



VDOT VISSIM User Guide

Version 2.0

VDOT Traffic Engineering Division

January 2020

Copyright 2020 by the
Virginia Department of Transportation
All rights reserved



This page is intentionally left blank.



VDOT Vissim User Guide

Version 2.0

Copyright 2020 by the Virginia Department of Transportation

All rights reserved

VDOT Traffic Engineering Division

January 2020

Notice and Disclaimer

This document is disseminated by the Virginia Department of Transportation (VDOT) for informational purposes only. No changes or revisions may be made to the information presented in this document without the express consent of VDOT.

The recommendations in this guidebook are based on VDOT experience, best practices, and engineering preferences. VDOT assumes no liability for the use of the information contained in this document nor endorses any products or manufacturers referenced in this document. Trademarks or manufacturers' names are used solely for reference purposes.

Copyright

The contents of all material available in this guidebook and its online version are copyrighted by VDOT unless otherwise indicated. Copyright is not claimed as to any part of an original work prepared by a US or state government officer or employee as part of that person's official duties or any information extracted from materials disseminated by a private manufacturer. All rights are reserved by VDOT, and content may not be reproduced, downloaded, disseminated, published, or transferred in any form or by any means, except with the prior written permission of VDOT, or as indicated below. Copies for personal or academic use may be downloaded or printed, consistent with the mission and purpose of VDOT (as codified in its governing documents). However, no part of such content may be otherwise or subsequently reproduced, downloaded, disseminated, published, or transferred, in any form or by any means, except with the prior written permission of and with express attribution to VDOT. Copyright infringement is a violation of federal law subject to criminal and civil penalties.

Table of Contents

Table of Contents	iii
Table of Tables	vi
Table of Figures.....	vi
1 Introduction.....	1
1.1 Purpose.....	1
1.2 VDOT Vissim User Guide Maintenance and Updates.....	1
1.3 Software Updates.....	1
1.4 User Guide Summary.....	1
2 Model Development	3
2.1 General network coding	3
2.1.1 Links and Connectors	3
2.1.2 2D/3D Models, Vehicle Types, and Vehicle Classes.....	4
2.1.3 Vehicle Compositions, Inputs, and Routings	6
2.1.4 Transit Routes and Transit Stops	7
2.1.5 Desired Speed Decisions and Reduced Speed Areas	8
2.1.6 Conflict Areas Versus. Priority Rules	9
2.1.7 Gradient.....	11
2.1.8 Simulation Setup	12
2.2 Freeway coding	13
2.2.1 Merge/Diverge/Weave Segments.....	13
2.2.2 Link Segmentation	15
2.2.3 Parallel/Concurrent Facilities (e.g., HOT/HOV)	17
2.2.4 Ramp Meters.....	19
2.3 Arterial coding.....	20
2.3.1 Intersection Approaches	20
2.3.2 Stop-Controlled Intersections	22
2.3.3 Signalized Intersections	23
2.3.4 Right-Turn-on-Red Coding.....	24
2.3.5 Pedestrian Crosswalks	24
2.3.6 Roundabouts.....	25
3 Model Review and Debugging.....	28
3.1 Model Debugging versus Calibration	28
3.2 Model Review Checklist	28
3.3 Identifying model Coding errors	29
3.3.1 Review of Model Input Data	29
3.3.2 Animation Review.....	29
3.3.3 Error Log.....	29

3.3.4	Common Coding Errors.....	30
4	Results and Presentation.....	32
4.1	Evaluation.....	32
4.1.1	Evaluation Setup	32
4.1.2	Node Evaluation	34
4.1.3	Data Collection Points	34
4.1.4	Queue Counter	36
4.1.5	Travel Times	36
4.1.6	Link Evaluation.....	39
4.1.7	Vehicle Network Performance	39
4.2	Output Results	39
4.2.1	Direct Output (Raw Data).....	39
4.3	Visualization of Simulation Results	41
4.3.1	Output Plots	41
4.3.2	Network Heat Maps	41
4.3.3	Ancillary Performance Visualization.....	44
4.4	Video Presentation	44
4.4.1	2D Vehicle Animations.....	44
4.4.2	Real-Time Network Performance Results	46
4.4.3	3D Animation.....	46
5	Calibration Guidance	49
5.1	Background Information.....	49
5.2	Data Collection for Calibration.....	49
5.2.1	Importance of Reliable Data.....	50
5.3	Calibration Parameters.....	50
5.3.1	Lane Change Parameters	51
5.3.2	Speed Distributions	51
5.3.3	Driving Behavior.....	51
5.3.4	Conflict Areas and Priority Rules	54
5.4	Calibration Plan.....	54
5.5	Pre-Calibration Considerations	55
5.5.1	Verify Model is Ready for Calibration	55
5.5.2	Driving Behavior Containers	55
5.5.3	Lane Change Distances.....	56
5.5.4	Terminal Conditions.....	56
5.5.5	Lane-By-Lane Traffic Conditions.....	56
5.6	Calibration Considerations.....	56
5.6.1	Calibration Checkpoints.....	56

