IDEA

Innovations Deserving
Exploratory Analysis Programs

Transit IDEA Program

Simulation and Animation Model for Planning and Designing Transit Terminals

Final Report for Transit IDEA Project 32

Prepared by:
Prianka N. Seneviratne
TransAn LLC
Silver Spring, MD

November 2005

TRANSPORTATION RESEARCH BOARD
OF THE NATIONAL ACADEMIES

Innovations Deserving Exploratory Analysis (IDEA) Programs Managed by the Transportation Research Board

This Transit IDEA project was funded by the Transit IDEA Program, which fosters development and testing of innovative concepts and methods for advancing transit practice. The Transit IDEA Program is funded by the Federal Transit Administration (FTA) as part of the Transit Cooperative Research Program (TCRP), a cooperative effort of the FTA, the Transportation Research Board (TRB) and the Transit Development Corporation, a nonprofit educational and research organization of the American Public Transportation Association (APTA).

The Transit IDEA Program is one of four IDEA programs managed by TRB. The other IDEA programs are listed below.

- NCHRP Highway IDEA Program, which focuses on advances in the design, construction, safety, and maintenance of highway systems, is part of the National Cooperative Highway Research Program.
- High-Speed Rail IDEA Program, which focuses on innovative methods and technology in support of the Federal Railroad Administration's (FRA) next-generation high-speed rail technology development program.
- Safety IDEA Program, which focuses on innovative approaches for improving railroad safety and intercity bus and truck safety. The Safety IDEA program is funded by the Federal Motor Carrier Safety Administration and the FRA.

Management of the four IDEA programs is coordinated to promote the development and testing of innovative concepts, methods, and technologies.

For information on the IDEA programs, look on the Internet at www.trb.org/idea or contact the IDEA programs office by telephone at (202) 334-3310 or by fax at (202) 334-3471.

IDEA Programs
Transportation Research Board
500 Fifth Street, NW
Washington, DC 20001

Simulation and Animation Model for Planning and Designing Transit Terminals

Final Report

Transit IDEA Project 32

Prepared for the
Transit IDEA Program
Transportation Research Board
National Research Council

Prepared by
Prianka N. Seneviratne
TransAn LLC
Silver Spring, MD

November 2005

ACKNOWLEDGEMENTS

TransAn LLC is grateful to its colleagues, partners, and the various professionals that provided advice and suggestions on matters ranging from programming tools to user interface, to appropriate calibration techniques. In particular, we appreciate the three members of the expert review panel for this project: Massoud Javid of Parsons Brinkerhoff in Las Vegas, Nathalie Martel of Tecsult Inc. in Montreal, and John Bergener of Ricondo Associates in Las Vegas, for the time and effort they volunteered to provide timely suggestions for improving the model throughout its development. The Transit IDEA Program Officer, Harvey Berlin, also deserves special mention and our gratitude for the guidance and encouragement he provided from the outset. Last, but not least, we wish to extend our appreciation also to Vince Jackson and Todd Siedman of the Washington Metropolitan Area Transit Authority for providing traffic and site data for model verification, and to TRB's Transit IDEA panel for the valuable feedback on model features, which have helped increase its versatility.

TABLE OF CONTENTS

EXECUTIVE SUMMARY

The purpose of this Transit IDEA project was to develop a rapid assessment tool for evaluating the impact of internal layouts and traffic management strategies on vehicular flow within and in the vicinity of transit terminals. The project's main output, the Intermodal Terminal Simulation Model (ITSIM), can be used by transit agencies or their consultants to simulate, animate, and compare approach and internal delays, queue lengths and durations, etc., under a variety of stochastic parameters, and different terminal configurations. Steps in installing and modeling a transit terminal are described in a separate User Manual prepared as part of this project.

The model is based on rule-based dynamics (or driver behavior logic), comprised of a specially developed bus parking algorithm as well as advanced car-following and lane-changing logic, to realistically animate vehicle movement within and on the approaches to transit terminals . Object-oriented modeling (OOM) technique used in ITSIM allows a service (bus bay parking) area to be examined as a composition of several objects and object types.

Development of ITSIM permitted previously used parking logic (rules) that enables the simulation and animation of parallel curb parking to be advanced to handle parking decisions that do not fall into the traditional groupings. Two types of parking decisions are considered in the model. One type is vehicles seeking to park closest to the destination (bus stop) without queuing, and the other type is, vehicles that would join a queue to reach the desired destination or parking space. The logic representing the former group's search for parking space and the ultimate parking and departure from the bus loading zone is unique, and ITSIM is the only model described in public domain literature that uses such logic.

