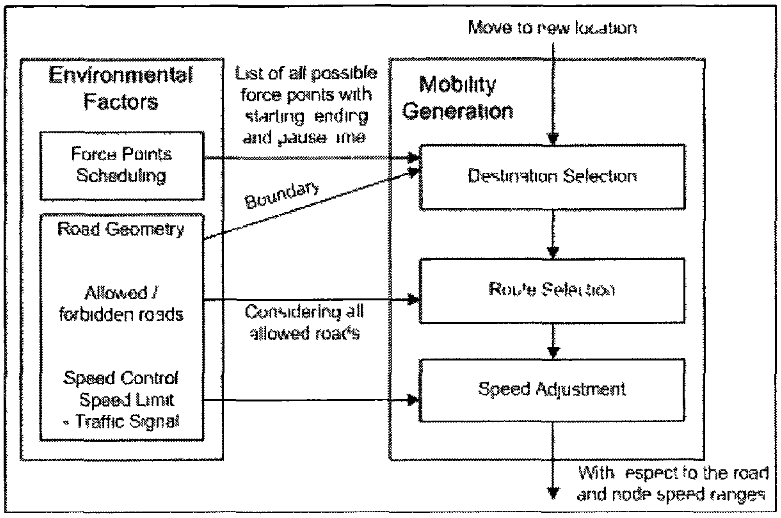

Move to new location
Environmental Factors
Force Points Scheduling
Road Geometry
Allowed / forbidden roads
Speed Control Speed Limit - Traffic Signal
List of all possible force points with starting ending and pause time
Boundary
Considering all allowed roads
Mobility Generation
Destination Selection
Route Selection
Speed Adjustment
With respect to the road and node speed ranges

Figure 3 2 Realistic Mobility

The next step in mobility generation is destination selection A node selects either a force point or a connecting point as its destination This becomes possible by the information provided by environmental factors Force point is selected based on the force value set for specific node class

The next step is to generate a route from current location to the destination We call the movement between current location and the destination the *trip* of the mobility Normally, the shortest route between these two points is calculated, but the force value of all the force points on the calculated route may change the path (the user prefers to go through other attraction points and avoid repulsion points on the way) Once the route selection is done, speed adjustment is carried out through out the trip Once again environmental factors like traffic lights, road limits and node's own speed capability are the deciding factors in speed selection There can be multiple intermediate points in a trip We call the movement between any two consecutive intermediate points the *leg* of the mobility On reaching the destination, the node selects another destination and this process repeats until the simulation ends

3.6 Derivation of Realistic Mobility Models

Realistic mobility models can be derived in several ways

- Deriving directly from real traces
- Introducing realistic aspects like making turns, directions, speeds, and pauses in constrained ways into random mobility models
- Introducing geometric objects such as roads, highways, obstacles, attraction and repulsion points randomly or based on some graphs in the simulation region
- Introducing group mobility based on social networks

In these, models derived from real traces would give highly realistic mobility model But, the availability of such traces are rare and even available historical data are remotely useful as they seldom reflect the future pattern Random mobility models are synthetic and, therefore, they may not reflect mobility of a practical system However, it was considered reasonable to use such models, and particularly one of the random mobility models called random waypoint model is expected to exist as the benchmark mobility model in MANET simulations[21] Random waypoint model is simple and supported by mathematical foundations It will remain natural choice for most proof of concept type simulations However, for simulation of more realistic MANETs, we believe that the rate of use of random mobility models in MANET simulations would decrease drastically if better alternates are available Practical systems will rarely have nodes with random dominated mobilities Therefore, we consider mobility models based on GIS would be more desirable in designing more realistic MANET simulations

Again, introducing geometric objects may be done in, at least, three ways

- Creating random objects and roads
- Extracting objects of real world scenarios from geographic information system (GIS) databases

- Combination of above two

3.7 Performance Metrics

In Mobile ad hoc network, the node's mobility plays a vital role that causes constant changes in ad hoc network topology and hence urges a need to study statistical properties of node mobility In order to evaluate mobility, several performance metrics are introduced in literature Performance metrics used to analyze mobility models fall mainly into three categories [7], movement metrics, connectivity metrics and coverage metrics

1 *Movement metrics* It can be computed for number of legs, leg distance, leg speed, leg duration, origin, destination, direction etc

2 *Connectivity metrics* It can be calculated to analyze the connectivity between number of nodes, which may include metrics like number of connections, connection duration, connection changes, connection availability, etc

3 *Coverage metrics* It can be calculated to analyze the netwoik graph coverage which may include metrics like node distribution, node coverage, etc

The above metrics, if applicable and meaningful, may be computed for minimum, maximum, average, total, standard deviation, rate, ratio, etc , and also at individual, group, or system level

RLMobiGen provides the following instant information for each node n at the sampling time

- node instant position (x, y)

- origin (x, y)

- destination (x, y)

- current speed (km/h)

- leg start time (seconds)

- leg duration (seconds)

- pause at destination (seconds)

In almost every networked system, communication is a fundamental problem for most applications and in dynamic system like mobile ad hoc network, even achieving effective communication between the mobile nodes is challenging In mobile networks, the nodes with transmission range form a dynamic graph known as *connectivity graph* The connectivity and the coverage of this graph influence the performance of most of the communication protocols

The following terminologies related to connectivity and coverage are introduced in [12]

Definition 3 7 *A* **link** *is said to exist or be ON between two nodes i and j if they are within each other's transmission range Link is a boolean function over time t and it is denoted by $link(i, j, t)$*

Definition 3 8 *A* **path** *is said to exist or be ON between two nodes i and j if there is a sequence of nodes and the links between consecutive nodes in the sequence are ON Path is also a boolean function over time t and it is denoted by $path(i, j, t)$*

Definition 3 9 *The interval between ON state and immediate OFF state of a path between the nodes i and j is a* **session** *Session is also a boolean function on time t and it is denoted by $session(i, j, t)$*

We define a new terminologies related to link and path which are termed as *directional link* and *directional path*

Definition 3 10 *A* **Directional link** *is said to exist or be ON between two nodes i and j if any of them is in each other's transmission range Link can be uni-directional (shown in Figure 3 3) or bi-directional (shown in Figure 3 4) based on the transmission range of two nodes Link is a function over time t and it is denoted by $link(i, j, t)$*

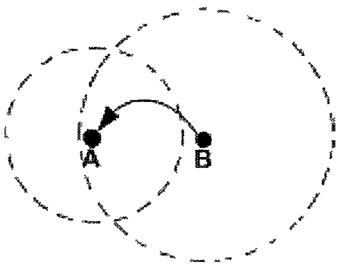


Figure 3 3 Uni-directional Link

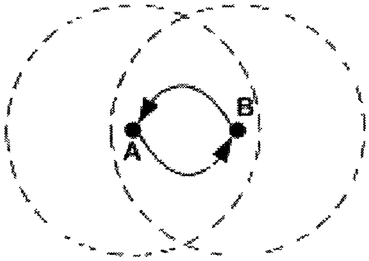


Figure 3 4 Bi-directional Link

$link(i, j, t)$ returns any of the following three values,

- 0 - Link is OFF
- 1 - Link is ON and it is uni-directional
- 2 - Link is ON and it is bi-directional

Figure 3 3 shows a uni-directional link between node A and node B Node B can send the data to node A but node B can not receive the data from node A, as node B is not in the range of node A

$$\mathrm{A} \nrightarrow \mathrm{B} \text{ but } \mathrm{B} \rightarrow \mathrm{A} \Longrightarrow \mathrm{A} \overline{\leftharpoondown} \mathrm{B}$$

Figure 3 4 shows a bi-directional link between node A and node B Node A can send the data to node B and also can receive the data from node B

$$\mathrm{A} \rightarrow \mathrm{B} \text{ and } \mathrm{B} \rightarrow \mathrm{A} \Longrightarrow \mathrm{A} \rightleftharpoons \mathrm{B}$$

Definition 3 11 *A* **Directional path** *is said to exist or be ON between two nodes i and j if there is a sequence of nodes and the links between consecutive nodes in the sequence are ON Path can be uni-directional or bi-directional based on the links Path is a function over time t and it is denoted by* $path(i, j, t)$

$path(i, j, t)$ returns any of the following four values,

- 0 - Path is OFF

- 1 - Path is ON and uni-directional (all links are uni-directional)

- 2 - Path is ON, bi-directional and one or more links are uni-directional

- 3 - Path is ON, bi-directional and all links are bi-directional

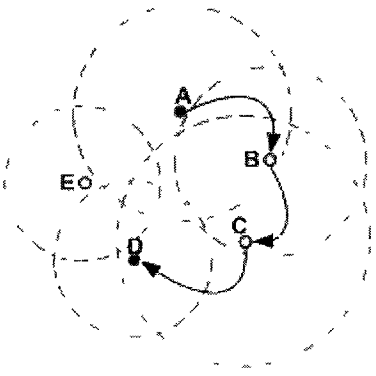
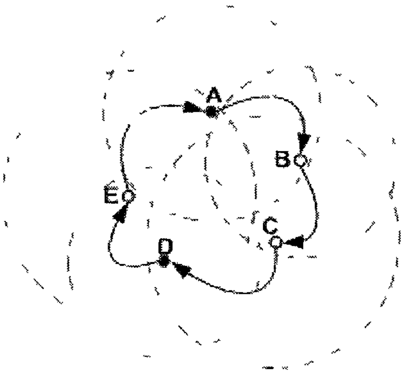


Figure 3 5 Uni-directional Path

Figure 3 5 shows a uni-directional path between node A and node D Node A can send the data to node D through node B and node C, however node D can not send the data to node A, as no node is in the range of node D

$$A \rightarrow B \rightarrow C \rightarrow D \text{ but } D \nrightarrow A \Longrightarrow A \nRightarrow D$$

Figure 3 6 shows a bi-directional path between node A and node D but through two different uni-directional paths Node A can send the data to node D through node B and node C however node A receives the data from node D through node E



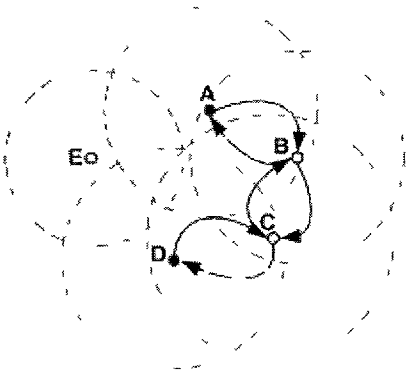


Figure 3 6 Bi-directional Different Path

Figure 3 7 Bi-directional Same Path

$$\mathrm{A} \rightarrow \mathrm{B} \rightarrow \mathrm{C} \rightarrow \mathrm{D} \text{ and } \mathrm{D} \rightarrow \mathrm{E} \rightarrow \mathrm{A} \Longrightarrow \mathrm{A} \rightleftarrows \mathrm{D}$$