5.6.2	Calibration Steps	57
5.6.3	Calibration Guidelines.....	58
5.7	Calibration Report	60
6	Model Scenarios.....	61
6.1	Existing Conditions Model Development.....	61
6.2	Future No-Build Model Development	61
6.3	Future Build Model Development.....	62
APPENDIX A	65
APPENDIX B	69

TABLE OF TABLES

Table 5-1. Various Calibration Measures and Potential Data Sources	50
--	----

TABLE OF FIGURES

Figure 2-1. Standard Taper Coding.....	4
Figure 2-2. Flow of Information Between 2D/3D Models, Vehicle Types, and Vehicle Classes.....	5
Figure 2-3. Partial Route Coding Example.....	8
Figure 2-4. Connector Entering Link Part Way Through Reduced Speed Area	10
Figure 2-5. Conflict Area Options for Intersecting Facilities	11
Figure 2-6. Simulation Parameters Window	13
Figure 2-7. Freeway Merge Segment Coding Example.....	15
Figure 2-8. Freeway Diverge Segment Coding Example.....	16
Figure 2-9. Freeway Weave Segment Coding Example.....	17
Figure 2-10. Freeway Link Segmentation Example for Merge Influence Area	18
Figure 2-11. Freeway Parallel Facility Link Coding Example	19
Figure 2-12. Freeway Concurrent HOV Facility Link Coding Example	20
Figure 2-13. Example of Intersection Approach Coding with No Barrier Separation.....	21
Figure 2-14. Example of Intersection Approach Coding with Barrier Separation for Left-Turn Lane	22
Figure 2-15. Example of Intersection Approach Coding with Channelized Right Turn.....	23
Figure 2-16. Right-Turn-on-Red Coding Example	25
Figure 2-17. Example of Intersection Pedestrian Crosswalk Coding.....	26
Figure 2-18. Two-Lane “Dog Bone” Interchange.....	27
Figure 4-1. Evaluation/Configuration Window.....	32
Figure 4-2. Evaluation Configuration Window.....	33
Figure 4-3. Node Defined for Intersection Movements	35
Figure 4-4. Node Evaluation Configuration and Settings	35
Figure 4-5. Data Collection Points and Data Collection Measurements Definition.....	36
Figure 4-6. Queue Counter Definition.....	37
Figure 4-7. Travel Time Collection Segment Definition	38
Figure 4-8. Travel Time Collection Parallel Routes Coding Example	38
Figure 4-9. Link Evaluation Activation	39
Figure 4-10. Attribute Result Tables.....	40
Figure 4-11. Direct Outputs.....	40
Figure 4-12. Output Plots.....	41
Figure 4-13. Generating Output Plots.....	42
Figure 4-14. Generating Link-Based Network Heat Maps	42
Figure 4-15. Speed Heat Map	43
Figure 4-16. Volume Heat Map	43
Figure 4-17. Density Heat Map	44
Figure 4-18. Node Performance LOS and Queue Length.....	45
Figure 4-19. Node Performance Turning Movement Volumes.....	45
Figure 4-20. Screen Captures of 2D Model Simulation.....	46
Figure 4-21. Screenshot of Real-Time Link Speed and Queue Length.....	47
Figure 4-22. Storyboard Screenshots of Vehicle Speed and Queueing During Simulation Period.....	47
Figure 4-23. Storyboard Screenshots of Vehicle Speed During Simulation Period.....	48
Figure 4-24. Screen Captures of 3D Model Simulation.....	48
Figure 5-1 Driving Behavior Container.....	52
Figure 5-2. Flow Chart of Calibration Steps.....	57

Glossary of Terms

***.ATT:** Vissim, Version 6.0 and later, output file containing simulation results.

Build Model: Vissim model developed to analyze the operation effects of a proposed improvement on a transportation facility. This model should be derived from a calibrated Existing Conditions Model and should account for any parallel projects within the study area that are included in that regions Long-Range Transportation Plan.