ITSIM is comprised of three *modules*. The *data input module* is used to: (i) construct the approach roadway, intersections, terminal configuration; and (ii) enter vehicle characteristics and mix, arrival rates, dwell times, and distribution of buses to the relevant stop within the terminal. The *computation module* is comprised of separate sub-modules to generate transit vehicles, assign characteristics to the vehicles, record the positions of vehicles, and computes basic statistics and performance measures. The third is the *animation and output module*, which displays (animates) the activities and produces summaries of instantaneous or overall performance as graphs and tables as well as EXCEL data files that could be further analyzed.

The model was calibrated and verified at the Silver Spring Metrorail station, in Maryland. Input data such as bus schedules, and AutoCAD maps of the terminal layout and approach roads were provided by the Washington Metropolitan Area Transit Authority (WMATA), while traffic flow (demand) on approach roads, distribution of buses among different loading/unloading bays (attractors), and bus dwell times were collected by the project team at the site. Test results were examined and verified by the expert review panel, and were found to closely meet the project objectives. The model was presented by the principal investigator to WMATA staff on June 21, 2005, and to TRB's Transit IDEA panel on June 25, 2005. In concert, the underlying Java and OpenGL programming tools, use of AutoCAD or real picture backgrounds of study area, low input data requirement, and driver decision-making rules that reflect human behavior, make ITSIM a versatile tool. The principal investigator plans to market ITSIM through a clearing house or a consulting firm.

INTRODUCTION

An essential part of planning transit and intermodal transportation terminals is determining the horizontal layout that would maximize the throughput of buses and other types of motor vehicles safely and with minimum conflict between vehicles and pedestrians. Even today, planning is mostly done under deterministic conditions, using architectural tools. However, with increasing spatial constraints and emphasis on emergency preparedness, it is imperative that the scope of terminal planning in the future will need broadening to include analyses of alternative horizontal layouts under stochastic bus and traffic flows, bus route and vehicle mixes, evacuation rates, etc.

Simulation tools, which can be used for traffic flow analyses under dynamic conditions, have been available for over two decades. Their growth has been fuelled by the rapid expansion of intelligent transportation systems (ITS) applications, advanced computing and programming tools, and by the refinements to traffic flow models based on hydrodynamics. As described by Duitt (*1*), microscopic simulation "schematically represents the individual vehicle driver in the individual vehicle reacting to other individuals and the immediate external constraints on freedom of movement". Thus far, they have largely been used to design traffic management systems for urban arterials, freeways, and specific sites. Macroscopic models have been used to estimate network-wide levels of service and congestion , whereas, microscopic models Have been used for project- or site-level analyses

In both types of models, system parameters such as vehicle arrival rates, vehicle mix, and dwell times can be varied over time to reflect the relevant stochasticity. For example, in the case of freeways, impact on demand and vehicle mix can be analyzed. Additionally, in microscopic models, driver preference for closer parking spaces and shorter walking distances can be incorporated by assigning vehicles, first to available spaces near the terminal entrances, and subsequently to spaces further away. In addition, the severity of congestion (and contributing reasons) at different locations along the network can be identified and evaluated.

Development of microscopic simulation tools continued rapidly on both sides of the Atlantic in the 1990s, with researchers searching for new means to represent driver behavior and include real time signal controls (*2,3,4*). Algers, et. al. (*5*) have identified 58 such simulation models. Fifteen of those, including the popular CORSIM, either evolved or are under development in the US. Most European models (total of 11), including the commercially advanced PARAMICS software, hail from the UK. Barcelo, et. al. (*4*) have advanced AIMSUN2 to a point where users can simulate a network in real time for traffic management purposes or using O-D matrices and route selection models for evaluation. There are also a dozen or so macroscopic and mesoscopic models in use today. The macroscopic models (e.g., METACOR) are used mostly for testing signal optimization and emergency vehicle response management. The mesoscopic models such as INTEGRATION are used for dynamic traffic assignment.

After evaluating 32 of the 58 models, Algers, et. al. (*5*) have made several broad conclusions about the features, capabilities and limitations of the current models. A critical and useful observation in the conclusions that pertain to this project is that "search for parking, bicycles/motorbikes,………, and parked vehicles are essentially not modeled". It was also concluded that model developers have assumed that "vehicles go from one point to another weaving across lanes and creating queues on roads and do not often park nor search for a parking space."