Figure 3 7 shows a bi-directional path between node A and node D where all links are bi-directional Node A can send and receive the data from node D through node B and node C

$$\mathrm{A} \rightarrow \mathrm{B} \rightarrow \mathrm{C} \rightarrow \mathrm{D} \text{ and } \mathrm{D} \rightarrow \mathrm{C} \rightarrow \mathrm{B} \rightarrow \mathrm{A} \Longrightarrow \mathrm{A} \leftrightarrow \mathrm{D}$$

3.7 1 Coverage Analysis

Coverage is influenced by both mobility and transmission range of the nodes Node distribution and the ratio of the simulation area covered by transmission range to the total area are useful metrics to be analyzed for coverage Since RLMobiGen is a discrete time based simulator, the metrics are computed over discrete times Here the objective is to study how the nodes spread in the simulation region during simulation

To compute the node spread or spatial distribution, the region is divided into small cells of equal height (h) and width (w), where h and w are two positive integer values and then the nodes are counted inside each cell after specific interval of time

3 7 2 Connectivity Analysis

For connectivity analysis, we mainly study two metrics the connection changes and the session duration Path duration is an important metric for testing communication protocols For example, some protocols referred as connection-oriented protocols require the path between source and destination to be ON throughout the communication

Chapter 4

RLMobiGen

This chapter presents the software architecture and design of RLMobiGen RLMobiGen is a tool to simulate, analyze, visualize, and generate the trace of realistic mobility models Beforc presenting RLMobiGcn, we briefly describc computer simulation

4.1 Computer Simulation

Computer simulation of a system is simply a computer program that runs on a computer and transforms the *state of the system* in discrete time points over a specified period of time Computer simulation can be divided into multiple categories If the simulation program solves mathematical equations (most dynamic systems are usually modeled using differential equations) periodically, and uses the solutions to change the state and output of the simulation, then this simulation is called *continuous simulation* Originally, analog computers were used to run these kind of simulations If the system changes it state at discrete points in time where events occur probabilistically and can not be described directly using mathematical equations is known as *discrete simulation* Discrete simulation can be furthei classified as *discrete-time* and *discrete-event* simulation

Discrete event simulation consists of event list, simulation clock and event scheduler In disciete event simulation the simulator maintains a queue of events (also called *event list*) sorted by the simulated time they should occur *Simulation clock* maintains the simulation time and it is advanced to the time of occurrence of next event in the event list Since, it is not important to execute the simulation in real-time, the advancement of the simulation time can be same, faster or slower than real-time For example, in the simulation of humans evacuating a building, the queues buildup can be visualized using faster than real-time simulations, simulation of current flow through an electric circuit can be shown using slower than real-time simulations, and in-training simulations (i e flight simulators) can be exhibited using real-time simulations *Event scheduler* executes event from the event list and the system state changes at the occurrence of each event in the system System state is represented by state variables For instance, in simulating the behavior of queue at the bank-teller, the number of customers arrived and the number of customers served are state variables and these will be updated on the occurrence of some event in the system The number of customers arrived will be updated when the customer arrives in the bank and the number of customers served will be updated when the bank-teller serves the customer Simulation continues until either event list becomes empty or simulation time ends

The performance metrics of the system are collected from the state variables, and also by using various ways of scientific visualization, simulation results are often aggregated into static images

4.2 Architecture of RLMobiGen

In higher level, the architecture of RLMobiGen is illustrated in figure 4 1 RLMobiGen has following five main logical components

- Simulation Initializer Simulation initializer (SI) is responsible for generating the basic simulation environment It initializes the simulation with parameters such as simulation area, simulation time, and boundary action

- Geometry Generator Geometry generator (GG) is responsible for generating

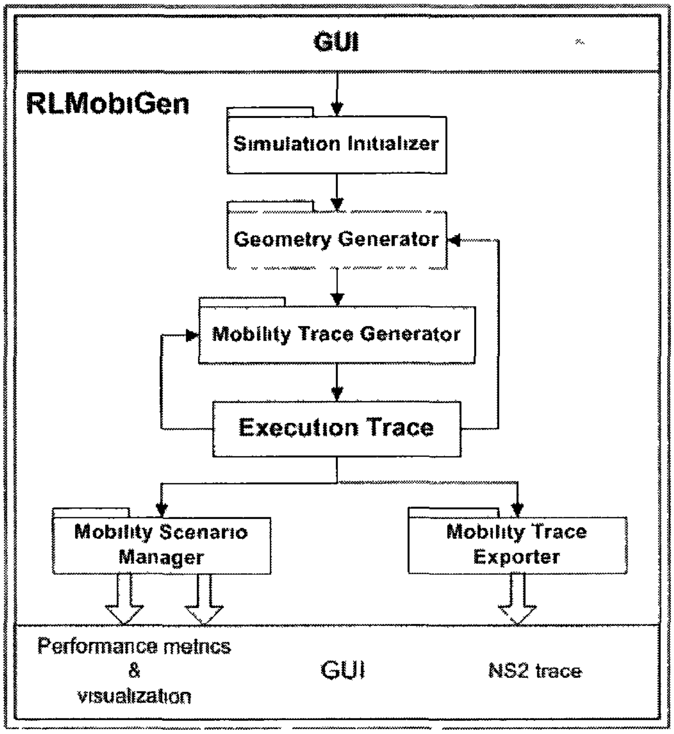


Figure 4 1 RLMobiGen Architecture

the geometry of real life artifacts such as roads, malls, and trails which guide the mobility of the nodes

- Mobility Generator Mobility generator (MG) is responsible for creating nodes and generating their mobility trace The trace also contains information required for visualization and statistical metrics computations based on the parameters set in SI, GG, and MG

- Mobility Scenario Manager This component is responsible for extracting various statistical insights and providing visualizations

- Mobility Trace Exporter RLMobiGen also allows user to export the mobility trace It is responsible for converting the mobility trace into following six formats

  1 NS-2

2 GloMoSim/QualNet

3 NAM

4 JPG

5 PDF

6 XML

The development of RLMobiGen involves four main tasks

(i) generating geometry for the simulation region,

(ii) characterizing mobile nodes and mobile locations,

(iii) modeling mobility, and

(iv) providing meaningful statistical analysis

4.2.1 Geometry Extraction

Generating the geometry for simulation region can be done in several ways Particularly, geometry can be generated randomly or extracted from maps of real world Again, generating random roads can be done in many ways generate vertical and horizontal roads or generate Voronoi tessellation on a set of random points in the given region Realistic geometry of real life artifacts such as roads, malls, and trails can only be generated by extracting the data from real world map Our software offers three main ways of generating geometry

(i) generate random geometry of vertical and horizontal roads,

(ii) generate geometry from real world map, and

(iii) combination of both - first generating from real world map and then editing that either by adding or deleting artifacts

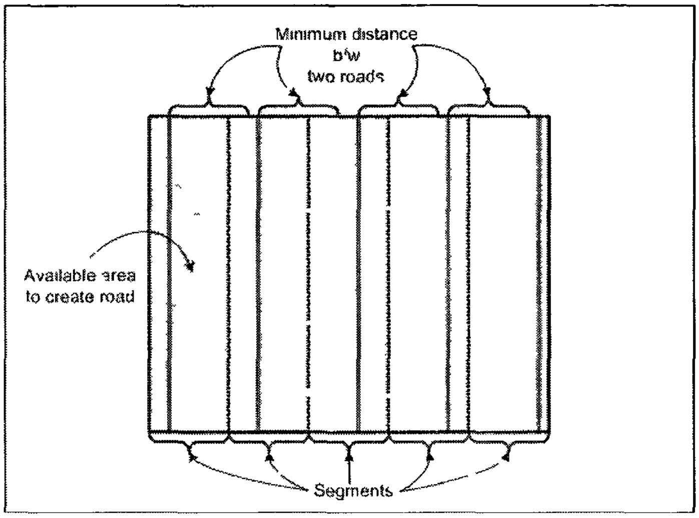


Figure 4 2 Vertical Roads

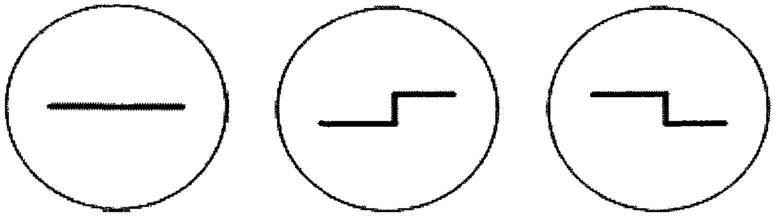
4 2.2 Geometry Generation by Vertical and Horizontal Roads

The first step in creating combination of vertical and horizontal roads is to create vertical roads and then connect all the vertical roads through multiple connecting roads To create vertical road, system requires an input for maximum vertical roads and minimum gap between two consecutive vertical roads Based on the total simulation width, system divides the simulation area into several horizontal segments, which is calculated as follows,

$$horizontal\ segment\ width = \frac{simulation\ width}{maximum\ vertical\ roads}$$

System randomly chooses "x" point within the available area of each segment to draw a straight road from top to the bottom of simulation area Each segment has only one vertical road as illustrated in figure 4 2

Once the vertical roads are placed in the simulation region, multiple connecting roads are created between each vertical road We have implemented three types of connecting roads as illustrated in figure 4 3 To create connecting roads, system requires an input for maximum connecting roads and minimum distance between two consecutive connecting roads Based on the the total simulation height, system divides



(a) (b) (c)

Figure 4 3 Connecting Roads

the simulation area into several vertical segments, which is calculated as follows,

$$vertical\ segment\ height = \frac{simulation\ height}{maximum\ connecting\ roads}$$

The system chooses any one of the three connecting roads randomly and then selects "y" point within each segment to draw a connecting road in between two consecutive vertical roads Each segment has only one connecting road as illustrated in figure 4 4

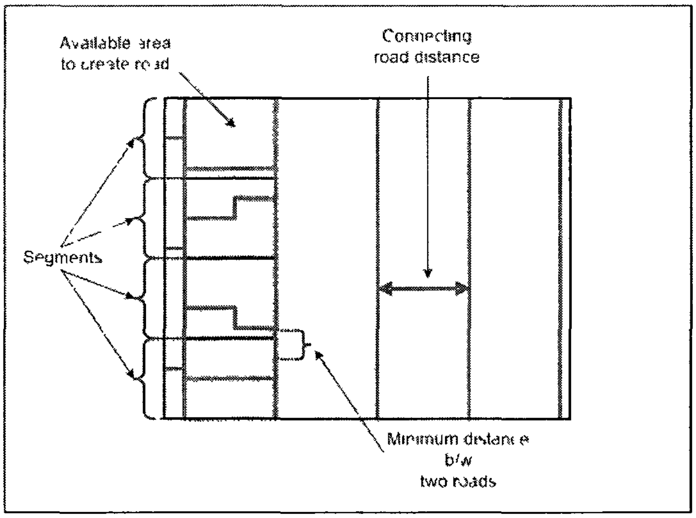


Figure 4 4 Horizontal Roads

The system also assigns a speed limit to each road from the range ($v_{r_{min}}$ - $v_{r_{max}}$) and sets the road type as "residential" Both the speed limits and road type for each road segment can later be changed by the user