Calibration: Process where the modeler modifies the parameters that cause the model to best reproduce field-measured and observed local traffic conditions.

Debugging: Processes by which model coding errors are identified and rectified. This step should be completed for every Vissim model scenario developed and should be completed before calibration for the existing conditions model.

Delay: Additional travel time experienced by a driver, passenger, bicyclist, or pedestrian beyond that required to travel at the desired speed (expressed in seconds).

Density: The number of vehicles occupying a given length of lane at a particular instant (expressed in passenger cars or vehicles per mile per lane).

Deterministic Traffic Tools: Traffic analysis tools that assume that there is no variability in the driver-vehicle characteristics (e.g., HCS 2010).

Existing Conditions Model: Vissim model reflecting existing conditions for a project study area. This model is used as a basis for future year scenario analyses and should be calibrated to *TOSAM* calibration thresholds using field data.

***.INP:** Vissim, Version 5.40 model file extension.

***.INPX:** Vissim, Version 6.0 model file extension.

Measure of Effectiveness (MOE): Quantitative or qualitative characterization of some aspect of the service provided to a specific road user group.

Microsimulation: Modeling of individual vehicle movements on a second or sub-second basis used to assess the traffic performance of a transportation network.

No-Build Model: Vissim model derived from the Existing Conditions Model which is often used as a benchmark for comparison against a Build Model. The No-Build Model should account for any parallel projects within the study area that are included in that regions Long-Range Transportation Plan.

Project Manager: Individual responsible for accomplishing the stated project objectives through planning, execution, and closing of a project.

Stochastic: having a random probability distribution or pattern that may be analyzed statistically but cannot be predicted precisely. Stochastic functions are utilized in Vissim to account for day to day variability of traffic conditions.

Synchro: Synchro is a deterministic tool that is primarily used for analyzing traffic flow, traffic signal progression, and optimization of traffic signal timing. Additionally, Synchro may be used to analyze arterials, signalized intersections, and unsignalized intersections.

TOSAM: Acronym for the *VDOT Traffic Operations and Safety Analysis Manual*

***.v3d:** File extension for 3D models used in Vissim microsimulation models

Validation: Process where the modeler checks the overall model-predicted traffic performance for a network against field measurements of traffic performance (using data not used in the calibration process).

Vissim Intersection Data Processing Tool (Version 2.0): Excel-based tool for processing Vissim intersection node results.

Wiedemann Car-Following Model: Car-following models originally formulated in 1974 and 1999 that were constructed based on conceptual development and limited data and must be calibrated to traffic conditions for individual facilities/traffic streams.

This page is intentionally left blank.

1 Introduction

1.1 PURPOSE

This user guide is intended to assist Vissim users with microsimulation model development, calibration, and post-processing using Vissim Version 11, but will also apply to Versions 6-10 unless specifically identified in this document. VDOT identified the need to develop the *Vissim User Guide* to inform modelers of VDOT preferred practices and best practices in the industry, understanding the increased use of this microsimulation tool in transportation projects and the continuous evolution of Vissim software. This user guide is intended to be used by VDOT staff, locality staff, and the consultant community for VDOT projects.

The *Vissim User Guide* is not intended to serve as a specific microsimulation tool manual; however, it will provide guidance on how to properly develop Vissim models to produce outputs compatible with the VDOT preferred practices. The *PTV Vissim User Manual* is recommended as a reference to supplement this user guide.

1.2 VDOT VISSIM USER GUIDE MAINTENANCE AND UPDATES

VDOT is responsible for maintaining and updating the user guide on a regular basis. Future updates are expected as Vissim and PTV Suite software change and the VDOT *Traffic Operations and Safety Analysis Manual (TOSAM)* evolves.

1.3 SOFTWARE UPDATES

Users should be aware of any software updates from PTV, who releases service packs to fix small bugs of the program on a continuous basis. The transition to a new build (e.g., Vissim 11.00-01 to Vissim 11.00-02) or version (e.g., Vissim 11 being updated to Vissim 12) of Vissim should be based upon approval by VDOT after VDOT reviews the new build or version. However, it is not recommended that users change the version or build from which the model is originally calibrated for any given project. It is critical to include the Vissim version and build in documentation for each submittal.