Compared to highway and arterial simulation models, intermodal and transit terminal simulation tools are few. The search undertaken as part of this project revealed two specific models for airport curbside simulation models (*6,7*). They are parts of passenger terminal simulation packages, and not independent terminal simulation models. There are also reported applications of microscopic models such as VISSIM and PARAMICS at airports and transit terminals. Additionally, Sarosky and Wilcox (*8*) have applied a discrete-event simulation model developed for rail freight terminal planning. Rizzoli, et. al. (*9*) have reported the development of a model to assess the impact of different intermodal rail terminals on delays to trucks and loading space requirements. A similar tool for seaport landside simulation has been developed by Gambardella, et. al. (*10*).

Although each of the above intermodal terminal simulation models treat arrival and service rates as random variables, and use object-oriented programming to efficiently describe other stochastic processes, the handling of the most important element, parking, is not explained. For example, there is no indication of how the choice of bus loading/unloading bus bays (parking spaces) are made by the bus operators, and the rules governing the movement of vehicles in and out of parking spaces, and the search for available parking spaces while approaching the destination area.

IDEA PRODUCT

The principal deliverable under this Transit IDEA project, the Intermodal Terminal Simulation Model (ITSIM), was developed with two goals in mind. One goal was to create a tool that can be used to evaluate the impact of alternative internal roadway layouts and traffic management strategies on traffic operations, within and around transit terminals,. The other goal was to advance and test the rule-based dynamics (or driver behavior logic) to realistically simulate vehicle movement, parking space assignment and selection, and parking maneuvers under variable demand conditions. Given that bus operator behavior and decisions, for example at a transit terminal or an airport curbside, are different, this project was used to further the parking logic developed by Prianka Seneviratne (*11*) in association with the advanced car-following and lane-changing logic developed for AIMSUN 2 (*4*) and MITSIM (*3*).

ITSIM is a dual-threaded application, meaning it has two simultaneous paths of execution. The first thread is the simulation model itself, which includes pseudo random number generation, vehicle generation, car following, lane changing, path finding, and other vehicle and object interactions. The second thread takes care of all of the visual aspects of the model. This includes the program menus, toolbars, view windows, object editing, and rendering of the simulation scene in 2-D and 3-D. The two threads execute independently of one another, but share the vehicle and network data. On a multi-processor system, one processor can execute the first thread, while the other processor can execute the second thread taking full advantage of these systems. A 2-D screen shot of Silver Spring Metrorail Terminal being prepared for modeling using ITSIM is shown below in Figure 1.