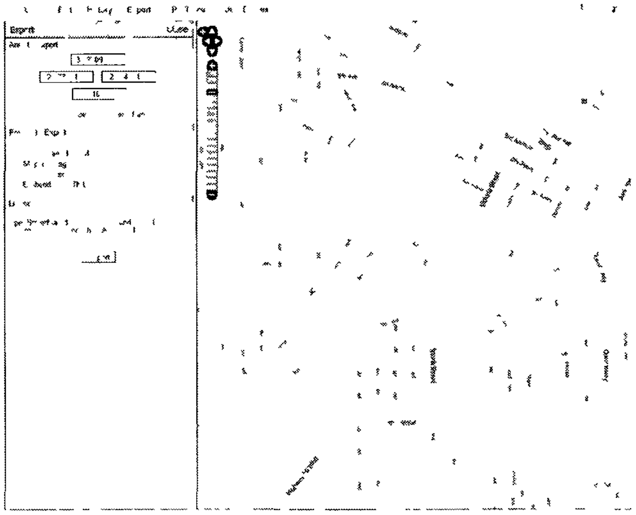


Figure 4 5 OpenStreetMap GIS Data Export Tool

4 2 3 Geometry Generation from Real World Maps

Open Street Map (OSM) website [54] allows everyone to use geographical data free of cost from anywhere on earth and can be exported from the website as shown in figure 4 5 User can use the *"Export"* tab on the main map display to export the map image or raw data of a particular area By default the export engine takes the area of the map currently within view, however user can also manually select the rectangular area (bounding box) in the map It allows user to export data in following formats,

- **OpenStreetMap XML Data** Allows export of the raw OpenStreetMap data in an XML format OpenStreetMap API retrieves the data of a bounding box region The XML data can be saved to a osm file and opened within tools like JOSM

- **Mapnik Image** Allows export of PNG, JPEG, SVG, PDF, and PostScript maps in the OpenStreetMap Mapnik, Osmarender, Cycle Map, and NoName style

- **Osmarender Image** Allows export of a map image in the style produced by tiles@home (Osmarender) by using the MapOf service

- **Embeddable HTML** Creates HTML code which can be embedded in any web page using *iframe*

We have used OpenStreetMap XML Data to create geometry OpenStreetMap uses a topological data structure *Nodes* are points with a geographic position *Ways* are lists of nodes, representing a polyline or polygon *Tags* can be applied to nodes or ways and consist of *tag-name = "value"* pairs For example, Table 4 1 shows a sample file which we used to explain basic file structure Among various information in the osm file, the essential information is discussed below

- *minlat* and *minlon* at line 3 refers to the latitude and longitude of the starting point for the bounding region, whereas *maxlat* and *maxlon* at line 4 refers to the latitude and longitude of the ending point for the bounding region respectively
- Line 5 to 8 contains the information about the point in a bounding region, whereas *lat* and *lon* refers to latitude and longitude of that point There can be multiple points in an OSM file
- "<way>" tag in line 13 to 20 contains the information about the road or building depending upon the value in "<tag>" as shown in line 18 (in this case, it is a road of type "footway") "<nd>" tag as shown at line 16 and 17 contains the information of the points in a road or building There can be multiple ways in an OSM file

In case of a road, first point is considered as starting point and last point is considered as ending point, whereas all points in between are considered as intermediate points in a road so that the precise data of the road curvature can be approximated by piecewise linear curve

The osm file also contains information about road type, number of lanes, speed range, traffic direction flow building types, railway tracks water ways etc

```
<?xml versıon=“1 0” encodıng=“UTF-8” ?>
<osm versıon=“0 6” generator=“CGImap 0 0 2” >
 <bounds mınlat=“53 90215” mınlon=“-122 77261”
    maxlat=“53 92255” maxlon=“-122 74051” />
 <node ıd=“55169025” lat=“53 9108884” lon=“-122 751186”
    user=“MıchaelCollınson” uıd=“308” vısıble=“true” versıon=“3”
    changeset=“685958” tımestamp=“2008-11-05T17 22 20Z”>
 </node>
 <node ıd=“55168450” lat=“53 910621” lon=“-122 751072”
    user=“Jochen Topf” uıd=“4499” vısıble=“true” versıon=“1”
    changeset=“184416” tımestamp=“2009-08-02T14 08 05Z”>
 </node>
 <way ıd=“7609060” user=“Adam” uıd=“5306” vısıble=“true”
    versıon=“3” changeset=“2914902”
    tımestamp=“2009-09-20T13 22 46Z”>
   <nd ref=“55169025” />
   <nd ref=“55168450” />
   <tag k=“hıghway” v=“footway” />
   <tag k=“created_by” v=“JOSM” />
 </way>
</osm>
```

Table 4 1 Open street map xml file (osm) format

Road Types

The following listing defines the different road types available in OSM GIS data

- track,
- trunk, trunk_link,
- secondary, secondary_link,
- primary, primary_link,
- service,
- motorway, motorway_link,
- tertiary,
- footway,
- unclassified,
- path,
- driveway,
- pedestrian,
- road,
- steps,
- cycleway, and
- residential

For the simplification of the data we have combined *trunk* and *trunk_link* as one road type (*trunk*) and the same is done with *secondary*, *primary* and *motorway*

4.3 Design

RLMobıGen ıs capable of generatıng complex scenarıos, ın whıch the mobılıty behavıor of all the nodes ıs dynamıc, each node ıs capable of exhıbıtıng dıfferent mobılıty behavıor at dıfferent ınstance of tıme ın the sımulatıon The sımulatıon parameters of RLMobıGen are sımple, ıntuıtıve, yet theır combınatıon provıdes a powerful mechanısm for sımulatıng realıstıc mobılıty scenarıos

4 3 1 Simulatıon Inıtializer Parameters

These parameters state the basıc sımulatıon characterıstıcs lıke sımulatıon area, duratıon and boundary actıon sımulatıon area ıs used to determıne the dıstance between one poınt to another Durıng the sımulatıon the number of mobıle nodes can ıncrease, decrease or remaın same based on the boundary actıon(ı e Exıt, Replace, and Enter & Exıt) In case of Enter & Exıt selectıon, system takes ınput of Enter & Exıt rate

Sımulatıon parameters are

- Sımulatıon area
- Sımulatıon start tıme
- Sımulatıon end tıme
- Boundary actıon
- Enter & Exıt Rate

4 3.2 Road Parameters

Road constıtutes the geometıy and can be extıacted from OSM XML file Each ıoad ıs a five-tuple consıstıng of name, type, mınımum and maxımum speed and traffic flow dırectıon

Road parameters are

- Road name
- Road type
- Road mın speed
- Road max speed
- Road traffic flow

Road name refers to the name of the road (ı e Unıversıty Way, Davıs road, etc) Road type refers to type of the road (ı e Motorway, Freeway, etc), multıple roads can have the same type Road mın speed ($v_{r_{min}}$) refers to the mınımum speed of the road Road max speed ($v_{r_{max}}$) refers to the maxımum speed of the road User can assıgn a speed lımıt to each road that lımıts the node's own capabılıty to move by adȷustıng ıts own speed v_n to the road speed range Road traffic flow refers to the dırectıon, nodes can move on the road ı e unıdırectıonal (road ıs one-way) or bı-dırectıonal (road ıs two-way) GG parses the OSM file and automatıcally ınıtıalızes all the roads wıth ıts attrıbutes accordıngly, that can later be changed by the user, ıf needed

4.3 3 Force Point Parameters

Force poınts are the regıons of ınterest that determıne the possıble destınatıons towards whıch mobıle nodes can move Each force poınt ıs a nıne-tuple consıstıng of x-y coordınates, type, avaılabılıty and pause duratıon

Force poınt parameters are

- x-coordınate
- y-coordınate
- Type
- Start tıme

- End tıme
- Mınımum pause tıme
- Maxımum pause tıme
- Mınımum speed
- Maxımum speed

Type refers to the type of force poınt (ı e restaurant, educatıonal buıldıng, fuel statıon, etc) Multıple force poınts can have the same type Start tıme refers to the tıme from where onwards, force poınt can be accessıble End tıme refers to the tıme tıll whıch force poınt ıs accessıble Mınımum pause tıme (t_{min}) refers to the mınımum pause tıme a mobıle node can stay after reachıng at the force poınt Maxımum pause tıme (t_{min}) refers to the maxımum pause tıme a mobıle node can stay after reachıng at the force poınt On reachıng the force poınt, node pauses for tıme t, chosen unıformly from (t_{min}, t_{max}) Force poınt can also move at the speed chosen from the range of mınımum speed and maxımum speed Mınımum speed and maxımum speed ıs set to zero, ıf force poınt ıs statıonary

Foıce poınt can be attached to a node carrıer wıth a force value whıch ıdentıfies the force poınt as attractıon poınt, obstacle and repulsıon poınt as mentıoned below

- -100 to -1 (attractıon poınt)
- 0 (obstacle)
- 1 to 100 (repulsıon poınt)

Force value also determınes the probabılıty of a force poınt been selected as a potentıal destınatıon One force poınt can act sımultaneously as attractıon poınt for one class of node and repulsıon poınt for the other class of node

4.3.4 Intersection Parameters

An intersection can have any one of the following two properties,

- Traffic lights
- Stop sign

Traffic Lights refers to red, yellow and green signal, where node stops at red light and moves on green light System captures green light time duration for individual intersections and simulates the traffic light for each individual road of the intersection using round-robin scheduling algorithm Stop Sign refers to the point where node comes to complete stop and then proceed towards its desired destination

4.3.5 Node Carrier Parameters

Each node carrier is a eight-tuple consisting of category, type, speed, rate of change in speed, allowed road types and population

Node carrier parameters are

- Node Category
- Node Type
- NC Min Speed
- NC Max Speed
- Acceleration Rate
- Deceleration Rate
- Allowed Road Types
- Total Nodes

Node category refers to the transporter of the node User selects node category from the predefined list of following six values,