1.4 USER GUIDE SUMMARY

The VDOT *Vissim User Guide* is arranged into the following five sections.

Section 2 – Model Development: The Model Development section highlights some important components of the Vissim model development process, including best practices for coding of key network objects in common network scenarios.

Section 3 – Model Review and Debugging: The Model Review and Debugging section will outline best practices for reviewing model development and debugging model coding errors. This section also includes several common coding errors and how they can be identified and rectified.

Section 4 – Results and Presentation: The Results Reporting section highlights different evaluation methods and the coding of evaluation parameters used to measure model performance. This section also provides guidance on how results can be visualized within Vissim.

Section 5 – Calibration Guidance: The Calibration Guidance section provides a discussion of the calibration process, including how and when different adjustments should be made to best replicate field conditions. This section also includes guidance on validation of Vissim model outputs based on the calibration criteria laid out in the *TOSAM*.

Section 6 – Model Scenarios: The Model Scenarios section discusses the typical model scenarios that are developed as part of a Vissim analysis and how those scenarios relate to each other.

This page is intentionally left blank.

2 Model Development

This section highlights some important components of the Vissim network development process. This manual is not meant to be a comprehensive manual on the coding of all network objects; rather, it is intended to provide guidance on best practices for coding of key network objects and common modeling scenarios.

Section 2.1 provides guidance on best practices and standard procedure for coding of key network objects and parameters required for most Vissim networks. **Section 2.1.7** focuses on objects and scenarios that apply principally to the coding of freeway networks, while **Section 2.3** focuses on objects and scenarios commonly used in the coding of arterial networks.

2.1 GENERAL NETWORK CODING

This section provides guidance on best practices and standard procedure for coding of network objects required for most Vissim networks.

2.1.1 *Links and Connectors*

Links and Connectors form the foundation of any Vissim network. Links function as stand-alone objects while Connectors must be attached to a link at either end to be added to the network. Best practice is to minimize the length of connectors whenever feasible during network coding.

2.1.1.1 *Link/Connector Curvature*

Spline points should be added to links and connectors to replicate curvature in the field/design plans; however, vehicle behavior is not affected by sharp versus gradual curves. In other words, a vehicle will maintain the same speed through a 1,000-foot radius curve as they would on a 100-foot radius curve unless Reduced Speed Areas are coded to account for reductions in speed due to curvature (Reduced Speed Areas are discussed in **Section 2.1.5.2**).

2.1.1.2 *Link Behavior/Link Display Types*

Link Behavior determines how vehicles interact within a given link. Link behavior and the attributes contained therein are discussed in **Section 5**; however, at the most basic level, freeway facilities should be coded using the default freeway Link Behavior Type (based on the Wiedemann 1999 car-following model), while arterial facilities should be coded using the default urban Link Behavior Type (based on the Wiedemann 1974 car-following behavior). Modifications are typically made to create custom Driving Behaviors during the model calibration process.

Link Display Types should be used to track where Link Behavior Types have been changed in the network. For each new Link Behavior Type that is created, a corresponding Link Display Type should also be developed and used with that Link Behavior Type on any link where it is implemented.

