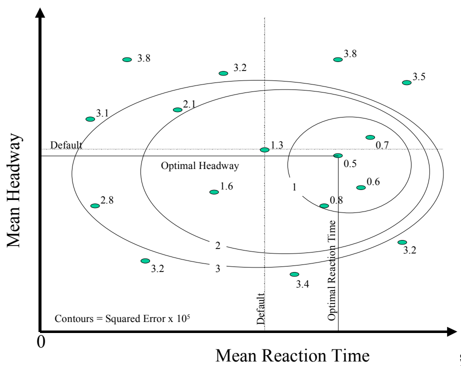
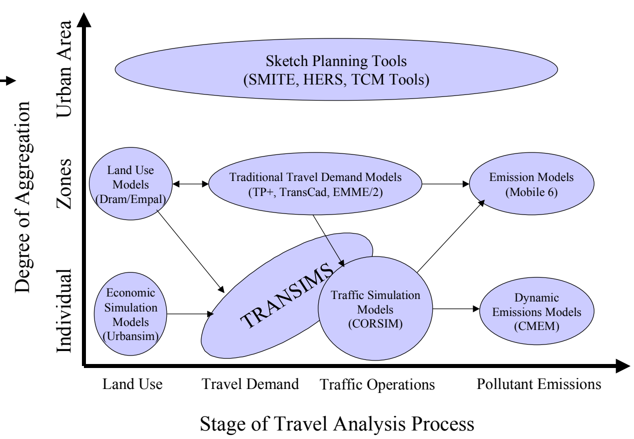
California Department of Transportation

Guidelines for Applying Traffic Microsimulation Modeling Software





September 2002

Richard Dowling, Ph.D., P.E.
Joseph Holland, P.E.
Allen Huang
Dowling Associates
180 Grand Avenue, Suite 250
Oakland, California 94612
510-839-1742
www.dowlinginc.com

Acknowledgements

The authors wish to thank the many people who informally contributed their advice to the authors on various issues related to microsimulation. They include

Mr. John Halkias, Federal Highway Administration
Mr. Gene McHale, Federal Highway Administration
Dr. Adolf D. May, University of California, Berkeley
Dr. Alexander Skabardonis, University of California, Berkeley
Mr. Thomas Bauer, Innovative Transportation Concepts
Mr. John Albeck, Trafficware
Mr. Scott Aitken, Quadstone Ltd.

These people have graciously shared insights from their own research and have pointed the authors to key papers on the topic. Their freely given advice and counsel have made this a better guidebook. However, this acknowledgement is not intended to imply that the above people reviewed or in any way endorsed the views expressed in this guidebook. The authors are the ones solely responsible for any errors in this guidebook.

The following people participated in reviewing the drafts of this guidebook and the authors would like to thank them for their time, efforts, and contributions to this guidebook:

Ms. Mary Rose Repine, California Department of Transportation, TSI
Mr. Steven Hague, California Department of Transportation
Mr. Leo Gallagher, California Department of Transportation, TSI
Mr. Al Arana, California Department of Transportation, Systems Planning
Mr. Rodney Oto, Caltrans District 04
Ms. Diane Jacobs, Caltrans District 07
Ms. Sharri Bender-Ehlert, Caltrans District 06
Mr. Cesar Castaneda, Caltrans District 06
Mr. Carlos Yamzon, Caltrans District 10
Dr. Henry X. Liu, California Partners for Advanced Transit and Highway (PATH)
Dr. Yonnel Gardes, California Partners for Advanced Transit and Highway (PATH)

Table of Contents

List of Exhibits

1 INTRODUCTION

This manual provides guidelines for the application of traffic micro-simulation modeling software to transportation project planning and development.

1.1 Overview

This manual covers the application of commercially available microsimulation software to traffic engineering problems typically encountered by Caltrans personnel. The emphasis is on training Caltrans personnel to recognize when and how to apply microsimulation in combination with or in parallel to other software tools to evaluate the traffic operations of project alternatives.

This manual and its associated training course do not provide training on the actual operation of the microsimulation software (Such training must be obtained from other courses). They focus instead on all of the steps leading up to operation of the software and all of the steps after application of the software.

The manual ranges from task budgeting (identifying when microsimulation is appropriate and the required data and staffing) to assessment of the technical validity of the results (calibration), to techniques for presenting the microsimulation results to management and the general public. The lectures in the associated training course are enhanced with laboratory sessions applying Paramics microsimulation models to real world projects.

The class is a mixture of lecture and "hands-on" computer laboratory working with previously prepared microsimulation model runs. Students will learn:

1. To compare the capabilities, limitations, and data requirements of traffic analysis approaches and to determine which approach is the best for the problem at hand.
2. How to assess the technical validity of their microsimulation models.
3. How to analyze and present the results produced by microsimulation models.

1.2 Required Training Course Facilities

The required facilities for this course are:
1. White board, computer, projector, and screen for instructor.
2. One computer per 2 to 3 students.
3. All computers networked for easy exchange of files.

Each computer must meet the following specifications:
- Windows 2000 operating system Pentium 600 Mz, 128 MB RAM, 100 MB free disk space, graphics card supporting OpenGL and 1280x 1024 screen resolution with 32 Mb RAM, 3-button mouse.

Each computer must have the following software pre-installed:
- Microsoft Office 2000 (WORD, EXCEL, POWERPOINT).

- Paramics Modeller, Processor, and Analyser software (Version 3, Build 7) installed with software key (dongle). Paramics EXCEL Wizard (creates spreadsheets for comparing Paramics output).
- X Windows (Exceed) and OpenGL (Exceed 3D) from Hummingbird Communications Ltd. (30 day demos available at http://www.hummingbird.com/products/exceed).
- Full Shot (screen capture utility)

Participants will find it useful to bring the following materials with them to the class:

- Notepad, pencil, floppy, and hand calculator for the lab problems.

1.3 Training Course Schedule

An approximate schedule for the class is provided below:

DAY ONE – OVERVIEW
The first day of the training course provides an overview of the entire manual. It is designed for modelers and supervisory personnel.

1. Introduction
2. What is Traffic Microsimulation?
3. When Is Traffic Microsimulation Appropriate?
4. Data Preparation

Lunch Break

5. Model Calibration
6. Alternatives Analysis
7. Presenting The Results To The Public
8. Microsimulation Project Management
9. Wrap-Up

DAY TWO – CALIBRATION LABORATORY
This second day of training focuses on gaining practical experience calibrating a microsimulation model for use in evaluating a freeway-to-freeway interchange improvement project. Participants are provided a coded network and asked to run the simulation model and adjust various model parameters until the model is calibrated.

DAY THREE – ALTERNATIVES ANALYSIS AND PRESENTATION LABORATORY
This third day of training focuses on gaining practical experience in the use of microsimulation models to evaluate project improvement alternatives. Participants, in addition, will prepare and give presentations highlighting the results of the analysis.

2 WHAT IS TRAFFIC MICROSIMULATION?

This chapter defines traffic microsimulation, describes traffic microsimulation models in general, and provides an overview of the steps involved in performing a microsimulation analysis.

2.1 Definition

Microsimulation is the dynamic and stochastic[1] modeling of individual vehicle movements within a system of transportation facilities. Each vehicle is moved through the network of transportation facilities on a split second by split second basis according to the physical characteristics of the vehicle (length, maximum acceleration rate, etc.), the fundamental rules of motion (e.g. acceleration times time equals velocity, velocity times time equals distance) and rules of driver behavior (car following rules, lane changing rules, etc.).

Examples of microsimulation software are: Aimsum, CORSIM, Paramics, Simtraffic, Transmodeller, VISSIM, WATSIM, etc[2]. They stochastically model individual vehicle movements as a function of time and space.

Examples of simulation software that are NOT microscopic: FREQ, PASSER, and TRANSYT7F. These tools are also designed to simulate traffic operations but they do it at the macroscopic level. They are deterministic[3] models that model the movement of groups of vehicles or the average behavior of all vehicles on a given section of facility for a given time period.

Additional definitions of microsimulation model terminology are provided in Exhibit 1. The exhibit presumes knowledge of general traffic engineering terms and focuses only on terms that are either specific to microsimulation or have definitions in the microsimulation environment that vary from common traffic engineering usage.

[1] Dynamic means that transitions and changes in conditions are modeled during the analysis period, rather than assuming a single steady state condition throughout the analysis period. Stochastic means that the predicted system performance is subject to random variations (The same input may result in slightly different results when a different random number seed is used).

[2] The listed software may be a registered trademark of their respective owner.

[3] Deterministic models always produce the identical output when given the same input.

Exhibit 1. Microsimulation Terms

Input Terms	Definition
Conditional Turn	The probability of a vehicle making a specific turn at a downstream intersection is influenced by (conditioned on) the movement made by the vehicle at the upstream intersection.
Driver Aggressiveness	A characterization of the degree to which drivers respond to traffic flow conditions. The characteristics of aggressiveness vary by software.
Driver Awareness	A characterization of the degree of driver perception of road conditions. The characteristics of awareness vary by software.
Driver Cooperation	A characterization of the degree to which drivers will forgo individual advantage and modify their driving behavior to assist other drivers in the traffic stream. Varies by software.
Free-Flow Speed	The mean speed of traffic at low flow conditions, where the flow rate has no significant effect on actual vehicle speeds. This can be, but is not usually, the speed limit.
Incident	A vehicle breakdown, accident, or other event that causes full or partial obstruction of vehicle movements in a lane during the simulation period.
Interchange	May be a freeway interchange or may be a pair of street intersections (no freeway) for which the user wishes to specify the origin-destination travel patterns.
Link	A section of street or highway where physical characteristics (e.g. lanes, grade, width, design speed, etc.) are generally (not absolutely) similar over the length of the section.
Node	The point where two links connect. Often an intersection of two or more streets.
Source/Sink	One or more points or links on the network where vehicles are created (source) or destroyed (sink) by the software during the simulation.
Zone	A subarea of the network where vehicles are assumed to be created or destroyed. A zone may be associated with one or more source/sink points or links.

Process Terms	Definition
Acceptable Gap	The minimum gap between vehicles that is accepted by the median driver.
Calibration	Calibration is a mathematical process to identify the global and link specific parameters for driver behavior and vehicle operation that cause the simulation model to best reproduce observed real-world behavior for local conditions. Calibration is performed locally by the analyst for each individual application of the simulation model to a real world road network..
Car Following Model	One or more equations that specify how a following vehicle adjusts its speed in relation to the lead vehicles
Gap	The time or distance between the tail end of the leading vehicle and the front end of the following vehicle.
Headway	The definition of headway varies by software product and researcher. Headway is usually the time between when the front end of the leading vehicle and the front end of the following vehicle pass a given point. Some software define it as the time between the back end of the lead vehicle and the front end of the following vehicle. Headway is sometimes defined on a distance basis (feet) rather than on a time basis (seconds).
Lane Changing Model	One or more equations that specify the likelihood that a vehicle will change lanes. Lane changes may be Mandatory, Discretionary, or Anticipatory. Mandatory lane changes are forced by changes in street geometry and signing/striping. Discretionary lane changes are optional, at the driver's discretion. Anticipatory lane changes are discretionary lane changes made in anticipation of a downstream traffic flow condition.
Seed	A starting series of digits used by the random number generator to generate a sequence of numbers used in the simulation process for random numbers. Any given random number generator will generate the same sequence of "random" numbers from the same seed.
Simulation Period	The length of time for which a simulation is performed.
Time Slice	A length of time into which the simulation period is divided. For example a one hour simulation period may be divided into four 15-minute long time slices.
Time Step	The length of time between computations of vehicle position and speed during the simulation.

Validation	Validation is used in the literature for two distinct phases of model acceptance testing. During the software development phase, validation is one or more tests of the ability of the theoretical equations and rules in the simulation model to imitate real world driver behavior. Validation during this phase is performed by researchers during the development of micro-simulation models and their associated software. Later, when the analyst is coding a specific local network using the software, validation is the step that follows model calibration. Specific components of the coded model for the local network are first calibrated against a detailed set of field data. The entire coded model is then validated against more global data for the local network during the validation step.
Verification	Verification is one or more tests to assure the fidelity of the software to the driver behavior theory. Verification is performed by software programmers and independent certification agencies.
Warm-up Period	The length of time between the start of the simulation and some criterion for stability is reached. For example, if the criterion of stability is the number of vehicles in the system, then the warm up time is the length of time that needs to be simulated before the number of vehicles in the system during each time step stops increasing.

Output Terms	Definition
Animation	The visual display of the movement of vehicles output by simulation software.
Capacity	The maximum sustainable discharge rate for traffic past a given point for a specific time period and specific geometric conditions. Typically a one hour flow rate.
Delay:	The Year 2000 Highway Capacity Manual defines delay as "The additional travel time experienced by a driver, passenger, or pedestrian". Unfortunately the HCM does not define the yardstick against which the "additional travel time" is measured in order to determine delay. So the various microsimulation software define and measure "delay" differently. For some, delay is the difference between the actual travel time for a link and the theoretical travel time at the coded free-flow speed for the link. For others, delay is the difference between the actual link travel time for each vehicle and the theoretical travel time if the vehicle had traversed the link at its desired speed (which can be different than the link free-flow speed).
Density	The mean number of vehicles present on a link during a simulation run, divided by the product of the number of lanes and the length of the link.
Occupancy	The percentage of time that a vehicle is present within the detection range of a detector. Computed by summing the amount of time that vehicles are present in the detector and dividing that by the product of the total simulation time and number of vehicles simulated.
Queue	The Year 2000 Highway Capacity Manual defines a Queue as: "A line of vehicles, bicycles, or persons waiting to be served by the system in which the flow rate from the front of the queue determines the average speed within the queue." Some software allow the user to set the speed limit below which a vehicle is considered to be queued, while others fix the threshold. Some software require the user to set a maximum vehicle spacing as well as a maximum vehicle speed for vehicles to be considered in a queue. The queue may be measured in terms of number of vehicles or length to back of queue. Simulation software generally limit the reported queue length for any individual link to the length of the link. The portions of the queues that overflow a link are reported as being present on the upstream link.
Reports	The text file outputs produced by simulation software.
Speed	The sum of the products of the number of vehicles on the link times the length traveled, divided by the sum of the vehicle travel times on the link.
Stops	The number of deceleration events to full stop or near full stop during the simulation period. The definition of "stop" may include vehicles not actually stopped, but moving at some low speed, such as 3 feet per second. Some software will count all stops by all vehicles regardless of how often each vehicle stops. Other software will count only one stopping event per vehicle per link. Thus some software will count a vehicle moving up in a queue as stopping several times, while other software will define the vehicle as stopping only once on that link.
Volume	Mean flow rate of vehicles. May be expressed as the average number of vehicles averaged over one hour, or the number vehicles per simulation period.

2.2 General Characteristics of Microsimulation Models

The typical microsimulation software implements a combination of procedures (models) for identifying the location, speed, and acceleration rate of vehicles in the highway network at each moment of time. A set of relatively simple rules is used to move the vehicles through the network. Statistics are tabulated on the vehicle activity and two outputs are typically produced: text reports, and visual animations.

2.2.1 Time Steps

The simulation of vehicle movements is done in a series of time steps. The vehicle position, velocity, and rate of acceleration/deceleration are computed at the end of each time step and statistics accumulated on the results.

The number of time steps per second influences both the accuracy and duration of a microsimulation model run.

The more time steps per second, the greater the potential precision of the model results. A model that computes the vehicle position only once every second will not be as precise as a model that has several time steps per second.

This precision becomes more important for freeway analysis, when vehicles travel at higher speeds. A vehicle moving at 30 mph on a city street travels 44 feet (2.5 car lengths) in one second. A vehicle moving at 60 mph on a freeway moves 88 feet (5 car lengths) in one second. If the time step resolution is only 1 time step per second, then a vehicle moving at 60 mph cannot safely follow another vehicle by closer than 5 car lengths and avoid a collision if the lead vehicle decides to slow down between one time step and the next.

It is also possible for the car to pass over the detector within a one second period, and unless the software is smart enough to deduce that the vehicle must have crossed the detector between time steps, the vehicle will not be detected by the simulation software (some software will only detect a vehicle if it is actually present on the detector at the end of a time step).

Time step frequencies of one step per second are not considered a very fine level of resolution for micro-simulation, and are therefore, not recommended.

However, the more steps there are per second the more computations required to perform a given time period of simulation. Doubling the number of time steps per second doubles the computations.

It is also possible to go too far in the search for increasing precision. Some driver behavior models require knowledge of historical conditions (what was the lead vehicle's speed 3 time steps before the current time step) to assess drivers' reactions. If the analyst makes the time steps exceedingly short, this may bias the computations.

Once the model has been calibrated, the number of time steps per second should not be changed. Changing the time steps per second will affect the simulation results.

2.2.2 Randomization

Micro-simulation models would produce unrealistically regimented simulations with all drivers moving at the same time and in the same way, if it were not for randomization. The simple rules used to move vehicles in a microsimulation do not realistically reproduce the wide range of human behavior observed in the real world. Random variables are used to produce a plausible range of human behavior from the simple rules.

Computer software uses a random number generator to generate the necessary set of random variables. The generator requires a starting number, or "seed" to produce a unique sequence of numbers. The same seed, used with the same generating routine, on the same computer will produce the same sequence of numbers for use in the random variables, every time.

Thus, a single microsimulation model run is like rolling the dice only once. In order to find out the average conditions it is necessary to operate the microsimulation model several times, with different random number seeds and then average the results of the different runs. A discussion of the required number of runs to estimate mean and 95 percentile results is provided in the Alternatives Analysis chapter.

2.2.3 Vehicle Generation

At the start of every time step in the simulation, the microsimulation software makes a decision about whether or not to release one or more vehicles onto the road system. Vehicles may be released from specified entry and exit points on the network or they may be released from zones that span several links. The decision to release is a random decision with an expected value specified by the mean trip generation rate (vehicles/time step) for the zone. The random number generator is used to decide whether or not a vehicle is released. Note that this process does not guarantee that the final number of vehicles released onto the network will actually match the value coded by the analyst. The average will be close to the input value, but not identical.

At the time the vehicle is generated it is randomly assigned a vehicle type, and a driver type (level of aggressiveness and awareness) based upon vehicle mix and driver mix percentages provided by the user. The vehicle type determines its length, weight, width, height, maximum speed, maximum acceleration rate and the braking rates.

If the software allows for the input of vehicle origin-destination tables, the vehicle will also be assigned a destination, also using a random number generator. The probability of a vehicle being assigned a particular destination is specified by the origin-destination table.

2.2.4 Path Choice

If the software provides for the input of origin-destination tables, then a path (series of connected links leading to the assigned destination) will be computed. This path can be updated dynamically during the simulation run as the vehicle moves along the network and congestion conditions change on the network.

Various methods are available for determining the path chosen by the vehicle including: Shortest Path, Stochastic, or Dynamic[4].

If the software does not use an origin-destination table then the vehicle's ultimate destination and path will be determined as it moves down the network according to the intersection turn probabilities coded by the analyst. The vehicle will be randomly assigned to each turn movement at each intersection, making it possible for individual vehicles to drive in circles, unless the user over-rides this option.

2.2.5 Vehicle Movement Rules on Links

A link is a section of street where street geometry and demand are sufficiently constant so that the section of street can be modeled as a pipe, with vehicles entering the pipe at one end and leaving at the other end. There are three basic vehicle movement rules on a link: single vehicle, car following, and lane changing. For most all models, the specifics of vehicle behavior will vary a bit between freeway links and non-freeway links. Driver awareness and aggressiveness will modify vehicle behavior as well.

Single Vehicle on Link

In the absence of any other vehicles, the microsimulation software will move the vehicle down the link at its desired speed. The desired speed of the vehicle is determined by the analyst coded free-flow speed for the link and the driver type (aggressiveness). More aggressive drivers may go faster than the coded free-flow speed, less aggressive drivers may go slower. Some software may insert random accelerations and decelerations into the vehicle speed as it travels down the link. Others will leave the speed fixed for the length of the link, in the absence of other vehicles or obstructions in the downstream link.

Car Following Rule

When a vehicle catches up to another vehicle on the link, a car following rule and a lane changing rule are used to decide how the vehicle will respond to the lead vehicle. The software will employ a mean target following headway which may be varied according to driver aggressiveness and vehicle type. The definition of headway does vary by software. All software generally set their car-following rules to guarantee that vehicles will not collide.

Lane Changing Rule

Lane changing requires that an acceptable gap be available in the traffic stream in the adjacent lane. The lane changing rules will vary according to the situation. Mandatory lane changes are made when the road geometry dictates the change (such as a designated turn lane, or a lane drop). Discretionary lane changes are made to pass slower moving vehicles. The probability of a vehicle making a discretionary lane change is usually a function of the difference in speed between the two vehicles and the available gaps between vehicles in the adjacent lane. Anticipatory lane changes may be made to pre-position the vehicle for an upcoming turn, or to avoid slower vehicles entering the freeway on an on-ramp. The upstream decision point where

[4] Shortest Path sends all drivers on the route with the shortest path. Stochastic Route Choice splits the drivers between alternate routes assuming random variations in the perceived travel time or each route. Dynamic Route Choice assumes that drivers will change their routes as traffic conditions change during the simulation period.

vehicles start to make anticipatory lane changes (sometimes called the signpost or warning sign location) is often difficult to model realistically in today's simulation software. Some software provide for the vehicle to look ahead a set number of links (which can become a problem when numerous short links are coded). Other software may employ "path-based" pre-positioning that is not constrained by the number of links.

2.2.6 Vehicle Movement Rules Within the Intersection (at the Node)

Gap acceptance is used to move left turning vehicles (with permitted phasing) through opposing through vehicles, or stop sign controlled vehicles through the other vehicle streams. Each vehicle movement at an intersection is assigned a priority ranking. Lower priority vehicle streams defer to higher priority movements.

No crosscheck is typically made by most simulation software of vehicle movement inside of the intersection (once it has accepted a gap) to ensure that two vehicles do not occupy the same space at the same point in time (an apparent collision). If the analyst were to code two conflicting green phases, the simulation software may well show two streams of traffic driving through each other.

The vehicle speed may be slowed inside an intersection while it is making a turn. The reduced speed may be fixed by turn movement type, or, in some software, it may vary according to the radius and angle of the turn.

2.3 Typical Microsimulation Analysis Steps

The major steps involved in a microsimulation analysis are:

1. Identification of Project Purpose, Scope, and Approach
   This scoping step is critical for determining the ultimate cost and schedule for the microsimulation analysis.

2. Data Collection
   This step involves the collection of input data for the microsimulation model as well selected output data for calibrating the model.

3. Coding
   The model coding step is where the analyst converts the field data into inputs for the microsimulation model.

4. Error Checking
   The error-checking step verifies the accuracy of the coded input data.

5. Calibration
   Calibration is where the analyst adjusts the default parameters in the standard behavioral models contained in the microsimulation software to local conditions.

6. Alternatives Testing
   This step is the purpose for which the microsimulation model was developed.

7. Documentation
   Documentation provides information on the inputs to the model, the validity of the model, and the results of the alternatives analysis.

8. Presentation of the Results
   This step is where the analyst presents the microsimulation analysis results to decision makers and the general public.

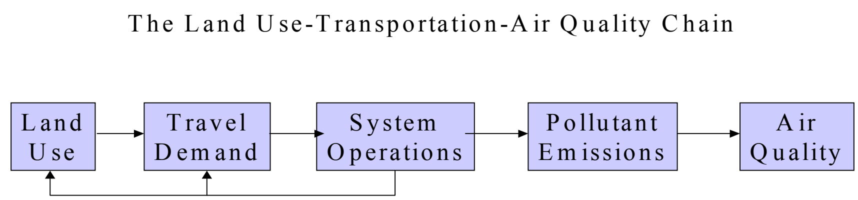
3 WHEN IS TRAFFIC MICROSIMULATION APPROPRIATE?

This chapter compares traffic microsimulation to other traffic analysis tools, and identifies the conditions when traffic microsimulation analysis is most appropriate.

3.1 The Spectrum of Traffic Analysis Tools

The available traffic analysis tools can be classified according to the portion (or portions) of the total traffic analysis process that each method is designed to address. The entire process can be divided into 5 major analytical steps (shown diagrammatically below): land use forecasting, travel demand estimation, transport system operations analysis, emissions estimation, and air quality forecasting.

Exhibit 2. The Traffic Analysis Process



Each analytical step in the process can be considered a "link" in the analysis chain. When all the links are completed and connected we have a comprehensive and complete procedure for analyzing the traffic operation and pollutant emission impacts of any policy or investment option. Each link in the traffic analysis chain is defined as follows:

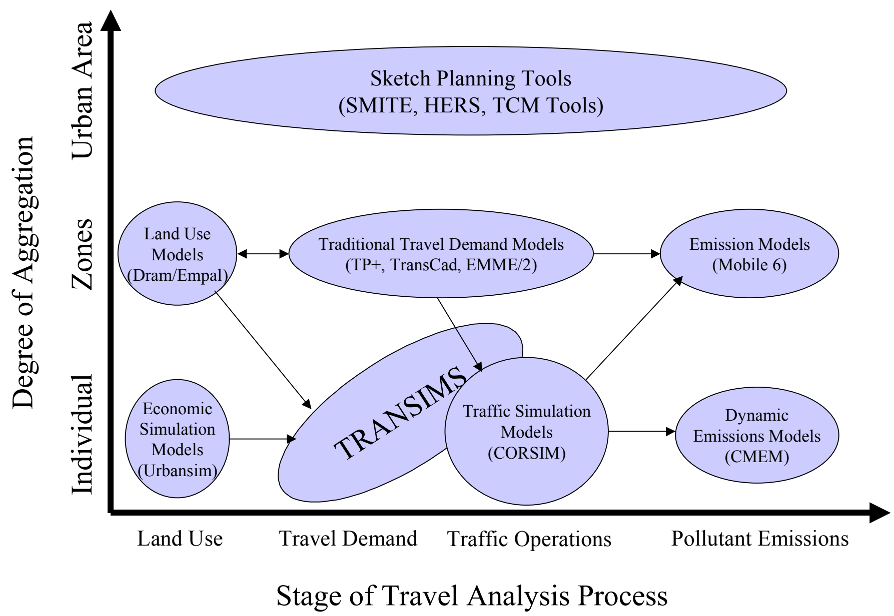
- The land use step forecasts growth in population, demographic and socio-economic changes, and spatially allocates people, households, and commercial activity within the air basin.
- The travel demand step converts the locational data generated by the land use model into estimates of travel activity.
- The systems operation step estimates the impacts of the forecasted travel activity on the operation of the region's transportation system. This step predicts travel time, speed, delay, and vehicle modal activity. The travel times predicted in this step influence the prior steps; land use and travel demand.
- The pollutant emissions step uses the vehicle modal activity data to predict emissions, which are in turn fed into an analysis of the basin's air quality.

A second dimension of this typology is the level of aggregation at which each analysis method is applied. The levels of aggregation are: urban area-wide, traffic analysis zones, and the level of the individual or household. They are explained below:

- The area-wide level of aggregation is typical of sketch planning models. These models and the methodologies behind them are designed to require and produce only basin-wide averages of vehicle-miles traveled, delay, and emissions.
- The traffic analysis zones level of aggregation is an intermediate level of aggregation typical of most all transportation forecasting models in the United States. Households and commercial activity are aggregated into geographic units, or zones. The real world transportation system is represented by a subset of key facilities and coded as "links". The models work with and produce results that reflect averages for each zone and link. An air basin is typically split into no more than 1,500 geographic analysis zones, which are often aggregates of census tracts (a few regions with GIS capabilities store their socio-economic data at a higher level of disaggregation, but travel models rarely can employ that full level of detail).
- The individual or household level is the disaggregate level of analysis. Each household or each person within the household is evaluated separately and the results summed to obtain estimates of aggregate behavior. In most cases the household level behavior forecasts are made for only a random sample of households in the region and the results expanded by a factoring process to represent all households in the region.