(i) bicycle

(ii) bus

(iii) car

(iv) motorcycle

(v) pedestrian

(vi) taxi

Node type is a user supplied value to differentiate between multiple node categories, multiple node carrier can have the same category (e g mobile phone of young pedestrian, mobile phone of old pedestrian, etc) NC minimum speed ($v_{n_{min}}$) refers to minimum speed a node carrier can travel NC maximum speed ($v_{n_{max}}$) refers to maximum speed a node carrier can travel irrespective of road maximum speed ($v_{r_{max}}$) For example, maximum allowed speed for cars in downtown is 50 km/h but car has capability to travel at 100 km/h The movement of each node carrier is restricted to the road it is moving on and the speed rules apply to node carrier i as mentioned in Table 4 2 Node carrier adjusts the speed based on acceleration or deceleration rate to achieve desired speed A node moves only on the allowed roads, selected by the

```
We must have $v_{min} \leq v_i(t) \leq v_{max}$ at any time $t$
  if $v_{n_{min}} \leq v_{r_{min}}$ and $v_{n_{max}} \geq v_{r_{min}}$ and $v_{n_{max}} \leq v_{r_{max}}$ then
    $v_{min} = v_{r_{min}}$
    $v_{max} = v_{n_{max}}$
  otherwise if $v_{n_{min}} \geq v_{r_{min}}$ and $v_{n_{min}} \leq v_{r_{max}}$ and $v_{n_{max}} \geq v_{r_{max}}$ then
    $v_{min} = v_{n_{min}}$
    $v_{max} = v_{r_{max}}$
  otherwise
    $v_{min} = v_{n_{min}}$
    $v_{max} = v_{n_{max}}$
```

Table 4 2 Node Carrier's Speed Range Selection user User may select more than one road type for a particular node carrier Multiple mobile nodes can be generated having same features of a particular node carrier

4.3.6 Node Parameters

Mobile node inherits mobility features from its node carrier However the transmission range is not dependent on the mobility, rather two or more mobile nodes with same mobility features can have different transmission ranges Node s transmission range is a user supplied non-negative value which determines the distance a mobile node can transmit the data

4.4 Implementation

We used Java Swing and AWT package with the help of NetBeans IDE to build the graphical user interface (GUI) for RLMobiGen

4 4.1 User Interface

Here, we present snapshots of the RLMobiGen graphical user interface (GUI) Building a useful and attractive interface is a complex task

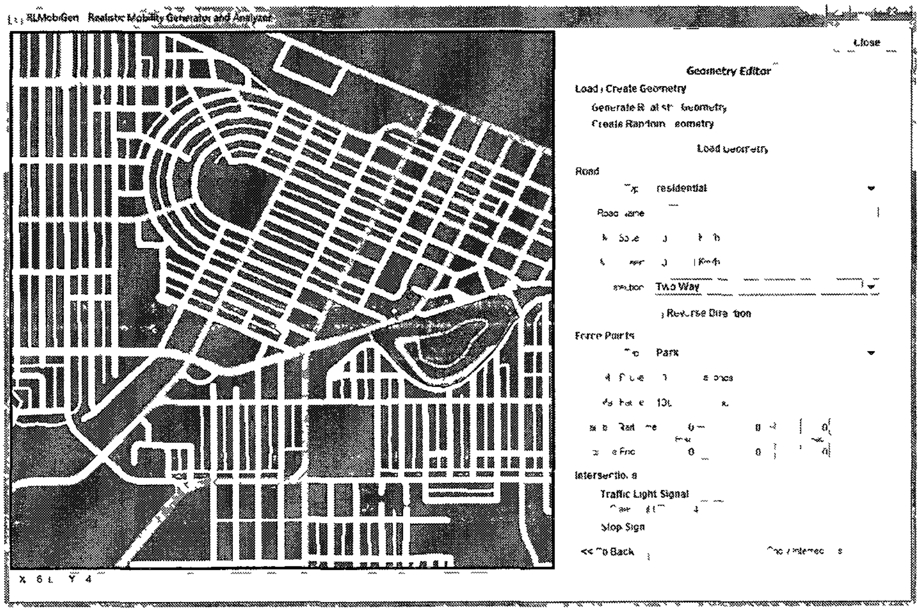
GUI of RLMobiGen is organized as hierarchical panels

- **Geometry Editor panel** (figure 4 6) Allows user to generate and edit geometry

- **Configure Simulation panel** (figure 4 7) Allows user to set simulation, node class and force point configuration

- **Animation panel** (figure 4 8) Traces animation

  Animation panel provides following subpanels to see the performance metrics and visualize the scenarios



RLMobiGen Realistic Mobility Generator and Analyzer

Close

Geometry Editor

Load / Create Geometry

Generate R al st Geometry

Create Random eometry

Load Geometry

Road

Type residential ▾

Road Name

Speed

Direction Two Way ▾

Reverse Direction

Force Points

Type Park ▾

Intersections

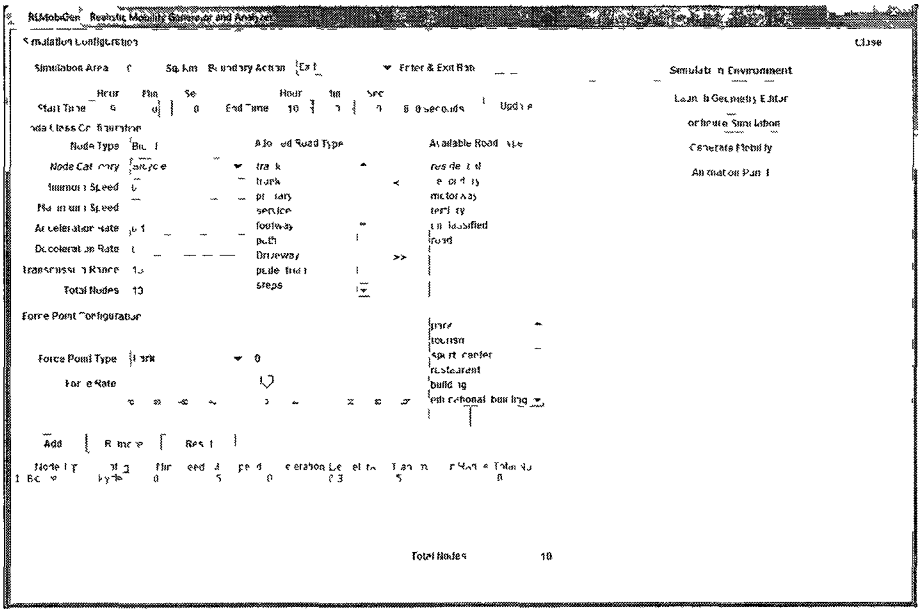
Traffic Light Signal

Stop Sign

<< Go Back

X 6 L Y 4

Figure 4 6 Geometry Editor Panel



RLMobiGen Realistic Mobility Generator and Analyzer

Simulation Configuration

Close

Simulation Area (Sq Km Boundary Action Exit ▾ Enter & Exit Rate

Simulation Environment

Launch Geometry Editor

Configure Simulation

Generate Mobility

Animation Panel

Start Time Hour 6 Min 0 Sec 0 End Time Hour 10 Min Sec 8 0 seconds Update

Node Class Configuration

Node Type Bic

Node Category bicycle ▾

Minimum Speed

Maximum Speed

Acceleration Rate 0 1

Deceleration Rate

Transmission Range 15

Total Nodes 10

Allowed Road Type: track, trunk, primary, service, footway, path, Driveway, pedestrian, steps

< >>

Available Road Type: residential, secondary, motorway, tertiary, unclassified, road

Force Point Configuration

Force Point Type Park ▾ 0

Force Rate

park, tourism, sport center, restaurant, building, educational building

Add Remove Reset

Node Type	Category	Min Speed	Max Speed	Acceleration	Deceleration	Transmission Range	Total Nodes
1 Bicycle	bycycle	0	5	0	0 3	5	8