2.1.1.3 *Coding Tapers*

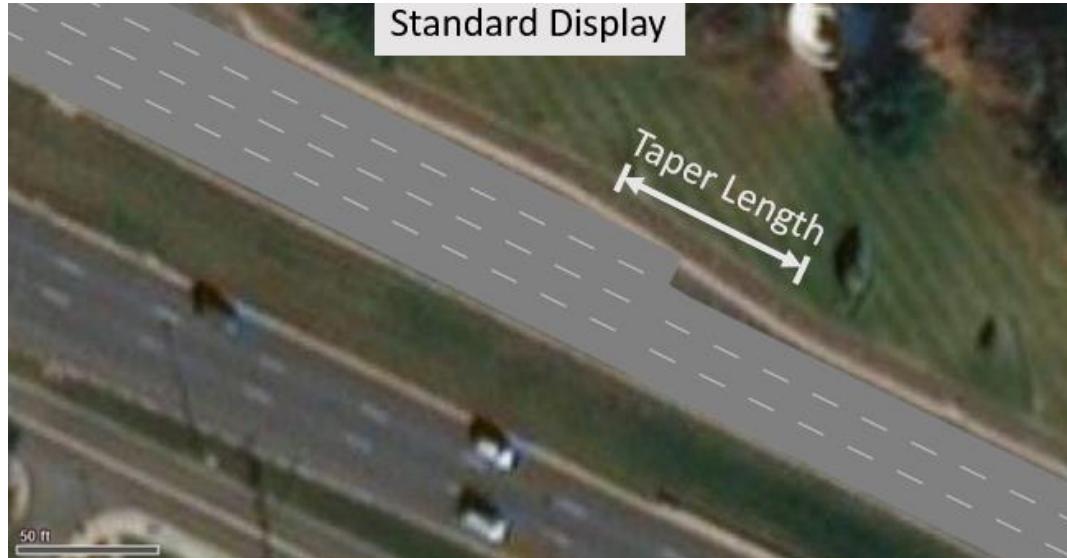
Lane drops and add lanes in the field typically have a taper to provide a smooth transition for vehicles as the total number of lanes on a facility decreases or increases. Vissim link and connector coding does not lend itself to replicating tapers, instead tapers are modeled in Vissim by assuming approximately 2/3 of the taper for auxiliary lanes and 1/2 of the taper for turning lanes. The connector should attach to the lanes which continue through the taper only and no connector should be attached to the lane that either begins or terminates at the taper. An example of this practice is shown in the diverge condition illustrated in **Figure 2-1**.

2.1.2 2D/3D Models, Vehicle Types, and Vehicle Classes

2D/3D Models, Vehicle Types, and Vehicle Class attributes allow the user to control the vehicle mix that is simulated in a Vissim network. The relationship between these attributes is illustrated in **Figure 2-2**.

Default 2D/3D Models, Vehicle Types, and Vehicle Classes are provided in Vissim; however, these defaults should be adjusted according to available field data to best replicate the vehicle mix present in the roadway network being analyzed.

Figure 2-1. Standard Taper Coding



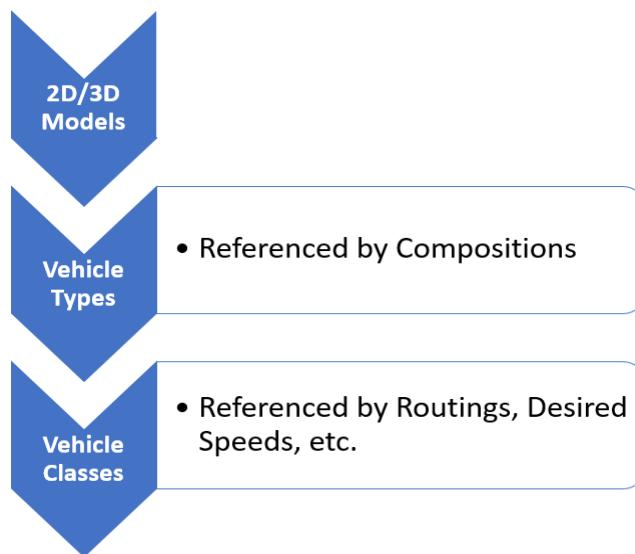
2.1.2.1 2D/3D Models

2D/3D Models act as the basis for determining the vehicle mix that is simulated in a Vissim network. Each 3D model is associated with a *.v3d model of a vehicle which, most relevant to network performance, will determine the length of the vehicles which are simulated in the network. These 2D/3D models are then placed

into 2D/3D model distributions, where the share of each 2D/3D model assigned to a given distribution is set relative to the total shares for that distribution (e.g., if the sum of the “Share” attributes of all 2D/3D models for a given distribution is 1, and the 2D/3D model for Car A has a “Share” value of 0.1, then 10% of the vehicles simulated under that distribution will be Car A).

2D/3D model distributions should be based on count data collected in the field. For example, the “Share” of tractor trailer versus single-unit trucks simulated in a network will have a significant effect on operational performance and so the correct 2D/3D distribution of HGVs should be applied to the network according to count data.

Figure 2-2. Flow of Information Between 2D/3D Models, Vehicle Types, and Vehicle Classes



A new Vissim file will reference a European vehicle fleet by default, which does not reflect the vehicle fleet typically found in Virginia. A default vehicle fleet for North America is provided by PTV in the Training Directory. This directory is accessible within Vissim 11 by navigating to “Help/ Examples/ Open Training Directory/ Vehicle Fleet & Settings Default/ USA” or this file can be found under the following computer path: *C:\Users\Public\Documents\PTV Vision\PTV Vissim 11\Examples Training\Vehicle Fleet & Settings Defaults\USA* with normal installations. The vehicle fleet contained in this Vissim file should be used as a starting point for model development per TOSAM guidelines.