Exhibit 3 illustrates this two-dimensional typology of transportation analysis tools. One or more examples are provided for each group of analysis tools. These examples are not the only or the best available tools for each group. The size of the chart did not permit a listing of all of the tools available. The tools at the top of the diagram are the most aggregated methods requiring the least amount of input data and computation time. The tools in the middle are typical of current practice. The tools at the bottom of the diagram are the most detailed and disaggregate tools available

Exhibit 3. The Spectrum of Traffic Analysis Tools



Traffic microsimulation falls at the bottom of the chart. It is the most detailed and data intensive of the available tools for analyzing traffic operations. CORSIM is one example of microsimulation tools.

3.1.1 Traffic Operations Analysis Tools

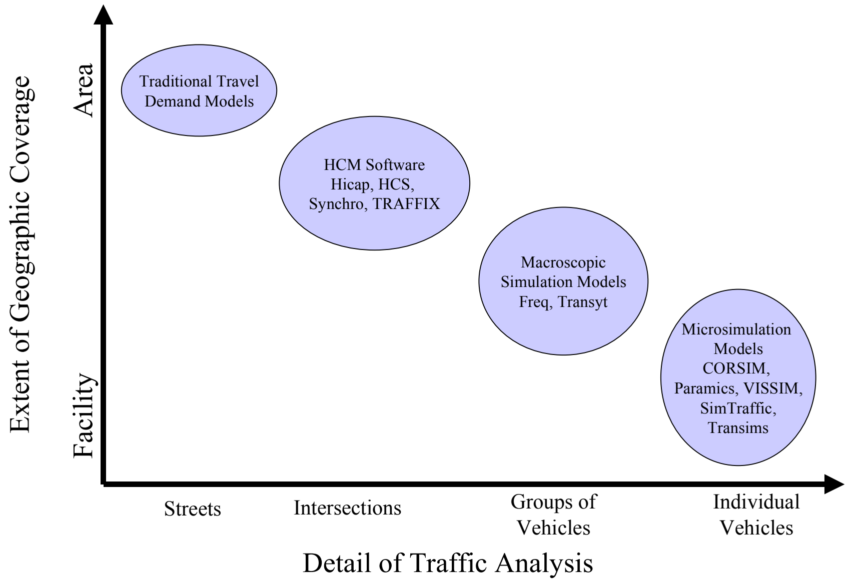
The previous exhibit showed how traffic operations analysis, and specifically traffic simulation models fit into the entire travel analysis process. The exhibit below focuses in on the traffic operations analysis tools in specific and shows how their geographic coverage varies with their degree of aggregation. The more aggregate the analysis tool, the broader the geographic coverage that can be accommodated by the tool. Note that many of the tools overlap each other in capabilities. The exhibit highlights the differences in the general capabilities of each tool.

Planning Models: Planning models are the most aggregate of the available traffic operations analysis tools. They are designed to cover large geographic areas such as entire urban areas. They contain sophisticated demand models but use relatively simplistic traffic operations models. They do not identify queues and cannot accurately identify bottlenecks. They are not good tools for highway design, but are useful for general sizing of facilities.

Highway Capacity Manual Models: Models based on the analytical techniques contained in the Highway Capacity Manual (HCM) are designed for operations analysis of isolated segments and points of transport system. Some system analysis is also possible. Demand is assumed to be a given. These models are good for deciding if the facility has sufficient capacity, but are less accurate at predicting the extent of congestion and queuing that may occur when demand exceeds capacity. These models fall between the traditional travel demand model and macroscopic simulation models in detail.

Macroscopic Simulation Models: Macroscopic simulation models are designed for operations analyses of systems of road facilities. They can identify optimal control strategies. Like Planning and HCM models, Macroscopic simulation models are deterministic. They provide only a single answer to any question. They fall between the HCM methods and microscopic simulation models in terms of their detail.

Exhibit 4. Traffic Operations Analysis Tools



Microscopic Simulation Models: Microscopic simulation models are designed for operations analyses of systems of road facilities. They address theoretical gaps in macroscopic models. Microsimulation models, however, are not designed for optimization of control strategies. These models show time-dependent operations results. They incorporate randomness. Instead of a single right answer, the user must deal with averages and confidence intervals.

3.2 When Is Microsimulation The Best Approach?

Microsimulation takes more data and staff resources than macroscopic simulation, and HCM type analyses. The analyst should employ only the level of effort required by the problem being studied.

If the average performance of the highway system is sufficient for the problem being studied, then the analytic approaches contained in HCM are usually adequate.
If conditions are very close to breakdown, then the greater accuracy of microsimulation may be required. For situations where the projected demand is greater than capacity, there may be the need for the better performance statistics produced by microsimulation.

3.2.1 Examples of Applications Best Suited for Microsimulation

There are several situations where microsimulation is the best technical approach for performing a traffic analysis.

Conditions that violate one or more basic assumptions of independence required by HCM models

Most of the HCM models assume that the operation of one intersection or road segment is not adversely affected by conditions on the adjacent roadway. Long queues from one location interfering with another location would violate this assumption. If any of the following conditions are present in the project study area and it is important that they be modeled accurately, then microsimulation will be the most appropriate analytical tool.

- Queues spill back from one intersection to another
- Queues overflow turn pockets
- Queues from city streets back up onto freeway
- Queues from ramp meters back up onto city streets

Conditions not covered well by available HCM models

There are several real-world situations not covered well by available HCM models:

- Multi-lane or two lane rural roads with traffic signals or stop signs.
- Truck climbing lanes.
- Short through lane adds or drops at a signal.
- Boundary points between different signal systems operating at different cycle lengths.
- Signal pre-emption (railroad crossings, fire stations, etc.).
- HOV Lane entry options, design options for starting or ending an HOV lane.
- Two-Way Left Turn lanes (however, at present, no commercially available microsimulation software can model this).
- Roundabouts.
- Tight Diamond Interchanges.
- Incident management options (since HCM and macroscopic models assume a steady state condition within each analysis period, they are not well suited to accurately track the build-up and dissipation of congestion related to random transitory conditions caused by incidents.).

Choosing Among Alternatives, None of Which Eliminate Congestion

Analytical HCM techniques are not very good at distinguishing between different levels of congestion (level of service “F”). They often assume no congestion at the start of the analysis period and do not consider congestion beyond the end of it. They cannot distinguish well between queues that interfere with facility operations and queues that do not. However, it is still tricky to correctly assess the statistics produced for super congested conditions, even when applying microsimulation models. Microsimulation models do NOT tally delay statistics for vehicles that are physically unable to enter the network.

Testing Options that Change Vehicle Characteristics and Driver Behavior

Microsimulation models, since they are sensitive to different vehicle performance characteristics and differing driver behavior characteristics, are ideal for testing ITS strategies designed to modify these characteristics. Driver information systems, automated vehicle guidance, triple or quadruple trailer options, new weight limits, new propulsion technologies etcetera, are all excellent candidates for testing with microsimulation models. HCM and other macroscopic models are not designed to be sensitive to new technology options, while microsimulation allows one to predict what the effect of new technology might be on capacity before it is actually in place.

3.3 Criteria for Evaluating/Selecting Software

This section suggests criteria for evaluating and selecting software for microsimulation. The basic criteria are: Technical Capabilities, Input/Output/Interfaces, User Training/Support, and On-going Software Enhancements.

3.3.1 Technical Capabilities

Is the software capable of handling the size of problems that you plan to deal with? Is it sensitive to the variables of concern to the study? Are the technical analysis procedures incorporated in the model state-of-the-art? A suggested checklist of technical capabilities:

1. Maximum Network Size
2. Car Following Logic
3. Lane Changing Logic
4. Sensitivity to Grades
5. Sensitivity to Horizontal Curvature
6. Sensitivity to Advanced Traffic Management Techniques
7. Available Global Calibration Parameters
8. Available Link Specific Calibration Parameters

3.3.2 Input/Output and Interfaces with Other Software

How difficult is it to input and edit data for the program? Can the data be imported from other software data files?

What standard output reports are available? What are the user's options for customizing output reports?

With what other software can the microsimulation software interface? Does it Import/Export to Popular Demand Model Software (Emme2, etc.)? Does it Import/Export to Popular Database/Spreadsheet Software (ACCESS, EXCEL, etc.)? Does it interface With Popular HCM Analysis Software (SYNCHRO, etc.)?

3.3.3 User Support and Training Requirements

What kind of training and support does the vendor provide? How much does it cost? How close is the vendor? Can you visit their office for direct help? Are there other users nearby that you can consult with?

How long will it take to learn to operate the program? What basic training is required of potential users before they can be trained on the specific software?

3.3.4 On-Going Software Enhancements

Generally, if the agency is going to invest the time and effort to train their people and set-up the necessary support structure, it is desirable that this investment have as long a payout period as possible. Thus, when selecting software, it is desirable if possible to avoid software that has been discontinued by the software vendor. It is certainly necessary to have software that keeps up with changes in computer operating systems so that the software can be run on new computers as they are acquired.

At the same time, one does not want to acquire software too early in the development process, when it is still undergoing numerous version changes to correct bugs.

3.3.5 Other Criteria

Shaw and Nam used the following criteria for evaluating microsimulation software.

1 **Network Size Limit:** Scale/extent of the network that can be modeled (in technical terms, the maximum number of links and nodes that can be accommodated in the model).
2 **Network Representation:** Extent to which the model can accurately represent the geometry of the road system.
3 **Traffic Flow Representation:** Number of traffic flow algorithms embedded in the model, such as car-following logic, lane-changing logic, and vehicle generation logic.
4 **Detail of Output:** Amount of information about network performance (measures of effectiveness) available after simulation, without post-processing the data.
5 **Network Merge:** Ability to combine separate areas into a single model and conversely, to divide an existing model into sub-models. A model with merge capability increases the efficiency of staffing utilization, since it allows various modelers to combine their work.
6 **3-D Modeling:** Ability of the model to represent and display vertical elements of roadway geometry, such as steep terrain and grade-separated interchanges.
7 **Traffic Composition:** Extent to which the model gives the user control over local traffic characteristics such as fleet composition, vehicle dimensions, and truck percentage.
8 **Animation:** Quality of the model's visual representation of traffic conditions.
9 **Input Data Requirements:** Amount of input data required to develop, calibrate, and validate the model successfully.
10 **Network Coding/Editing:** Amount of time and effort required to enter roadway geometry, traffic control, and other roadway characteristics into the model; make changes in these characteristics; and quickly create scenarios representing potential alternatives.
11 **Input/Output Review:** Extent to which the model can easily display various input/output data graphically for the purpose of network review and analysis.
12 **VISTA (GIS) Interface:** Anticipated amount of effort required to share data from the model with "VISTA", the GIS-based tools the FSOA project will use to overlay various roadway data.
13 **Economic Analysis Interface:** Anticipated amount of effort required to share data from the model with engineering economic analysis (benefit/cost) software.
14 **Incident Management Analysis:** Potential ability of the model to simulate real-time adjustments in vehicle routing resulting from incidents.
15 **Actuated Signal Control Devices:** Potential ability of the software to model semi-actuated, fully-actuated and interconnected traffic signal systems and ramp meters.
16 **User-Defined Traffic Control (API):** Potential ability of the software to model user-defined traffic control devices such as variable message signs. Also includes ability to import real-time traffic data from a Traffic Operations Center and ability to accommodate custom-developed software modules through macros and/or an Application Programming Interface (API).
17 **Public Transportation:** Ability of the program to model the interaction between road traffic and public transportation modes such as bus and rail.
18 **Calibration/Validation Results:** Ability of the model to meet (with reasonable effort) the Validation Acceptability Guidelines contained in the UK Highways Agency *Design Manual for Roads & Bridges (9*).
19 **Program Integrity:** Overall quality of the software engineering, including freedom from program crashes and freedom from unexpected, illegal, or illogical behavior in the modeled traffic.
20 **Technical Support:** Quality, accessibility, timeliness, and accuracy of the vendor when handling technical support questions from the model evaluation team.
21 **Documentation:** Quality, readability, reliability, and usefulness of the program's technical manual, error messages, on-screen help files, and web-based user assistance documents.
22 **Record of Large-Scale Freeway Applications:** Sufficient number of other transportation agencies and consultants that are using the program to model large urban freeway networks.
23 **Software Cost per Cop**y: Number of dollars charged for the first single license

Source: John W. Shaw, Do H. Nam; "Microsimulation, Freeway System Operational Assessment, and Project Selection in Southeastern Wisconsin: Expanding the Vision", Paper presented at Transportation Research Board Annual Meeting, Washington, D.C. 2002.

3.4 Discussion Session: Selection of Traffic Modeling Approach

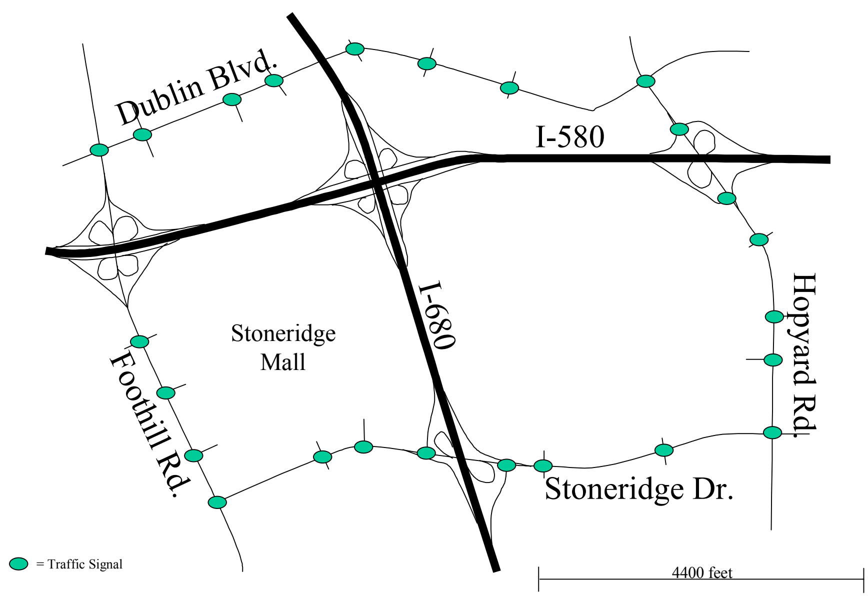
This discussion will walk through the decision making process involved in selecting the appropriate traffic analysis approach for each stage of a freeway interchange improvement project.

This case study involves the planning, project development process for a project to relieve on-going traffic congestion at the interchange of the I-580 and I-680 freeways in Pleasanton, California. The interchange is a full cloverleaf interchange with collector-distributor roads (see exhibit below). The demands for the southbound to eastbound and northbound to eastbound movements routinely exceed the available capacity of the collector distributor road and the freeway ramp merge during the morning and evening peak hours.

We will walk through each stage of the planning and project development process and identify the appropriate traffic modeling approach for each stage.

The purpose of this discussion is to identify the appropriate traffic analysis approach (planning model like Emme2, HCM analysis tool like HCS, macroscopic simulation like FREQ, or microscopic simulation tool like CORSIM) for performing the traffic analyses at each stage of the project development process.

Exhibit 5. I-680/I-580 Interchange Project Study Area



3.4.1 System Planning Documents

What types of traffic analyses are required for the development of a District Strategic Plan, a Route Concept Report, a System Development Program, and an Inter-Regional Road System Plan? What level of detail and precision is required? What resources are available? Which analysis tools are best for this type of traffic analysis?

3.4.2 Project Planning Concept and Scope

What types of traffic analyses are required for the development of a project planning concept and scope? What level of detail and precision is required? What resources are available? How long can you wait for the answers? Which analysis tools are best for this type of traffic analysis?

3.4.3 Programming

Is any traffic analysis required for programming?

3.4.4 Project Study Report (PSR):

What types of traffic analyses are required for the initial engineering studies and alternatives analyses that go into a PSR? What level of detail and precision is required? What resources are available? How long can you wait for the answers? Which analysis tools are best for this type of traffic analysis?

3.4.5 Project Report (PR)

What types of traffic analyses are required for the engineering studies and environmental analyses that go into a PSR? What level of detail and precision is required? What resources are available? Which analysis tools are best for this type of traffic analysis?

3.4.6 Preliminary and Final Design

Are any additional traffic analyses needed for preliminary and final design beyond those already completed for the Project Report? Do some studies need to be updated?

4 DATA PREPARATION

This chapter provides guidance on the identification, collection, and preparation of the data sets needed to develop a microsimulation model for a specific project analysis, and the data needed to evaluate the calibration and fidelity of the model to ‘real-world’ conditions present in the project analysis study area.

4.1 Required Data

Microsimulation models require the following input data:

1. Geometry (lengths, lanes, curvature)
2. Controls (signal timing, signs)
3. Existing Demands (turn volumes, OD table)
4. Calibration Data (performance data: speeds, queues)
5. Future Demands (turn volumes, OD table)

4.2 Geometric Data

Geometric data consists of the number of lanes, width, length, grade, design speed, and horizontal curvature of street segments. For intersections, the necessary geometric data also includes the angle of intersection, the designated turn lanes and their vehicle storage lengths, and curb return radii. This data can usually be obtained from construction drawings (As-Built’s) or aerial photos. The data should be accurate to the nearest foot. For large geographic areas or areas with widely varying elevations, aerial photos may have distortion in them that should be corrected with ortho-rectification.

Load and height limit data may also be required by some software. Vehicles that exceed these limits will not be routed on these road segments.
For some software the horizontal curvature must be accounted for with a series of short straight links with free-flow speeds reduced to account for the effect of the curvature.

So far, no software employs vertical curvature information. Vertical curves must be converted to a mean straight-line grade. The ability of software to account for the effects of long grades on truck operation varies by software package. The analyst should review the software documentation before investing a great deal of time coding vertical grade information. A reduced free-flow speed for some vehicle types may be used to account for grade effects.

4.3 Control Data

Control data can best be obtained from the files of the agencies operating the traffic controls (for traffic signals and ramp meters), or from field inspection (for signs, although some agencies maintain excellent sign inventories). Windshield survey photo-logs are also an excellent source of sign and striping information, if current.

4.4 Existing Demand Data

Demand data for existing conditions can best be obtained from traffic counts obtained from manual or automatic count stations. The Freeway Performance Measurement System (PeMS) is an excellent source of count and speed data for California freeways. The Highway Performance Monitoring System (HPMS) may be a good source of point specific demand data for existing conditions on conventional highways in rural areas.

4.4.1 Count Locations and Duration

Traffic counts should be conducted at key locations within the microsimulation model study area for the duration of the proposed simulation period.
If the simulation is for the afternoon peak period, then the counts should cover the entire peak period starting from before congestion starts on any section of the study area and lasting until beyond the end of any congestion in the study area. Often, available resources (and lack of complete information on the start and end of congestion) will limit the data collection to a two to three hour period without any assurance that there is no lingering congestion before or after the end of the count period.

The counts should be conducted simultaneously if resources permit so that all count information is consistent with a single simulation period. Often, resources do not permit this for the larger simulation areas, so the analyst must establish one or more control stations where a continuous count is maintained over the length of the data collection period. The analyst then uses the control station counts to adjust the counts collected over several days into a single, consistent set of counts representative of a single day within the study area.

4.4.2 Estimating Origin-Destination Tables

For some simulation models (Paramics, for example) the counts must be converted into an estimate of existing origin-destination patterns. VISSIM can work with either turn movement counts or an origin-destination (OD) table, however; the OD table is required if route choice shifts are to be modeled.

Local regional transportation planning agency (RTPA) travel demand models are generally <u>not a good source of existing demand</u> data because their data sets are generally limited to the nearest decennial Census year and their zone system is usually too macroscopic for microsimulation. The analyst must usually estimate the existing OD table from other data sources.

A license plate matching survey is the best and most accurate method for estimating an OD table. The analyst establishes checkpoints within and on the periphery of the study area and notes down the license plate numbers of all vehicles passing by each checkpoint. A matching program then is used to find out how many vehicles traveled between each pair of checkpoints.

License plate surveys, however, are quite expensive (sometimes requiring specialized high speed video cameras to record license plates at high speed locations). For this reason, most analysts estimate the OD table from less expensive traffic counts.

The fundamental problem of estimating OD tables from traffic counts is that there are mathematically an infinite number of OD tables that can match any given set of counts (even if counts are available for 100% of the street segments in the study area). The OD table has more degrees of freedom then there are street links to constrain it. So any number of OD tables can be created that fit the counts.

The solution is to make an additional assumption to constrain the solution to a single OD table. This is usually done by starting with a "seed" OD table and then modifying it as little as possible until an assignment of the OD table to the street network matches the traffic counts.

For the small study areas (under 5 miles in length) typically used in microsimulation studies it is safe to assume that relative distances have little effect on trip distribution. The traditional gravity model used in regional transportation demand models then collapses to a simple proportional model. Thus for the "seed" table, the analyst can use the model below which assumes that the number of trips between two points is directly proportional to the number of trips originating at one point and the number of trips destined to the other point.

$$T_{ij} = \frac{T_i \bullet T_j}{\sum_j T_j} \qquad \textit{Equation 1}$$

Where:
T_{ij} = the estimated number of trips between zone "i" and zone "j".
T_i = The total number of trips originating at zone "i".
T_j = The total number of trips destined to zone "j".

For a freeway microsimulation model the zones will typically be the freeway mainline and the on and off-ramps. The freeway mainline counts and the ramp counts then become the T_i's and the T_j's for estimating the trip table.

The above proportional model is designed so that all the entries in the seed table will add up (row-wise) to the traffic counts for the T_i's (the row totals in the OD table). However, the entries will not sum up column-wise to the counts for the T_j's (the column totals). So it is necessary to factor the table entries to better match the column totals. The process of factoring the table to better match the column and row totals is called the Furness process.
The Furness row and column factoring process is illustrated below. First the ratios of the desired to actual column totals are computed. The entries in each column are then multiplied by these column ratios. The table entries are then summed by each row and the ratio of desired to actual row total is computed for each row. The entries in each row are then multiplied by these row ratios. The process is then repeated on the columns and rows until the analyst specified number of iterations is reached or the analyst specified closure criterion (maximum remaining difference between the desired and actual row and column totals) is achieved. The Furness process can be implemented in a spreadsheet as illustrated in the tables below.

1. Initial Seed Table

	Zone A	Zone B	Row Total	Desired Total	Ratio
Zone A	114	86	200.00	200	1.00
Zone B	4	196	200.00	200	1.00
Column Total	118.00	282.00	400.00	400.00	
Desired Total	100	300			
Ratio	0.85	1.06			

Since the column totals do not match the desired attraction totals for each zone, each entry in the table is multiplied by the appropriate column ratio (which is the ratio of the desired column total divided by the actual column total).

2. Results of First Iteration on Column Totals

	Zone A	Zone B	Row Total	Desired Total	Ratio
Zone A	97	92	189.00	200	1.06
Zone B	3	208	211.00	200	0.95
Column Total	100.00	300.00	400.00	400.00	
Desired Total	100	300			
Ratio	1.00	1.00			

But now the row totals no longer match the desired production totals. So multiply all cell entries by row ratios.

3. Results of First Iteration on Row Totals

	Zone A	Zone B	Row Total	Desired Total	Ratio
Zone A	103	97	200.00	200	1.00
Zone B	3	198	201.00	200	1.00
Column Total	106.00	295.00	401.00	400.00	
Desired Total	100	300			
Ratio	0.94	1.02			

But now the column totals no longer match the desired attraction totals. (note that the column ratios are now closer to one. So we are getting closer). Typically the analyst would repeat the iterations 2 to 3 more times like this sufficient closure has been reached.

For most OD tables it is impossible to get 100% closure on all column and row totals by this method, because the adjustment factors will start repeating themselves. Also, the common sense requirement that only "whole integer" trips be distributed will cause rounding to interfere with the adjustment process as it gets closer to the desired totals.

4.5 Calibration Data

Calibration data consists of measures of capacity and measures of system performance such as travel times, speeds, delays and queues. To be valid and useful, <u>the calibration data must be</u>

gathered simultaneously with the traffic counts. If one has one or more continuous count stations in the study area, it may be possible to adjust the count data to match conditions present when the calibration data was collected, but this introduces the potential for additional error into the calibration data and weakens the strength of the conclusions that can be drawn from the model calibration step.

4.5.1 Field Inspection

It is extremely valuable to observe existing operations in the field during the time period to be simulated. Simple visual inspection can identify behavior not apparent in counts and floating car runs.

Video images may be useful, but may not focus on the upstream conditions causing the observed behavior, which is why a field visit during peak conditions is always important.

A field inspection is also valuable for aiding the modeler in identifying potential errors in the data collection.

4.5.2 Travel Time Data

The best source of point to point travel time data is “floating car runs”. One or more vehicles are driven the length of the facility several times during the analysis period and the mean travel time is computed. The number of vehicle runs required to establish a mean travel time within a 95% confidence level range depends on the variability of the travel times measured in the field. Free-flow conditions may require as few as 3 runs to establish a reliable mean travel time. Congested conditions may require 10 or more runs.

The minimum number of floating car runs needed to determine the mean travel time within a desired 95% confidence interval depends upon the width of interval that is acceptable to the analyst. If the analyst wishes to calibrate the model to a very tight tolerance, then a very small interval will be desired and a large number of floating car runs will be required. The analyst might aim for a confidence interval on the order of plus or minus 10% of the mean travel time. Thus, if the mean travel time were 10 minutes, the target 95% confidence interval would be 2 minutes.

$$N = \left(2 * t_{0.025,N-1} \frac{s}{R} \right)^2 \qquad \textit{Equation 2}$$

Where:

R = the 95% confidence interval for the true mean.

$t_{0.025,N-1}$ = the Student’s “t” statistic for two-sided error of 2.5% (sums to 5%) with N-1 degrees of freedom. (for 4 runs, t = 3.2; for 6 runs, t = 2.6; for 10 runs, t = 2.3)(Note there is one less degree of freedom than car runs when looking up the appropriate value of “t” in the statistical tables.)

s = the standard deviation of the floating car runs.

N = the number of required floating car runs.

For example, if the floating car runs showed a standard deviation of 1.0 minutes, a minimum of 7 floating car runs would be required to achieve a target 95% confidence interval of 2.0 minutes (plus or minus 1.0 minutes) for the mean travel time.

4.5.3 Point Speed Data

The Caltrans Freeway Performance Monitoring System (PeMS) is a good source of simultaneous speed and flow data for urban freeways in California. The loop detectors though may be subject to failures so the data must be reviewed carefully to avoid extraneous data points.

Loop detectors are typically spaced one-third to half-mile apart and their detection range is limited to a dozen feet. Under congested conditions, much can happen in between detectors, so the mean speeds produced by the loop detectors cannot be relied upon to give system travel times under congested conditions.

The loop measured free-flow speeds may be reliable for computing facility travel times under uncongested conditions, but care should be taken when using this data. Many locations have only single loop detectors in each lane, so the free-flow speed must be estimated from an assumed mean vehicle length. The assumed mean vehicle length may be automatically calibrated by PeMS, but this calibration requires some means of identifying which data points represent free-flow speed, which data points do not, and which ones are aberrations. The decision process involves some uncertainty. In addition, the mix of trucks and cars in the traffic stream varies by time of day, thus the same mean vehicle length cannot be used throughout the day. The bottom line here is that loop estimated/measured free-flow speeds should be treated with a certain amount of skepticism. They are precise enough for identifying the on-set of congestion, but may not be reliable to the nearest one mile per hour.

4.5.4 Capacity and Saturation Flow Data

Capacity and saturation flow data are particularly valuable calibration data since they determine when the system goes from uncongested to congested conditions.

Capacity can be measured in the field on any street segment immediately downstream of a queue of vehicles. The queue should ideally last for one full hour, but reasonable estimates of capacity can be obtained if the queue lasts only half an hour. The analyst simply counts the vehicles passing a point on the downstream segment for one hour (or lesser time period if the queue does not persist for a full hour) to obtain the segment capacity.