Total Nodes 10

Figure 4 7 Simulation Configuration Panel

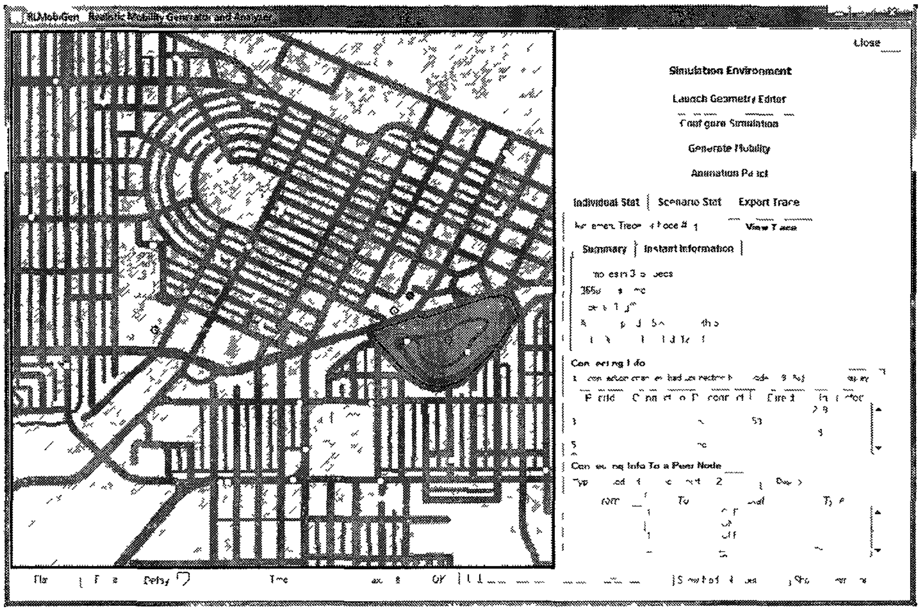


Figure 4 8 Animation Panel

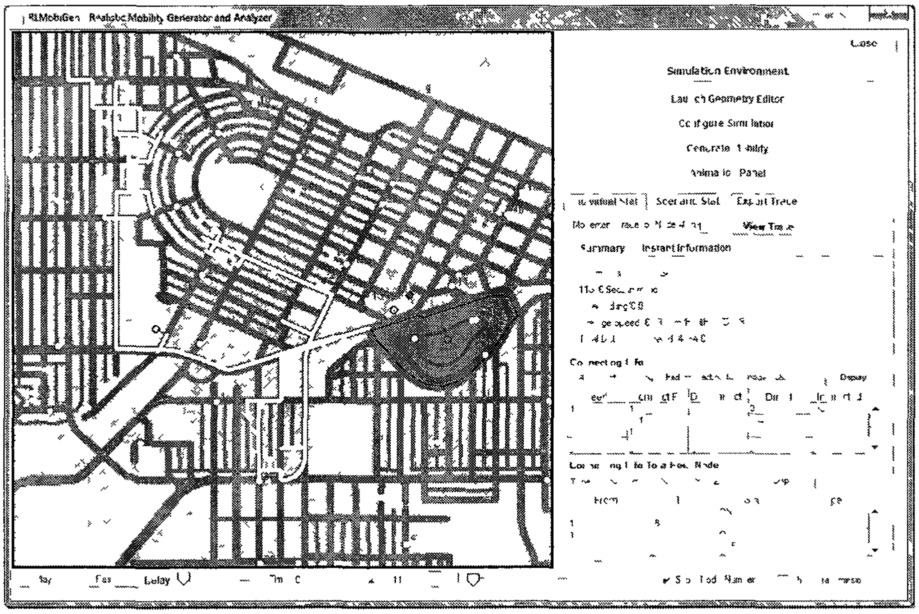


Figure 4 9 Animation Panel with Trace

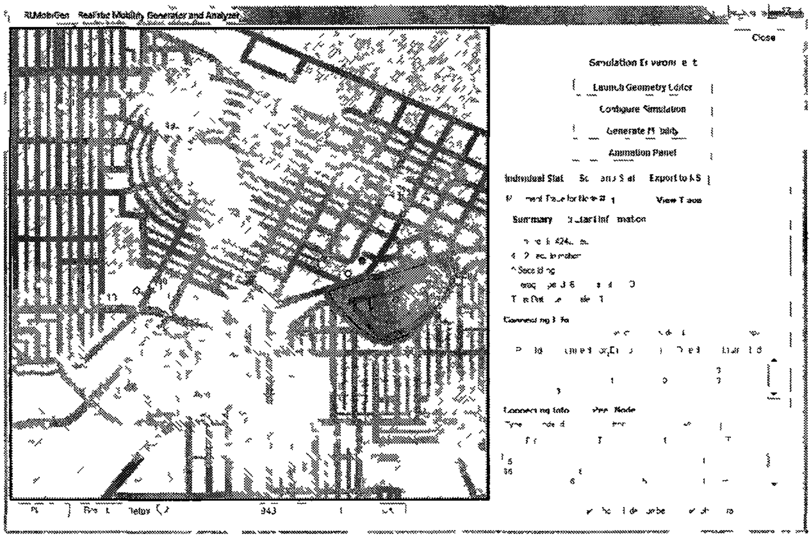
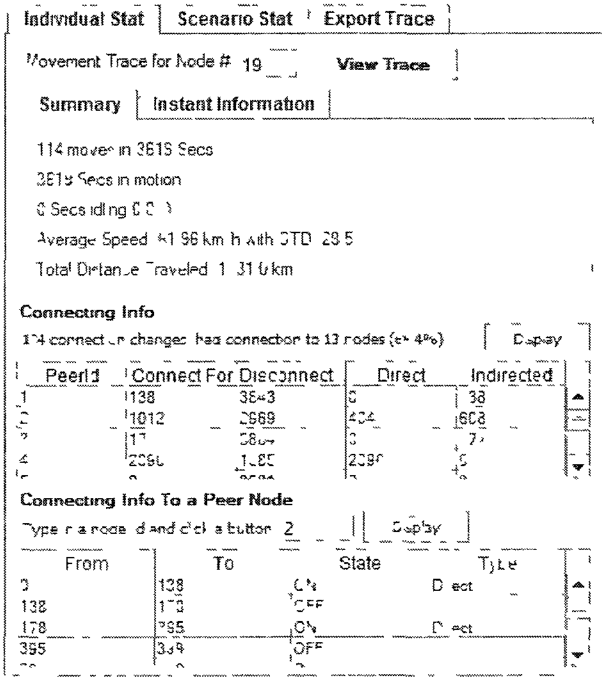


Figure 4 10 Animation Panel with Transmission Range



Movement Trace for Node # 19

114 moves in 3619 Secs

3619 Secs in motion

0 Secs idling 0 0

Average Speed 41 96 km h with STD 28 5

Total Distance Traveled 1 31 6 km

Connecting Info

174 connection changes has connection to 13 nodes (6 4%)

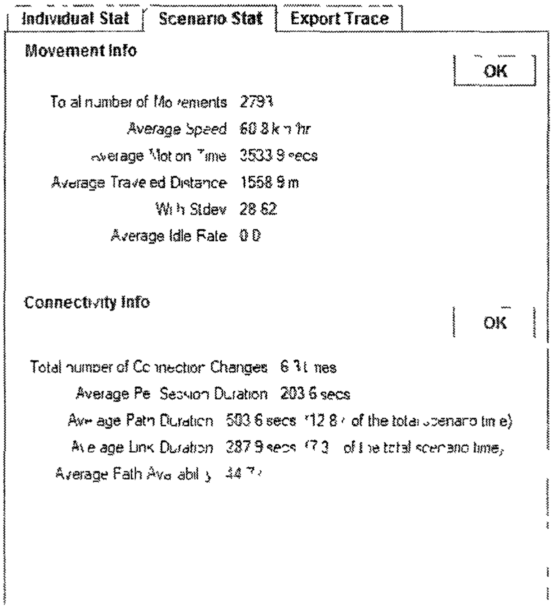
PeerId	Connect For Disconnect		Direct	Indirected
1	138	3543	0	38
2	1012	2969	404	608
3	17	38[illegible]	0	7[illegible]
4	2096	1085	2090	6

Connecting Info To a Peer Node

Type in a node id and click a button 2

From	To	State	Type
0	138	ON	Direct
138	178	OFF	
178	395	ON	Direct
395	349	OFF	

Figure 4 11 Individual Stat subpanel



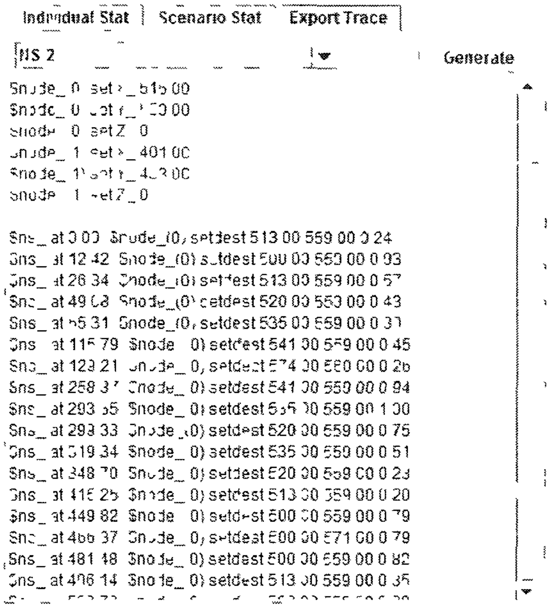
Movement Info

```
        Total number of Movements  2793
                    Average Speed  60 8 km/hr
              Average Motion Time  3533 9 secs
        Average Traveled Distance  1558 9 m
                       With Stdev  28 62
                Average Idle Rate  0 0
```

Connectivity Info

```
Total number of Connection Changes  631 times
     Average Per Session Duration  203 6 secs
         Average Path Duration  503 6 secs (12 8% of the total scenario time)
         Average Link Duration  287 9 secs (7 3% of the total scenario time)
     Average Path Availability  44 7%
```

Figure 4 12 Scenerio Stat subpanel



```
$node_(0) set X_ 515 00
$node_(0) set Y_ 530 00
$node_(0) set Z_ 0
$node_(1) set X_ 401 00
$node_(1) set Y_ 423 00
$node_(1) set Z_ 0

$ns_ at 0 00 $node_(0) setdest 513 00 559 00 0 24
$ns_ at 12 42 $node_(0) setdest 500 00 559 00 0 93
$ns_ at 26 34 $node_(0) setdest 513 00 559 00 0 57
$ns_ at 49 08 $node_(0) setdest 520 00 559 00 0 43
$ns_ at 65 31 $node_(0) setdest 535 00 559 00 0 37
$ns_ at 115 79 $node_(0) setdest 541 00 559 00 0 45
$ns_ at 123 21 $node_(0) setdest 574 00 580 00 0 26
$ns_ at 258 37 $node_(0) setdest 541 00 559 00 0 94
$ns_ at 293 55 $node_(0) setdest 535 00 559 00 1 00
$ns_ at 298 33 $node_(0) setdest 520 00 559 00 0 75
$ns_ at 319 34 $node_(0) setdest 535 00 559 00 0 51
$ns_ at 348 70 $node_(0) setdest 520 00 559 00 0 23
$ns_ at 415 25 $node_(0) setdest 513 00 559 00 0 20
$ns_ at 449 82 $node_(0) setdest 500 00 559 00 0 79
$ns_ at 466 37 $node_(0) setdest 500 00 571 00 0 79
$ns_ at 481 48 $node_(0) setdest 500 00 559 00 0 82
$ns_ at 496 14 $node_(0) setdest 513 00 559 00 0 35
```

Figure 4 13 Export Trace subpanel

- **Individual Stat subpanel** (figure 4 11) Presents performance metrics of individual nodes

- **Scenario Stat subpanel** (figure 4 12) Provides statistical summary of over all nodes

- **Export Trace subpanel** (figure 4 13) Converts the mobility trace into NS-2, GloMoSim/QualNet, NAM, JPG, PDF, and XML formats and displays

4 4.2 User Manual

User clicks *launch geometry editor* button from the main panel to create the geometry User can create either random geometry by selecting *create random geometry* option or realistic geometry by selecting *generate realistic geometry* option User then clicks *load geometry* button In case of realistic geometry, user imports the OSM file *Geometry editor panel* then parses the data file and illustrates the real geometry User may change the parametric values related to road, force points and intersections User then clicks the *go back* button to return to the main panel

User clicks the *configure simulation* button and inputs the specification for simulation and mobile nodes After the specifications are entered, the user clicks *generate mobility* button to generate the mobility trace *Generate mobility* button is provided to regenerate mobility trace with the same geometry When *generate mobility* button is clicked, the nodes will be randomly repositioned in the simulation region

Once the mobility is generated, user may click *animation panel* button to visualize the generated mobility During animation, the transmission range of mobile nodes can be displayed by selecting *show transmission* option to illustrate the connectivity between nodes as shown in figure 4 10 Users can see the performance metrics and visualize the scenarios in the respective panels In *individual stat* subpanel, user may enter node number and click *view trace* button to visualize the generated trace as illustrated in figure 4 9 User can also visualize the node s connectivity information with all other nodes and detailed information to a specific peer node in *individual stat* subpanel by clicking the *display* button In *scenario stat* subpanel, user may click *OK* button to get the summary of movement and connectivity information These metrics are extracted from the mobility trace by the *MetricsGenerator* using time-step based technique to collect statistics In *Export Trace* subpanel, user selects *format type* and clicks *Generate* button to generate the mobility trace in selected format

4.5 Mobility Trace Files

Generated mobılıty trace from RLMobıGen can be exported ın followıng formats,

4 5.1 NS-2 Mobility Trace Format

Mobıle nodes ınıtıalızatıon ın ns-2 ıs set by the followıng command,

$node_(<*node ıd*>) set X_ <*x axıs*>

$node_(<*node ıd*>) set Y_ <*y axıs*>

$node_(<*node ıd*>) set Z_ <*z axıs*>

and the future destınatıons of the mobıle node ıs set by the followıng command,

$ns_ at <*tıme*> "$node_(<*node ıd*>) setdest <*x axıs*> <*y axıs*> <*velocıty*>"