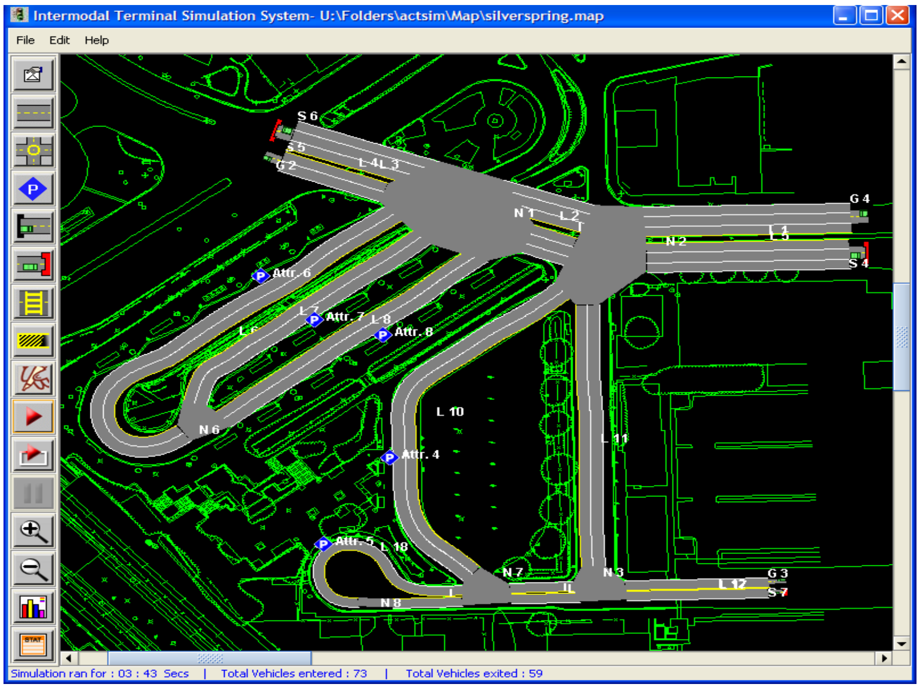

FIGURE 1 Animation of a typical loading/parking area

CONCEPT AND INNOVATION

The concept of simulating and animating the process of buses, taxis, kiss-n-ride vehicles, etc., entering an intermodal transportation terminal, queuing if necessary, moving to the designated stops to pick up and/or drop off passengers, and leaving the terminal required each vehicle and pedestrian to be represented as an object with certain properties and behaviors. The properties include: terminal layout (comprised of a series of nodes and links) vehicle type, dimensions, performance characteristics such as acceleration/deceleration rates, color code, etc. The three fundamental types of behavior include; *interaction with other vehicl*es (passing, following, and parking), *selection of destination* (i.e., loading/unloading within the internal road network in a terminal), *and determination of dwell time* (i.e., the time spent at each loading/unloading bay) had to be simulated. The interaction with other vehicles needed to be defined further in terms of *car- following, finding space within the designated or permitted loading/unloading area*, *lane changing*, and *parking maneuvers*. These interactions among objects add up to an insurmountable number of combinations, and a decentralized approach, which allows a service (parking) area to be examined as a composition of several objects and object classes, was needed.

Therefore, the present researchers decided to employ object-oriented programming (OOP) to model those interactions using a language that permit the gathering of ideas into a comprehensive framework. Java, where the properties of each vehicle and pedestrian class can be described using physical characteristic data, and behaviors can be described by using methods, was found to be best suited for the needs. In Java, classes can be arranged in a treelike hierarchy, so that a child class is able to inherit properties and behaviors from its parent class. It also contains an extensive set of pre-defined classes, grouped in packages that can be used in programs. Java is much simpler than C++, which is another popular object-oriented programming language, because it uses automatic memory allocation and garbage collection where as C++ requires the programmer to allocate memory and to collect garbage. Also, the number of language constructs in Java is small for such a powerful language. The clean syntax makes Java programs easy to write and read. Java is also portable, which means it can be run on any platform. Last, but not least, it provides extensive multimedia facilities that enable powerful multimedia applications such as Graphics Interchange Format (.GIF) and Joint Photography Experts Group (.JPEG). This was essential for importing layouts AutoCAD or picture files of the terminals and approach roads to be used as the backgrounds for visual representation (animation).

INVESTIGATION

Although the description of object classes was straight forward, the behavioral characteristics of the object classes, vehicles and pedestrians, had to be represented using fundamentals of traffic flow theory and typical decision-making processes and rules of drivers in the given environment. The following is a broad overview of the assumptions and features, including the sets of rules underlying ITSIM:

ITSIM comprises of three main modules: *data input*, *computation*, and *animation and output*. The *data input* module is used to define and create the terminal layout and the objects. Layouts are defined and created as links (travel lanes), nodes (intersections connecting links), loading/unloading bays (attractors), crosswalks, etc. within and in the vicinity of the terminal. The user can create the network as a rough sketch or on an imported AutoCAD drawing to the real-life scale. The objects or the vehicles and pedestrians in the network are defined and stored in a separate file. The most complex as well as critical is the *computation module*. It is made up of several sub-modules that generate vehicles, assign characteristics to the vehicles, simulate vehicle behavior as they move through the road network, record the positions of vehicles in small time increments, and performance measures. The *animation module* visually renders the simulated data and displays basic statistics or measures of effectiveness (MOEs). The modular design allows future expansion and improvement to the program without extensive re-coding. Moreover, it enables viewing in 3-D format and real-time model calibration.

DATA INPUT MODULE

The principal input data are characteristics of: links, nodes, vehicles, generators, attractors (the destination or loading/unloading bay within the terminal), and sinks (exit point from the terminal). ITSIM permits users to define *arrival (trip generation) rates* at each generator (entry node) in the form of demand files (i.e., arrival rates over the simulated period) or as forms and parameters of the arrival probability distributions at specified times during the simulated period. *Dwell times* at the attractors are based on whether a vehicle is dropping off or picking up a passenger, which in turn is defined as part of the input.

At each generator, the user defines the percent of vehicle entering the network (terminal) that will go to each attractor. At each attractor, the percent of each type of vehicle that goes to each of the subsequent attractors or sinks are defined. In the vicinity of each attractor, vehicles are distributed to a series of spaces centered at the main point loading bay or parking space such that the utility of the spaces decline as one moves upstream and downstream from the target. The percentages are input via an on-screen data form. The physical demarcation of zones is done graphically on-screen by selecting the appropriate icon from the drawing toolbox and moving the cursor over the area with the left mouse button pressed.

The terminal is laid out as a series of links and nodes. Each link may be a one-way or two-way single or multi-lane facility. Lanes are numbered sequentially to the left and right of the centerline. The type of lane is either parking or through, and can have a separate speed limit. Certain sections of parking lanes can be designated as special zones which may only be accessible to certain types of vehicles. They may be special parking zones or areas reserved for taxis, buses, limousines, etc. Through-lanes are for traffic moving towards a designated zone, for traffic leaving the curbside area, and/or for double-parking when the curb lane is occupied.