Saturation flow rate is: "The equivalent hourly rate at which previously queued vehicles can traverse an intersection approach under prevailing conditions, assuming that the green signal is available at all times and no lost times are experienced, in vehicles per hour or vehicles per hour per lane" (Source: Year 2000 Highway Capacity Manual). The saturation flow rate should be measured (using procedures specified in the Highway Capacity Manual) at all signalized intersections which are operating at or over 90% of their existing capacity. At these locations the estimate of saturation flow, and therefore, capacity will critically affect the predicted operation of the signal. Thus, it is cost-effective to accurately measure the saturation flow and therefore capacity at these intersections.

4.5.5 Delay and Queue Data

Delay can be computed from floating car runs or from delay studies at individual intersections.

Floating car runs can provide satisfactory estimates of delay along the freeway mainline, but are usually too expensive to make all of the necessary additional runs to measure all of the ramp delays.
Floating cars are somewhat biased estimators of intersection delay on surface streets since they reflect only those vehicles traveling a particular path through the network. For an arterial street with coordinated signal timing, the floating cars running the length of the arterial will measure delay only for the through movement with favorable progression. Other vehicles on the arterial will experience much greater delays. This problem can be overcome by running the floating cars on different paths, but the cost is generally prohibitive.

Comprehensive measures of intersection delay can be obtained from surveys of stopped delay on the approaches to an intersection. The number of stopped cars on an approach is counted every 30 seconds or so. The number of stopped cars times the counting interval (30 seconds) gives the total stopped delay. Dividing the total stopped delay by the total number of vehicles that crossed the stop-line (a separate count) during the survey period gives the mean stopped delay per vehicle. The stopped delay is converted to total delay by increasing it by 30% (a rule of thumb used in past editions of the Highway Capacity Manual, however, its applicability to all conditions is uncertain and untested, and therefore, should be used with caution).

4.6 Future Demand Forecasts

Forecasts of future demand are best obtained from the local regional transportation planning agency (RTPA) and its travel demand model. Trend line forecasts based on historic data are a reasonable second choice source of forecast demands.

4.6.1 Constraining Demand to Capacity

Regardless of which method is used to estimate future demand (regional model or trend line), care must be taken to ensure that the forecasts are a reasonable estimate of the actual amount of traffic that can arrive within the analysis period at the study area. Regional model forecasts are usually not very well constrained to the system capacity, and trend line forecasts are totally unconstrained. The result can be that the analyst is attempting to model a future condition that cannot happen.

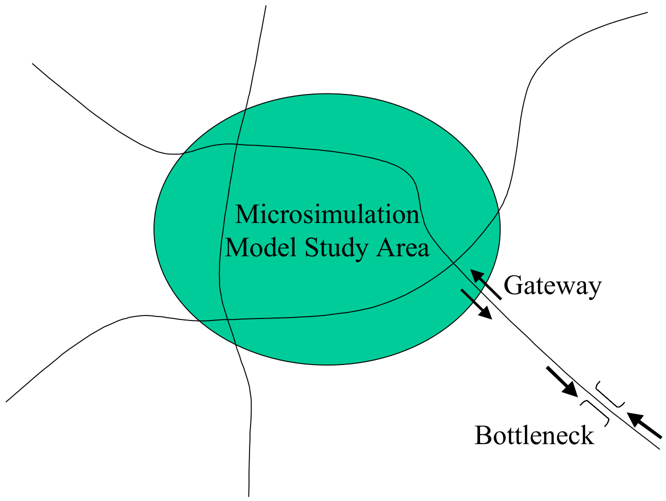
Microsimulation results are highly sensitive to the amount by which the demand exceeds the capacity of the facility, so it is vital that realistic demand forecasts be used in the analysis. The following steps outline a short procedure for manually reducing the forecasted demands in the study area to better match the capacity of the facilities feeding the study area.

Step 1. Identify Gateway Bottlenecks.

The analyst should first identify the critical bottlenecks on the facilities feeding the traffic to the boundaries of the microsimulation study area. Bottlenecks are sections of the facilities feeding the model study area that either have capacities less than other sections of the freeway or demands greater than the other sections. These are the locations likely to be the first ones to experience congested conditions as traffic grows.

These bottlenecks may be located on the boundary of the microsimulation study area, in which case they are identical to the gateway zones on the boundary of the microsimulation model study area. Bottlenecks within the microsimulation model study area can be neglected since they will be taken care of by the microsimulation model.

Exhibit 6. Bottleneck, Gateway, and Study Area



Inbound bottlenecks are congested sections feeding traffic to the microsimulation model area. Outbound bottlenecks are congested sections affecting traffic leaving the microsimulation model area. If an outbound bottleneck is likely to create future queues that back up into the microsimulation model study area, then the model study area should be extended outwards to include them. If the future outbound queues will not back up into the model study area then these bottlenecks can be safely neglected.

Step 2. Estimate Excess Demand at Inbound Bottlenecks

If the forecasted hourly demand at a bottleneck (in the inbound direction to the model) exceeds its capacity, the proportion of the demand that is in excess of the available hourly capacity should be computed. $X = \frac{D - C}{C}$ *Equation 3*

Where:
X = proportion of excess demand
D = Forecasted demand (vehicles per hour)
C = Estimated capacity (vehicles per hour)

Step 3. Reduce Forecasted Demand Inbound at Gateways

The forecasted hourly demands for the off-ramps between the bottleneck and the gateway entering the microsimulation study area should be reduced in proportion to the amount by which the forecasted bottleneck demand exceeds its capacity.

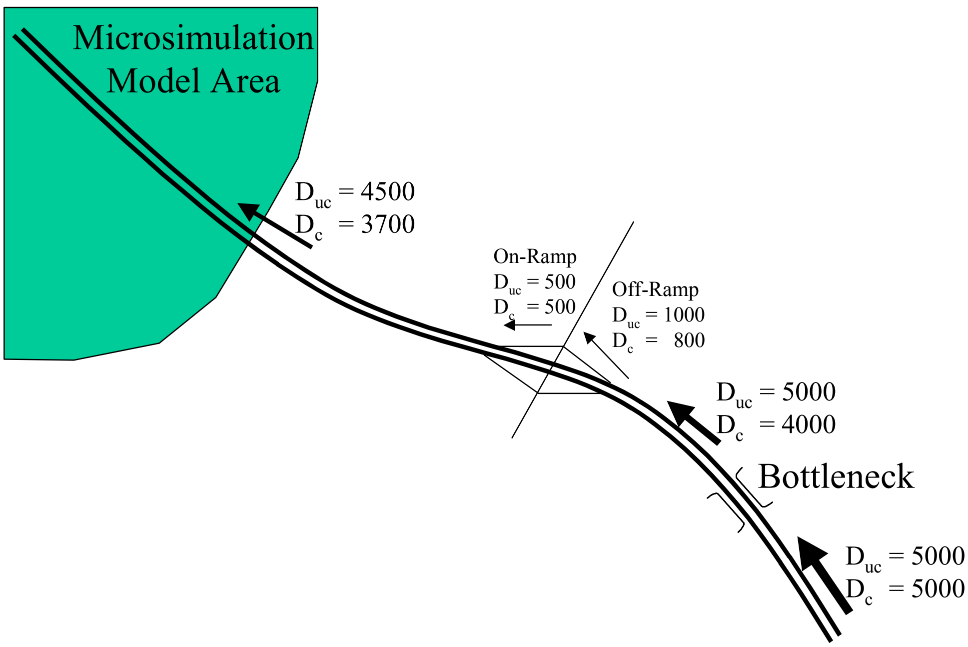
$D_{const} = D_{unconst} * (1 - X)$ *Equation 4*

Where:
X = proportion of excess demand
D_{const} = Constrained demand (vehicles per hour) for a downstream off-ramp or exit point.
$D_{unconst}$ = Unconstrained demand forecast (vehicles per hour)

It is suggested that the off-ramp demand be reduced in proportion to the reduction in demand that can get through the bottleneck on the assumption that the amount of reduction in downstream flows is proportional to the reduction in demand at the bottleneck. If the analyst has superior information (such as an OD table) then the assumption of proportionality can be over-ridden by the superior information. The constrained downstream gateway demand is then obtained by summing the constrained bottleneck, off-ramp, and on-ramp volumes between the bottleneck and the gateway to the study area.

The exhibit below illustrates how the proportional reduction procedure would be applied for a single inbound bottleneck that reduces the peak hour demand that can get through it from 5000 vph to 4000 vph. Since there is an interchange between the bottleneck and the entry gate to the microsimulation study area, the actual reduction is somewhat less (800 vehicles per hour) at the gate.

Exhibit 7. Example Proportional Reduction of Demand for Capacity Constraint



Starting upstream of the bottleneck there is an unconstrained demand for 5000 vph. Since the bottleneck has a capacity of 4000 vph, the downstream capacity constrained demand is reduced from the unconstrained level of 5000 vph to 4000 vph. Thus, one thousand vehicles are stored at the bottleneck during the peak hour. Since it is assumed that the stored vehicles are destined to downstream destinations in proportion to the exiting volumes at each off-ramp and freeway mainline, the downstream volumes are reduced the same percentage as the percentage reduction at the bottleneck (20%). A 20% reduction of the off-ramp volume results in a constrained demand of 800 vehicles per hour. The on-ramp volume is unaffected by the upstream bottleneck, so it's unconstrained demand is unchanged at 500 vph. The demand that enters the microsimulation study area is equal to the constrained demand of 4000 vph leaving the bottleneck, minus the 800 vph leaving the freeway on the off-ramp, plus 500 entering the freeway at the onramp which results in a constrained demand of 3700 vph.

4.6.2 Margin of Safety for Demand Forecasts

All forecasts are subject to uncertainty. It is risky to design a road facility to a precise future condition, given the uncertainties in the forecasts. There are uncertainties in both the likely growth in demand and the available capacity that might be present in the future. Slight changes in the timing or design of planned or proposed capacity improvements outside of the study area can dramatically change the amount of traffic delivered to the study area during the analysis period. Changes in future vehicle mix and peaking can easily affect capacity by 10%. Similarly, changes in economic development and public agency approvals of new development can dramatically change the amount of future demand. Thus it is good practice to explicitly plan for a certain amount of uncertainty in the analysis.

The analyst should plan for some selected percentage above and below the forecasted demand to allow for these uncertainties in future conditions. The analyst might consider at least a 10% margin of safety for the future demand forecasts. A larger range might be considered if the analyst has evidence to support the likelihood of larger variances in the forecasts.

To protect against the possibility of both under estimates and over estimates in the forecasts, the analyst should perform two sensitivity tests; one with 110% of the initial demand forecasts, and the other with 90% of the initial demand forecasts for establishing a confidence interval in the likely future conditions. Note that the percentage confidence interval (such as 95% confidence interval) has not been stated here, so it cannot be claimed that there is a certain probability of the true value falling in this +/- 10% range. This is merely a sensitivity test of the impacts of demands being 10% lower or 10% higher than forecast, without knowing the likelihood of it actually happening.

4.7 Data Collection Short Cuts

If one or more agencies in the study area currently has one of the following types of models already coded for the study area, they may be good and quick sources of the necessary data for microsimulation. Remember though, that calibration data should be matched to the input data. When mixing field measurements and data obtained from other sources, extra effort should be applied to ensure that the final data set is internally consistent (geometry, controls, flows, and performance measurements).

1. Regional Model (EMME/2, TP+/Viper, etc.): A regional travel demand model can provide existing and future vehicle origin-destination information, and crude road geometry. However, the road geometry contained in planning models usually cannot be relied upon for microsimulation work. Planning models do not usually have turn lane geometry, and lengths are often distorted to facilitate better display of the results. Signal timing data is usually completely lacking.

2. Synchro™ Model: A Synchro™ model for the study area can provide turning movements, intersection geometry, and signal timing. Origin-destination information would have to be synthesized from the Synchro turn movements.

3. TRAFFIX™ Model: A TRAFFIX™ model can provide turning moves, intersection geometry, and signal timing. Origin-destination information would have to be synthesized from

the turn movements. Actuated controller settings are approximated in TRAFFIX™ by an equivalent fixed time setting. Link lengths may not be coded accurately by the user since link lengths are critical in TRAFFIX™ only for arterial level of service analysis.

4. TRANSYT Model: A TRANSYT model can provide turning moves, intersection geometry, and signal timing. Origin-destination information would have to be synthesized from the turn movements. Actuated controller settings are approximated in TRANSYT by an equivalent fixed time setting.

5. FREQ Model: A FREQ model can provide an origin-destination table and road geometry for the freeway. The analyst can enter ramp and freeway mainline volumes into FREQ and use its OD estimation routine to estimate the OD table for each time slice of the peak period.

6. HCS Model: AN HCS model can provide turn moves, ramp merge/diver geometry, intersection geometry, and signal timing. Link lengths will generally not be available unless the user has performed an arterial level of service analysis.

5 MODEL DATASET CODING

The coding of input data into a model is a complex software specific task that is best covered in training courses on the specific software. It is only briefly summarized in the most general form here. This chapter therefore provides only the briefest highlights of coding, and emphasizes instead error checking strategies.

5.1 Coding the Model

Coding the input data into the model is a major task. During this task the data on network geometry, control, and demands is input into the microsimulation model.
Three basic types of data are coded:

- Network geometry (lanes, lengths, etc.)
- Control data (signs, signal timing)
- Demands

The steps involved in coding are:

1. Import and size overlay image (aerial photo or As-Built CAD file) for network coding
2. Set up coding templates (standardized correspondence tables between facility type, area type, and other link characteristics to expedite coding of standard link types).
3. Rough in links and node locations over aerial photo
4. Code link attributes (lanes, free-flow speeds)
5. Code intersection attributes (control type, control parameters, turn lane designations, stop bars, turn pockets)
6. Code source/sink zones or nodes
7. Code vehicle types and origin-destination table(s).
8. Review/Revise default global parameters (vehicle characteristics, vehicle mix, etc.).

5.2 Error Checking

Before proceeding to calibration it is necessary to ensure that the model input data has been entered correctly. Error checking involves various tests of the coded network. The steps involved in error checking are:

1. Color code links by attribute (lanes, facility type, free-flow speed, etc.) and identify discrepancies.
2. Review intersection attributes.
3. Review demand inputs
4. Run model at very low volumes to identify errors.
5. Trace selected vehicles through the network.
6. Plot OD Desire Lines to visually verify demand entries.

5.2.1 Error Checklist

Here is a simple checklist to go through to verify that the input data has been coded accurately.

1. Software Errors
   - For input compilation and run time error messages consult the software user's guide about the appropriate correction.
   - For software implementation errors check user group and software vendor websites for listings of known software errors and suggested work arounds.
2. Network
   - Check basic network connectivity. Are all connections present? Watch for "super-imposed" links and nodes (two or more nodes placed in the same location will look like a single node. The links may connect to one of the nodes, but not the other). Look for isolated nodes (no links connected to them). They should usually be eliminated.
   - Check link geometry (lengths, number of lanes, free-flow speed, facility type, etc.). Overlaying the network over an aerial photo is very helpful.
   - Check intersection controls (control type, control data)
   - Check for prohibited turns, lane closures and lane restrictions at intersections and on links
3. Demand
   - Check the identified sources and sinks (zones) for traffic. Verify zone volumes against traffic counts.
   - Check vehicle mix proportions and vehicle characteristic descriptions.

5.2.2 Error Checking Strategies

Here are some strategies that can be used to increase the efficiency and effectiveness of the error checking process.

1. Use color codes to identify links by the specific attribute being checked (for example: links might be color coded by free-flow speed range). Out of range attributes can be identified quickly if given a particular color. Breaks in continuity can also be spotted quickly (for example: a series of 35 mph links with one link coded as 25 mph).

2. Load 50% or less of the existing demand and observe vehicle behavior as the vehicles move through the network. Look for congestion that shows up at these unrealistically low demand levels. The congestion is often due to coding errors.

3. Follow a single vehicle through the network (again at very low demand levels) and look for unexpected braking and/or lane changes. Repeat for other O-D pairs.

4. Check entry and exit link flows at the 50% demand level to verify that all demand is being correctly loaded and moved through the network.

5. If a software error is suspected, code simple test problems (such as a single link or intersection) where the solution can be computed manually and compare the model and manually computed solutions.

6 MODEL CALIBRATION

This chapter addresses the steps, computations, and criteria for determining if the simulation model is reasonably consistent with local reality. The presumption of calibration is that the behavioral models and application software have already been validated and verified by the model and software developers. The only task left to the analyst is to adapt the behavioral models to local conditions.

6.1 Validation, Verification, and Calibration

A lengthy series of steps are required to ensure the accuracy of a microsimulation model. The series starts with researchers and programmers and finishes with the model user.

6.1.1 Model Theory Validation

The researchers must first conduct research and, based on that, specify a reasonable model of driver behavior and vehicle performance. The first step is therefore to determine that the proposed theoretical model of driver behavior and vehicle performance reasonably represents reality and is consistent with the general body of theory on traffic behavior. Fellow researchers review this work and publish papers critical or supportive of the new model. This step is sometimes called in the literature, simply “model validation”, however; to reduce confusion, this step will be called here “model theory validation”.

6.1.2 Model Software Verification

The programmer must then implement this theoretical model in software. The second step is therefore a “software verification” step. The programmer must develop tests and provide documentation to potential buyers and users that the software faithfully implements the theory. This is called “model verification”. Users may resort to professional societies, universities, public agencies, and each other for “independent” information on the veracity of the software.

6.1.3 Model Application Calibration

The software user then codes and calibrates a simulation model for a specific location. The programmer often provides defaults for the numerous model parameters in order to make the software easier to learn and apply. The user must then modify some of the program supplied default parameters to better match local conditions. This procedure is called model “calibration”.

This chapter deals only with the model application calibration step, adjustment of the default model parameters to cause the model reproduce more precisely local conditions.

6.2 Calibration Strategy

Calibration involves potentially hundreds of model parameters each of which impacts the simulation results in a manner that is often highly correlated with that of the others. The analyst can very easily get trapped in a never ending circular process fixing one problem only to find a new one pops up somewhere else. Therefore it is essential to break the calibration process into a series of logical, sequential steps, a strategy for calibration.

6.2.1 Fundamental Assumption for Calibration

Every model of travel behavior includes what are considered to be the most important factors for predicting travel behavior, but these models necessarily exclude some factors that are considered to be less important. The effect of these excluded factors will vary from community to community. It is therefore the objective of calibration to slightly adjust the basic behavioral models to account for the influence of the excluded factors to better replicate local behavior.

The fundamental assumption of calibration is therefore that the travel behavior models in the simulation model are essentially sound[5]. They have been both validated and verified. There is no need to verify that they produce the correct delay, travel time, and density, when they are given the correct input free-flow speed and capacity for a link. The only remaining task for the analyst is therefore slightly adjust these models to predict the correct capacity for local conditions.

6.2.2 Calibration Strategy

There are literally hundreds of often highly correlated parameters that might be modified by the user. In order to make calibration practical, the parameters must be divided into categories and each category dealt with separately.

Typology of Calibration Parameters

The hundreds of available calibration parameters should be divided by the analyst into two basic categories:

1. Parameters that the analyst is reasonably certain about and does not wish to adjust, and
2. Parameters that the analyst is less certain about and willing to adjust.

The analyst should attempt to keep the set of adjustable parameters to as small a set as possible in order to minimize the effort required to calibrate them. However, the trade-off is that more parameters allow the analyst more degrees of freedom to better fit the calibrated model to the specific location.

The set of "adjustable" parameters is then further subdivided into those that directly impact capacity (such as mean headway) and those that directly impact demand (such as route choice parameters). The capacity adjustment parameters are calibrated first (holding demand fixed), the demand adjustment parameters are then calibrated second (holding the capacity adjustment parameters fixed).

Each set of adjustable parameters can be further subdivided into those that affect the simulation on a global basis and those that affect the simulation on a more localized basis. The global

[5] The analyst determines the soundness of the simulation model during the stage of software selection and evaluation.

parameters are calibrated first. The "local" link-specific parameters are left for the last fine-tuning step of each stage of the calibration.

Exhibit 8. Division of Model Parameters Into Three-Stage Calibration Strategy

	Capacity Related Parameters		**Demand Related Parameters**	
	Global Parameters	**Link-Specific Parameters**	**Global Parameters**	**Link-Specific Parameters**
Parameters with values certain	Stage I Error check values.	Stage I Error check values.	Stage I Error check values.	Stage I Error check values.
Parameters with uncertain values	Stage II Global Calibration of Capacity	Stage II Fine Tuning of Link Capacities	Stage III Global Calibration of Demands	Stage III Fine Tuning of Link Volumes

Recommended Strategy
The following practical calibration strategy is recommended:

- **Stage I:** First an error checking process is performed on the coded input data and parameters. The coded network and demand data is reviewed along with the input and default values of the "fixed" parameters identified by the analyst.

- **Stage II:** Second, an initial calibration process is run to identify the values for the capacity adjustment parameters that cause the model to best reproduce observed traffic capacities. A global calibration is first performed, followed up by link specific fine-tuning. At this initial stage, all demands are held fixed (or artificially increased if necessary to cause congestion) and only traffic operations parameters are calibrated.

- **Stage III:** A second calibration process is performed, but this time with demand and route choice allowed to vary (The capacity parameters are now held fixed). The only parameters calibrated at this stage are those that specifically impact demand and route choice. First, a global calibration is performed, followed by link specific fine-tuning of parameters that affect route choice.
- **StageIV**: Finally, the overall model results are reviewed for realistic performance (travel times and queue lengths. Fine tuning adjustments are made to better match field measurements of travel times and queues.

The recommended calibration strategy is thus:

1. Check for and eliminate all obvious coding errors.
2. Calibrate the capacity related adjustment factors.
3. Calibrate the demand estimation parameters.
4. Review of overall realism of model results (queues and travel times).

The error correction stage is essential so that the calibration process does not result in parameters that are distorted to compensate for coding errors. The specified vehicle mix and vehicle performance characteristics are reviewed at this time.

The capacity calibration consists of the adjustment of global parameters in the simulation model to best replicate local field measurements of capacity. This is an important step because capacity has a profound effect on predicted system performance (delay and queues).

During this stage of the calibration the volumes used in the model are taken straight from field counts or temporarily artificially increased at key locations so as to be able to observe capacity conditions at these locations. If possible, the model is not allowed to re-route the traffic. Only the model parameters related to car following, lane changing, etcetera are calibrated to match field-measured capacity.

Once the parameters that influence capacity have been calibrated then the only remaining parameters to calibrate are those that affect demand, including those that affect route choice. During this last stage the demand estimation and route choice algorithms in the model are turned on again. The model predicted volumes are then compared to field counts and the analyst adjusts the link speeds, route choice parameters, and demand estimation algorithm until the best volume fit is achieved.

A final review is then made of overall model performance in terms of mean travel time and queues. The capacity and parameters and free-flow speeds are fine tuned cause the model to better match field measurements of travel times and queues. This last stage may involve compromising some of the calibration effort in the prior two stages, so this last fine-tuning should be done carefully, trying to preserve the stage II and stage III calibrated parameter values as much as possible.

6.3 First Stage: Error-Checking

Before proceeding to calibration it is necessary to ensure that the model input data has been entered correctly. Error checking involves various reviews of the coded network, the coded demands, and the default parameters. The steps involved in error checking are:

1. Review of vehicle parameters.
2. Review link attributes.
3. Review intersection attributes.
4. Review demand inputs (including vehicle occupancy)
5. Run model at very low volumes to identify errors.
6. Trace selected vehicles through the network.

6.3.1 Review of Vehicle Parameters

The vehicle parameters include vehicle mix, vehicle dimensions, and vehicle performance characteristics (maximum acceleration, etc.)[6]. The default values provided with the software are reviewed at this stage and corrected as necessary to match local conditions.

Vehicle Mix

The vehicle mix is defined by the analyst, often in terms of the percentage of total vehicles generated. Typical vehicle types in the vehicle mix might be passenger cars, single-unit trucks, semi-trailer truck, and bus.

Default percentages are usually included in most software packages, but the vehicle mix is something that is highly localized, and national default values will rarely be valid for specific locations. For example, the percentage of trucks in the vehicle mix can vary from a low of 2% on urban streets during rush hour to a high of 40% of daily weekday traffic on an intercity interstate freeway.

It is recommended that the analyst obtain one or more vehicle classification studies for the study area for the time period being analyzed. Vehicle classification studies can often be obtained from nearby truck weigh station locations.

Vehicle Dimension/Performance Parameters

The analyst should attempt to obtain the latest available vehicle fleet data (vehicle mix, dimensions, and performance) from the local state department of transportation or air quality management district/agency. National data can be obtained from Motor & Equipment Manufacturers Association, various car manufacturers, the Federal Highway Administration, and/or the Environmental Protection Agency.

As an illustration of how to develop updated vehicle performance specifications from these sources, the table below shows the vehicle specifications developed in 2002 by the Wisconsin Department of Transportation. They obtained vehicle performance data from the Ford Motor Corporation and the AASHTO Green Book. Test track performance data on maximum acceleration and deceleration rates were reduced by Wisconsin DOT personnel to rates presumed by them to be more typical of on-street driver behavior.

Exhibit 9. Example Wisconsin DOT Vehicle Performance Specifications

Vehicle Type	Length (meters)	Width (m)	Height (m)	Weight (metric tonne)	Top Speed (km/h)	Maximum Acceleration (mpss)	Maximum Deceleration (mpss)
Passenger Cars							
Mustang Convertible	4.7 m	1.9 m	1.4 m	1.5	224	3.6	-3.9
Crown Victoria	5.4 m	2.0 m	1.4 m	1.8	171	2.4	-3.8
Focus Sedan	4.4 m	1.7 m	1.4 m	1.1	169	2.3	-4.1
Kia Rio Sedan	4.2 m	1.7 m	1.4 m	1.0	177	2.1	-3.0

[6] The software supplied default vehicle mix, dimensions, and performance characteristics should be reviewed in particular for simulation software developed outside the United States. The software supplied default values frequently are not appropriate for United States conditions.

Vehicle Type	Length (meters)	Width (m)	Height (m)	Weight (metric tonne)	Top Speed (km/h)	Maximum Acceleration (mpss)	Maximum Deceleration (mpss)
Taurus Sedan	5.0 m	1.9 m	1.4 m	1.5	180	2.6	-3.9
Pickups, Vans and Sport Utility Vehicles							
F-150 Pickup	5.3 m	2.0 m	1.8 m	1.8	171	2.3	-3.7
Windstar Mini-Van	5.1 m	1.9 m	1.7 m	1.8	182	2.3	-3.1
Econoline Van	5.4 m	2.0 m	2.1 m	2.7	158	2.2	-3.2
Ford Explorer	4.8 m	1.8 m	1.8 m	1.9	158	2.4	-3.4
Motorcycles							
BMW Motorcycle	2.0 m	0.7 m	1.1 m	0.2	265	5.5	-4.5
Buses							
School Bus	13.4 m	2.4 m	3.0 m	12.9	100	0.5	-3.2
New Flyer Transit Bus	12.2 m	2.6 m	3.8 m	17.7	105	0.4	-3.2
Prevost Motorcoach	13.7 m	2.6 m	3.7 m	18.9	135	1.7	-3.7
Trucks							
Single Unit Truck	12.2 m	2.6 m	4.1 m	32.6	100	1.7	-3.7
55 Foot Semi-Trailer Truck	16.7 m	2.6 m	4.1 m	35.7	125	1.4	-3.5
Max. Wisconsin Single Trailer Semi-Truck	19.8 m	2.6 m	4.1 m	35.7	125	1.4	-3.5

Source: Wisconsin Department of Transportation, District 2, FSOA Team, Waukesha, WI, June 2002.

In the absence of more recent data from these sources, the analyst may use the following defaults taken from FHWA's CORSIM program and from Trafficware's SimTraffic™ program.