For sample file, see Table 4 3

```
# Node(s) Inıtıalızatıon
$node_(0) set X_ 460 00
$node_(0) set Y_ 36 00
$node_(0) set Z_ 0

# Node(s) movements
$ns_ at 0 00 "$node_(0) setdest 467 00 25 00 66 00"
$ns_ at 19 80 "$node_(0) setdest 420 00 1 00 107 00"
$ns_ at 68 43 "$node_(0) setdest 467 00 25 00 46 00"
$ns_ at 181 50 "$node_(0) setdest 481 00 1 00 36 00"
```

Table 4 3 Example of trace file ın the ns-2 format

4.5.2 GloMoSim Mobility Trace Format

GloMoSim reads mobility traces from two different files which can be specified with the parameter NODE-PLACEMENT and MOBILITY To read the node initial positions from the file, “FILE” value is mentioned in “NODE-PLACEMENT” parameter whereas file path (i e NODES IN) is mentioned in NODE-PLACEMENT-FILE parameter the format of the data stored in NODES IN file is as follows,

```
<node id> 0 (<x axis>, <y axis>, <z axis>)
```

where “0” represents the starting time

The future destinations of mobile node is stored in MOBILITY IN file in the following format,

```
<node id> <time>S (<x axis>, <y axis>, <z axis>)
```

For sample file, see Table 4 4

```
# NODES IN file
0 0 (467 00, 25 00, 0)

# MOBILITY IN file
0 19 80S (420 00, 1 00, 0)
0 68 43S (467 00, 25 00, 0)
0 181 50S (481 00, 1 00, 0)
```

Table 4 4 Example of trace file in the GloMoSim format

4.5 3 NAM Mobility Trace Format

NAM is an animation tool that is used to visualize the mobility trace and real world packet trace data Mobile nodes initialization in NAM is set by the following command,

n -t <*time*> -s <*node id*> -x <*x axis*> -y <*y axis*> -Z <*z axis*> -z <*node size*> -v <*shape*> -c <*color*>

and the future destinations of the mobile node is set by the following command,

n -t <*time*> -s <*node id*> -x <*x axis*> -y <*y axis*> -U <*x velocity*> -V <*y velocity*>

For sample file, see Table 4 5

```
# Node(s) Initialization
n -t * -s 0 -x 460 00 -y 36 00 -Z 0 -z 15 -v circle -c black

# Network configuration
# Setting trace file version
V -t * -v 1 0a5 -a 0
# Setting Wireless range
W -t * -x 1000 -y 1000

# Node(s) movements
n -t 0 00 -s 0 -x 467 00 -y 25 00 -U 66 0 -V 66 0
n -t 19 80 -s 0 -x 420 00 -y 1 00 -U 107 0 -V 107 0
n -t 68 43 -s 0 -x 467 00 -y 25 00 -U 46 0 -V 46 0
n -t 181 50 -s 0 -x 481 00 -y 1 00 -U 36 0 -V 36 0
```

Table 4 5 Example of trace file in the NAM format

4.5.4 XML Mobility Trace Format

The XML based mobılıty trace format ıs not bound to any specıfic network sımulator but can be used to extract varıous statıstıcal outputs

```
<?xml versıon="1 0" encodıng="UTF-8" ?>
<rmg versıon="0 1" generator="RLMobıGen 0 1" >
  <bounds startx="0" starty="0" endx="650" endy="650" />
  <node ıd="0" x="516 00" y="559 00"></node>
  <node ıd="1" x="401 00" y="433 00"></node>
  <nodelegs nodeıd="0">
    <leg destx="513 00" desty="559 00" speed="38 0"
      starttıme="0 00" endtıme="12 41" />
    <leg destx="500 00" desty="559 00" speed="46 0"
      starttıme="12 42" endtıme="26 33" />
    <leg destx="513 00" desty="559 00" speed="36 0"
      starttıme="26 34" endtıme="49 07" />
  </nodelegs>
  <nodelegs nodeıd="1">
    <leg destx="382 00" desty="433 00" speed="33 0"
      starttıme="0 00" endtıme="19 32" />
    <leg destx="361 00" desty="412 00" speed="39 0"
      starttıme="19 33" endtıme="33 24" />
    <leg destx="327 00" desty="315 00" speed="48 0"
      starttıme="33 25" endtıme="48 13" />
  </nodelegs>
</rmg>
```

Table 4 6 RLMobıGen xml tıace file (ımg) foımat

4.5.5 JPG Mobility Trace Format

RLMobiGen allows us to visualize mobility traces in the JPG format as shown in Figure 4 14 which can be easily attached in any document file

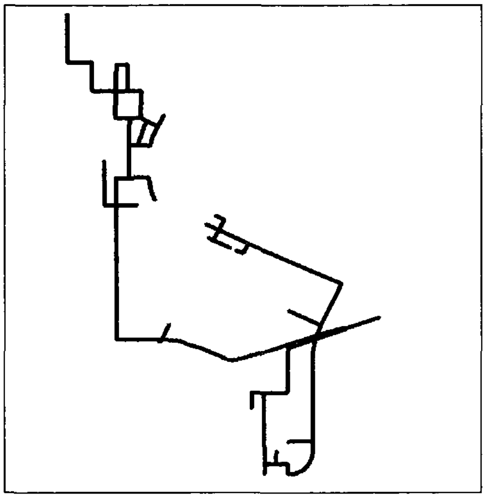


Figure 4 14 Mobility trace for a single node in JPG format

4 5 6 PDF Mobility Trace Format

RLMobiGen also allows the visualization of mobility traces in the PDF format as shown in Figure 4 15

4.6 Summary

In this Chapter we presented the architecture, parameters and implementation of RLMobiGen Also, we explained how RLMobiGen can be used to extract the data from a real world map, generate the mobility trace, visualize various performance and behavioral aspects and then export the trace into six different formats

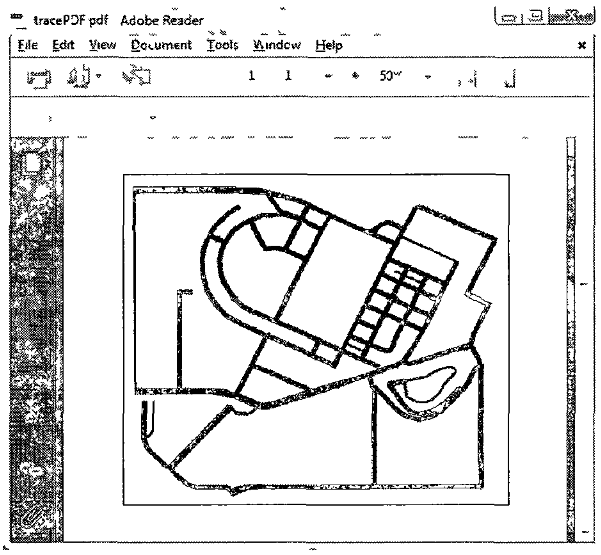


Figure 4 15 Mobility trace for a single node in PDF format

Chapter 5

Experiments

There are several simulation experiments that can be conducted using RLMobiGen. Having the capability to get the parametric values of geometry and mobile nodes from the user, numerous experiments can be performed with following possibilities:

- same geometry features but different mobile node characteristics
- different geometry features but same mobile node characteristics
- and experiments with different geometry and mobile node characteristics

5.1 Experimental Setup

Unless otherwise mentioned, the following are the default values used for the input parameters:

- Simulation Time - [09:00:00 – 09:25:00], 1500 seconds
- Size of the simulation region - 600 sq. m
- Pause range - attached to the force points, traffic lights, and stop signs
- Number of node carrier - 5 (each mobile node has parametric values as shown in Table 5.1)

Node Carrier	Total Nodes	Speed Range (km/hr)	Transmission Range (meters)	Acceleration Rate	Deceleration Rate
1	10	0 - 15	10	0 01	0 05
2	10	0 - 30	25	0 03	0 06
3	10	0 - 50	50	0 05	0 10
4	10	0 - 100	75	0 10	0 20
5	10	0 - 120	100	0 12	0 25

Table 5 1 Node Carrier Parameters

Transmission range is not relevant for node movement and distribution analysis Speed ranges are attached to the roads and node carrier

5.2 Movement Analysis

In this section, we present simulation results of movement related metrics

5 2 1 Leg Calculation

We conducted the experiment in both random (Non-GIS) and realistic (GIS) geometry to calculate the number of legs performed by all the nodes In this experiment we have used 100 nodes The observation of leg calculation for Non-GIS and GIS is shown in figure 5 1 and figure 5 2 respectively

The number of legs are considerably less in Non-GIS model as compared to the number of legs in GIS model This is because, there are high chances that the node may change its speed or direction more frequently because of the curves in a path as compared to the path which is straight and has less intermediate points

Ten experiments were conducted by varying the speed of nodes to analyze the effect of speed on the leg movement and on the average distance traveled by all the nodes The experiments were conducted for 100 nodes The results are shown in

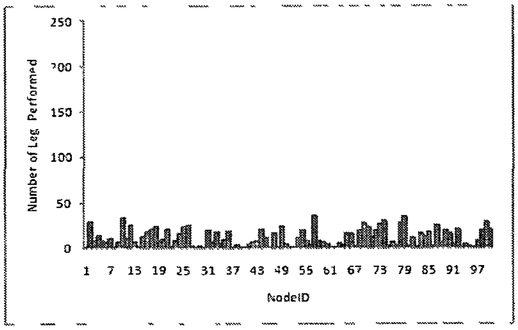


Fıgure 5 1 Number of Legs ın Non-GIS Model

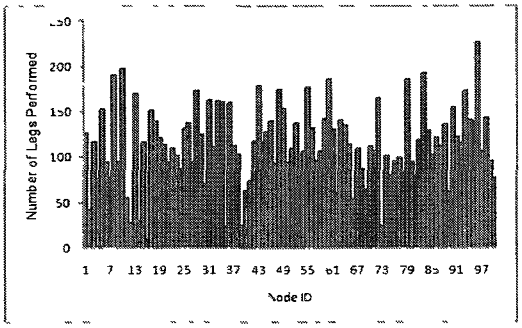


Fıgure 5 2 Number of Legs ın GIS Based Model

figure 5 3 and figure 5 4

Fıom thc figuıcs 5 3 and figuıc 5 4, we make thc followıng obseıvatıons

- Average number of legs ıncreases sıgnıficantly ın GIS based model wıth the ıncrease ın speed as compared to the Non-GIS based model where the ıncrease ın average number of legs ıs very mınımal

- By comparıng both the figures, we can easıly say that there ıs a sıgnıficant change ın number of legs performed, among the two models However, the ratıo of ıncrease ın average dıstance traveled to the ıncrease ın speed appears to be the same