2.1.2.2 Vehicle Types

Vehicle Types are the next step in setting the vehicle mix that will be simulated in a given Vissim network. For each Vehicle Type, a 2D/3D model distribution will be assigned, along with four other important attributes which will affect the behavior of simulated vehicle types: “Desired Acceleration Function”, “Desired Deceleration Function”, “Maximum Acceleration Function”, and “Maximum Deceleration Function”. These four attributes, in combination with Driving Behavior parameters (discussed in more detail in **Section 5**), will affect the way different Vehicle Types behave during simulation. For example, a passenger car will typically have a higher desired and maximum acceleration than a Heavy Goods Vehicle (HGV). It is important that these attributes reflect observed field conditions whenever that information is available.

Finally, for Vehicle Types to be simulated during a Vissim run, they need to be assigned to a specific Vehicle Composition. Vehicle Compositions are discussed in greater depth in **Section 2.1.3.1**.

2.1.2.3 Vehicle Classes

Vehicle Classes do not directly determine the vehicle mix simulated in a Vissim network, however they do play an important role in the Vissim network coding process by allowing users to group multiple Vehicle Types together to assign them to network objects. For example, if a given set of Vehicle Routings (discussed in **Section 2.1.3.3**) or Desired Speed Decisions (discussed in **Section 2.1.5.1**) should apply to all Vehicle Types in the network, a Vehicle Class can be created which groups together multiple Vehicle Types. This new Vehicle Class can then be assigned to the Vehicle Routings or Desired Speed Decisions (and several other network objects).

2.1.3 Vehicle Compositions, Inputs, and Routings

Vehicle Compositions, Inputs, and Routings act to determine: what Vehicle Types are simulated (Compositions), how many vehicles are simulated (Inputs), and where those vehicles go in the network (Routings). These three network parameters are closely related and should be considered together when developing a Vissim network.

2.1.3.1 Vehicle Compositions

Vehicle Compositions are assigned to each individual Vehicle Input and determine the relative flow of different Vehicle Types which enter the simulation. Each Vehicle Composition will have one or more vehicle types assigned to it, along with a Desired Speed Distribution and Relative Flow for each Vehicle Type.

The Desired Speed Distribution (discussed in **Section 2.1.5**) determines the range of desired speeds that individual simulated vehicles will have upon entering the network. In general, the same Desired Speed Distribution should be used for each Vehicle Type within a given Vehicle Composition unless field data dictates otherwise.

The Relative Flow determines what portion of Vehicle Input Volume should be simulated as each Vehicle Type included in a given Vehicle Composition. For example, if a Vehicle Composition has three Vehicle Types with a total Relative Flow of 1 and a Relative Flow for Vehicle Type A of 0.25, then 25% of the Vehicle Input Volume associated with that Vehicle Composition will be simulated as Vehicle Type A.

2.1.3.2 Vehicle Inputs

Vehicle Inputs determine the actual volume of vehicles entering the Vissim network in Vehicles per Hour. Vehicle Inputs are assigned to specific Links within the Vissim network and different flow rates can be set for specific Time Intervals relative to the Simulation Period. Typically, Vehicle Inputs should be coded on “entry links” (i.e., Links with no upstream connectors) as these links represent the outer bounds of the Vissim network. *TOSAM* dictates that inputs should be developed for every 900 second (15 minute) interval of simulation. Different Vehicle Compositions can be assigned to each Time Interval for a given Vehicle Input, which should be done if sufficient field data is available. Otherwise a single Vehicle Composition can be assigned to every Time Interval for a given Vehicle Input.

Volumes for vehicle inputs are always in Vehicles per Hour, even when Time Intervals are less than or greater than 3600 seconds (1 hour). For example, an input with a Time Interval of 0-900 seconds with a volume of 2,000 vehicles will simulate—500 vehicles during that 900 second interval, provided there is no downstream impedance to restrict vehicles entering the network.

Vissim also gives an option for Vehicle Inputs to be treated as either “Stochastic” or “Exact”, where “Exact” Vehicle Inputs simulate the exact number of vehicles indicated by the flow rate while “Stochastic” Vehicle Inputs vary according to stochastic functions based on the Seed Number for a given run. *TOSAM* specifies that “Exact” Vehicle Inputs be used in Vissim (setting Vehicle Inputs to “Exact” will cause them to be highlighted yellow in the Vehicle Inputs list).