Exhibit 10. Suggested Vehicle Characteristics Defaults to be Used in Absence of Better Data

Vehicle Type	Length (ft)	Max. Speed (mph)	Max. Accel. (ft/ sec^2)	Max.Dec. (ft/sec^2)	Jerk (ft/sec^3)
Passenger Cars	14	75	10	15	7
Single-Unit Truck	35	75	5	15	7
Semi-trailer truck	53	67	3	15	7
Double-Bottom Trailer Truck	64	61	2	15	7
Bus	40	65	5	15	7

Notes:
Max. Speed = Maximum sustained speed on level grade in miles per hour.
Max. Accel. = maximum acceleration rate in feet per second squared when starting from zero speed.
Max. Dec. = maximum braking rate in feet per second squared (vehicles can actually stop faster than this, but this is a mean comfort based maximum).
Jerk = maximum rate of change in acceleration rates in feet per second cubed.
Sources: CORSIM and SimTraffic™

6.3.2 Error Checklist

Here is a simple checklist to go through to verify that the input data has been coded accurately.

1. Software
    - Check the software and user group websites for known errors.
2. Network
    - Check basic network connectivity. Are all connections present?
    - Check link geometry (lengths, number of lanes, free-flow speed, facility type, etc.).

- Check intersection controls (control type, control data)
- Check for prohibited turns, lane closures and lane restrictions at intersections and on links

3. Demand
    - Check the identified sources and sinks (zones) for traffic.
    - Verify zone volumes against traffic counts.
    - Check vehicle occupancy distribution (if modeling high occupancy vehicles).
    - Check vehicle mix proportions and vehicle characteristic descriptions.

6.3.3 Error Checking Techniques

Here are some techniques that can be used to increase the efficiency and effectiveness of the error checking process.

1. Definitely check the software website for user questions and comments. Sometimes known software errors and work arounds are listed.
2. Overlay the coded network over aerial photos of the study area to quickly verify the accuracy of the coded network geometry.
3. Turn on the node numbers and look for superimposed numbers. They are an indication of "super-imposed" links and nodes. Two or more nodes placed in the same location will look like a single node. The links may connect to one of the nodes, but not the other).
4. Use color codes to identify links by the specific attribute being checked (for example: links might be color coded by free-flow speed range). Out of range attributes can be identified quickly if given a particular color. Breaks in continuity can also be spotted quickly (for example: a series of 35 mph links with one link coded as 25 mph).
5. Run the animation at an extremely low demand level, so low that there is no congestion. The analyst should then trace single vehicles through the network and see where they unexpectedly slow down. These will usually be locations of minor network coding errors that disturb the movement of vehicles over the link or through the node. This test should be repeated for several different origin-destination zone pairs.
6. Run the simulation at 50% demand level and look for unexpected congestion locations at this low demand level. Check entry and exit link flows to verify that all demand is being correctly loaded and moved through the network.
7. If a software error is suspected, code simple test problems (such as a single link or intersection) where the solution can be computed manually and compare the model and manually computed solutions.

6.4 Second Stage: Calibration for Capacity

The objective of the second stage of the model calibration is to find a set of model parameters that cause the model to come as close as possible to matching field measurements of traffic capacity.

6.4.1 Collection of Field Measurements of Capacity

The choice of locations for field measurements of capacity will depend upon the existing traffic conditions within the study area.

For unsignalized facilities (freeways, rural highways, and rural roads), the analyst should identify locations where queues persist for at least 15 minutes and measure the flow rate at the point where the queue discharges. This observed flow rate is measured only while an upstream queue is present. It is summed across all lanes and converted to an equivalent hourly flow rate. This is the field measured capacity of the facility at this point. Several measurements should be made and averaged.

For signalized intersections, the analyst should identify the approach legs that frequently have queues of at least 10 vehicles per lane and measure the saturation flow rate per hour per lane using the procedures outlined in the appendix to the signalized intersection chapter of the Highway Capacity Manual. Several measurements should be made and averaged.

Ideally, there should be at least 3 or 4 locations where capacity can be measured for unsignalized facilities (if part of the proposed simulation model network) and 3 to 4 locations where saturation flow can be measured for signalized intersections (if part of the proposed simulation model network).

If there are insufficient locations in the field where the analyst can measure capacity then the methods of the Highway Capacity Manual can be used to estimate capacity. However, these methods should not be considered as accurate as direct field measurements.

The locations selected for calibrating the model to capacity should ideally be geographically separated to ensure the robustness of the results. Three street links in linear sequence do not provide as sturdy a basis for calibration as widely dispersed geographic locations.

The analyst may choose to perform the capacity calibration on a subnetwork of the total network to save on model run times and evaluation effort.

6.4.2 Obtaining Model Estimates of Capacity

Microsimulation models do not output a number called capacity. Instead they output the number of vehicles that pass a given point. So the analyst must manipulate the input demand as necessary to create a queue upstream of the target section to be calibrated.

For unsignalized facilities (freeways, rural highways, and rural roads), the simulated queue should persist for at least 15 minutes of simulated time, across all lanes and links feeding the target section. The simulated capacity is then the mean flow rate of the target section (measured at a detector placed in the target section and summed across all lanes) averaged over the 15 minute or longer period that the queue is present. The result is then divided by the number of lanes and converted to an hourly flow rate.

For signalized intersections, the coded demand should be increased as necessary to ensure the necessary queues of at least 10 vehicles at the start of green. A detector is placed in the model at the stopline to measure the discharge headways (On a per lane basis) of the first 10 vehicles crossing the detector in each lane. The per-lane headways are averaged for each lane and then averaged across lanes. The result is then converted to an hourly flow rate per lane.

6.4.3 Choice of Calibration Parameters

The objective of this stage of calibration is to reproduce field measurements of capacity, not demands, thus only parameters that affect maximum flow rates (such as car following, lane changing, and gap acceptance) should be calibrated. Parameters that affect demand and route choice, such as trip generation rates, familiarity, cost functions, etcetera, should be untouched at this stage.

For most microsimulation models, the mean headway and the mean reaction time (or their equivalent) are satisfactory capacity calibration parameters.

6.4.4 Calibration Objective Function

It is recommended that the analyst seek to minimize the Mean Square Error (MSE) between the model estimates of maximum achievable flow rates and the field measurements of capacity. The MSE is the sum of the squared errors averaged over several model run repetitions. Each set of repetitions has a single set of model parameter values (p) with different random number seeds for each repetition within the set[7].

Objective Function is as follows: Select a set of model parameters "p" so as to:

$$\text{Minimize } MSE = \frac{1}{R}\sum_{r}(M_{ltpr} - F_l)^2 \qquad \text{Equation 5}$$

Subject to:

$p_m^{\min} \le p_m \le p_m^{\max}$ for all user adjustable model parameters "p_m"

where:

MSE = Mean squared error.

M_{lpr} = Model estimate of queue discharge flow rate (capacity) at location "l" and time "t" using parameter set "p", for repetition "r".

F_{il} = Field measurement of queue discharge flow rate (capacity) at location "l".

R Number of repetitive model runs with fixed parameter values p_m and different random number seeds[8].

$p_{m,r}$ Value of model parameter number "m" for model run number "r"

$p^{min}{}_{m}$, User specified limits to the allowable range of parameter values (p_m). Limits are

[7] Some researchers have calibrated models using the percent mean square error in order to avoid the unintended weighting effect when combining different measures of performance (such as volumes and travel time) into one measure of error. The percent MSE divides each square error by the field measured value. The effect of using percent MSE is to place greater weight on large percentage errors rather than on large numerical errors. Since it seems desirable to place greater weight on high volume errors than on high percent errors on low volume streets, the simple MSE was selected (with weights to reduce the biasing effect of mixing different performance measures).

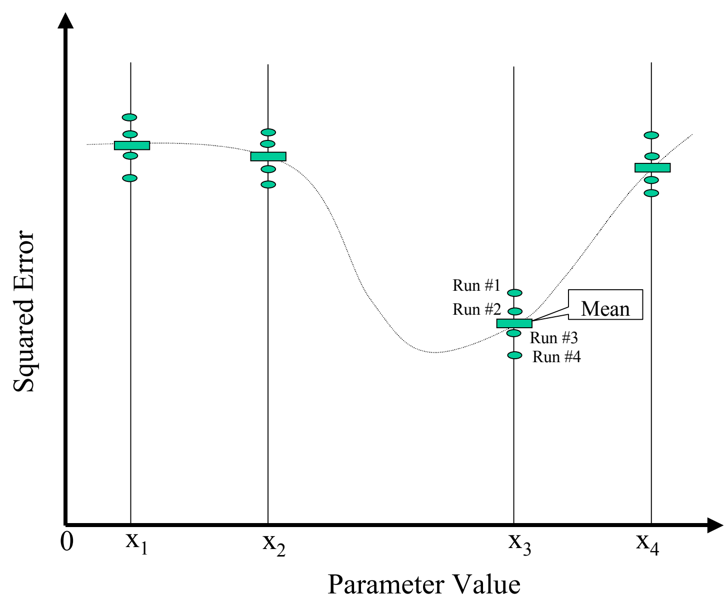
[8] Since the objective is to minimize the error, dividing by a constant, R (the number of repetitions), would have no effect on the result. However, "R" is included in the objective function to emphasize to the analyst the necessity of running the model several times with each parameter set.

p^{max}_{m} necessary to avoid solutions that violate laws of physics, vehicle performance limits, and driver behavior extremes.

The number of model runs "R" required to determine the mean squared error for each set of parameters (p) must be determined by the analyst by reviewing the variance in the estimate of the squared error and comparing it to the difference in the mean square errors for two sets of model parameters. The number of runs should be set high enough so that the analyst's confidence in the mean square error is sufficient to select between two different parameter sets.

The exhibit below illustrates how a single parameter search (m=1) might look for the number of model runs (R) set to 4.

Exhibit 11. Minimization of the Mean Square Error



The analyst must employ a search strategy for identifying the optimal combination of parameters (p_m) for minimizing the squared error. Various software and heuristic search methods are available (see the Optimization Technology Center Website of the Argonne National Laboratory and Northwestern University located at:
http://www.ece.nwu.edu/OTC/).

6.4.5 Fine Tuning - Calibration of Link Specific Parameters

Once the optimal global capacity calibration parameter values have been identified, there will be still some locations where model performance deviates a great deal from field conditions. The next step is therefore to fine-tune the predicted capacity to match the location specific measurements of capacity as closely as possible (within 1 percent).

Link specific capacity adjustments account for the roadside factors that affect capacity (such as presence of on-street parking, numerous driveways, or narrow shoulders), which are not typically coded in the microsimulation network input data set.

Most simulation software packages have link specific capacity (or headway) adjustment factors that apply only to the subject link. These capacity adjustment factors are used to fine-tune the model calibration.

Link specific adjustment factors should be used sparingly since they are not behavior based. They are fixed adjustments that will be carried through all future runs of the simulation model. Alternatives that might change the vehicle mix or driver behavior would not affect these link adjustments.

6.5 Third Stage: Calibration for Demand

Once the analyst is satisfied that the model (when given the correct demands) reproduces as closely as possible field measured capacities, the next step is then to calibrate the demand parameters in the model to best match observed demand levels and route choice.

6.5.1 5.5.1 Choice of Calibration Parameters

Since the objective of the second step of calibration is to reproduce observed traffic volumes, not operating conditions, then only parameters that affect traffic demand and route choice, such as trip generation rates, peaking of demand, driver familiarity, link cost functions, link free-flow speeds, etcetera, should be calibrated at this stage. Parameters affecting predicted traffic capacity should remain untouched at this stage.

6.5.2 Choice of Field Measurements for Calibrations

Since the ability of the model to predict local capacity was calibrated in the previous stage, there is no need to revisit that at the second stage. Only demand is uncertain now. Therefore, for this stage, the choice of field measurements for model calibration is simple: volumes. The analyst tries to match the predicted point or link volumes at different locations in the network.

6.5.3 Calibration Objective Function

For the second step of the calibration it is recommended that the analyst seek to minimize the Mean Square Error (MSE) between the observed and estimated volumes. The same basic MSE formula can be used, but with different variables.

6.5.4 Fine Tuning

Fine-tuning consists of the application of link specific route choice factors that affect the proportion of demand assigned to each link. These link specific factors may include free-flow speed, and link specific travel costs used in route selection.

6.6 Fourth Stage: Overall Review

In this last stage of the calibration the overall traffic performance predicted by the fully functioning model is compared to field measurements of travel time, queue lengths, and duration of queuing. The analyst very gingerly touches up link free-flow speeds and link capacities to better match field conditions. Since changes made at this stage may compromise the prior two stages of calibration, these changes should be made very sparingly. The next section suggests calibration targets for this last stage of review.

6.7 Calibration Targets

The objective of model calibration is to get the best match possible between model performance estimates and field measurements of performance. However, there is a limit to the amount of time and effort anyone can put into eliminating error in the model. There comes a point of diminishing returns where large investments in effort yield small improvements in accuracy. The analyst needs to know when to stop. The following table suggests potential stopping points for calibration. They were developed by the Wisconsin Department of Transportation based upon guidelines developed in the United Kingdom.

Exhibit 12. Wisconsin DOT Freeway Model Calibration Criteria

Criteria & Measures	Acceptability Targets
Hourly Flows, Model vs. Observed	
Individual Link Flows	
Within 15%, for 700 vph < Flow <2700 vph	> 85% of cases
Within 100 vph, for Flow < 700 vph	> 85% of cases
Within 400 vph, for Flow > 2700 vph	> 85% of cases
Total Link Flows	
Within 5%	All Accepting Links
GEH Statistic – Individual Link Flows	
GEH < 5	> 85% of cases
GEH Statistic – Total Link Flows	
GEH < 4	All Accepting Links
Travel Times, Model vs. Observed	
Journey Times Network	
Within 15% (or one minute, if higher)	> 85% of cases
Visual Audits	
Individual Link Speeds	
Visually acceptable Speed-Flow relationship	To analyst's satisfaction
Bottlenecks	
Visually acceptable Queuing	To analyst's satisfaction

Source: FREEWAY SYSTEM OPERATIONAL ASSESSMENT, Technical Report I-33, Paramics Calibration & Validation Guidelines, DRAFT , Wisconsin Department Of Transportation, District 2, June 2002

The GEH statistic is computed as follows:

$$GEH = \sqrt{\frac{(V-C)^2}{(V+C)/2}} \qquad \textit{Equation 6}$$

where:
GEH = The statistic
V = model estimated directional hourly volume at a location.
C = directional hourly count at a location.

A GEH value of less than 5 for more than 85% of the locations is considered acceptable in United Kingdom practice[9].

6.8 Review of Animation Output

Animation output enables the analyst to "see" the vehicle behavior that is being modeled and assess the reasonableness of the microsimulation model itself. The animation should be observed in close detail at key congestion points to determine if the animated vehicle behavior is realistic.

If the observed vehicle behavior appears unrealistic, the analyst should search through the following potential causes of the unrealistic animation in the order shown below:

1. Error in Analyst Expectations: The analyst should first verify in the field the correct vehicle behavior for the location and time period being simulated before deciding that the animation is showing unrealistic vehicle behavior. Many times, analyst expectations of realistic vehicle behavior are not matched by actual behavior in the field[10]. Field inspection may also reveal causes of vehicle behavior not apparent when coding the network from plans and aerial photos. These causes need to be coded into the model if the model is to be expected to produce realistic behavior.

2. Analyst Data Coding Errors: The analyst should next check for obvious and subtle data coding errors that may be causing the simulation model to represent the behavior incorrectly. Subtle data coding errors are the most frequent cause of unrealistic vehicle behavior in commercial microsimulation models that have already been subjected to extensive validation. Subtle coding errors include apparently correctly coded inputs that are incorrect because of how they are used in the model to determine vehicle behavior.[11]

3. Animation Errors: The analyst should recognize that often a different software module is used to display the vehicle movement on the screen, using different algorithms than those used to determine vehicle movement in the simulation model. Unrealistic movement may originate from constraints on the visual display capabilities of the computer that do not apply to the mathematical computations of vehicle movement inside the simulation model. Animation error can only be resolved by the software writer of the animation module.

---

[9] Traffic Appraisal in Urban Areas, Highways Agency, Manual for Roads & Bridges Volume 12 Section 2. Department for Transport [Formerly Department of Environment, Transport & the Regions], London, May 1996. (http://www.archive.official-documents.co.uk/document/ha/dmrb/index.htm).

[10] Analysts should not expect classic macroscopic traffic flow concepts to apply at the microscopic, individual vehicle level. Macroscopic flow concepts, such as no variance in mean speed at low flow rates, do not apply to the behavior of an individual vehicle over the length of highway. An individual vehicle's speed may vary over the length of the highway, and between vehicles, even at low flow rates. Macroscopic flow theory speaks of the average speed of all vehicles being relatively constant at low flow rates, not individuals.

[11] For example, it could be that the warning sign for an upcoming off-ramp is posted in the real world ¼ mile before the off-ramp, but because the model uses warning signs to identify where people start positioning themselves for the exit ramps, the analyst may have to code the warning sign at a different location, the location where field observations indicate where the majority of drivers start positioning themselves for the off-ramp.

4. Vehicle Behavior Model Errors: All vehicle behavior models, by definition, account for only a few of all the potential factors than can affect driver decision making. In addition there may be software errors in the application of the model theory. Vehicle behavior model errors can only be resolved by the software writer who wrote the software application and the theorist who created the behavior model.

6.9 Validation Against Fundamental Traffic Flow Relationships

The analyst should be cautious about comparing mean results across several model repetitions against fundamental traffic flow relationships. This is because the basic traffic flow relationship between density, volume, and speed applies to point measurements and not to averages of a series of point measurements.

According to the fundamental equation of traffic flow, the flow rate Q at any particular point (i) is equal to the density divided by the speed at that point.

$$Q_i = k_i \bullet s_i \qquad \textit{Equation 7}$$

However, this does not mean that the mean flow rate over time is equal to the mean density over time divided by the mean speed over time (except under very limited circumstances[12]).

$$\overline{Q} = \frac{1}{N}\sum k_i \bullet s_i \neq \tfrac{1}{N}\sum k_i \bullet \tfrac{1}{N}\sum s_i \qquad \textit{Equation 8}$$

Where:
Q-bar = the mean flow rate
N = number of observations
k_i = the i'th observation of density.
s_i = the i'th observation of speed.

This does not reduce the validity of the mean speed, flow, or density. The mean values of these traffic measures just do not follow the fundamental traffic flow equation.

In order to compare model performance to the fundamental traffic flow equation the analyst should select one model repetition for evaluation and gather point data on speed, headways, and flow. The data should be as temporally and geographically disaggregate as possible.

6.10 Laboratory Session: Microsimulation Model Calibration

Class participants will exercise a simulation model for a freeway interchange. They will error check it and then calibrate the simulation model parameters to field measurements of volume and travel time. The detailed write-up for this laboratory is contained in the appendix.

[12] All uniform density and uniform speed over the time period

7 ANALYSIS OF RESULTS

This chapter provides guidance on the analysis and interpretation of microsimulation model output. Microsimulation models are notorious for generating reams of output. This chapter explains how to determine the number of required model runs, pick out the key numerical outputs in each microsimulation model run, and interpret their meaning.

7.1 Microsimulation Output

Microsimulation models typically produce two types of output, animation displays and numerical output in text files. The animation display shows the movement of individual vehicles through the network over the simulation period. Text files report accumulated statistics on the performance of the network. It is crucial that the analyst review both numerical and animation outputs, and not just one or the other, in order to gain a complete picture of the results.

Animation output shows the results from just one run of the simulation model. Drawing conclusions about traffic system performance from reviewing just the one animation result is like trying to decide if the dice are fair from just one role. One needs to role the dice several times, tabulate the results, and compute the mean and standard deviation of the results in order to have the information needed to determine if the dice are fair.

7.1.1 Animation Output

Animation output is very powerful in that it enables the analyst to see the modeled vehicle behavior and assess the accuracy of the analyst's coding of the network. The analyst can even begin to assess the reasonableness of the behavioral models inside the microsimulation model itself.

However, reviewing animation results can be time consuming and tedious for large networks and long simulation periods. So the analyst should focus on the times and locations where congestion appears in the simulation. Tools that help the analyst in this review include those that help the analyst jump to different time points in the simulation (snapshots), those that help the analyst spot congestion problems (hotspots and dynamic link coloring), and diagnostic tools like "vehicle traces" that enable the analyst to follow the movement of individual vehicles through the network.

Snapshots record the status of the simulation at selected time points (for example, every 5 minutes) in a simulation period. The analyst can start the simulation from any snapshot time point to view vehicle behavior at that time. Snapshots can be a very cost effective tool for reviewing animation for lengthy simulation periods. The analyst can peruse the snapshots and quickly identify the rough starting times and locations of congestion. A more detailed review can then be performed by running the animation from the last snapshot before the onset of congestion.

Hotspots provide a visible mark on the network highlighting locations of congestion as the animation proceeds. These can be a very useful tool for identifying the locations and starting/ending times of congestion in large networks over long simulation periods.

Dynamic link coloring changes the color of the link according to a user selected performance measure, such as density, speed, or flow. For example, as density changes during the simulation period from low to high, the color of the link may be changed from blue to red. If the software lacks hotspot identification in its animation, then dynamic link coloring can be used to find problem spots in the network.

Vehicle traces enable the analyst to follow individual vehicles through the network to better assess the reasonableness of vehicle behavior during the simulation.

The analyst may be tempted to base their analysis entirely on a detailed review of the results of the animation, but there are two potential pitfalls with this approach:

1. The animation is only "one roll of the dice". A congestion problem may not show up in model runs using one random number seed, but may show up in other runs with different random number seeds.
2. The software module used to animate the movement of vehicles on the screen is not usually the same software module used to actually simulate the vehicle behavior on the network. The text output file summarizes vehicle behavior coming directly out of the simulation of the vehicles, while the animation is a translation of this information into a format that the computer can display.

For these reasons the analyst should not only review the animation output to verify detailed vehicle behavior, but should also review the text outputs.

7.1.2 Numerical Output

Microsimulation software reports the numerical results of the model run in text output files called reports.

Unless the analyst is reviewing actual vehicle trajectory outputs, the output reports are almost always a summary of the vehicle activity simulated by the model. The results may be summarized over time and or space. It is critical that the analyst understand how the software has accumulated and summarized the results to avoid pitfalls in interpreting the numerical output.

Microsimulation software may report instantaneous rates (such as speed) observed at specific instances of time, or may accumulate the data over a longer time interval and either reports the sum, the maximum, or the average. Depending on the software package, vehicle activity that occurs between time steps (such as passing over a detector) may not be tallied, accumulated, or reported.

Microsimulation software may report results for specific points of a link in the network or aggregated for the entire link. The point specific output are similar to what would be reported by detectors in the field. Link specific values of road performance are accumulated over the length of the link, and therefore will vary from point data.

The key to correctly interpreting the numerical output of a microsimulation model is to understand how the data was accumulated by the model and summarized in the report. The report headings may give the analyst a clue as to the method of accumulation used, but these short headings cannot usually be relied upon. The method of data accumulation and averaging can only really be determined through a detailed review of the model documentation on the reports it produces, and, if the documentation is lacking, querying the software developers themselves.

A healthy skepticism is valuable when reviewing reports. It helps to cross check outputs to see if the analyst understands correctly how the data is accumulated and reported.

Volumes

Volumes may be reported as an accumulation (number of vehicles observed since start of time period), or a rate (number of vehicles observed since start of time period divided by the length of the time period, in hours)

Depending on the software, vehicles that start the time period outside of the link, but end the time period inside the link, may or may not be accumulated in the total count for that time period. Similarly, vehicles that start out on the link, but exit the link before the end of the time period may or may not be counted. Since more or less vehicles may be stored on the link at the end of the time period than at the beginning (due to changes in density), the link volume count may be different than a detector count at a specific point on the link.

Depending on the software, vehicles that pass over a detector between time steps may not be counted by the software.

Travel Time, Speed, and Delay

Travel time, speed, and delay are all travel time based measures of system performance. Delay isolates that portion of the travel time that is most objectionable to drivers. Mean system speed is a useful tool for normalizing the travel time results into an index of overall system performance.

<u>Travel time</u> may be accumulated in terms of vehicle-hours traveled (which is the sum of all the travel times for all vehicles during the simulation period) or an average travel time may be reported. Some software will report the travel time by components. The analyst should read the software documentation carefully to understand the distinctions between the components and whether they overlap. Some software will report time in units of minutes; others will report it in units of hours.

<u>Speed</u>, being a rate, may be measured and averaged in several ways in microsimulation software.

1. The average speed over a section of roadway may be computed by summing the total travel times for all the vehicles traversing the full length of the roadway section, dividing by the number of vehicles, and then dividing the result into the total length of

the roadway section. This method excludes vehicles present on portions of the roadway section that do not travel the full length of the section.

2. The average speed of all vehicles present on any portion of a section of roadway may be computed by summing the vehicle-miles traveled (VMT) and dividing by the vehicle-hours traveled (VHT). This method includes vehicles that do not traverse the full length of the roadway section.
3. The average speed may be measured at a point on the link and averaged. A simple arithmetic mean of the observed point speeds will produce a higher mean speed than a harmonic mean (which averages the inverse of the speeds). The harmonic mean comes closer to the average speeds produced by the previously described methods. However, a point estimate of mean speed will never be the same as the mean speed measured over a distance.

Delay depends upon how the software defines free-flow travel time. Some software define the free-flow travel time as the analyst coded free flow speed for each link divided into the length of the link. Other software define the free-flow travel time as the length of the link divided by the randomly selected desired speed of each driver on the link. Delay may be split into various components, such as control delay and stopped delay, which may overlap.

Delay is generally reported by microsimulation models only for the link and time period for which is occurs. Thus delay that spills over beyond the end of the simulation period is not accumulated. Control delay associated with a traffic signal may be reported for several upstream links rather than accumulated only on the approach links to the signal.

Stops

The number of stops is a useful indicator of the quality of signal progression. The fewer stops there are, the better the progression.

Depending on the software, a stop may be defined as a vehicle coming to a full stop (traveling at zero miles per hour for at least one time step), or a stop may be defined as a vehicle coming almost to a stop (traveling below some defined speed and acceleration for at least one time step).

A vehicle moving up in a queue may accelerate and stop several times on the same link. Some software will accumulate each stop, other software may accumulate only the first stop on the link.

Density

Density is used as a measure of the quality of service on freeways and highways.

Density may be reported as an instantaneous "snapshot" value averaged over the length of a link at different time points during the simulation. It is computed as the length of the link multiplied by the number of lanes, divided by the number of vehicles present on the link when the snapshot is taken.

Density may be computed from the mean headway (measured in distance) of vehicles crossing over a point detector on the link. The point measurement of mean density at a detector will be different from the mean density measured over the length of a link. The amount of the difference will depend upon how much spot densities vary over the length of the link.

Queues

Queue lengths are important for identifying locations of heavy congestion in the system. Queue overflows indicate locations needing more storage.

The identification of a queue varies by software. Some allow the analyst to define the maximum speed, and maximum headway below which a vehicle is considered to be in a queue. Other software set a maximum speed and maximum acceleration below which a vehicle is considered to be in a queue. Some require that two vehicles be present to form a queue.

The mean queue, maximum queue, maximum back of queue, or 95 percentile queue may be reported.

The mean queue may be computed by summing up the number of queued vehicles present on a link at each time step and then dividing by the total number of time steps (including time steps when zero queue is present). This results in much lower average queue than the traditional macroscopic traffic engineering definition of mean queue for a signal that excludes all time steps when the light is green.

Note that if the queue at any point exceeds the length of the link or the turn pocket storage, then queue length is reported as the length of the link or the turn storage pocket. The excess queue is assigned to the upstream link or (for a turn pocket) to the through lanes upstream of the turn pocket.