COMPUTATION MODULE

This module executes the rules on vehicle dynamics and vehicle steering according to the characteristics defined stored in the first module, and store data at one second intervals for retrieval by the animation module.

The rules governing vehicle and pedestrian dynamics (movement) are described below. They were adapted from several sources (*3,4,11,12,14*), and were modified to represent driver behavior in roadways within intermodal terminals.

Random number generation algorithm used in ITSIM has been coded in assembly language, and is among the fastest pseudo random number generator algorithms. It has a period of 3×10^{47}, and very good randomness. Arrival rates, dwell times, pedestrian crossing rates, travel speeds, vehicle mix, and assignment of vehicles to attractors may be treated as fixed or random quantities following distributions specified by the user. Therefore, the distribution parameters for each class and behavioral variable must be input by the users.

VEHICLE DYNAMICS

Vehicle dynamics in the ITSIM can be divided into three parts; *forward movement, lateral movement (lane changing),* and *parking*. *Forward movement* dynamics is what makes vehicles accelerate or decelerate in the normal traffic stream, and maintain safe distances according to the speed of leading vehicle and spacing (gap).

The basic premise behind the *lateral movement* rules is that a driver will change lanes according to the availability of a gaps and/or speeds of the adjacent lanes in relation to the present lane. If the gap is inadequate and/or relative speed is lower, then the driver will continue in the present lane, except in cases when the lane change is forced (lane end/closure or merging). In the latter case, the lane with the first available gap will be chosen. The set of rules for *forward* and *lateral movement* were adapted from HUTSIM (*14*).

Parking dynamics also consist of several rules. The basic underlying premise there is that each driver seeks a parking space as soon as he or she reaches a user defined point upstream from the attractor it is assigned. Any space within a reasonable distance upstream and downstream of the attraction (i.e., within the user specified loading zone) will be chosen. However, the objective of a driver is to find a space closest to the attraction, and therefore, the space directly in front of an attractor has the highest preference. Preference for spaces upstream and downstream of the most preferred space is assumed to uniformly decrease. When a driver cannot find a space within the loading zone, he or she is assumed to join the re-circulation traffic.

When there is only one lane of curb parking, ITSIM can be used to analyze situations where drivers park on the lane adjacent to the curb lane (unauthorized parking). In this case, it is assumed that the loading zone is an elliptical area centered at the attraction. To keep the computations simple, a lane distance weight factor (LDWF) is used to rank order the parking spaces in the elliptical zone. Accordingly, the preference for space on the adjacent lane is equal to a curb lane space at a distance (LDWF**a*) from the attractor. The value of "*a*" is equal to the distance from the curb to the centerline of the adjacent lane. For example, if LDWF=3, it means that parking space nearest to the door in the adjacent lane and a parking space at a distance three times the perpendicular distance are ranked equally.

Vehicles parked along the curbside or on the opposite side of a median, adjacent to it, are counted as single parked vehicles. A vehicle will start looking for a parking space as soon as it is at a specified distance of the door. This distance is called the 'upstream' distance, and is specified in the vehicle arrivals input form. The distance at which a vehicle will give up parking is the 'downstream' distance.

VEHICLE STEERING

Vehicles traveling across a link or node steer by aiming at a point directly ahead of them. Vehicles changing lanes will change their destination point from one in their own lane to one in the adjacent lane. The vehicle will then change direction by rotating the front tires and 'steering' toward the new destination point. Vehicles can only change direction while they are moving. Tire angular acceleration is assumed constant. The vector representing the direction the vehicle is moving is broken into two components, the direction of the tires, and the vector normal to the tires. It is assumed that the tires never slip sideways, and that this component of the forward force is lost. The force vector in the direction of the tires (minus the rolling resistance) is then broken into two components, a force in the direction the car is moving, and a force normal to the direction the car is moving. This normal force creates a moment about the point between the back tires. The vehicles moment of inertia is taken into account, its angular acceleration is calculated, and the vehicle is rotated.

ANIMATION AND OUTPUT MODULE

The primary function of this module is render the location and status (activity performed) of each object in the network, which was computed and stored by the previous module, in an advance graphics format at one second intervals (*15*). Open GL API developed by Silicon Graphics is used for the animation. It reads the object properties and positions from the computation module and displays them as pixels. In Open GL, the links and nodes are represented by 2D strips and polygons respectively, and vehicles, generators, attractors, and sink are represented as 2 or 3D boxes. Animation includes the grouping of vehicles behind slower vehicles, vehicle accumulation at crosswalks, locations where double parking occur, and where parked vehicles are blocked from entering the adjacent traffic stream. A particular section of the network may also be chosen for representation in 3-dimensions. A real-time 3-dimensional rendering of the simulated traffic at the entrance to Silver Spring Metrorail terminal is shown in Figure 2.