2.1.3.3 Vehicle Routings

Vissim provides several different Vehicle Routing types. This user guide will focus on the two most common Vehicle Routing types: Static and Partial Routes. All Vehicle Routings use Relative Flows to set how many vehicles should be assigned to each route based on the total number of vehicles that arrive at the Routing Decision during the time interval over which that Routing Decision is active. Vehicle Routings can also be set to apply to all Vehicle Types or to Specific Vehicle Classes.

Static Routings act as the base routing for the network and Routing Decisions for Static Routings should be coded on the same link as the Vehicle Input, as close to the beginning of the link as possible. When coding Static Routes, users can either code routes as “relay routes”, where a new routing decision is provided at each decision point (e.g., every off-ramp on a freeway or every intersection on an arterial facility), or as “end-to-end routes”, where a single continuous route is provided from the link where vehicles enter the network to the link where vehicles exit the network. “Relay routes” are often easier to code and do not require the user to account for travel patterns within the Vissim network beyond the total demand volume on each link. “End-to-end routes” will account for travel patterns within a Vissim network and are often based on trip tables obtained from external tools such as travel demand models or other origin-destination data sources. In either case, it is important to make sure that Static Routes are provided for vehicles at all points in their path within a Vissim network. If Static Routes are not provided, vehicles can either: become “rogue vehicles” wherein an un-routed vehicle will be randomly assigned a path by Vissim at each decision point it encounters, or the un-routed vehicle may exit the simulation altogether if the downstream link has less lanes than the upstream link (i.e., a lane drop).

Partial Routes act as a secondary routing system in conjunction with Static Routes if the user wants to control the relative flow of vehicles where multiple paths are available for vehicles from a given origin to a given destination. For example, if the collector-distributor road shown in **Figure 2-3** is known under existing conditions to carry 20% of the through traffic, while the other 80% of traffic stays on the mainline, then Partial Routes can be coded as shown to account for that split during simulation.

2.1.4 Transit Routes and Transit Stops

Vissim provides the ability to code Transit lines that operate, or are planned to operate, within a Vissim network study area. Transit vehicles operate different than personal vehicles, instead of being dictated by Compositions, Inputs, and Routings transit vehicles run according to Transit Routes and Transit Stops.