- The number of legs performed does not affect the average dıstance traveled

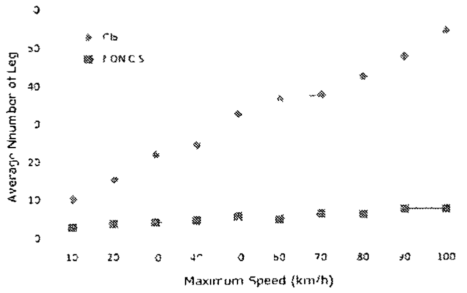


Fıgure 5 3 Number of Legs

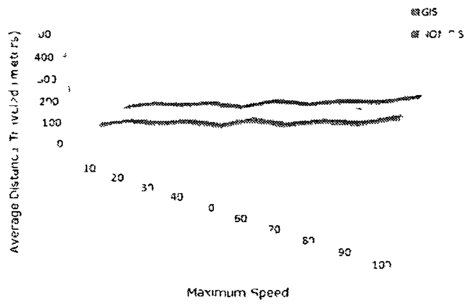


Fıgure 5 4 Average Dıstance Traveled

5.3 Coverage Analysis

In this section, we present simulation results of coverage related metrics

During the experiment, the simulation region is divided into small cells of equal size (10 by 10) and 1000 nodes are uniformly placed inside the simulation region After the nodes move for 500 seconds, a snapshot of the positions of the nodes is taken and the number of nodes at each cell is counted for their spatial distribution

5.3 1 Node Placement

The distribution of mobility nodes in the simulation region highly depends on the placement of node at any given time t The placement of node is also governed by the geometry of the mobility model The node placement in Non-GIS based mobility model and GIS based mobility model is shown in figure 5 5 and figure 5 6 respectively

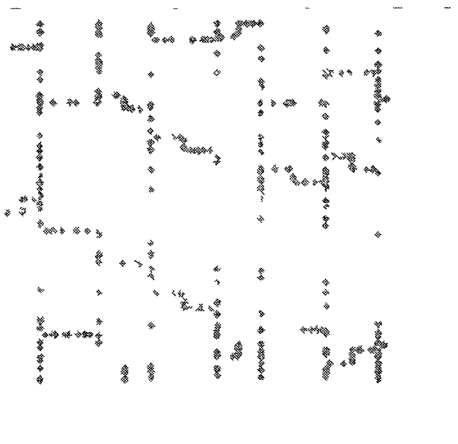


Figure 5 5 Node Placement in Non-GIS Model

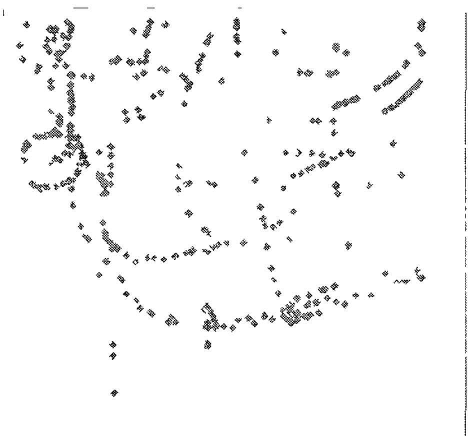


Figure 5 6 Node Placement in GIS Based Model

5.4 Connectivity Analysis

In this section, we present simulation results of connectivity related metrics We studied mainly two metrics the connection changes and session duration We conducted the experiment by varying the transmission range of the mobile nodes The

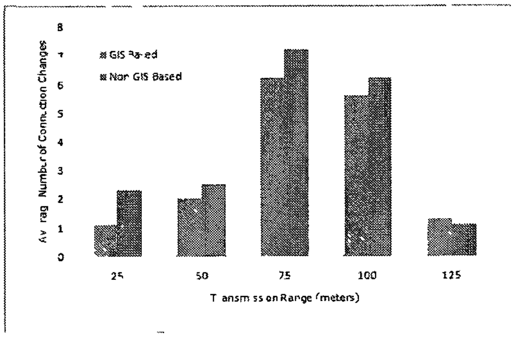


Fıgure 5 7 Connectıon Changes

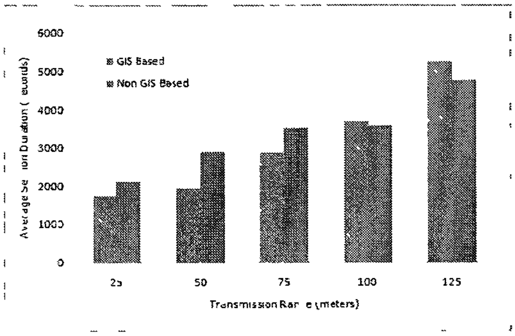


Fıgure 5 8 Sessıon Duratıon

observatıon for connectıon change and sessıon duratıon ıs presented ın figure 5 7 and figure 5 8 respectıvely The change ın connectıvıty ıs hıgh when the transmıssıon range ıs medıum but ıt decreases sıgnıficantly when the transmıssıon range reaches to eıther low or hıgh level However sessıon duratıon ıncreases wıth the ıncrease ın transmıssıon regıon ın both the models

Chapter 6

Conclusion and Future Directions

6.1 Conclusion

Random mobility models used in MANET and realistic mobility models specifically designed for vehicular ad hoc network (VANET) are two extreme cases Between these two simple and sophisticated modeling extremes, there is a spectrum of mobility models in which several useful realistic models can be chosen with varying levels of detail

In this thesis, we presented a unified approach to generate realistic mobility models and presented a software called RLMobiGen that we designed to do the same Using the proposed software, several realistic mobility models with varying characteristic can be generated and analyzed Our software offers several tools to carefully visualize and analyze the generated mobility trace before it is used in actual simulation Such deeper insights into mobility trace would be extremely beneficial for understanding the simulation behavior of the system and will be helpful to interpret the results in a more meaningful way

Though there is always a room for improvement, we believe that our framework provides reasonable reliability of mobility modeling to perform the performance evaluation of protocols and services in MANET systems

The final comparison of RLMobiGen and other mobility generation tools is provıded ın table 6 1

As shown ın table 6 1, the mobılıty tools such as GMSF, GrooveSım, MOVE, STRAW, SUMO, and VanetMobıSım allow the user to ımport the real road topology from the databases On the other hand, tools lıke FreeSım and MobıReal only supports the user defined graph, whereas, CıtyMob and IMPORTANT only generates random geometry Among all these mobılıty generatıon tools, MOVE, SUMO, and VanetMobıSım provıdes the feature of extractıng real world roads, generatıng random roads, and creatıng user defined roads However, RLMobıGen also allows the user to alter the extracted real world geometry or randomly created geometry by addıng or deletıng the roads

To sımulate the realıstıc behavıor of mobıle nodes, obstacles (O) and attractıon poınts (AP) were ıntroduced ın the mobılıty generatıon tools such as GEMM, MobıReal, STRAW, and VanetMobıSım whıch tends to bound the mobılıty of the node However, ın RLMobıGen, we have ıntroduced the unıfied characterızatıon of obstacles, attractıon poınts, and repulsıon poınts (RP), along wıth theır mobıle characterıstıcs as force poınts ın a common framework to sımulate more realıstıc mobılıty patterns

Currently, the mobıle nodes ın all the sımulators perform a trıp by usıng dıffeıent mobılıty models such as random waypoınt model (RWM), random walk (RW), and Dıjkstra shortest path algorıthm In these sımulators, destınatıon selectıon ıs done based on eıther obstacle, attractıon poınt or a random poınt ın a graph However, ın RLMobıGen mobıle node selects eıther a force poınt or a connectıng poınt as a destınatıon

The mobılıty tools such as FreeSım, GrooveSım, and SUMO do not provıde the facılıty to export ıts generated mobılıty trace to already exıstıng network sımulators Other tools lıke VanetMobıSım, MOVE, and GMSF provıde good trace export facılıty and our software RLMobıGen out beats the prevıous sımulators by provıdıng the export facılıty of most trace formats

Our software ıs more generıc, provıdes the facılıty to create ınfinıte node types, and applıcable to a wıde range of systems ıncludıng a class of hıgh level VANETs Also, our tool is more comprehensive in creating, visualizing, and analyzing the mobility models

We hope that RLMobiGen will be helpful and practical within the MANET research community to carry out the performance evaluation of protocols and services in MANET systems

6.2 Future Directions

Although our software is fairly comprehensive, it can be improved in several directions We outline some of them next

- Road speed can be associated with time For example, road speed in school zones may vary depending on the school timings
- RLMobiGen assumes node's transmission range as in 2-dimensional region It can be extended to 3-dimensional to bring more realism into mobile node communication
- Also, RLMobiGen can be improved to enhance the sophistication in the statistical analysis domain such as confidence interval

		Graph			Force Points								
	Input Database	User Defined	Random	Geo Graphical	O	AP	RP	Trip	Speed	Acceleration	Intersection	Visualization Tool	Trace Export Format
CityMob	no		Grid					RWM	random		no	✓	NS-2
FreeSim	no	✓						Dijkstra	road dependent		no	✓	none
GEMM	no					✓		RWM	uniform		no		NS-2
GMSF	Landeskarte			✓				Dijkstra	smooth	✓	traffic lights	✓	NS-2, GloMoSim, QualNet, NAM, XML PDF
GrooveSim	Tiger			✓				RW, Dijkstra	uniform, road dependent		no	✓	none
IMPORTANT	no		✓					RWM, RW	smooth	✓	no		NS-2
MobiReal	no	✓			✓			RW	uniform		no	✓	GTNetS
MOVE	Tiger	✓	Grid, Spider	✓				RW, Dijkstra	uniform		stoch turns	✓	NS-2, GloMoSim, QualNet
STRAW	Tiger			✓	✓			RW, Dijkstra	smooth	✓	traffic lights		Swans
SUMO	Tiger, OSM	✓	✓	✓				RWM	smooth	✓	traffic lights	✓	none
VanetMobiSim	Tiger	✓	Voronoi	✓		✓		RWM, Dijkstra	uniform, road dependent	✓	traffic lights	✓	NS-2, GloMoSim, QualNet, XML
RLMobiGen	OSM	✓	✓	✓	✓	✓	✓	Dijkstra	uniform, road dependent, node carrier dependent	✓	traffic lights, stop sign	✓	NS-2, GloMoSim, QualNet, NAM, XML PDF, JPG

Table 6.1 Comparison of RLMobiGen and Other Mobility Generation Tools

Bibliography

[1] A Aravind, and H Tahir Towards Modeling Realistic Mobility for Performance Evaluations in MANET, *Proc of the 9th International Lecture Notes in Computer Science (LNCS) Conference on Ad Hoc Networks and Wireless (ADHOC-NOW)*, 109-122, 2010