The animation module can display a chosen set of MOEs (statistics) ranging from the total vehicle arrivals and departures during that interval (accumulation) to delays (total time spent by a vehicle in a given section minus the actual dwell time minus the time to traverse that section at the maximum allowable speed) in real time. A typical view of instantaneous performance indicators generated is shown in Figure 3.

The MOEs over the whole simulation period is the final output. It could be viewed as graphical or tabular summaries. The latter could be saved as Microsoft EXCEL files for further analyses by the user.

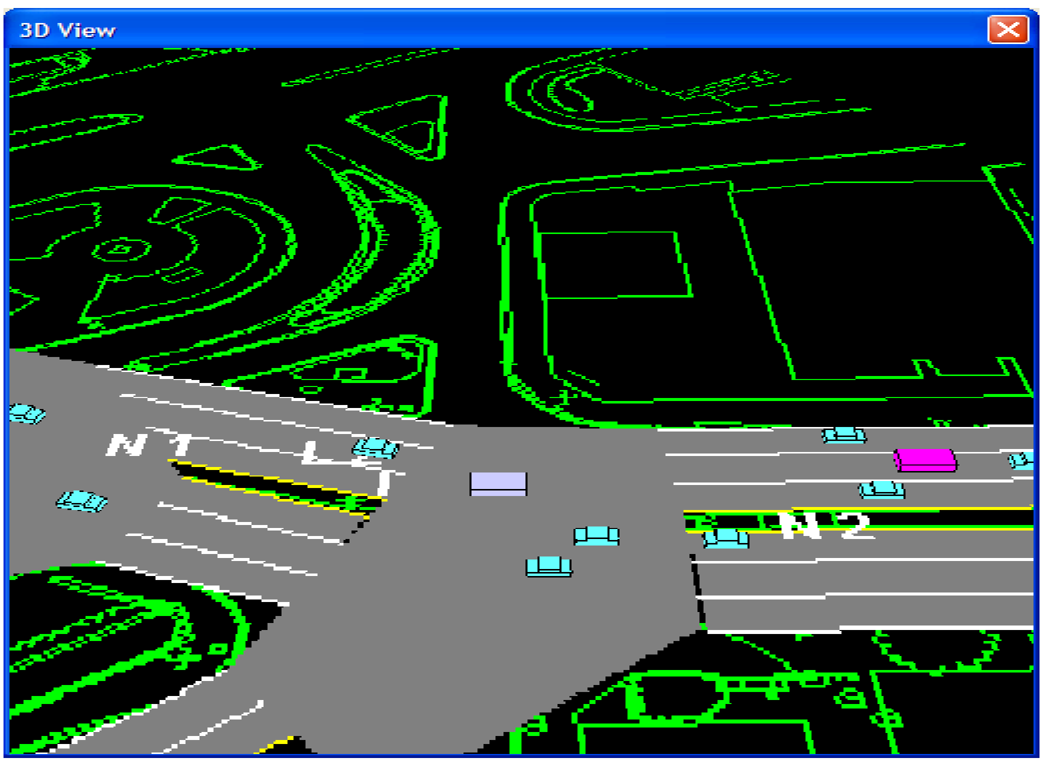

FIGURE 2 OpenGL 3-dimensional view

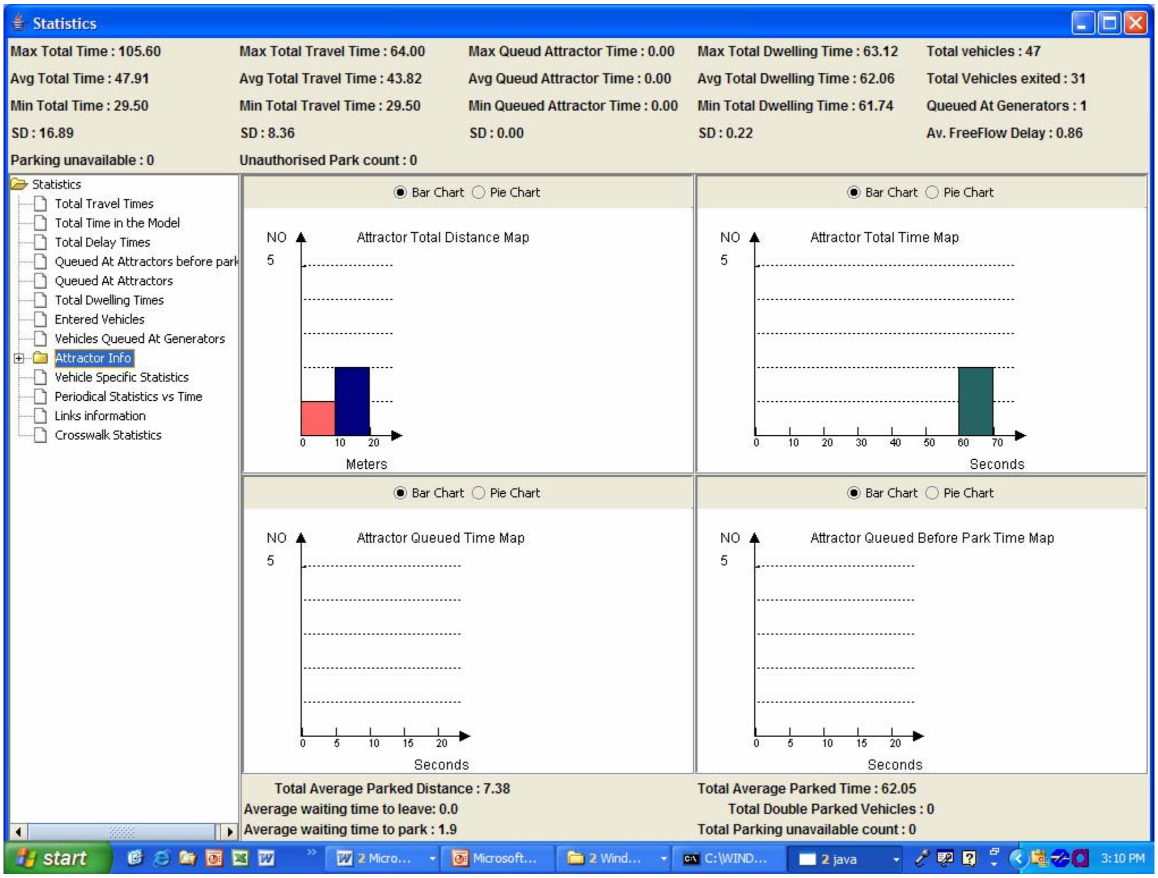

FIGURE 3 Typical instantaneous performance indicators generated for each section

USING ITSIM

ITSIM is user friendly, and requires no programming or special computer skills to use. Steps in installing and modeling a transit terminal are described in a separate User Manual.

Due to the Java portability, it can be run on any platform. However, the following hardware and software characteristics are recommended as a minimum:

- Windows XP or 2000 platform (can be run on any OS that supports Java, Java graphics and OpenGL graphics engine for Java)
- 256 MB of RAM
- NVidia GForce MX440 graphics card with 64MB

ITSIM was calibrated and verified at the Silver Spring Transit Terminal, in Maryland. Input data such as bus schedules, and AutoCAD maps of the terminal layout and approach roads were provided by WMATA. Traffic flow (demand) on approach roads, distribution of buses among different loading/unloading bays (attractors), and bus dwell times were collected at the site by the present research team. Test results were examined and verified by the expert review panel, and were found to be satisfactory and in line with the project objectives.

PLANS FOR IMPLEMENTATION