The maximum queue is often the most number of vehicles observed to be queued on a link or a turn pocket lane. Note that the reported maximum queue will not exceed the length of the link or the length of the turn pockets.

At a traffic signal, vehicles start moving at the front of a queue when the signal turns green. During this time the queue of vehicles may continue to build at the back of the queue while the total number of vehicles in the queue is actually decreasing. For this reason, some software will report the maximum "back of queue" in addition to the maximum queue.

The maximum back of queue may be reported as the stopping point of the vehicle in the queue farthest back from the exit point of the link. If this length is greater than the length of the link, only the link length is reported. Similarly, the reported maximum back of queue for a turn pocket will not exceed the available storage length of the turn pocket.

The 95 percentile queue is defined as the queue length that has a 95% probability of not being exceeded during the analysis period (It might be exceeded more or less often on other days and outside the analysis period). Microsimulation software do not usually report the 95 percentile queue lengths because it requires multiple model repetitions with different random number seeds to estimate the probability of a queue of a given length. Some software may estimate the 95 percentile queue from a single run based upon an assumed distribution of arrival patterns and the mean arrival rate on red within the simulation period, however; this assumes that the single microsimulation model run with one random number seed is average (rarely true).

7.2 Summarization of Results

Even though the numerical reports produced by microsimulation software are summaries of the vehicle trajectories predicted by the simulation model, the numerical reports represent only the results of "one roll of the dice". The model runs must be repeated with other random number seeds and the numerical results summarized across the various repetitions.

To keep the analysis of results tractable, the analyst must select a few key summary statistics that best characterize the results of each model run for each alternative, and then select a method for aggregating the summary statistics for each alternative (mean, standard deviation, and 95 percentile).

7.2.1 Selection of Key Summary Statistics

Microsimulation, by its very nature, can bury the analyst in detailed, microscopic output. The key is to focus on a few key indicators of system performance and localized breakdowns in the system.

Key Indicators of Overall System Performance

The key indicators of overall system performance are:
1. Vehicle-Miles Traveled (VMT) or Vehicle-Kilometers Traveled,
2. Vehicle-Hours Traveled (VHT), and
3. Mean System Speed.

Total system delay is another useful overall system performance measure, when it is available. The number of stops is also useful for signal coordination studies.

VMT provides an indication of total travel demand (in terms of both the number of trips and the length of the trips) for the system[13]. Increases in VMT generally indicate increased demand. VMT is computed as the product of the number of vehicles traversing a link and the length of the link, summed over all links.

Since VMT is computed as a combination of the number of vehicles on the system and their length of travel, it can be influenced both by changes in number vehicles and by changes in the trip lengths during the simulation period. Changes in VMT between one run and the next can be caused by:

- Random variations between one run and the next (use of the same random number seed in both runs can reduce the variation, but might not eliminate it for some software).
- Changed demand (higher demands will increase VMT, lower demands will reduce VMT)

[13] Person-miles traveled (PMT), if available, is a superior measure of travel demand since it takes into account the number of people in each vehicle.

- Changed congestion causing vehicles to take different paths (Increased congestion will usually increase VMT, but sometimes, as in the case of ramp metering, increased congestion can cause a modest drop in VMT. Increased congestion may also reduce the number of vehicles that can complete their trip during the simulation period, also decreasing VMT.).
- Inability of the model to load the coded demand onto the network within the simulation period (Increased congestion may force the model to store some of the vehicle demand off-network due to bottlenecks at loading points. In this situation, increased congestion may actually lower VMT, since stored vehicles cannot travel any distance during the simulation period).

VHT provides an estimate of the amount of time expended traveling on the system[14]. Decreases in VHT generally indicate improved system performance and reduced traveling costs for the public. VHT is computed as the product of the link volume and the link travel time, summed over all links.

Since VHT is computed as a combination of the number of vehicles and the time spent traveling, it can be influenced both by changes in demand (the number of vehicles) and changes in congestion (travel time). Changes in VHT between one run and the next can be caused by:

- Random variations between one run and the next.
- Changed demand (higher demands will increase VHT, lower demands will lower VHT).
- Changed congestion (increased congestion will usually increase VHT by reducing speeds).
- Demand stored off-network due to excessive congestion at load points (increased congestion that causes demand to be stored off-network may reduce VHT if the software does not accumulate delay for vehicles stored off the network).

Mean system speed is an indicator of overall system performance. Higher speeds generally indicate reduced travel costs for the public. The mean system speed is computed from the VMT and VHT as follows:

Mean System Speed = VMT/VHT *Equation 9*

Changes in mean system speed between one run and the next can be caused by:

- Random variations between one run and the next.
- Changed link speeds and delays due to congestion.
- Changes in vehicle paths due to congestion.
- Changes in vehicle demand
- Changes in the number of vehicles stored off-network due to excessive congestion at load points.

[14] Person-hours traveled (PHT), if available, provides a superior measure of travel delay since it takes into account the number of people delayed in each vehicle. This is especially important for comparing the performance of high occupancy vehicle alternatives.

Total system delay, if available, is useful because it reports the portion of total travel time that is most irritating to the traveling public. However its definition is quite tricky. It depends upon what the analyst or the software writer consider to be ideal (no delay) travel time. Some sources consider delay to include only the delay due to increases in demand above some base uncongested (free-flow) condition. Others add in the base delay occurring at traffic control devices even at low flow conditions. Some include acceleration and deceleration delay. Others include only stopped delay. The analyst should consult the software documentation to ensure the appropriate use and interpretation of this measure of system performance.

Key Indicators of Localized Problems

The key indicators of localized problems are:
1. Link Queue Overflow Reports
2. Hot Spots Report

The link queue overflow report produced by most software identifies the links and times during the simulation period when the computed queue of vehicles equaled (and therefore probably actually exceeded) the storage capacity of the link. Queue overflows typically interfere with the operation of upstream facilities and are worth investigation. This report may be less useful if the analyst has split long sections of roadway into many short links. The result may be many "false alarms" for short links.

Depending on the software, the analyst may be able to define hotspots as queues of a minimum length that persist for a minimum time. The hot spots report then points the analyst to locations of persistent long queues (even those that do not overflow beyond the end of a link) during the simulation period.

7.2.2 Summarizing the Key Statistics

The analyst needs to determine how the key statistics produced by multiple simulation model run repetitions should be summarized.

Requirement for Multiple Repetitions

Microsimulation models rely upon random numbers to generate vehicles, select their destination, select their route, and to determine their behavior as they move through the network. No single simulation run can be expected to reflect any specific condition, such as the mean, the minimum, the maximum, or the 95 percentile. It is necessary to run the model several times with different random number seeds[15] in order to get the necessary output to determine mean, minimum, and maximum values. The analyst must then post-process the runs to obtain the necessary output statistics.

Summarizing the Results: Mean, Mode, Median

The mean is the average value of a variable over the multiple repetitions of the simulation model runs with different random number seeds.

[15] Those with more sophisticated statistical aptitude may elect to use "variance reduction techniques" employing a single, common random number seed to reduce the number of required repetitions. These techniques are described in: Joshi, S.S., Ajay K. Rathi, "Statistical Analysis and Validation of Multi-population Traffic Simulation Experiments", Transportation Research Record 1510, Transportation Research Board, Washington, D.C., 1995

Two other well-known statistical measures of a distribution are the mode and the median. The mode is the most frequently observed value in the distribution. The median is the value for which half the distribution falls above it and half falls below it.

Symmetric distributions, like the Normal distribution will tend to have identical values for the mean, median, and mode (see Exhibit 13). Skewed distributions, like the "F" distribution will have different values for the mean, median, and mode (see Exhibit 14). Many traffic observations (speeds, delay, travel time, and queues in particular) will tend to have skewed distributions because negative values are not possible and their mean values are not much greater than their standard deviations.

Exhibit 13. Normal Distribution

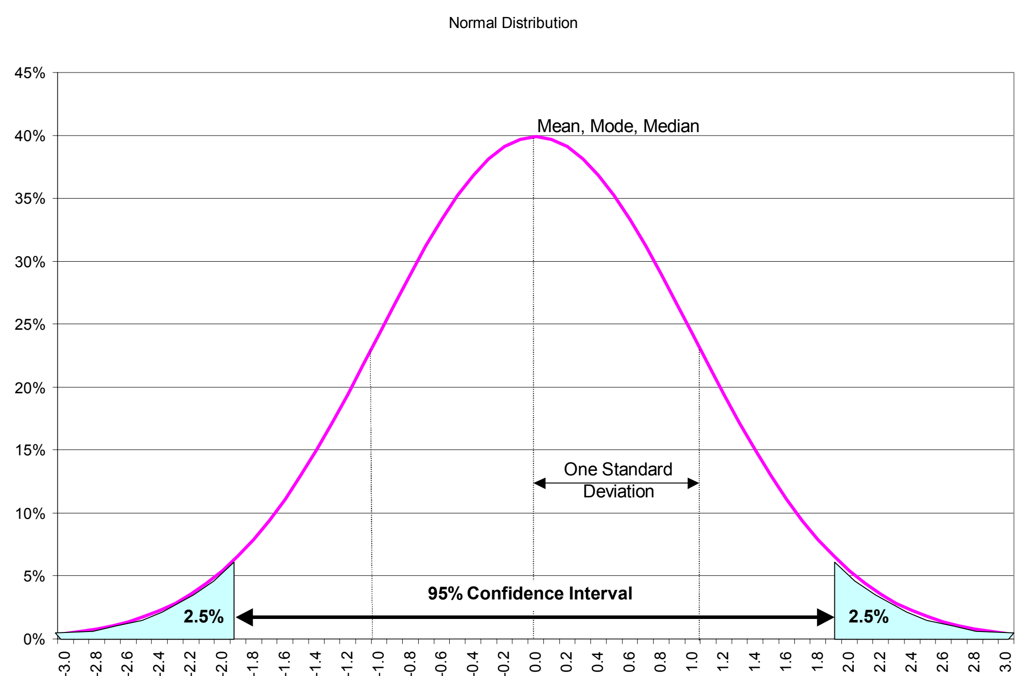
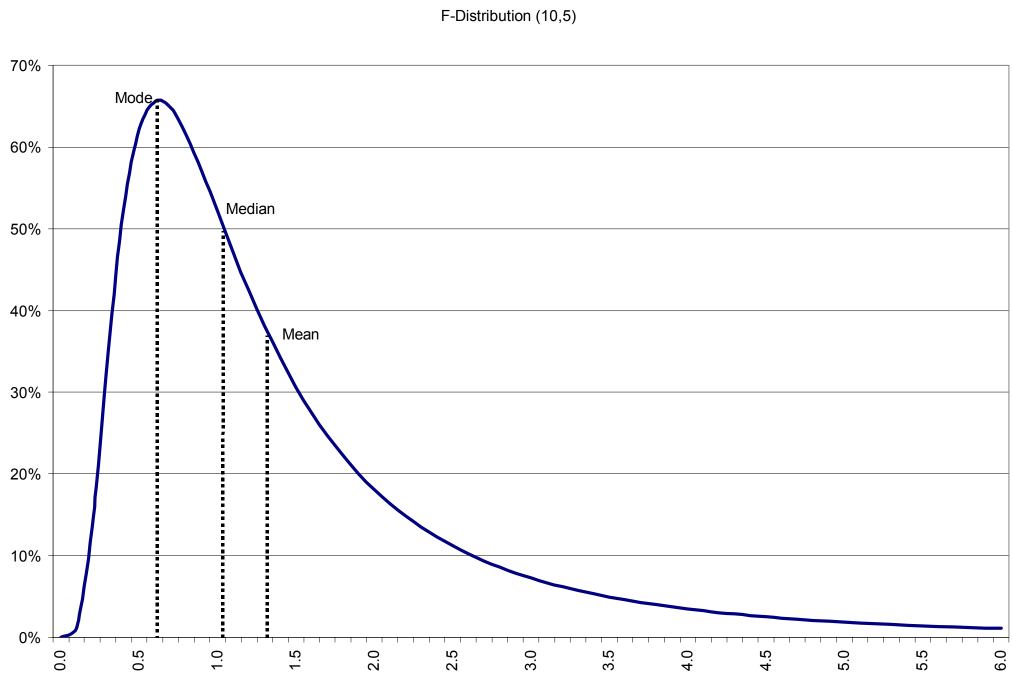


Exhibit 14. The "F" Distribution



The mean values of random samples taken from even skewed distributions will tend to be normally distributed, which allows the analyst to easily draw various statistical conclusions about the probability of obtaining the observed results. Less is known about the distribution of the median and modes of skewed distributions, so more advanced statistical techniques are required to draw conclusions about those statistical measures.

7.2.3 Mean and Standard Deviation

The most useful statistics for summarizing the results of many simulation model run repetitions are the mean and standard deviation of the results. The mean gives an indication of the center point of the results (and if the results are Normally distributed, the mean also is the most frequently observed result). The standard deviation gives an indication of how much the results varied. The two statistics together can be used to determine confidence intervals for the results and to test various hypotheses about the results.

Both the mean and the standard deviation should be reported in any summary of the results of multiple microsimulation runs. Readers of the report can then use these two statistics to perform various statistical tests and draw conclusions about the significance of the results.

The Number of Repetitions Required to Achieve a Target Precision

In order to determine the number of microsimulation model runs (with different random number seeds) needed to estimate the mean values (such as mean flow rate, mean speed, mean delay), the analyst must have a rough idea of the likely variance in the results produced by the model and a target interval in which the analyst wants the answer to lie with a certain amount of probability. Three pieces of information are required: The sample standard deviation, the desired length of the confidence interval, and the desired level of confidence.

The required number of repetitions of model runs (with different random number seeds) must be estimated by an iterative process.
A preliminary set of repetitions is usually required to make the first estimate of the standard deviation for the results.
The first estimate of the standard deviation is then used to estimate the number of repetitions required to make statistical conclusions about alternative highway improvements.
When all of the repetitions have been completed, the analyst then goes back and recomputes the standard deviation and required number of repetitions, based upon all of the completed repetitions.
If the required number of repetitions is less than or equal to the completed number of repetitions, then the analysis is complete. If not, then the analyst either relaxes the desired degree of confidence in the results or performs additional repetitions.

Estimation of Sample Standard Deviation

The standard deviation (an estimate of the variance) is required to estimate the number of repetitions. However, it takes a certain minimum number of repetitions to estimate the standard deviation in the first place! So either the analyst estimates the standard deviation directly (based upon past experience) or executes a few model run repetitions (each run using a different random number seed) and uses the equation below to compute an initial estimate of the sample standard deviation.

$$s^2 = \frac{\sum \left(x - \bar{x}\right)^2}{N - 1}$$

Equation 10

where:
s = the standard deviation
x = the variable (such as delay) for which the sample variance is desired.
$\bar{x}$ = the average value of the variable produced by the model runs.
N = number of model runs.

Unless the analyst already knows from experience the standard deviation, it is recommended that four repetitions be performed for the initial estimation of the standard deviation. This initial estimate is then revisited and revised later if and when additional repetitions are performed for the purposes of obtaining more precise estimates of mean values or for alternatives analysis.

Selection of Desired Confidence Level

The confidence level is the probability that the true mean lies within the target confidence interval. The analyst must decide to what degree he or she wishes to know the interval in which the true mean value lies. The usual approach is to pick a 95% level of confidence, but analysts may choose higher or lower levels of confidence. Higher percent levels of confidence require more repetitions.

Selection of Desired Confidence Interval

The confidence interval is the range of values within which the true mean value may lie. The length of the interval is at the discretion of the analyst and may vary according to the purpose for which the results will be used. For example, if the analyst is testing alternatives that are very similar, then a very small confidence interval will be desired to distinguish between the alternatives. If the analyst is testing alternatives with big differences, then a larger confidence interval can be tolerated. Smaller confidence intervals require more repetitions to achieve a given level of confidence. Confidence intervals less than half the value of the standard deviation will require a large number of repetitions to achieve reasonable confidence levels.

Computation of Minimum Repetitions

It is impossible to know in advance exactly how many model runs will be needed to determine a mean (or any other statistical value) to the analyst's satisfaction. However, after a few model runs, the analyst can make an estimate of how many more runs may be required to obtain a statistically valid result.

The required minimum number model repetitions is computed using the equation below:

$$CI_{1-\alpha\%} = 2 * t_{(1-\alpha/2),N-1} \frac{s}{\sqrt{N}}$$ *Equation 11*

Where:

$CI_{(1\text{-alpha})\%}$ = the (1-alpha)% confidence interval for the true mean, where "alpha" equals the probability of the true mean not lying within the confidence interval.

$t_{(1\text{-alpha}/2),N-1}$ = the Student's "t" statistic for the probability of two-sided error summing to "alpha" with N-1 degrees of freedom, where "N" equals the number of repetitions.

s = the standard deviation of the model results.

Note that when solving this equation for "N", it will be necessary to iterate until the estimated number of repetitions matches the number of repetitions assumed when looking up the "t" statistic. The exhibit below shows the solutions to the above equation for in terms of the minimum number of repetitions for various desired confidence intervals and desired degrees of confidence.

Exhibit 15. Minimum Repetitions to Obtain Desired Confidence Interval

Desired Range (CI/S)	Desired Confidence	Minimum Repetitions
0.5	99%	130
0.5	95%	83
0.5	90%	64
1.0	99%	36
1.0	95%	23
1.0	90%	18
1.5	99%	18
1.5	95%	12
1.5	90%	9
2.0	99%	12
2.0	95%	8
2.0	90%	6

Notes to Exhibit:

1. Desired Range = desired confidence interval (CI) divided by standard deviation (S).

2. For example if the standard deviation in the delay is 1.5 seconds, and the desired confidence interval is 3.0 seconds at a 95% confidence level, then it will take 8 repetitions to estimate the mean delay to within plus or minus 1.5 seconds.

7.2.4 The 95 Percentile Result

The 95 percentile result is also a very useful statistic for indicating how bad conditions can get. It is especially useful for assessing the likelihood of queues overflowing the available storage.

The analyst might be tempted to select the model run repetition with the worst result to evaluate whether or not queue overflows will be a problem. However, since the model runs are only a sample of the universe of possible outcomes (the worst outcome in the sample may not truly be the worst possible outcome), it is better to compute the 95% probable worst outcome.

The 95 percentile result is computed based upon the mean of the observed sample runs and an assumed distribution of the results. It is convenient to assume that the maximum queues observed in each model run are normally distributed with mean and standard deviation equal to the sample mean and sample standard distribution[16] [17].

$$95\%Queue = m + 1.64 \bullet s$$ *Equation 12*

Where:

95%Queue =	The queue length that has a 95% probability of not being exceeded during the analysis period (may be exceeded more or less often on other days under different conditions).
m =	The maximum observed queue length for each model repetition, averaged over all the repetitions.
s =	The standard deviation of the maximum observed queue lengths for the model repetitions.

Note that this equation results in a longer predicted queue length than the traditional 95 percentile design criteria for left turn pockets at traffic signals, since that design criterion is based upon 95 percent of the signal cycles within a simulation period. If the traditional design criterion is desired, then the analyst should substitute the mean (and standard deviation) of the observed maximum queue length per cycle into the equation.

7.2.5 Note on Comparison of Aggregate Statistics to Fundamental Traffic Flow Relationships

The analyst is going to run into trouble when comparing mean results across several model repetitions against fundamental traffic flow relationships. This is because the basic traffic flow

[16] Many statistical phenomena approximate a Normal distribution at large sample sizes. Even though most analysts in microsimulation usually work with relatively few model repetitions, the Normal distribution assumption is usually good enough for most analyses.

[17] Note that when computing the 95 percentile queue at the macroscopic level it is typical to assume that the arrivals of vehicles are Poisson distributed. Microsimulation models predict the arrival patterns of vehicles, so the Poisson distribution assumption is not necessary when estimating 95 percentile queues using microsimulation data.

relationship between density, volume, and speed applies to point measurements and not to averages of a series of point measurements.

According to the fundamental equation of traffic flow, the flow rate Q at any particular point (i) and time is equal to the density divided by the speed at that point.

$$Q_i = k_i \bullet s_i \qquad \textit{Equation 13}$$

However, this does not mean that the mean flow rate over time is equal to the mean density over time divided by the mean speed over time (except under very limited circumstances[18]).

$$\overline{Q} = \frac{1}{N}\sum k_i \bullet s_i \neq \tfrac{1}{N}\sum k_i \bullet \tfrac{1}{N}\sum s_i \qquad \textit{Equation 14}$$

Where:
Q-bar = the mean flow rate
N = number of observations
k_i = the i'th observation of density.
s_i = the i'th observation of speed.

This does not reduce the validity of the mean speed, flow, or density. The mean values of these traffic measures just do not follow the fundamental traffic flow equation.

In order to compare model performance to the fundamental traffic flow equation the analyst should select one model repetition for evaluation and gather point data on speed, headways, and flow. The data should be as temporally and as geographically disaggregate as possible.

7.3 Correction of Biases in Results

In order to make a reliable comparison of the alternatives, it is very important that the vehicle congestion under each alternative be accurately tabulated by the model. This means that congestion (vehicle queues) should not extend physically or temporally beyond the geographic or temporal bounds of the simulation model. The tabulated results should also exclude the initial and unrealistic warm-up period when vehicles are first loaded on the network. Congestion that overflows the time or geographic limits of the model will not normally be reported by the model, which can bias the comparison of alternatives.

Ideally, the simulation results for each alternative would have the following characteristics:
1. The warm-up period before the system reaches equilibrium for the simulation period is excluded from the tabulated statistics;
2. All of the congestion begins and ends within the simulation study area;
3. No congestion begins or ends outside of the simulation period; and
4. No vehicles are unable to enter the network from any zone (source) during any time step of the simulation.

[18] All uniform density and uniform speed over the time period

7.3.1 Isolating the Warm-up Period

Simulation model runs usually start with zero vehicles on the network. If the simulation output is being compared to field measurements (as in calibration), then the artificial period where the simulation model starts out with zero vehicles (the warm-up period) must be excluded from the reported statistics for system performance.

The warm-up period may be specified by the analyst or it may be determined by the software.

Typically, the number of vehicles present at any one time on the network is used to determine if the model has reached equilibrium and therefore can start tallying performance statistics for the network. Once the number of vehicles present on the network ceases to increase by a minimum specified amount, then the warm-up period is deemed to be concluded.

If the number of vehicles and the mean speed do not level off within the first 15 minutes, it could be that the demand coded by the analyst for the system is greater than the system capacity. In this case congestion will never level off. This will result in less accurate congestion statistics since the system never clears the congestion. The analyst should consider extending the starting and end times of the simulation to incorporate lower demand periods before and after the peak period.

If it is not feasible to extend the simulation period to uncongested time periods, the analyst should choose a warm-up period that is equal to twice the estimated travel time at free-flow conditions to traverse the length of the network. For example: if the freeway being modeled is 5 miles long, it takes roughly 5 minutes to traverse the length of it at free-flow speed, so the warm-up period is set at 10 minutes.

7.3.2 Correction of Output for Blocked Vehicles

If either or both alternatives are severely congested then the simulation may be unable to load vehicles onto the network. Some may blocked from entering the network on the periphery. Some may be blocked from being generated on internal links. These blocked vehicles will not typically be included in the travel time (VHT) or delay statistics for the model run[19].

The best solution is to extend the network back to include the maximum back of the queue. If this is not feasible, then the analyst should correct the reported VHT to account for the unreported delay to blocked vehicles.

Microsimulation software will usually tally the excess queue that backs up outside the network as "blocked" vehicles (vehicles unable to enter the network) for each time step. The analyst sums up the number of software reported blocked vehicles for each time step of the simulation and multiplies by the length of each time step (in units of hours) to obtain the vehicle-hours of delay. The delay due to blocked vehicles is added to the model reported vehicle-hours traveled (VHT) for each model run.

---

[19] The analyst should verify with the software documentation or developer how statistics on blocked vehicles are accumulated in the travel time and delay summaries.

7.3.3 Correction of Output for Congestion Extending Beyond End of Simulation Period

Vehicles queues that are present at the end of the simulation period may affect the accumulation of total delay and distort the comparison of alternatives (Cyclical queues at signals can be neglected.). The "build project" alternative may not look significantly better than the "no-build" option if the simulation period is not long enough to capture all of the benefits.

The best solution is to extend the simulation period until all of the congestion that built up over the simulation period is served. If this is not feasible, the analyst can make a rough estimate of the uncaptured "residual" delay by computing how many vehicle hours it would take to clear the queue using the equation given below.

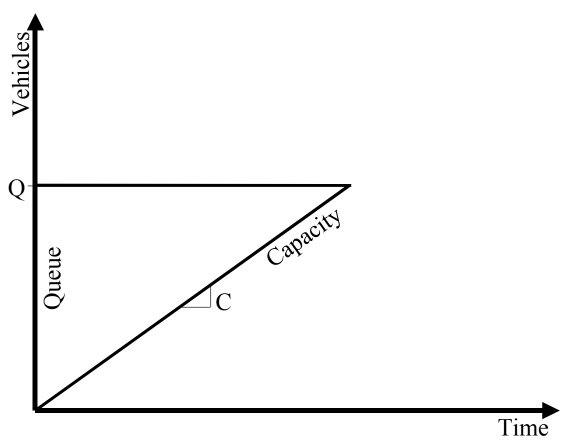
$$VHT(Q) = \frac{Q^2}{2 \bullet C}$$ *Equation 15*

Where:

- VHT(Q) = The extra vehicle-hours traveled of delay attributable to a queue present at the end of the simulation period.
- Q = The number of vehicles remaining in the queue at the end of the simulation period.
- C = The discharge capacity of the bottleneck in vehicles per hour

The equation computes the area of the triangle created by the queue and the discharge capacity after the end of the simulation period (see exhibit below).

Exhibit 16. Computation of Uncaptured Residual Delay at End of Simulation Period



Note that this is not a complete estimate of the residual delay since it ignores the interaction of left over vehicles from the simulation period interfering with traffic arriving in later time periods.

7.4 Interpretation of Animation Results

The analyst is unable to sum or average animation results, so it is necessary to take a different approach when reviewing animation results. The analyst must select one or more model run repetitions to review and then focus his or her review on the key aspects of each animation result.

7.4.1 Selection of Representative and/or Worst Case Repetition

The analyst has to decide if he or she will review the "typical" case output, or the worst case output, or both. The "typical" case might give an indication of "average" conditions for the simulation period. The "worst case" is useful for determining if the transportation system will experience a failure and for viewing the consequences of that.

The next question that the analyst must decide is then how to identify which model repetition represents "typical" conditions and which repetition reflects "worst case" conditions. The total vehicle travel time (VHT) may be a useful indicator of typical and worst case conditions. The analyst might also select other measures, such as number of occurrences of blocked links (links with queue overflows), or delay.

If VHT (vehicle-hours traveled) is selected as the measure and the analyst wishes to review the "typical" case, then the analyst would pick the model run repetition that had the total VHT that came closest to falling in the median of the repetitions (50% of the repetitions had VHT less than that, and 50% has more VHT). If the analyst wished to review the worst case, then they would select the repetition that had the highest VHT.

The pitfall of using a global summary statistic (like VHT) to select a model run repetition for review is that while the system may be performing on average, this does not mean that each link and intersection in the system is experiencing average conditions. The median VHT repetition may actually have the worst performance for a specific link.

If the analyst is focused on a specific link or intersection, then the analyst should select some statistic related to vehicle performance on that specific link or intersection for selecting the model run repetition for review.

7.4.2 Review of Key Events in Animation

The key events to look for in reviewing animation are the formation of persistent queues. Cyclical queues at signals that clear each cycle are not usually as critical, unless they block some other traffic movement.

The analyst should not confuse the secondary impacts of queues (one queue blocking upstream movements and creating a secondary queue) with the root cause of the queuing problem. Eliminating the cause of the first or primary queue may eliminate all secondary queuing. Thus the analyst should focus on the few minutes just prior to formation of a persistent queue to identify the causes of the queuing.