[2] K Maeda, A Uchiyama, T Umedu, H Yamaguchi, K Yasumoto, and T Higashino Urban Pedestrian Mobility for Mobile Wireless Network Simulation *Ad Hoc Network* 7(1) 153-170, 2009

[3] F Martinez, C Toh, J Cano, C Calafate, and P Manzoni A Survey and Comparative Study of Simulators for Vehicular Ad hoc Networks (VANETs) *Wireless Communication and Mobile Computing*, 2009

[4] R Baumann, F Legendre, and P Sommer Generic Mobility Simulation Framework (GMSF) *In Proc of the 1st ACM SIGMOBILE workshop on Mobility models*, 49-56, 2008

[5] I Stojmenovic Simulations in Wireless Sensor and Ad Hoc Networks Matching and Advancing Models, Metrics, and Solutions, *IEEE Communications Magazine*, 102-107, 2008

[6] F Martinez, J Cano, C Calafate, and P Manzoni CityMob A Mobility Model Pattern Generator for VANETs, *Proc of the IEEE International Conference on Communications (ICC)*, 370-374, 2008

[7] A Aravind, and X Cui RMobiGen A Trace Generation, Visualization, and Performance Analysis Tool for Random Mobility Models, *Proc of the High Performance Computing and Simulation Symposium*, (ACM/SIGSIM,SpringSim), 403-412, 2008

[8] A Aravind, H Tahir, and Baldeep RealMobiSim realistic mobility simulator and analyzer *Proc of the 3rd ACM Workshop on Performance Monitoring and Measurement of Heterogeneous Wireless and Wired Networks*, 186-189, 2008

[9] P Sommer Design and Analysis of Realistic Mobility Models for Wireless Mesh Networks, *Msc Thesis*, Communication Systems Group, Computer Engineering and Networks Laboratory (TIK), Institute of Technology Zurich, 2007

[10] F Karanadi, Z Mo, and K -C Lan Rapid Generation of Realistic Mobility Models for VANET *IEEE Proc of the wireless Communications and Networking Conference*, 2506-2511, 2007

[11] A E Kamal, and J N Al-Karaki RMM A Realistic Mobility Model for Mobile Ad hoc Networks, *Proc of the IEEE International Conference on Communications (ICC)*, 2007

[12] X Cui Development of A Trace Generation and Analysis Software For Random Mobility Models, *MSc Thesis*, University of Northern British Columbia, Prince George, Canada, 2006

[13] T R Andel, and A Yasinsac On the Credibility of Manet Simulations, *IEEE Computer*, 48-54, 2006

[14] N Potnis, and A Mahajan Mobility models for vehicular ad hoc network simulations In *ACM-SE 44 Proceedings of the 44th annual Southeast regional conference*, pages 746747, New York, NY, USA, 2006

[15] J Harrı, F Fılalı, C Bonnet, and M Fıore VanetMobıSım Generatıng Realıstıc Mobılıty Patterns for VANET *Proc of the 3rd ınternatıonal workshop on Vehıcular ad hoc networks (ACM VANET)*, 96-97, 2006

[16] J Harrı, F Fılalı, C Bonnet, and M Fıore Mobılıty Models for Vehıcular Ad-hoc Networks A Survey and Taxonomy, *Research Report RR-06-168*, Department of Mobıle Communıcatıons, France, 2006

[17] H Morı, H Kıtaoka, and E Teramoto Traffic Sımulatıon for Predıctıng Traffic Sıtuatıons at expo 2005 *R&D Revıew of Toyota CRDL*, 41(4) 45-51, 2006

[18] F Baı, and A Helmy A Survey of Mobılıty Models ın Wıreless Ad-hoc Networks, *In Wıreless Ad Hoc and Sensor Networks*, Sprınger-Verlag, 2006

[19] R Mangharam, D S Weller, D D Stancıl, R Raȷkumar, and J S Parıkh GrooveSım a topography-accurate sımulator for geographıc routıng ın vehıcular networks, *2nd ACM Workshop on Vehıcular Ad Hoc Networks (VANET 2005)*, 2005

[20] S Kurkowskı, T Camp, and M Colagrosso Manet Sımulatıon Studıes The Incredıbles, *ACM Mobıle Computıng Communıcatıon Revıew*, 9(4) 50-61, 2005

[21] T Chu, and I Nıkolaıdıs Node Densıty and Connectıvıty Propertıes of the Random Waypoınt Model, *Computer Communıcatıons*, 27 914-922, 2004

[22] M Feeley, N Hutchınson, and S Ray Realıstıc Mobılıty for Mobıle Ad-hoc Network Sımulatıon, *Ad-Hoc, Mobıle, and Wıreless Networks*, 324-329, 2004

[23] D Kotz, C Newport, R S Gray, J Lıu, Y Yuan, and C Ellıott Experımental Evaluatıon of Wıreless Sımulatıon Assumptıons, *Proc of the ACM MSWıM*, 78-82, 2004

[24] S Ray Realistic Mobility for Manet Simulation, *MSc Thesis*, University of British Columbia, Vancouver, Canada, 2004

[25] A K Saha, and D B Johnson Modeling Mobility for Vehicular Ad-hoc Networks, *Proc of the ACM Workshop on Vehicular Ad Hoc Networks (VANET)*, 2004

[26] F Bai, N Sadagopan, and A Helmy The IMPORTANT Framework For Analyzing The Impact of Mobility on Performance of Routing Protocols for Ad-hoc Networks, *Proc of IEEE INFOCOM*, 825-835, 2003

[27] A P Jardosh, E M Belding-Royer, K C Almeroth, and S Suri Towards Realistic Mobility Models for Mobile Ad-hoc Networks, *Proc of the 9th Annual International Conference on Mobile Computing and Networking*, 2003

[28] T Camp, J Boleng, and V Davies A survey of Mobility Models for Ad-hoc Network Research, *Wireless Communication and Mobile Computing*, 438-502, 2002

[29] L Blazevic, S Gioidano, and J Y L Boudec Self Organized Terminode Routing *Journal of Cluster Computing*, 5(2) 205-218, 2002

[30] J Tian, J Hahner, C Becker, I Stepanov, and K Rothermel Graph Based Mobility Model for Mobile Ad-hoc Network Simulation, *Proc of the 35th Annual Simulation Symposium*, 2002

[31] D Krajzewicz, G Hertkorn, C Rossel, and P Wagner SUMO (Simulation of Urban Mobility) - an open-source traffic simulation *In A Al-Akaidi, editor, 4th Middle East Symposium on Simulation and Modeling*, 183-187, 2002

[32] D Cavin, Y Sasson, and A Schiper On the Accuracy of MANET Simulators, *ACM Proc of POMC*, 38-43, 2002

[33] Z J Haas, and M R Pearlman The Performance of Query Control Schemes for the Zone Routing Protocol, *IEEE/ACM Transactions on Networking*, 9(4) 427-438, 2001

[34] K Pawlikowski, H D J Jeong, and J S R Lee On Credibility of Simulation Studies of Telecommunication Networks, *IEEE Communications Magazine*, 132-139, 2001

[35] E M Royer, P M Melliar-Smith, and L E Moser An Analysis of the Optimum Node Density for Ad Hoc Mobile Networks, *In Proc of the IEEE International Conference on Communications*, 857-861, 2001

[36] C Bettstetter Smooth is better than sharp a random mobility model for simulation of wireless networks *In Proc of the 4th ACM international Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems MSWIM '01 ACM*, 19-27, 2001

[37] V Davies Evaluating Mobility Models Within an Ad-hoc Network, *Master's Thesis*, Colrado School of Mines, 2000

[38] M D Berg, M V Kieveld, M Overmars, and O Schwarzkopf Computational Geometry Algorithms and Applications, *Springer Verlag*, 2000

[39] X Hong, M Gerla, G Pei, and C Chiang A group mobility model for ad hoc wireless networks *In Proc of the ACM International Workshop on Modeling and Simulation of Wireless and Mobile Systems (MSWiM)*, 1999

[40] J Scourias, and T Kunz An Activity-based Mobility Model and Location Management Simulation Framework, *Proc of the 2nd ACM international workshop on modeling, analysis and simulation of wireless and mobile systems*, 61-68, 1999

[41] C Hartman, and H J Vogel Teletraffic Analysıs of SDMA-Systems wıth Inhomogeneous MS Locatıon Dıstrıbutıon and Mobılıty, *Kluwer Wıreless Personal Communıcatıons*, 11(1) 45-62, 1999

[42] B Lıang, and Z J Haas Predıctıve Dıstance-Based Mobılıty Management for PCS Networks *In Proceedıngs of IEEE Informatıon Communıcatıons Conference (INFOCOM 1999)*, 1999

[43] J Broch, D A Maltz, D B Johnson, Y -C Hu, and J Jetcheva A Performance Comparıson of Multı-Hop Wıreless Ad Hoc Networkıng Routıng Protocols *In Proc of the Fourth Annual ACM/IEEE Intl Conf on Mobıle Computıng and Networkıng*, 85-97, 1998

[44] Z J Haas, and M R Pearlman The Performance of Query Control Schemes for the Zone Routıng Protocol, *Proc of SIGCOMM*, 167-177, Vancouver, BC, 1998

[45] Z Haas A New Routıng Protocol foı Reconfigurable Wıreless Networks *In Proc of IEEE conference ICUPC*, pages 562-565, 1997

[46] R A Guerın Channel Occupancy Tıme Dıstrıbutıon ın a cellular radıo System *IEEE Transactıons on Vehıcular Technology*, 36(3) 89-99, 1987

[47] L Bajaȷ, M Takaı, R Ahuȷa, K Tang, R Bagrodıa, and M Gerla GloMoSım A Scalable Network Sımulatıon Envıronment, Report, Unıversıty of Calıfornıa, Los Angeles, CA 90095

[48] CanuMobıSım - CANU Mobılıty Sımulatıon Envıronment, http //canu ınformatık unı-stuttgart de/mobısım/

[49] FreeSım - Freeway Sımulator, http //www freewaysımulator com/

[50] GloMoSım - Global Mobıle Informatıon Systems Sımulatıon Lıbrary, http //pcl cs ucla edu/proȷects/glomosım/

[51] Geo Base, http //www geobase ca/geobase/en/ındex html

[52] The Network Sımulator ns-2, http //www ısı edu/nsnam/ns/ındex html

[53] OPNET Sımulator, http //www opnet com/

[54] Open Street Map, http //www openstreetmap org/

[55] QualNet Network Sımulator, http //www scalable-networks com/

[56] STRAW - Street Random Waypoınt mobılıty model, http //www aqualab cs northwestern edu/proȷects/STRAW/

[57] SWAN - Sımulator for Wıreless Ad-hoc Networks, http //www eg bucknell edu/swan/

[58] Tıger Database, http //www census gov/geo/www/tıger/

[59] VISSIM Sımulator, http //www ptvamerıca com/software/ptv-vısıon/vıssım/