Based on the comments and suggestions provided by the expert review panel, WMATA and TRB's Transit IDEA panel, several modifications to the model are planned. Firstly, the network creation feature will be modified to allow all types of files to be imported as backgrounds, and adjusted to the preferred size and shape. Secondly, signal control at intersection on the approach roads to the terminal will be introduced. That would allow ITSIM to be used to evaluate the impacts of changes to approach street layout and cycle times on bus delays. In the interim, additional coding has been done to allow a user to run ITSIM over extended time periods up to three hours at thrice the real speed, and it has been referred to a Canadian consulting firm that has been awarded a contract to analyze an intermodal terminal design. Once the above modifications are made, the principal investigator plans to market ITSIM through a clearing house or a consulting firm.

CONCLUSIONS

This Transit IDEA project has been successful in creating a low-cost alternative tool that can enhance the ability of transit agencies to rapidly perform evaluations of intermodal transit terminal development and improvement plans. Although the concept of object oriented programming has been employed by traffic engineers for nearly three decades, the surveys undertaken as part of this project revealed that the advent of powerful tools such as Java has allowed it to be furthered to simulate and animate complex traffic networks. Several new and modified tools have emerged in the last five years, but they were found to have two limitations. One is, many are better suited for simulating and animating freeways, urban arterials, intersections, and choke points such as parking garage entry/exit points, border crossings, and toll plazas. The second is that, most require costly licensing agreements, extensive training, and massive amounts of input data.