7.5 Interpretation of Numerical Results

When interpreting the numerical output produced by microsimulation models it is crucial to understand the methods used by the models to accumulate the results and how these methods may differ from traditional methods of estimating performance measures.

7.5.1 Delay and Intersection Level of Service

It is often valuable when explaining microsimulation model results to the general public to report the estimated delays in terms of Highway Capacity Manual (HCM) levels of service. Delay is used in the HCM to estimate the level of service for signalized and unsignalized intersections. The analyst though should account for the distinctions between the ways microsimulation software and the HCM define delay and accumulate it for the purpose of assessing level of service.

Accumulation of Mean Delay Statistics

The Highway Capacity Manual bases its level of service grades for intersections on estimates of mean control delay for the highest consecutive 15-minute period within the hour. The microsimulation output for each run must be accumulated over a similar 15 consecutive minute time period and averaged over several runs with different random number seeds in order to achieve a comparable result.

All microsimulation models assign delay to the segment in which it occurs. For example, the delay associated with a traffic signal may be parceled out over several approach links to the intersection, if the queues extend beyond one link upstream from the intersection. Thus, when analysts seek to accumulate the delay at a signal they should investigate whether they should be accumulating it for more than just the single approach links to the signal.

Definition of Delay

There are differences in how the HCM and the microsimulation software define delay, some of which the analyst may be able to correct for, and some of which the analyst will have to recognize when comparing level of service results between the Highway Capacity Manual and microsimulation models.

The Year 2000 Highway Capacity Manual divides delay into various components and defines them as follows:

Delay:	The additional travel time experienced by a driver, passenger, or pedestrian.
Total delay:	The sum of all components of delay for any lane group, including control delay, traffic delay, geometric delay, and incident delay.
Control delay:	The component of delay that results when a control signal causes a lane group to reduce speed or to stop; it is measured by comparison with the uncontrolled condition.
Traffic delay:	The component of delay that results when the interaction of vehicles causes drivers to reduce speed below the free-flow speed.
Geometric delay:	The component of delay that results when geometric features cause vehicles to reduce their speed in negotiating a facility.
Incident delay:	The component of delay that results from an incident, compared with the no-incident condition.
Stop time:	A portion of control delay when vehicles are at a complete stop.

The HCM does not define the yardstick against which the "additional travel time" is measured in order to determine delay.

All microsimulation software compute total delay, but they vary as to its division into components, and the measurement of free-flow time against which total delay is computed. Most microsimulation software compare the actual travel time to the theoretical mean travel time at the user coded free-flow speed to obtain total delay. One software package computes total delay on a vehicle-by-vehicle basis by comparing actual travel times to individual drivers' desired free-flow speeds (as opposed to the mean free-flow speed used by other software). The analyst needs to review the software documentation and seek additional documentation from the software vendor to understand how delay is computed by the software.

Estimating Control Delay from Total Delay or Stopped Delay

Control delay, a subset of total delay, is currently not computed by most software packages. The difficulty is isolating control related delay from other causes of delay (such as traffic and geometric delay) on the approach and departure legs of an intersection. A second problem for microsimulation models for accumulating control delay is determining how many street segments upstream and downstream from a signal to include in the computation.

If the software does not output total delay, then the analyst can use the "rule of thumb" that total delay is 30% greater than stopped delay to estimate total delay (if not reported by software).

Total delay on all approach links occupied by queues of vehicles leading up to the signal must be accumulated[20].

The control delay can then be estimated if the analyst can segregate the geometric and traffic flow related delay components from total delay. The analyst can conservatively assume these other components are negligible and use total delay as control delay for the purposes of computing intersection level of service. The result will be to under estimate the level of service (predict it will be worse than it really is).

7.5.2 Density and Freeway/Highway Level of Service

If microsimulation model reports of vehicle density are to be reported in terms of their level of service implications, it is important to first translate the densities reported by the software into the densities used by the Highway Capacity Manual to report level of service for uninterrupted flow facilities. Density is NOT used as a level of service measure for interrupted flow facilities such as city streets with signals and stop sign intersections.

The Year 2000 Highway Capacity Manual defines freeway and highway level of service based upon the average density of passenger car equivalent vehicles in a section of highway for the peak 15-minute period within an hour. For ramp merge and diverge areas, only the density in the rightmost 2-lanes is considered for level of service. For all other situations, the density across all lanes is considered. Trucks and other heavy vehicles must be converted to passenger car equivalents using the values contained in the HCM according to vehicle type, facility type, section type, and grade.

7.5.3 Queues

The Year 2000 Highway Capacity Manual (HCM) defines a queue as: "*A line of vehicles, bicycles, or persons waiting to be served by the system in which the flow rate from the front of the queue determines the average speed within the queue. Slowly moving vehicles or people*

---

[20] This can be a complex exercise if the queues back up through an upstream signal. The analyst must then arbitrarily assign portions of the delay to each signal. One option would be to compute an average delay per vehicle for both signals and report an average level of service.

joining the rear of the queue are usually considered part of the queue. The internal queue dynamics can involve starts and stops. A faster-moving line of vehicles is often referred to as a moving queue or a platoon."

The HCM defines the "Back of Queue" as: "*The distance between the stop line of a signalized intersection and the farthest reach of an upstream queue, expressed as a number of vehicles. The vehicles previously stopped at the front of the queue are counted even if they begin moving.*"

These definitions are not implementable within a microsimulation environment, since "waiting to be served" and "slowly" are not easily defined. Consequently alternative definitions based upon maximum speed, acceleration, and proximity to other vehicles have been developed for use in microsimulation.

There are a wide variety of definitions as to what constitutes a queue in a microsimulation environment. Most software define a vehicle to be in a queue if it is completely stopped or moving below some threshold speed. Some software allow the user to set the threshold speed. Some software require in addition that at least 2 slow moving vehicles be present in close proximity before they are counted as being in a queue. Other software also require that the acceleration/deceleration rate be below a certain threshold.

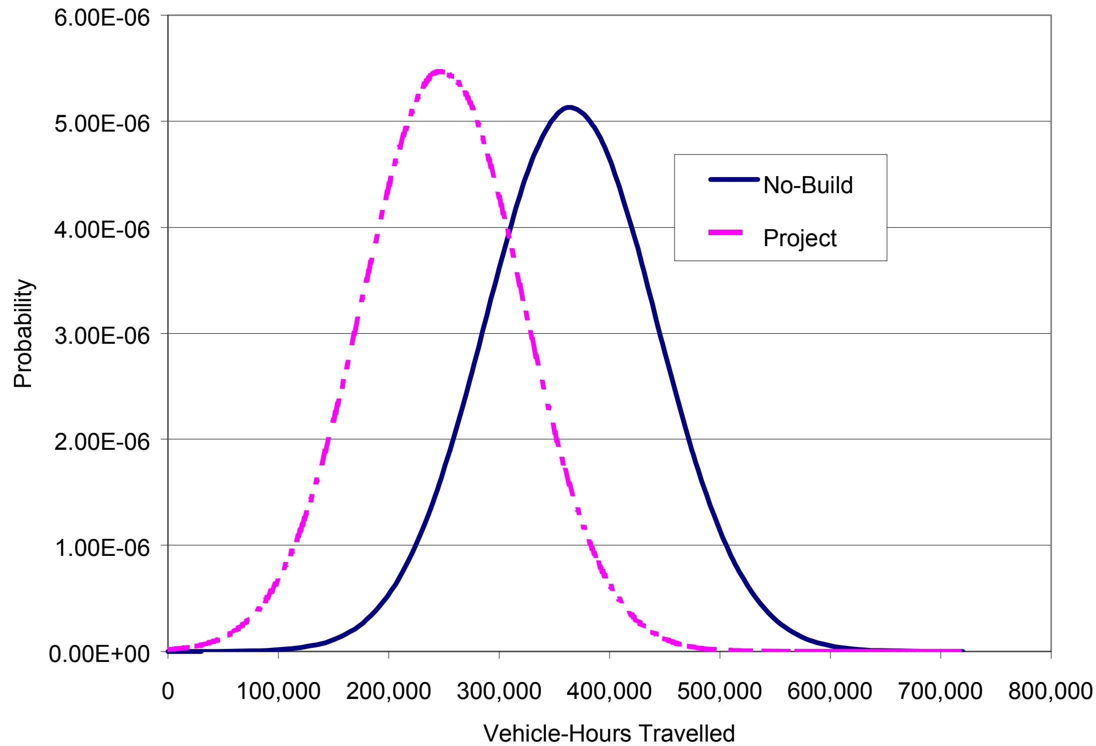
Note that for all microsimulation programs the number of queued vehicles counted as being in a particular turn pocket lane or through lane cannot exceed the storage capacity of that lane. Any overflow is assigned to the upstream lane and link where it occurs, not to the downstream cause of the queue. Unlike macroscopic approaches that assign the entire queue to the bottleneck that causes it, microsimulation models can only observe the presence of a queue; they currently do not assign a cause to it. So in order to obtain the 95% queue length it may be necessary to temporarily increase the length of the storage area so that all queues are appropriately tallied in the printed output.

7.6 Hypothesis Testing of Alternatives

When the microsimulation model is run several times for each alternative, the analyst may find that the variance in the results for each alternative is close to the difference in the mean results for each alternative (See example plot of the simulation results for two very similar alternatives in Exhibit 17).

How is the analyst to determine if the alternatives are significantly different? To what degree of confidence can the analyst claim that the observed differences in the simulation results are due to the differences in the alternatives and not just the result of using different random number seeds? This is the purpose of statistical hypothesis testing. Hypothesis testing determines if the analyst has performed an adequate number of repetitions for each alternative to truly tell the alternatives apart at the analyst's desired level of confidence.

Exhibit 17. Illustration of Variance in Simulation Results for Two Very Similar Alternatives



7.6.1 Estimation of the Required Number of Model Repetitions

This section gives the analyst an estimate of the minimum number of model run repetitions that would be required to determine if two alternatives with results a given distance apart are significantly different. This estimate however requires a preliminary estimate of the standard deviation of the model run results for the alternatives, which in turns requires a few preliminary runs in order to estimate the standard deviation.

The procedure therefore is to
1. Perform a preliminary set of model run repetitions for each alternative.
2. Estimate the standard deviation and the mean difference between the alternatives from the preliminary runs, and then compute the required number of runs using the equations in this subsection.
3. If the required number of runs is greater than the preliminary number of runs, the analyst performs the additional repetitions for each alternative, and recomputes the mean difference and standard deviation using the augmented set of model run repetitions.

Estimation of Pooled Standard Deviation

The analyst should first perform about 4 model repetitions of each alternative to estimate the pooled standard deviation for all alternatives according to the following equation:

$s_p^2 = \frac{s_x^2 + s_y^2}{2}$ *Equation 16*

Where:
s_x = the standard deviation of model run results for alternative "x"
s_y = the standard deviation of model run results for alternative "y"

The preliminary model repetitions used to estimate the pooled estimate of the standard deviation of the model run results can also be used to estimate the likely difference of the means for each alternative.

Selection of Desired Confidence Level

A 95% confidence is often selected for hypothesis testing. This means that there is a 5% chance (alpha) that the analyst will mistakenly reject the null hypothesis that the means come from identical distributions with the same mean, when they are indeed identical (This is known as a Type I error). If a higher confidence level is desired, it comes at the cost of increasing the likelihood of making a Type II error (accepting the null hypothesis when it is really false).

The project objective may determine the desired confidence level. If the objective is to design an alternative with a 95% probability that it will provide significant improvements over the current facility, then this is the appropriate confidence level for the determination of the number of model repetitions required.

Selection of Minimal Difference in Means

The likely minimal difference in means between alternatives must be selected by the analyst. This is the target sensitivity of the simulation tests of the alternatives. Alternatives with mean results farther apart than this minimal difference will be obviously different. Alternatives with mean results closer together than this minimal difference will be considered to be indistinguishable.

The project objectives have some bearing on the selection of the minimal difference to be detected by the simulation tests. If the project objective is to design a highway improvement that reduces mean delay by at least 10%, then the tests should be designed to detect if the alternatives are truly at least 10% apart.

The preliminary model repetitions used to estimate the pooled estimate of the standard deviation of the model run results can also be used to estimate the likely difference of the means for each alternative. The smallest observed difference in these preliminary runs would be the selected minimal difference of means to be used in determining the required number of repetitions.

Computation of Minimum Repetitions

Assuming that the analyst wishes to reject the null hypothesis that the means of the two most similar alternatives are equal with only an "alpha"% chance of error against the counter hypothesis that the mean of alternative "x" is different than "y", then the number of repetitions required can be computed according to the following equation[21]:

[21] If the analyst intends to perform hypothesis tests on only a few pairs of alternatives, then the equation provided should be sufficiently accurate. However, if the analyst plans to perform hypothesis testing of all possible pairs of alternatives, then this equation will underestimate the required number of repetitions needed to achieve the desired confidence level. The analyst

$$|\bar{x} - \bar{y}| > t_{(1-\alpha/2);2n-2} \bullet s_p \sqrt{\tfrac{2}{n}}$$ *Equation 17*

where:

$\lvert\bar{x} - \bar{y}\rvert =$	The absolute value of the estimated difference between the mean values for the two most similar alternatives "x" and "y".
$s_p =$	the pooled estimate of the standard deviation of model run results for each alternative.
n =	number of model repetitions required for each alternative
t =	the student's "t" statistic for a confidence level of (1-alpha) and "2n-2" degrees of freedom[22].

Note that when solving this equation for "N", it will be necessary to iterate until the estimated number of repetitions matches the number of repetitions assumed when looking up the "t" statistic. The table below provides solutions to this equation for various target mean difference ranges and levels of confidence.

Exhibit 18. Minimum Repetitions to Distinguish Alternatives

Minimum Difference of Means	Desired Confidence	Minimum Repetitions Per Alternative
0.5	99%	65
0.5	95%	42
0.5	90%	32
1.0	99%	18
1.0	95%	12
1.0	90%	9
1.5	99%	9
1.5	95%	6
1.5	90%	5
2.0	99%	6
2.0	95%	4
2.0	90%	4

Notes to Exhibit:

1. The minimum difference in means is expressed in units of the pooled standard deviation: $\frac{|\bar{x} - \bar{y}|}{S_p}$.

2. For example if the pooled standard deviation in the delay for two alternatives is 1.5 seconds, and the desired minimum detectable difference in means is 3.0 seconds delay at a 95% confidence level, then it will take 4 repetitions of each alternative to reject the hypothesis that the observed differences in the simulation results for the two alternatives could be due to random chance.

should consult: Chapter 12, Introduction to Between Subjects ANOVA, "All pairwise comparisons among means" in David M. Lane, Hyperstat OnLine, An Introductory Statistics Book and Online Tutorial for Help in Statistics, Rice University. (http://www.davidmlane.com/hyperstat).

[22] Note this is a two-sided "t" test for the null hypothesis that the means are equal versus the hypothesis that they are different.

7.6.2 Hypothesis Testing For Two Alternatives

To determine whether the simulation outputs provide sufficient evidence that one alternative is better than the other (for example: Build Project versus No-Build), it is necessary to perform a statistical hypothesis test of the difference of the mean results for each alternative. A null hypothesis is specified, "The model predicted difference in VHT for the two alternatives occurred by random chance. There really is no significant difference in the mean travel time between the alternatives". A statistic is computed for a selected level of confidence, and if the difference between the two means is less than that statistic, then the null hypothesis is accepted and it is concluded that there is insufficient evidence to prove that the one alternative is better than the other. The analyst either makes more model repetitions of each alternative (to improve the sensitivity of the test) or relaxes his or her standards (confidence level) for rejecting the null hypothesis.

The specification of the problem is:

Null Hypothesis:

$$H_0 : \mu_x - \mu_y = 0$$

against

$$H_1 : \mu_x - \mu_y \neq 0$$

where:
μ_x = the mean VHT (or some other measure) for alternative "x".
μ_y = the mean for alternative "y".

This is a two-sided "t" test with the following optimal rejection region for a given alpha (acceptable Type 1 error).

$$\bar{x} - \bar{y} > t_{(1-\alpha/2);(n+m-2)} \bullet s_p \sqrt{\tfrac{1}{n} + \tfrac{1}{m}} \qquad \textit{Equation 18}$$

$\bar{x} - \bar{y} =$ The absolute value of the difference in the mean results for alternative "x" and alternative "y".
s_p = the pooled standard deviation
t = the Student's "t" distribution for a level of confidence of (1-alpha) and (n+m-2) degrees of freedom.
n = sample size for alternative "x"
m = sample size for alternative "y"

$$s_p^2 = \frac{(n-1)s_x^2 + (m-1)s_y^2}{(m+n-2)} \qquad \textit{Equation 19}$$

s_p = the pooled standard deviation
s_x = the standard deviation of results for alternative "x".
s_y = the standard deviation of results for alternative "y".
n = sample size for alternative "x"

m = sample size for alternative "y"

The probability of mistakenly accepting the Null Hypothesis is alpha (usually set to 5%, to get a 95% confidence level test). This is Type I error.

There is also the chance of mistakenly rejecting the null hypothesis. This is called Type II error and it varies with the difference between the sample means, their standard deviation, and the sample size[23].

7.6.3 Hypothesis Testing For Multiple Alternatives

When performing hypothesis testing on more than one pair of alternatives it is most efficient to first determine if <u>any</u> of the alternatives are significantly different from the others. An analysis of variance (ANOVA) test is performed to determine if the mean results for any of the alternatives are significantly different from the others.

1. If the answer is "yes", the analyst goes on to test specific pairs of alternatives.
2. If the answer is "no", then the analysis is complete, or the analyst runs more model repetitions for each alternative to improve the ability of the ANOVA test to discriminate among the alternatives.

Analysis of Variance (ANOVA) Test

Analysis of variance (ANOVA) has three basic requirements:

1. Independence of Samples (random samples)
2. Sampling distribution of means is normal
3. Equal variances of groups

Levine's test of heteroscedasticity can be used for testing whether or not the variances of model run results for each alternative are similar. Less powerful non-parametric tests, such as Kruskal-Wallis test (K-W Statistic) can be performed if the requirements of ANOVA cannot be met.

However, ANOVA is tolerant of modest violations of these requirements, and may still be a useful test under these conditions. ANOVA will tend to be conservative if its requirements are not completely met (less likely to have Type I error with a lower power of the test to correctly reject the null hypothesis).

To perform the ANOVA test first compute the test statistic:

$$F = \frac{MSB}{MSW} \qquad \textit{Equation 20}$$

Where:
F = the test statistic
MSB = the mean square error between alternatives (formula provided below)
MSW = the mean square error among model results for the same alternative (within alternatives)

[23] Analysts should consult standard statistical textbooks for tables on the Type II errors associated with different confidence intervals, and sample sizes.

The formulae below show how to compute MSB and MSW.

$$MSB = \frac{\sum_{i=1}^{g} n_i \bullet (\bar{x}_i - \bar{x})^2}{g-1} \qquad \textit{Equation 21}$$

where:
MSB = Mean square error between alternatives (i=1 to g)
n_i = number of model runs with different random number seeds for alternative "i".
$\bar{x}_i =$ the mean value for alternative "i".
$\bar{x} =$ the mean value averaged across all alternatives and runs.
g = the number of alternatives

And:

$$MSW = \frac{\sum_{i=1}^{g} (n_i - 1) \bullet s_i^2}{N-g} \qquad \textit{Equation 22}$$

where:
MSB = Mean square error between alternatives (i=1 to g)
n_i = number of model runs with different random number seeds for alternative "i".
$s_i^2 =$ the variance of the model run results for alternative "i".
N = the total number of model runs summed over all alternatives.
g = the number of alternatives

The null hypothesis of equal means is rejected if:

$$F > F_{1-alpha,g-1,N-g} \qquad \textit{Equation 23}$$

Where
$F_{1-alpha,g-1,N-g}$ = the "F" statistic for a Type I error of "alpha" (alpha is usually set at 5% for a 95% confidence level) and g-1 and N-g degrees of freedom. N is the total number of model runs summed over all alternatives. "g" is the number of alternatives.

The job is then to identify which alternative is best, and that is described in the next step below.

If the null hypothesis cannot be rejected then the analyst is either done (there is no statistically significant difference between any of the alternatives at the 95% confidence level) or the analyst should consider reducing the level of confidence to below 95% or implementing more model runs per alternative to improve the sensitivity of the ANOVA test.

Pairwise Tests of Some Pairs of Alternatives

If the analyst wishes to perform hypothesis tests for only a few of the potential pairs of alternatives, it is possible to get by using the standard "t" test described above for comparing a single pair of alternatives.

$$\bar{x} - \bar{y} > t_{(1-\alpha/2);(n+m-2)} \bullet s_p \sqrt{\tfrac{1}{n} + \tfrac{1}{m}}$$ *Equation 24*

$\bar{x} - \bar{y} =$ The absolute value of the difference in the mean results for alternative "x" and alternative "y".

s_p = the pooled standard deviation

t = the Student's "t" distribution for a level of confidence of (1-alpha) and (n+m-2) degrees of freedom.

n = sample size for alternative "x"

m = sample size for alternative "y"

$$s_p^2 = \frac{(n-1)s_x^2 + (m-1)s_y^2}{(m+n-2)}$$ *Equation 25*

s_p = the pooled standard deviation

s_x = the standard deviation of results for alternative "x".

s_y = the standard deviation of results for alternative "y".

n = sample size for alternative "x"

m = sample size for alternative "y"

In one merely wishes to test that the "best" alternative is truly superior to the next best alternative, then the test need be performed only once.

If one wishes to test other possible pairs of alternatives (such as second best versus third best) it is possible to still use the same "t" test, but the analyst should be cautioned that the level of confidence diminishes each time the test is actually performed (even if the analyst retains the same nomimal 95% confidence level in the computation, the mere fact of repeating the computation reduces its confidence level. For example a 95% confidence level test repeated twice would have a net confidence level for both tests of $(0.95)^2$, or 90%.

Some experts however have argued that the standard "t" test is still good enough for multiple paired tests, even at its reduced confidence level.

Pairwise Tests of All Pairs of Alternatives

If the analyst wishes to preserve the high confidence level for all possible paired tests of alternatives, then they should adopt the more conservative John Tukey "Honestly Significantly Different" (HSD) test to determine if the null hypothesis, that the two means are equal, can be rejected[24]. The critical statistic is:

[24] Adapted from Reference: "Hyperstat OnLine, An Introductory Statistics Book and Online Tutorial for Help in Statistics", by David M. Lane, Associate Professor of Psychology, Statistics, and Administration, Rice University. (www.davidmlane.com/hyperstat)

$$t_s = \frac{|\bar{x}_i - \bar{x}_k|}{\sqrt{\frac{MSE}{n}}}$$ *Equation 26*

Where:

t_s =	the studentized "t" statistic
$\bar{x}_i$ =	the mean value for alternative "i".
$\bar{x}_k$ =	the mean value for alternative "k".
MSE =	the mean square error = MSB + MSW
N =	the number of model runs with different random number seeds for each alternative. (If the number of runs for each alternative is different then use the harmonic mean of the number of runs for each alternative.)

Reject the null hypothesis that the mean result for alternative "i" is equal to that for alternative "k" if:

$$t_s > t_{1\text{-}alpha,\, g\text{-}1}$$ *Equation 27*

Where

$t_{1\text{-alpha, g-1}}$ = The student's "t" statistic for a desired Type I error of "alpha" (alpha is usually set at 5% to obtain a 95% confidence level) and "g-1" degrees of freedom, with "g" equal to the total number of alternatives tested, not just the two being compared in each test.

Some experts consider the HSD test to be too conservative, failing to reject the null hypothesis of equal means when it should be rejected. The price of retaining a high confidence level (same as retaining a low probability of a Type I error) is a significantly increased probability of making a Type II error (accepting the null hypothesis when it is really false).

7.6.4 What To Do if the Null Hypothesis Cannot Be Rejected

If the null hypothesis of no significant different in the mean results for the alternatives cannot be rejected then the analyst has several options:

1. Increase the number of model run repetitions per alternative until the simulation performance of one alternative can be distinguished from the other.

2. Reduce the confidence level from 95% to a lower level where the two alternatives are significantly different, and report the lower confidence level in the results.

3. Accept the results that the two alternatives are not significantly different.

7.7 Sensitivity Analysis

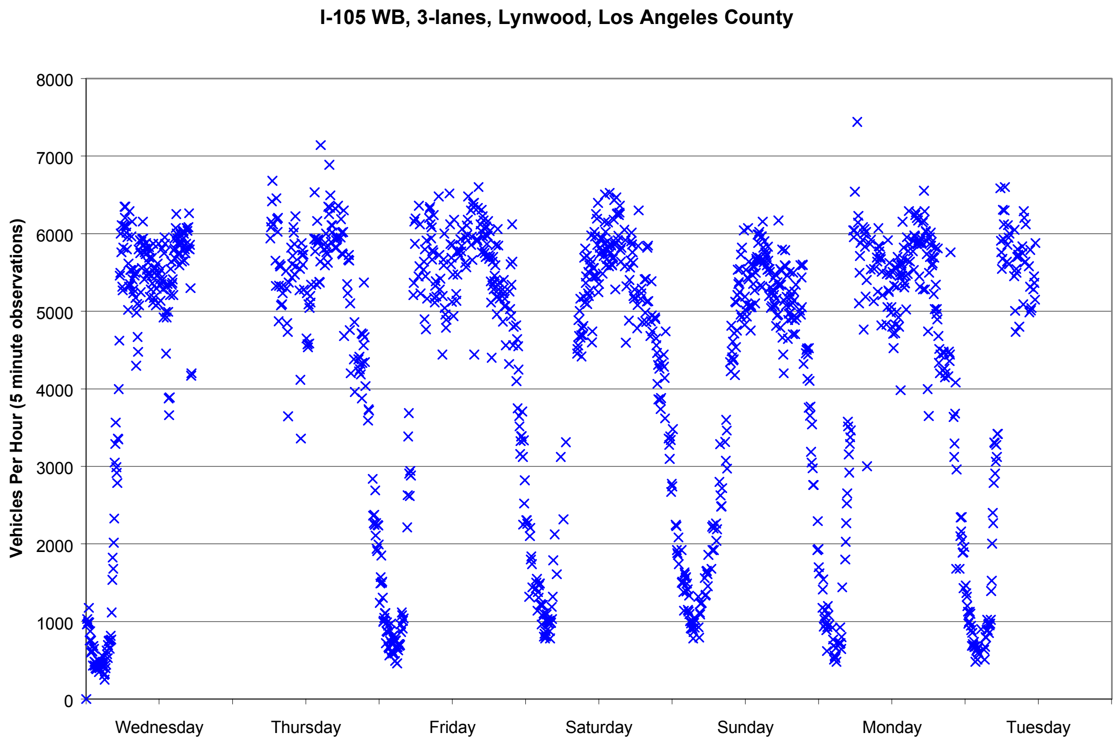
A sensitivity analysis is performed to develop an understanding of the sensitivity of the conclusions of the study to changes in the underlying assumptions behind the analysis. Additional model runs are made with changes in demand levels and key parameters in order to determine the robustness of the conclusions of the alternatives analysis. The analyst may vary:

1. Demand,
2. Street improvements assumed to be in place outside the study area, and/or
3. Parameters for which the analyst has little information.

A sensitivity analysis of different demand levels is particularly valuable when evaluating future conditions. Demand forecasts are generally less precise than the ability of the microsimulation model to predict their impacts on traffic operations. A 10% change in demand can cause a facility to go from 95% of capacity to 105% of capacity with a concomitant massive change in the predicted delay and queuing for the facility. The analyst should estimate the confidence interval for the demand forecasts and test the microsimulation at the high end of the confidence interval to determine if the alternative still operates satisfactorily at the potentially higher demand levels.