The principal output of the project, ITSIM, is based on the same classical traffic engineering principles, advanced vehicle control logic, and programmed using state-of-the-art object oriented programming languages. It is especially designed to simulate and animate traffic flow through transit and intermodal transportation terminals that involve parallel and angle parking at designated stops within its confines and the approach roads. ITSIM includes a wide distribution of population behavior, and can handle time-variant demand or short-run traffic flow variations. The parking logic in ITSIM is unique and, at the time of writing, it is the only model described in public domain literature that uses such logic. Another, perhaps a more generic advancement, is the on-screen network editing capability. In comparison to the currently available public domain traffic simulation software, ITSIM offers a unique interface for data entry and editing that is both user-friendly and graphical. Point and click capability for editing and tracking simulation data enables a user to test alternatives rapidly or note possible errors in collected and entered data.

INVESTIGATOR PROFILE

Principal Investigator: Prianka N. Seneviratne, PhD

Name of Firm: TransAn LLC

Contact Information: 314 Ellsworth Drive
Silver Spring, MD 20910
Tel: (301) 588-0714
Fax: (301) 562-7913
E-mail: transanllc@cs.com
Web: http://translanllc.com

GLOSSARY

ITSIM	–	Intermodal Terminal Simulation Model
MOE	–	measures of effectiveness
TCRP	–	Transit Cooperative Research Program
TRB	–	Transportation Research Board
WMATA	–	Washington Metropolitan Area Transit Authority

REFERENCES

1. Druitt, Stephen (2004). Introduction to microsimulation. http://www.sias.com/sias/s-paramics/articles/article2/article2.html

2. Zhou, B., Blum, J., And Eskandarian, A. (2005). Virtual Reality Visualization of Microscopic Traffic Simulations. 84th Annual Meeting of The Transportation Research Board. Washington. Dc. January.

3. Ben Akiva, Moshe, et. al. (1998). A simulation laboratory for evaluating dynamic traffic management systems. Paper presented at *INFORMS/CORS* Montreal 1998 Cluster: Advances in Microscopic Traffic Simulation.

4. Barcelo, J. et. Al. (1999). Modeling advanced transport telematics applications with microscopic simulators: the case of AIMSUN 2.

5. Algers, Stefan, et. al. (1999). Review of micro-simulation models. SMARTEST project deliverable. http://www.its.leeds.ac.uk/smartest/deliv3.html.

6. Reed, K.A. (1994). Landside computer simulation using the airport landside planning system (ALPS). JKH Mobility Services, Inc.

7. Bender, G.G. and Chang, K.Y., (1997). Roadway and curbside traffic simulation for Las Vegas McCarran International Airport, *ASCE Conference on Airport Modeling and Simulation*, May.

8. Saroksy, Thomas and Wilcox, Terry. (1994). Simulation of a railroad intermodal terminal. Proceedings of the 1994 Winter Simulation Conference. J. D. Tew, S. Manivannan, D. A. Sadowski, and A.F. Seitz (ed). Association of Computing Machinery. December.

9. Rizzoli A.E., Fornara N., Gambardella L.M. (1999). A Simulation Tool for Combined Rail-Road Transport in Intermodal Terminals, MODSIM99, International Congress on Modeling and Simulation, Hamilton, New Zealand, 6-9th December.

10. Gambarella, L. M. et. al. (1998). Simulation and planning of intermodal container terminals. Special issue on Harbor and Maritime Simulation in *Simulation Journal*. Vol. 21. No. 2. pp. 107-116.

11. Seneviratne, Prianka (2001). Access Traffic Simulation Model (ACTSIM). Final Report. ITS-IDEA Project 78. National Research Council. Transportation Research Board. Washington, DC. January.

12. Kraus, Stefan. (1998). Microscopic modeling of traffic flow: investigation of collision free vehicle dynamics. Doctoral Thesis. University of Cologne. DLR-Forschungsbericht 98-08.

13. Nagel, Kai. (1998). From particle hopping models to traffic flow theory. *Transportation Research Record 1331*. Transportation Research Board. National Research Council. Washington, D.C. pp. 1-9.

14. Koskonen, I. (1996). HUTSIM – Simulation Tool for Traffic Signal Control Planning. Department of Civil & Environmental Engineering. Helsinki University of Technology. Laboratory of Transportation Engineering. Publication # 89.

15. Tan, Aaron C. and Bowden, Royce O (2004). The virtual intermodal transportation system (VITS). Final Report submitted to the University Transportation Centers Program. Department of Industrial Engineering Mississippi State University. May.