A sensitivity analysis of demand is also valuable when evaluating existing traffic conditions, because of fluctuations in peak hour demands for different days of the same week. The day of the traffic count may have been the average peak hour, the lowest peak hour, or the highest peak hour that week. Exhibit 19 illustrates the cyclical fluctuation in demand for an urban freeway in Los Angeles for one week in May 2002. The peak flow rate can vary from 6100 vehicles per hour up to 7200 vehicles per hour depending on which weekday was counted for the purposes of the microsimulation.

Exhibit 19. Illustration of fluctuations in demand within same week

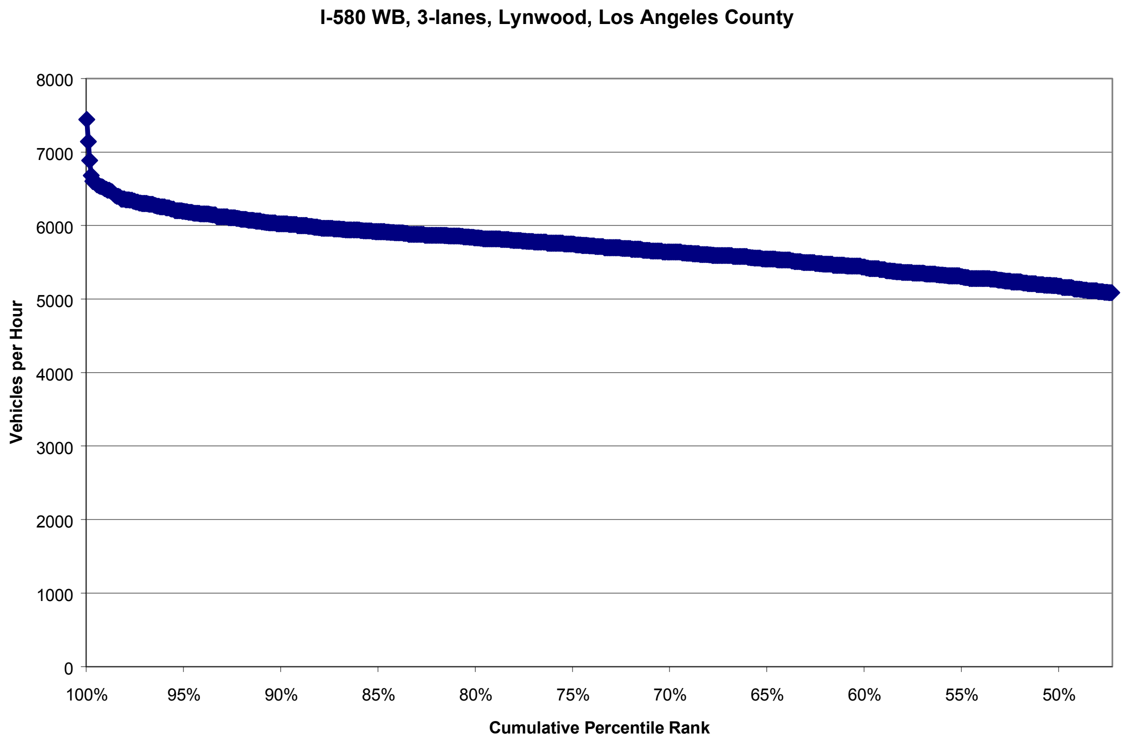


A cumulative percentile ranking of the counts for the week (see Exhibit 20) shows that if one selected a design volume of 6100 vehicles per hour that would rank as the 95 percentile highest volume of the week. A sensitivity test might then look at a demand level of 6,700 vehicles per hour, the 99 percentile highest volume for the week.

Street improvements assumed to be in place outside the simulation study area can also have major impacts on the simulation results by changing the amount of traffic that can enter or exit the facilities in the study area. Sensitivity testing would change the assumed future level of demand entering the study area, and the assumed capacity of facilities leaving the study area to determine the impacts of changes in the assumed street improvements.

The analyst may also run sensitivity tests to determine the effect of various assumptions about parameter values used in the simulation. If the vehicle mix was estimated, variations in the percentage trucks might be tested. The analyst might also test the effects of different percentages of familiar drivers in the network.

Exhibit 20. Cumulative Percentile Ranking of Demands Within Same Week



7.8 Optimization

Simulation models do not optimize. They do not determine the best traffic signal and ramp meter control settings to maximize or minimize some user specified objective function. Simulation models determine the performance of a given set of signal settings.

The analyst can run a simulation model multiple times with different signal settings and manually hunt for the signal settings that give the best performance, but this is usually very time consuming. Alternatively the analyst might code the data into a macroscopic model (one that is designed for optimization) and run that model to determine the optimal signal settings. The optimized signal settings are then input into the microsimulation model to verify the performance of the optimized setting.

7.9 Laboratory Session: Assessment of Microsimulation Results

Class participants will review the outputs of various simulation model runs for the I-650/I-580 Interchange Flyover Project and compute various summary statistics for the alternatives.

8 PRESENTING THE RESULTS TO THE PUBLIC

Where the use of simulation is warranted, it is useful for obtaining analysis results, gaining an understanding of operational characteristics, and providing a way to visually demonstrate complex traffic operational concepts to non-technical (as well as technical) audiences. A picture (or movie) is indeed worth thousands of words when conveying the results of technical analyses, and this is where one of the key values of simulation is realized.

Simulation in its essence is a series of time-sequence pictures showing the position of vehicles on a roadway network. When displayed sequentially they become a movie or video of the vehicular action on that network. The various display speeds - slow motion, real time, and accelerated motion - all have their function for analysis and presentation purposes. This chapter presents guidelines for designing and creating simulation presentations to managers and/or the public. These guidelines are necessarily broad-brush in nature due to the variety of traffic issues that might be evaluated by microsimulation.

A very professional presentation can be created using a software program such as Microsoft® PowerPoint®, which is specifically designed to produce such presentations. Such programs create slides that are projected on the computer screen. Each slide can contain text, tables, images, movie clips, or combinations of these. For presentation to audiences, the slides can be projected onto a meeting room screen using a projector attached to a laptop computer. The presentation can be controlled manually from the computer keyboard, manually from a remote mouse control, or automatically by the presentation software.

8.1 Setting Presentation Objectives

In setting the objectives of simulation presentations, one should review the key issues that prompted the use of microsimulation in the first place. What were the goals of using simulation to analyze this traffic situation? How do the simulation results fulfill those goals? With the answers to these questions at hand, one can then consider what portions of the simulation results best demonstrate the findings that address the original goals. For instance, if a key issue is identifying the effects of alternate roadway improvements on peak hour conditions, then an effective presentation would be a side-by-side movie simulation of the same system, with the same traffic loadings, for the same peak period without the improvements and with the alternative improvements. In addition, still shots at critical times with added annotations (arrows, circles, labels, etc.) could be used to focus the audience's attention on the most important differences between the alternatives.

Other factors to be considered in designing a presentation are the amount of operation time to simulate and the speed of the simulation to use. These factors are related, since the speed of simulation will control how much operation time is simulated in a movie of a given length. The controlling consideration may well be the amount of time allowed for your entire presentation. The shorter the allowed presentation time the more concise the simulation portion will need to be. The simulation needs to cover enough operation time to show the points you want to make.

If presentation time is limited, you may need to edit the simulation movie to show the changes over time with "snapshots" of simulation from different portions of the overall simulation time period. Real time speed may be appropriate for relatively close-up views of the system; however, most views that show the overall area of a roadway system are best viewed at a somewhat accelerated speed. The choice of speed will depend on how large the displayed network is and how detailed the operational features are that you are trying to illustrate. If the speed is too slow, the simulation can appear as a series of snapshots rather than as a movie, and if the speed is too fast, the audience can't follow the action very well.

If you expect your audience to be new to simulation, you should plan to have an initial part of the presentation that allows them to become familiar with what they are viewing. You should plan on pointing out basic elements of the simulation display and perhaps even identifying some of the limitations of the display for representing actual on-the-ground conditions. This will enhance the audience's orientation and answer in advance some of the common questions people have about what they are viewing during simulations. When viewing simulations for the first time, it is common for people to become quite enthralled with the minute details of vehicle movements and apparent interactions with other vehicles, thereby overlooking the main points that the simulation is intended to convey.

Another factor to consider is what view of the roadway system will best convey the simulation results. The most common view is an aerial (or plan) view, which places the viewer directly above the roadway system, or 90 degrees from horizontal. Technical audiences who are used to using plan views readily comprehend this view. Non-technical audiences can usually understand these views as well, but may need more orientation cues to get their bearings. Another view is oblique (slanted), in which the operations are shown from an angle of less than 90 degrees from horizontal. Another view is a windshield view, in which the operations are seen from the vantage point of a driver in one of the vehicles. Other combination views can be created by panning along the roadway network or by panning between alternate views (such as from an aerial view to an oblique view). The view(s) to be used in a given presentation will depend on what you are trying to convey to the audience and which view(s) will best accomplish that for you. Avoid including unnecessary or showy effects that contribute little or nothing to the clarity of your presentation. There is a tendency of some people to view simulations as nothing more than clever cartoons, and an excess of slick video editing effects can tend to support that opinion.

8.2 Writing the Presentation Script

Creating a simulation presentation is somewhat like creating a movie. There will be a series of scenes, each with a specific purpose, that need to be designed and created. Like any good presentation there should be a clear introduction, a series of items conveying the information to the audience, and a meaningful conclusion that summarizes the presentation and leaves the audience with the main idea to be conveyed.

Plan your presentation before capturing the images and movie clips so that your effort is focused and you avoid inefficiency. Outline your presentation as a series of slides, each containing text, images, video clips, or combinations of these. Use text judiciously so that the audience is not required to do a lot of reading during the presentation. Instead use text as labels, headings, and bullet points. Convey your message verbally, using the presentation text as an outline. Images and movie clips will mostly speak for themselves, but you may need to add some labels or pointers to focus the audience's attention. Make your verbal explanations of the images and movies succinct and to the point. Excess wordiness is distracting and annoying during such presentations. Don't repeat yourself unless it is necessary for clarity, which it seldom is.

For each slide identify what you desire to convey with it and how best to get that across to your audience. Decide whether a text slide is adequate or if visual images are needed. If visual images are called for, decide whether a snapshot image or a movie clip would best serve your purposes. The result will be a "story board" outline of your presentation, which will set the stage for the next step, capturing images and move clips.

8.3 Capturing the Images and Movie Clips

To capture images, run the simulation to the point where the image you want is visible. Pause the simulation and then use a screen capture program to capture the portion of the image you want. Save the image as a separate image file for later use. Giving each file a name that is descriptive of its contents and/or its sequence in the presentation will greatly facilitate presentation editing at a later time.

To capture movie clips, decide which portion of the simulation is to be included in the clip and run the simulation to a point a few seconds before the desired portion begins. Use a video capture program to define which part of the screen or which window you want to capture. Set the program to create a movie clip file (AVI type file). Start the simulation and when the beginning of the desired portion is reached, start the video capture process. When the end of the desired portion is reached, stop the video capture process. Save the video clip as a separate file for later use. Again, using descriptive names will greatly facilitate the presentation editing process. You may find that while the video capture process is running, the speed of the simulation slows down. This may produce a movie clip that runs slower than you want. If so, you will have to experiment with faster simulation speeds so that the resulting movie clip runs as desired.

You are now ready to insert the images and movie clips into your presentation to produce the finished product.

8.4 Assembling the Presentation

The steps involved in assembling the presentation can be illustrated with an example that assumes the Microsoft® PowerPoint® presentation software is used.

Step 1: Open a blank presentation, add a title slide, and enter the title text

Step 2: Insert new slides as required to build the presentation. Pre-designed slide layouts are available for the user to select, or the user can custom design slides. Let's start with an "Object" layout slide, which includes a title and an object box. The object box can contain an image of a movie clip.

Step 3: Click on the title box and enter the title for the slide.

Step 4: Double click on the object box to add an image or a video clip. Select the "Create from file" option button and use the "Browse" button to find the image file or the movie clip file that you want to place in this object box. Click "OK" to close the dialog box. The desired image should now be shown in the object box. Step 5:For images, no further settings are needed. For move clips the RightClick on the object box and select "Custom Animation". The following settings produce good presentation results.

Dialog Tab	Item	Setting
Timing	Start animation	Animate (on click or automatic)
Effects	Entry animation & sound	No Effect
Play Settings	Object action	Play

Step 6: Continue adding slides to complete the presentation. This might include bullet list slides, table slides with tabulations of key measures of effectiveness (average delay, levels of service, etc.), or combination slides.

Step 7: Test-run the presentation for time and effectiveness. Make edits as required to fine tune the presentation.

8.5 Hardware/Software Requirements

The hardware requirements for the simulation software should adequately accommodate the software needed to produce a professional looking presentation.

Microsoft® PowerPoint® presentation software is ideal for creating the formal electronic presentation. It allows the user to include snapshot images as well as movie clips on the presentation slides. It also has features to automatically display the presentation, including slide display timing, and transition effects between slides.

Presentations can also be created using Microsoft® Word software by imbedding images (as picture objects) and movie clips (as video clip objects) in a document. The results are functional in the absence of software designed specifically for presentations, but not as professional looking when presented.

There are various products available to capture snapshot images or movies from the simulation screen displays. A few are listed below.

Exhibit 21. Image Capture Software

Software Products That Capture Snapshot Images and Movies			
For Snapshots	✓	✓	✓
For Movies	✓		✓
Product	SnagIt	FullShot	CapturePro
Developer	TechSmith Corporation	InBit Incorporated	Creative Softworx, Inc.
Available at	www.techsmith.com	www.inbit.com	www.creativesoftworx.com
Single-user license	$39.95	$49.99	$34.95
Multi-user license	10 for $200.00	5 for $224.99	10 for $249.95
			needs Windows 2000 or Windows XP operating system

For making the presentation to an audience a laptop computer and a computer projector are required. Computer screen and projection resolution of at least 1024 x 768 is recommended.

8.6 Laboratory Session: Presentation of Microsimulation Results

Problem: Prepare a PowerPoint® presentation that shows the simulation results for the I580/I680 interchange before and after the flyover ramp improvement. Include data tabulations (placeholders only for this exercise), still images, and movie clips in the presentation. The presentation will be given at a public hearing and it must not exceed five minutes in total.

Resources: Two simulations showing the before and after improvement traffic flows with the same traffic demands.

Software tools: Microsoft® PowerPoint® and freeware CamStudio program.

Steps:

1. Identify the objectives of the presentation and the audience characteristics
2. Review the simulation to determine the best portions to use for presentation
3. Write the presentation script
4. Plan the views and viewing speeds
5. Create the images and movie clips using SnagIt
6. Assemble the presentation using PowerPoint® and test run for time and effectiveness; edit as required

Step 1. Identify the objectives of the presentation and the audience characteristics

The objective of this presentation is to demonstrate the difference in traffic flow conditions before and after the flyover improvements to the interchange. The audience will be a mix of elected and appointed officials, interested parties and local stakeholders, and members of the public at large.

Step 2. Review the simulation to determine the best portions to use for presentation

To show the operational differences, the area of the cloverleaf ramps is most useful. This is where the existing bottleneck problems exist and where the problematic southbound to eastbound traffic queuing problems develop. The traffic operation within the same fixed view window(s) in both simulations will best show the differences. Define view windows for use in capturing still frames and movies and save these in both simulations. In Paramics, the easiest way to do this is to store a view in one simulation and then manually add that view to the VIEWS file in the other simulation. To store a view use: View/Views/Store, and give the view a name such as 5 AVI View or 6 Still View.

Pick a time period of the simulation (about one-minute in length) that is after the warm-up period and that shows the congested conditions in the Before case. Note the start time, and make sure the times of the still frames and movies you create match in both simulations. Let's use 16:45:00 as our starting time.

Step 3. Write the presentation script

Title Slide		I580/I680 Interchange Simulation Results
Slide 2	still	Network overview - showing the entire network - Before
Slide 3	still	Network overview - showing the entire network - with Flyover
Slide 4	still	Focused view of the cloverleaf interchange alone - Before
Slide 5	still	Focused view of the interchange area - with Flyover
Slide 6	still	Same as Slide 4 but at 16:45 with traffic - Before
Slide 7	still	Same as Slide 5 but at 16:45 with traffic - with Flyover
Slide 8	movie	Focused view of the interchange area - Before - 1-minute movie starting at 16:45 starting at 16:45
Slide 9	movie	Focused view of the interchange area - with Flyover - 1-minute movie starting at 16:45 starting at 16:45
Slide 10	table	Tabulation of pertinent statistics for the Before and After cases (just create a placeholder slide for this exercise - no data required, but the students are to suggest the kinds of statistics that might be used in an actual presentation)

Step 4. Plan the views and viewing speeds

All views will be *plan* views and the viewing speed will as fast as the simulation will produce under the movie capture process.

Step 5. Create the images and movie clips

There are commercially available programs that specialize in capturing images and movies from the computer screen. Two excellent ones are FullShot for capturing still image files (.JPG) and SnagIt for capturing movie files (.AVI) as well as image files.

For this lab exercise, however, we will use the Windows screen capture feature to get our still images and we will use a freeware program called CamStudio (by RenderSoft Software, http://www.atomixbuttons.com/vsc).

To capture a still image of the computer screen, hold down the Alt-key and press the PrintScreen-key. This places an image of the screen on the Windows clipboard. Switch the PowerPoint program and paste the image onto a slide (Edit/Paste or Ctrl-V). Size the image to suit your slide layout. Note that you can use the Crop tool on the PowerPoint Picture Toolbar to remove potions of the screen image that you don't want to include.

To capture a movie clip with the CamStudio program, first run the simulation to the desired movie starting point, then launch the CamStudio program. Set the program configuration options to suit your needs. The following are suggested:

Region/Fixed Region ...	Width 450, Height 400
Options/Video Options ...	Quality=100 Check the Auto Adjust box Set Time Lapse slider to Max Framerate
Options/Cursor Options ...	Hide Cursor
Options/Program Options ...	Hide flashing rectangle during recording Play AVI file when recording stops
View/Compact View	

Click the record button.

Place the recording window crosshairs in the center of the screen area to be recorded, and click the left mouse button.

Immediately start the simulation. After about a minute, click the Stop button on the CamStudio toolbar.

Save the AVI file under a new name for later insertion into the PowerPoint presentation.

Step 6. Assemble the presentation using PowerPoint® and test run for time and effectiveness; edit as required

Use the procedures detailed in section ***8.4 Assembling the Presentation*** to build a presentation according to the outline in *Step 3*, above.

9 MANAGEMENT OF MICROSIMULATION PROJECTS

This chapter provides an overview of the steps involved in accomplishing a microsimulation project and how a microsimulation project is managed.

9.1 Scoping a Microsimulation Project

The major tasks of a microsimulation project are:

A. Identification of Project Purpose, Scope, and Approach
B. Data Collection
C. Coding
D. Error Checking
E. Calibration
F. Alternatives Testing
G. Documentation
H. Presentation of the Results

Each of these tasks is discussed in more detail below.

9.1.1 Identification of Project Purpose, Scope, and Approach

Before embarking on any major analytical effort it is wise to assess exactly what it is the analyst hopes to accomplish. One should identify the project objectives, the scope of the project, and the appropriate analytical approach to accomplish those objectives.

Project Objectives

Project objectives should answer the following questions.

Why is the analysis needed? What questions should the analysis answer?
Who is the intended recipient/decision maker for the results?

A generic set of project objectives might be: A traffic impact analysis is required of a proposed project and several alternatives for an environmental impact report. The analysis needs to identify the likely traffic impacts caused by a proposed project, the comparative impacts if one of various alternatives is built, and the recommended mitigations for mitigating the project impacts to a less than significant level. The analysis results are targeted to the local transportation planning agency commissioners.

While these generic objectives answer the three basic questions, the answers are not specific enough yet to help us decide on the analytical approach, and specifically whether or not microsimulation will be required. The project objectives set general bounds on the required analysis, but some additional specificity is required to develop the analytical approach.

Project Scope (Study Area Boundaries)

Once the project objectives have been identified, the Project Scope provides the additional detail required to select an analytical approach. The project scope sets the geographic bounds of the analysis, the temporal bounds of the analysis, the range of alternatives to be tested, and the numerical output to be produced. The project scope answers the following questions:

- What is the project being analyzed? (How large and complex is it.)
- What are the alternatives to be analyzed (How many, how large and complex are they)?
- What is the likely geographic scope of the impacts of the project (How far does the congestion to be mitigated by the project extend? How many square miles, which facilities, which people)?
- What is the likely temporal scope of the impacts of the project (how many hours of the day does congestion last, what days of the year, how many years into the future)?
- What degree of precision do the decision makers expect? Is 10% error tolerable? Are hourly averages satisfactory? Are the impacts of the alternatives likely to be very similar or very different from those of the proposed project? How disaggregate of an analysis is required? Is the analysis likely to produce a set of alternatives where the decision makers must choose between varying levels of congestion, as opposed to the situation where one or more alternatives eliminate congestion while others do not?)?

A scope that has a tight geographic focus, little tolerance for errors, and little difference in the performance of the alternatives will tend to favor microsimulation. A scope with a moderately greater geographic focus and a 20 years in the future time frame will tend to favor a blended travel demand model and microsimulation approach. A scope focused on large geographic areas and long time frames in the future will tend to rule out microsimulation and favor instead a combination of travel demand models and Highway Capacity Manual analysis techniques.

Project Approach

The project approach spells out the tools and steps involved in performing the technical analysis. A typical approach involving microsimulation will have the following steps:

1. Data Collection
2. Model Coding
3. Error Checking
4. Calibration
5. Alternatives Analysis
6. Documentation
7. Presentation of Results

The following sections describe each of these steps.

9.1.2 Data Collection

The data collection step involves collecting (either from the field or other sources) the necessary input data for the microsimulation model and the necessary output data (such as travel times and delays) for calibrating the coded model.

The data collection task involves the collection of two types of data for the simulation model. One set of data is gathered for the purposes of coding the model. The second set of data, best if collected at the same time as the first set, is used to calibrate the model's output.

The required model input data are: Geometry (s, lanes, curvature), Controls (signal timing, signs), and Existing Demands (turn volumes, OD table). The scope of work should identify the sources (such as aerial photos, agency files, field measurements) that will be used to obtain each of the input data. The scope should identify the study area boundaries, intersections, and highway segments for which the data will be gathered.

Intersection and highway segment geometry can be obtained from aerial photos and spot checked in the field.

Signage can best be verified in the field. Signal control data is best obtained from the operating agency files.

Existing demand should be measured in the field with machine counts to obtain 24 hour flow profiles. All critical intersections should have peak period turning movement counts. The count data should be obtained for all locations for as close to the same day and time as possible. If counts must be spread over many days (this is usually the case), then it is helpful to have a permanent count station in the field to monitor changes in overall demand levels between counting days.

Before converting the counts into estimates of demand, inconsistencies between locations and dates should be resolved as much as possible by the analyst. Once a consistent set of count data has been obtained then it will probably be necessary (depending on the software) to convert the counts into estimates of origin-destination demands. Allow a fair amount of staff time for the demand estimation step.

The required model calibration data consists of field measurements of travel time and delay. In the absence of a fairly comprehensive system of loop detectors, this data is most efficiently obtained through floating car studies. The cars are run approximately 10 times each peak period on each major route in the study area for which model calibration is critical. The floating car studies should be done at the same time as the volume counts to ensure consistency between the demand inputs to the model and the travel time measurements against which the model will be calibrated.

It is extremely valuable to observe existing operations in the field during the time period to be simulated. Simple visual inspection can identify behavior not apparent in counts and floating car runs. Video images may be useful, but may not focus on the upstream conditions causing the observed behavior, which is why a field visit during peak conditions is always important.

9.1.3 Coding the Model

During this step the data on network geometry, control, and demands is input into the microsimulation model. The basic steps of coding are:

1. Code network geometry (lanes, lengths, etc.)
2. Code control data (signs, signal timing)
3. Code demands

This is a complex software specific task that is best covered in training courses on the specific software. It is only briefly summarized in the most general form here.

Coding the input data into the model is a major task. The steps involved in coding are:

1. Import and size overlay image (aerial photo or As-Built CAD file) for network coding
2. Set up coding templates (standardized correspondence tables between facility type, area type, and other link characteristics to expedite coding of standard link types).
3. Rough in links and node locations over aerial photo
4. Code link attributes (lanes, free-flow speeds)
5. Code intersection attributes (control type, control parameters, turn lane designations, stop bars, turn pockets)
6. Code source/sink zones or nodes
7. Code vehicle types and origin-destination table(s).
8. Review/Revise default global parameters (vehicle characteristics, vehicle mix, etc.).

9.1.4 Error Checking

During this step the initial network coding for the microsimulation model is proofed. Error checking involves various tests of the coded network. The steps involved in error checking are:

1. Color code links by attribute (lanes, facility type, free-flow speed, etc.) and identify discrepancies.
2. Review intersection attributes.
3. Review demand inputs
4. Run model at very low volumes to identify errors.
5. Trace selected vehicles through the network.

9.1.5 Calibration

During this step the model is calibrated to better match existing conditions. Model calibration involves identifying a few global parameters for calibration and the operation of the model numerous times to identify the optimum values for those parameters. This can be a very time consuming process since one is never done. There is always something else that could be fixed. It is wise to set a budget and schedule for this task and stop when either the model is good enough or when time and money is up, whichever comes first.

9.1.6 Alternatives Analysis

During this step the microsimulation model is run to test the alternatives. Alternatives coding and testing involves first developing some baseline forecasts of future demand. The existing network is then edited for each alternative. Finally, the model is run numerous times for each alternative and statistics are gathered from the model reports. Allow sufficient time for post-processing the model outputs to obtain good comparative data for the alternatives.

9.1.7 Documentation

This step involves documenting the assumptions, analytical steps, and results of the analysis. It is important for ensuring that the decision makers understand the assumptions behind the results and for enabling other analysts to reproduce the results. Good documentation of a microsimulation analysis should include the following:

1. Project Objectives and Scope
2. Overview of Project Approach (tools used, rationale)
3. Data Collection (sources and methods)
4. Calibration Tests and Results
5. Forecast Assumptions (assumed growth inside and outside of study area, street improvements, etc.)
6. Description of Alternatives
7. Results

Documentation is a vital step to preserve the rationale for the various decisions that are made in the process of developing, calibrating, and operating a microsimulation model. The documentation should be sufficient so that given the same input files, another analyst can understand the calibration process, and repeat the alternatives analysis.

9.1.8 Presentation to Public

During this step the results of the microsimulation analysis are presented to supervisors and to the general public. The presentation task includes summarization of the model report in a form suitable for public presentation. It will usually involve a mixture of still slides and movie clips.

9.2 Estimating Staffing Requirements

A rough rule of thumb for staffing a microsimulation project is about 8 person hours of effort per freeway interchange (or signalized intersection not at a freeway interchange) to set up and error check the coded network. Figure that is about 50% of the budget. Figure another 25% of the budget for calibration, leaving 25% for alternatives analysis, documentation, and presentations. Add any data collection costs on top of that.

In terms of training, the person responsible for roughing in the network and the initial round of coding can be a beginner or immediate level in terms of knowledge of the software. They should have supervision though from a more experienced person with the software. Error checking and calibration are best done by a person with advanced knowledge of the microsimulation software. Model documentation and public presentations can be done by a person with intermediate level of knowledge of microsimulation software.

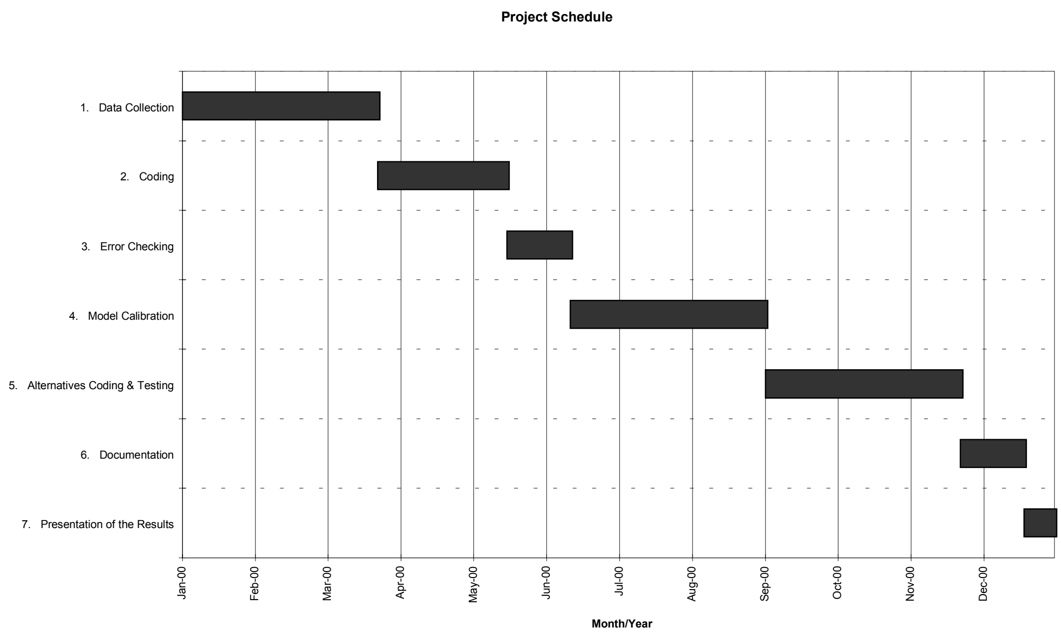
There is no fixed staffing requirement for the development of microsimulation models because the analyst can continue to invest time and effort continually refining the model. There is always something else that can be fixed or improved.

It is like buying a car. You can get basic wheels for under $5,000 or you can spend $50,000 to get a luxury car. It is up to the buyer to determine how much they are willing to invest in a car and whether the extra pleasure is worth the extra investment of a luxury car.

9.3 Estimating Time Schedule Requirements

In determining schedule, it should be recognized that only one person at any one time can truly create, calibrate, and operate the microsimulation model on the computer. Many non-computer tasks can be done in parallel, but most computer tasks are best done by a single person, since so many aspects of microsimulation models are so closely interrelated. A generic time schedule is shown in the chart below.

Exhibit 22. Prototypical Time Schedule for Microsimulation Project



9.4 Management of a Microsimulation Project

Much of the management of a microsimulation project is the same as managing any other highway design project: establish clear objectives; define a solid scope of work and schedule, monitor milestones, review deliverables. The subsections below discuss some of the management issues and solutions that are specific to microsimulation analysis projects.

9.4.1 Milestones and Deliverables

The key milestones for a microsimulation project are described in more detail in other parts of this chapter and in later chapters. The key milestones and recommended deliverables are summarized below:

Milestone	Deliverable
A. Definition of Project Scope	1. Project Scope and Schedule 2. Proposed Data Collection Plan
B. Data Collection	3. Data Collection Results Report 4. Proposed Model Coding Quality Assurance Plan
C. Model Coding and Quality Assurance	5. Coded and Error Checked Model files 6. Proposed Calibration Procedures with Calibration Targets.
D. Calibration	7. Calibration Test Results Report 8. Proposed Alternatives Analysis Procedures
E. Alternatives Analysis	9. Alternatives Analysis Report
F. Documentation	10. Final Report compiling all prior reports.

Depending on the number of alternatives being analyzed, the first four milestones (Scope, Data, Coding, and Calibration) will consume one-half to three-quarters of the project budget and time.

9.4.2 Methods for Speeding Up Delivery of Results

The surest way to speed up development and calibration of a microsimulation model is to trim the geographic scope and temporal breadth (hours of the day) of the model. Cutting the number of alternatives analyzed can also yield significant savings.

Alternatively, there are some microsimulation tasks that can be done in parallel with some advanced planning.

1. Dividing Up Network Coding Chores

The master network can be coded quickly in skeleton form (nodes and links, but no geometry or signal timing information). This master network can then be split into subareas and a different subarea assigned to each engineer/planner for data entry of the more detailed geometric and signal timing information. With careful control and recognizing the features and limitations specific to each microsimulation software package, the subarea networks can then be pasted back into the master network (the specifics of each software package will determine whether copying subsets of links from subnetworks onto the original skeleton master network, or stitching together the subnetworks into a new master network is most cost effective).

2. Performing Demand Estimation Independent of Network Coding

The estimation of existing and future origin-destination tables can usually be assigned to a different team than the one performing the network coding. Demand estimation involves a different type of expertise (working with regional travel demand models for example) than network coding and thus it is often preferable to assign demand estimation to a separate team.

The key is to define the zone system for the network early on. Then the demand estimation team can focus on creating the necessary OD tables while the network coding team works on the network.

Another option is to employ various techniques to speed up model run times. These include turning off various display features during the simulation run (options vary by software), plus running the calibration for only a portion of the network and a portion of the entire simulation period.

The project manager might consider first coding the study area into a macroscopic model like Synchro (for urban streets) and FREQ (for freeways) to obtain rough results for the highway improvement project and its alternatives. These rough results can provide preliminary information for later stages of the environmental analysis while the microsimulation model is still under development. These macroscopic models can also provide performance targets for evaluating the accuracy of the microsimulation model outputs. Synchro has the advantage of being able to export to CORSIM (but is of no use except to provide mean green times for actuated signals in Paramics models).

Finally, additional timesavings can be achieved with better computer equipment. Simulation models may run no faster than real time for large networks on slow computers. During this time the analyst is relatively unproductive, waiting for the computer to complete the run. Two solutions are to: one, buy a faster computer, or two, make two computers available to the analyst, one for running the model, the other for performing other duties while waiting for the simulation model to complete the run.

Overnight model runs can also cut down on analyst down time, but waiting 24 hours between runs will significantly delay the delivery of a completed model.

9.4.3 Cost Cutting Options

The surest way to cut model development costs is to cut its geographic and temporal scope, and to reduce the number of alternatives evaluated.

The most cost-effective method for building a microsimulation model is to have a single experienced person do model coding, calibration, and alternatives analysis from start to finish. In this case, a more highly paid experienced person can be more cost-effective than a lesser-paid inexperienced person.

Microsimulation models are notorious for absorbing all available staff resources. They are a painting that is never finished. The artist can always add a few more brushstrokes of refinement to any microsimulation model. The key to managing costs on a microsimulation project is therefore to establish definite and realistic goals for the microsimulation model.
Reasonable amounts of error have to be tolerated in any microsimulation model. The manager must establish the maximum tolerable level of error consistent with the goals of the project and the agency's sensitivity to investment decision mistakes.

For example, if the agency is designing a new freeway, the microsimulation model should be accurate enough to predict whether or not the facility will have enough capacity. The microsimulation model need not predict the speed of traffic on the facility to the nearest 5 mph.

Perfection should be avoided in microsimulation, since it is unachievable at any cost and results in unfinished projects.

9.4.4 Managing Consultants

The management of consultants is pretty much the same regardless of the type of project: establish scope, budget, milestones, and deliverables; and monitor progress. This subsection identifies some specifics related to microsimulation projects.

Two problems a manager often encounters when reviewing consultant products developed using microsimulation models are:

1. Insufficient in-house expertise to verify the technical validity of the model.
2. Consultant provides insufficient data to verify the validity of the model.

The manager may bring more expertise to the review of the model by resorting to a technical advisory panel composed of one or more experts in traffic microsimulation. The panel may be drawn from experts at other agencies, other consultants, or a nearby university. The experts should have had some prior experience developing simulation models with the software being used for the particular model.

Secondly, the manager (and the technical advisory panel) must have access to the input files for the microsimulation model. They must also have access to the simulation software and the experience with the specific software to test model.
There are many hundred parameters involved in the development and calibration of a simulation model. Consequently, it is impossible to assess the technical validity of a model based solely upon its printed output and visual animation of the results. There are infinite combinations of right and wrong parameter values that can still yield a plausible animation result. The manager must have access to the model input files so that he or she can assess the veracity of the model by reviewing the parameter values that go into the model as well as looking at its output.

9.5 Laboratory Session: Scoping, Budgeting, and Scheduling a Microsimulation Project

Class participants will work with the instructor to prepare analysis objectives, scope, and a scope of work for a prototypical microsimulation analysis for the I680/I-580 Interchange Flyover Project EIR/EIS.

10 APPENDICES

10.1 Search Algorithms for Calibration

Since simulation models are complex models, it is not usually possible to formulate the models as a closed form equation for which traditional calculus techniques can be applied to find a minimum value solution. It is necessary to use some sort of search algorithm that relies upon multiple operations of the simulation model, plotting of the output results as points, and searching between these points for the optimal solution. Search algorithms are required to find the optimal solution to the calibration problem.

The calibration problem is a "non-linear" (because simulation results are non-linear), "least-squares" (because the squared error is being minimized) optimization problem.

There are many software packages available for identifying the optimal combination of calibration parameters for minimizing the squared error between the field observations and the simulation model. The Argonne National Laboratory (http://www-fp.mcs.anl.gov/otc/Guide/SoftwareGuide/Categories/nonlinleastsq.html) lists several software packages for "non-linear least squares" parameter estimation. MATLAB, and MAPLE are two examples of commercial software.

The sections below illustrate a couple of simple approaches available for single parameter estimation and dual parameter estimation when working with a standalone simulation model. Estimation of three or more parameters however; would require the use of software.

10.1.1 Single Parameter Search Algorithm (Golden Section)

Several methods are available for finding the value of a single parameter that minimizes the squared error between the model and the observations. These methods include Newton's Method, Secant Method, Quadratic Approximation Methods, and the Golden Section Method. One of these, The Golden Section Method is illustrated in the example below.

Example of Golden Section Method

Objective: Find the global value of the mean headway between vehicles that minimizes the squared error between field counts of traffic volumes and model estimates.

Approach: We will use the Golden Section Method to identify the optimal mean headway.

Step 1. Identify the maximum and minimum acceptable values for the parameter to be optimized.

This step brackets the possible range of the parameter in which the optimal solution is presumed to lie. The user can select whatever he or she believes is appropriate for the range. For this

example we will set the minimum acceptable value for the mean headway at 0.5 seconds and the maximum at 2.0 seconds.
The larger the range the more robust our search, but it will take longer to find the optimum. The smaller the range of acceptable values, the greater the likelihood that the best solution lies outside the range.

Step 2. Compute Squared Error for Maximum and Minimum Values

The simulation model is run to determine the volumes predicted when the maximum acceptable mean headway is input, and again for the minimum acceptable headway. Either randomization should be turned off, or the model needs to be run several times with each mean headway and the results averaged for each mean headway. The squared error produced by the model for each mean headway is computed.

Step 3. Identify Two Interior Parameter Values for Testing

This step is what gives the "Golden Section" method its name. The two interior points (x1 and x2) are selected according to very specific ratios of the total range that preserve these ratios as the search range is narrowed in subsequent iterations. The formulae for selecting the two interior mean delays for testing are:

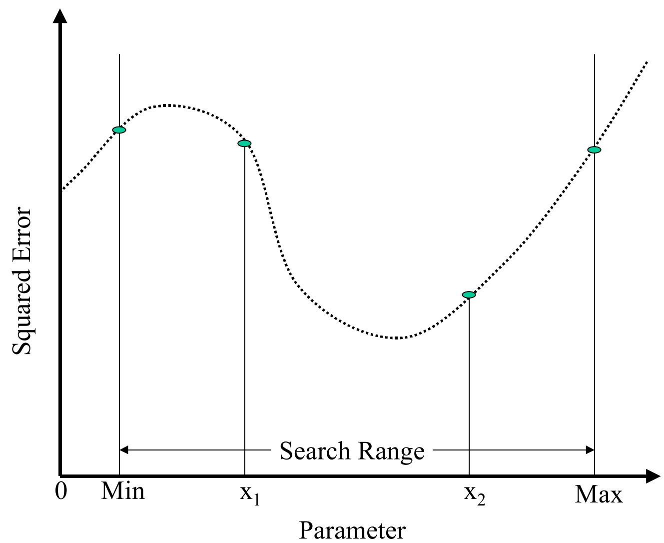
$$x_1 = \min + 0.382 \bullet (\max - \min)$$ *Equation 28*

$$x_2 = \min + 0.618 \bullet (\max - \min)$$ *Equation 29*

where:
x1 = the lower interior point value for the mean delay to be tested.
x2 = the upper interior point value for the mean delay to be tested.
min = the lower end of the search range for the mean delay
max = the upper end of the search range for the mean delay

The minimum and maximum ends of the search range are initially set by the user (based on the acceptable range set in Step 1), but then the search range is gradually narrowed as each iteration is completed.

Exhibit 23. Golden Section Method



Step 4. Compute Squared Error for Two Interior Parameter Values.

The simulation model is run for the new values of mean delay (x1 and x2) and the squared errors are computed.

Step 5. Identify the three parameter values which appear to bracket the optimum

This step narrows the search range. The parameter value (x1 or x2) that produces the lowest squared error is identified. (If either the minimum or the maximum parameter values produce the least squared error, the search range should be reconsidered.) The parameter values to the left (lower) and right (higher) of that point become the new minimum and maximum values for the search range. For instance in Figure 19, parameter value x2 produces the lowest squared error, so ....

Step 6. Go to Step 3, repeat until uncertainty in location of optimal parameter value is satisfactory.

The golden section search is repeated until the range of values in which the optimum parameter value lies is small enough to satisfy the user. After about 10 iterations, the uncertainty in the optimal value of the parameter is reduced by a factor of 100. So if an initial range of 0.5 to 2.5 is specified for the mean headway (a range of 2.0 seconds), this range will be reduced to 0.2 seconds after 10 iterations of the golden section method. The user will know the optimal value of the mean headway to within plus or minus one tenth of one second.

References:

Frederick S. Hillier, Gerald J. Lieberman, Introduction to Operations Research, 6th Edition, McGraw-Hill, New York, 1995.

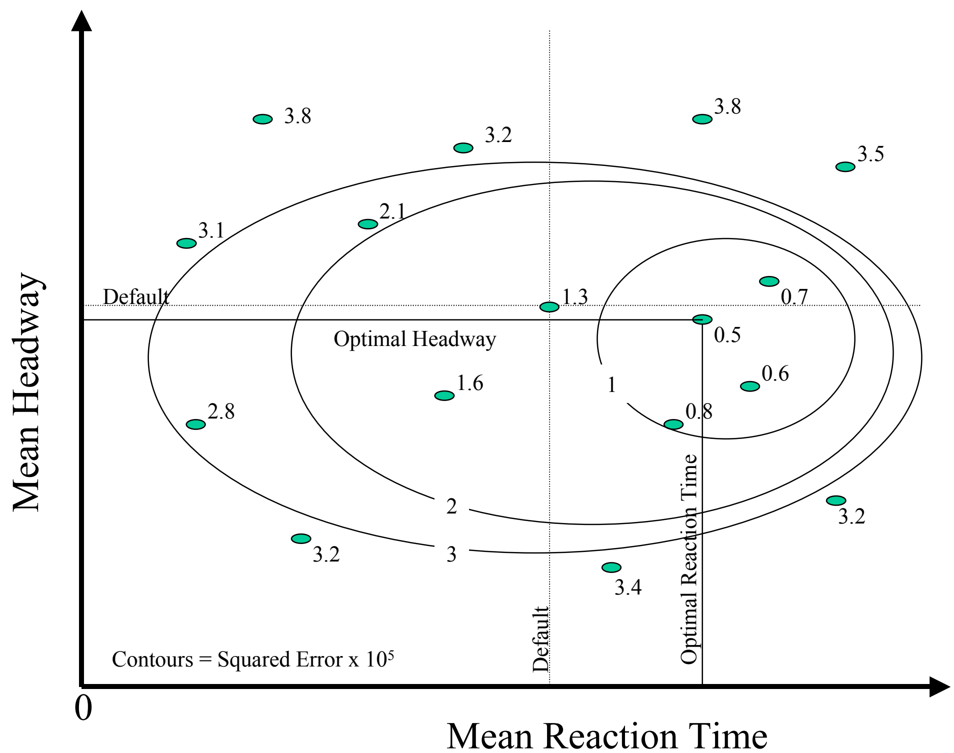
Hamdy A. Taha; Operations Research, An Introduction, 7th Edition, Prentice-Hall, New York, 2003. (look for 6th edition published in 1996, if 7th edition not available).

10.1.2 Simple Two Parameter Search Algorithm

In the case where two model parameters are to be optimized the user can use a contour plot approach to identifying the optimal values of the parameters. One first identifies the acceptable ranges of the two parameters and then exercises the model for pairs of values of the parameters. The squared error is computed for each pair of parameter values and plotted in a contour plot to identify the value pairs to result in the lowest squared error.

An example of this approach is shown below for optimizing the mean headway and mean reaction time parameters. The search starts with the default values, blankets the region with a series of tests of different values, and then focuses in more detail on the solution area.

Exhibit 24. Example Contour Plot of Squared Error



10.1.3 A Dual Objective, Two Parameter Search Algorithm

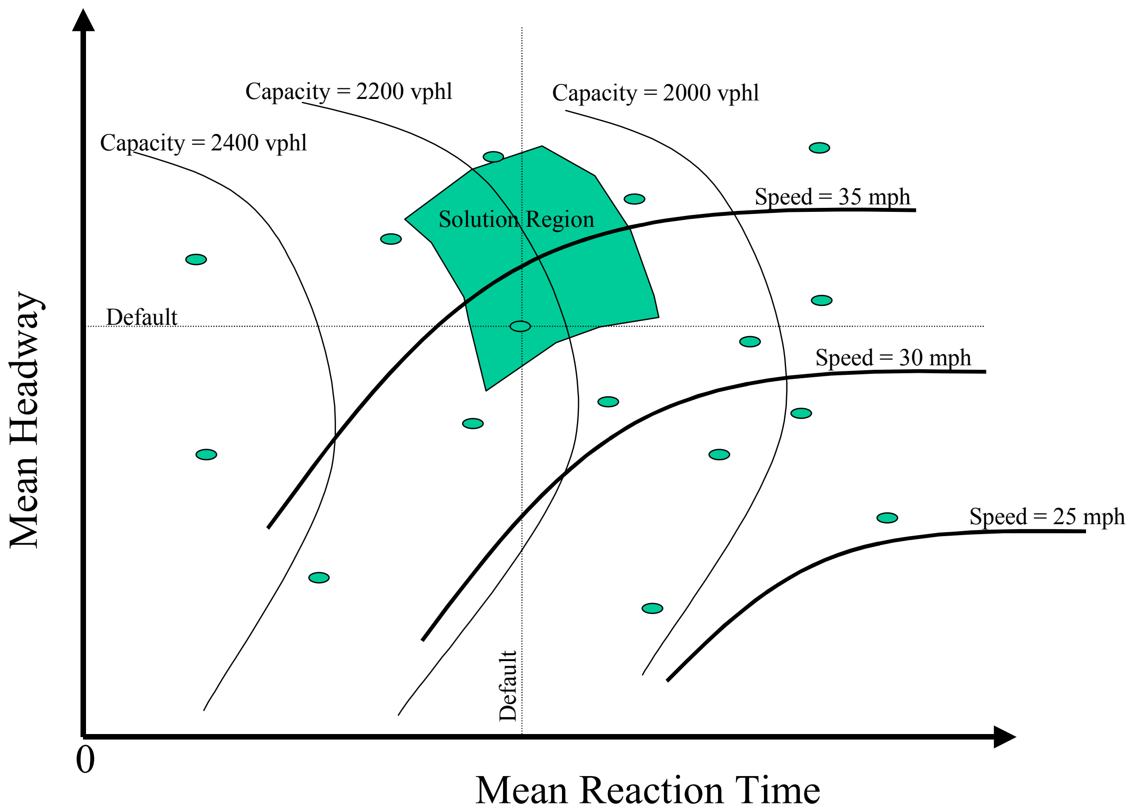
This section addresses the case where the analyst wishes to consider two model calibration objectives separately, rather than as the weighted sum of the square errors (single objective) proposed above. The analyst in this case can use a variation of the two dimensional search algorithm identified above with up to two parameters. Instead of plotting a single set of contour lines for the squared error, the analyst plots two sets of contour lines (one set for each objective) for each pair of parameters. The analyst then visually identifies the acceptable solution range where both objectives are satisfied to the extent feasible.

The example below illustrates this algorithm. The analyst has identified two objectives for the model calibration.

1. The model should predict a mean speed over the of the freeway of 35 mph (plus or minus 2 mph), and
2. The model should predict the maximum observed flow rate for the bottleneck section to be 2200 vehicles per hour per lane (plus or minus 100 vph).

The diagram below shows how changing the mean headway and the mean reaction time changes the values of these two objectives (speed and capacity). The solution region shows the pairs of values of mean headway and mean reaction time that meet both objectives.

Exhibit 25. Dual Objective, Dual Parameter Search



Adapted from: Yonnel Gardes, Adolf D. May, Joy Dahlgren, Alex Skabardonis, Bay Area Simulation and Ramp Metering Study, UCB-ITS-PRR-2002-6 California PATH Research Report, University of California, Berkeley, February 2002.

10.2 Further Reading

This appendix provides a few key references on microsimulation applications for further reading.

10.2.1 Traffic Microsimulation Fundamentals

Traffic Appraisal in Urban Areas. Highways Agency, Manual for Roads & Bridges Volume 12. Department for Transport, London, May 1996.
(http://www.archive.official-documents.co.uk/document/ha/dmrb/index.htm).

Provides guidance on the development, calibration, and validation of traffic models for urban areas.

Revised Monograph on Traffic Flow Theory, Nathan H. Gartner, Carroll J. Messer, Ajay Ruthi, editors; Federal Highway Administration, 1994
(available at: **http://www.tfhrc.gov/its/tft**)

This extensive report summarizes the state of the art for traffic flow theory: Human Factors, Car Following Behavior; Macroscopic Flow Models, and Traffic Simulation Models.

May, A.D., Traffic Flow Fundamentals, Prentice Hall, Englewood Cliffs, New Jersey. 1990.

This book provides an introduction to microscopic and macroscopic traffic flow characteristics.

Ahmed, Kazi Iftekhar, Modeling Drivers' Acceleration and Lane Changing Behavior, Ph.D. Dissertation, Massachusetts Institute Of Technology, Cambridge, MA, February 1999

This dissertation provides a review of the state of the art in driver acceleration, car following, and lane changing models and tests a suggested improved set of models.

10.2.2 Calibration and Validation of Microsimulation Models

FREEWAY SYSTEM OPERATIONAL ASSESSMENT, Technical Report I-33, Paramics Calibration & Validation Guidelines, DRAFT , Wisconsin Department Of Transportation, District 2, Milwaukee, WI., June 2002

This report provides the proposed guidelines for determining the satisfactory validation of a simulation model for the freeway system.

Gardes,Y., May,A.D.,Dahlgren, J.,Skabardonis, A., "Bay Area Simulation and Ramp Metering, California PATH Research Report".

This research report illustrates a two parameter, two objective search for calibration of the Paramics model to the I-680 freeway in Pleasanton, California. The two objectives were mean travel time over the length of the freeway, and the maximum observed flow rate at the key bottleneck section on the freeway. The two parameters to be optimized were mean headway and mean reaction time.

Jayakrishnan, R., Jun-Seok Oh, Abd-El Kader Sahraoui, "Calibration and Path Dynamics Issues in Microscopic Simulation for Advanced Traffic Management and Information Systems", UCI-ITS-WP-00-22, Institute of Transportation Studies, University of California, Irvine, CA, December 2000

The authors identify two phases of model validation: conceptual and operational. The conceptual phase compares the model concept to relevant theory. The operational phase compares model output to real wolrd measurements.

They suggest a generalized calibration objective function of minimizing the squared error of a combination of error measures (such as volume and occupancy). They suggest that a normalizing parameter be included in the objective function to counter-weight each of the error measures so that one (such as volume) does not dominate the other simply because of the units in which it is expressed. Thus all volume errors are divided by the maximum observed volume, all occupancy errors are divided by the maximum observed occupancy.

Project Suite Calibration and Validation Procedures, Quadstone, Ltd., Edinburgh, UK, February 2000
(http://www.paramics-online.com/tech_support/customer/cust_doc.htm password required).

10.2.3 Analysis of Results

Joshi, S.S., Ajay K. Rathi, "Statistical Analysis and Validation of Multi-population Traffic Simulation Experiments", Transportation Research Record 1510, Transportation Research Board, Washington, D.C., 1995

The authors present a "common random number" (CRN) strategy for reducing the number of model runs necessary to distinguish between alternatives. The CRN strategy is a means of reducing the variance of the estimates produced by the simulation model. With reduced variance, fewer runs are needed to accept or reject a hypothesis of no difference between the alternatives. The CRN strategy, however; by inducing correlation between the alternatives, invalidates standard statistical analysis assumptions of independence between tests and requires special techniques to determine the rejection regions.

Lane, David M., Hyperstat OnLine, An Introductory Statistics Book and Online Tutorial for Help in Statistics, Rice University. (www.davidmlane.com/hyperstat)

This text provides information on the appropriate statistical approach for the testing of multiple pairs of alternatives.

Gerstman, B., and Marg Innovera, "StatPrimer Version 5.1", California State University, San Jose, (http://www.sjsu.edu/faculty/gerstman/StatPrimer/)

This text provides a good introduction to the analysis of variance (ANOVA) for accepting or rejecting the hypothesis that all alternatives have the same traffic performance.

10.2.4 Software Evaluation

Courage, K. and C. Wallace, NGSIM Feasibility Study Final Report, Federal Highway Administration, Washington, D.C., 2001

The report compares the technical modeling capabilities of microsimulation software. They identified the following functions: Coordinated traffic signals, Adaptive traffic signals, Priority to public transport vehicles, Ramp metering, Motorway flow control, Incident management, Zone access control, Variable message signs, Regional traffic information, Static route guidance, Dynamic route guidance, Parking guidance, Public transport information, Automatic debiting and toll plazas, Congestion pricing, Adaptive cruise control, Automated highway system, Autonomous vehicles, Support for pedestrians and cyclists, Probe vehicles, Vehicle detectors.

They also looked at software capabilities to model following phenomena: Weather conditions, Incidents, Search for a parking space, Public transports, Parked vehicles, Traffic calming measures, Elaborate engine model, Queue spill back, Commercial vehicles, Weaving, Bicycles / motorbikes, Roundabouts, Pedestrians.

Nam,D.H.,Shaw,J.W.,"Microsimulation, Freeway System Operational Assessment, and Project Selection in Southeastern Wisconsin: Expanding the Vision", Presentation, Annual Meeting, Transportation Research Board, Washington, D.C., 2002.

The authors identified 23 criteria for assessing the suitability of microsimulation software for their freeway improvement project ranking process. The criteria include: network size limit, fidelity of network representation, traffic flow representation, detail of output, network merge capability, 3-D modeling, adjustable traffic composition, animation, input data requirements, difficulty of network coding/editing, ease of input/output review, GIS interface, economic analysis interface, incident management analysis, actuated signal control devices, user defined traffic control (API), public transportation, calibration and validation results, program integrity, technical support, documentation, record of large scale freeway applications, software cost per copy.

Skabardonis, A., Assessment of Traffic Simulation Models, Final Report, Office of Urban Mobility, Washington State Department of Transportation, Seattle, WA, May 1999.

The report evaluates macroscopic and microscopic simulation models against the following criteria:

1. Model capabilities and features.
2. Modeling of traffic flow (accuracy of modeled variability of traffic demand in time and space, growth and decay of queues, capacity reductions at incidents and bottlenecks.).
3. Input and calibration data requirements.
4. Output options (performance measures reported, linkage to HCM for level of service analysis. Quality of graphic displays and animation.).
5. Computational aspects (computer platform required, software run times).
6. Cost (staff time and training, software acquisition, technical assistance).