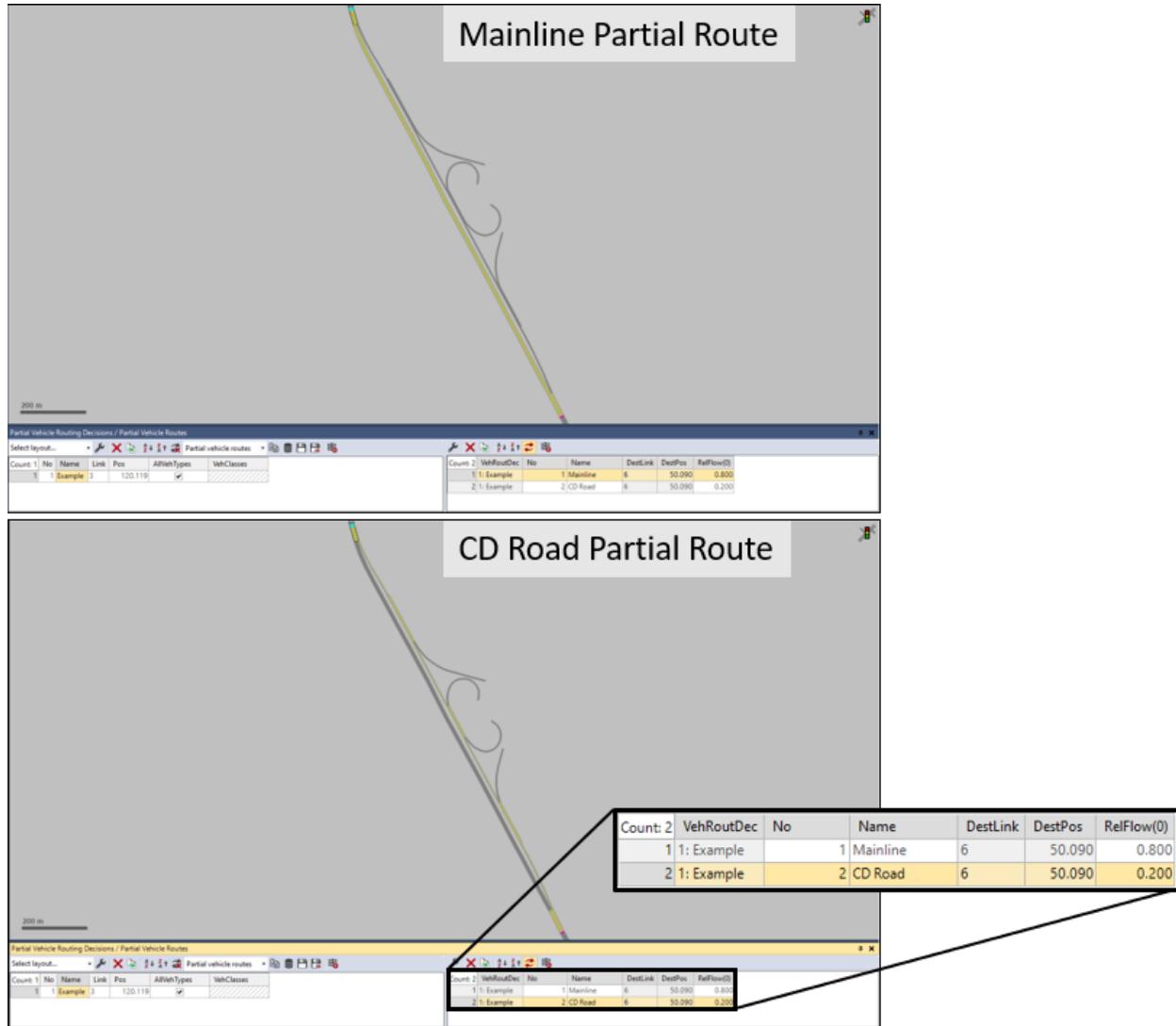
2.1.4.1 Transit Routes

Transit Routes determine the route that the transit vehicle will take through the network. All Transit Routes originate from the start of a link, like Vehicle Inputs. Each Transit Route is given departure times which determine the precise time that the transit vehicle will arrive in the network. Vehicles are assumed to be exactly “on-time” upon entry into the network.

2.1.4.2 Transit Stops

Transit Stops determine where on a Transit Route path a given transit vehicle will stop and for how long. The length of each stop can be controlled in two ways: Predetermined Dwell Times or Passenger Boarding. Predetermined Dwell Times are set according to Dwell Time Distributions. Passenger Boarding is modeled by adding passengers into the network at each stop to simulate boarding and alighting conditions more accurately in congested networks.

Transit Stops can also be toggled active/inactive for any line that traverses the location of that stop. This functionality is useful for areas where multiple transit groups are operating and may not share stops, or in situations in with specialty services such as employer shuttles or school buses are provided.

Figure 2-3. Partial Route Coding Example


2.1.5 Desired Speed Decisions and Reduced Speed Areas

Desired Speed Decisions and Reduced Speed Areas assign speeds to vehicles at different points along their path (as dictated by their route) during simulation. Both Desired Speed Decisions and Reduced Speed Areas operate based on Desired Speed Distributions which provide a distribution of potential Speeds that can be assigned to each Vehicle Class during simulation. Vissim provides a set of default Desired Speed Distributions, but TOSAM provides specific distributions to be implemented for various posted speeds, right turns, and left turns which should be implemented in lieu of the Vissim provided defaults.

2.1.5.1 Desired Speed Decisions

Desired Speed Decisions permanently update the Desired Speed of a vehicle. Vehicles enter the network during simulation with an assigned Desired Speed according to their associated input composition. Once a vehicle encounters a Desired Speed Decision, the Desired Speed will be updated according to the Desired Speed Distribution associated with that Desired Speed Decision for that Vehicle Class. This process will continue for each subsequent Desired Speed Decision encountered by that vehicle if the Desired Speed Decision encountered has a Desired Speed Distribution assigned to the class of that vehicle.

A vehicle will not accelerate or decelerate prior to encountering a Desired Speed Decision in anticipation of a change in its Desired Speed. For Example, if a vehicle with a Desired Speed of 40 mph is approaching a Desired Speed Decision which will update its Desired Speed to 60 mph, that vehicle will continue to travel at 40 mph until it crosses the Desired Speed Decision, at which point it will begin to accelerate up to its new Desired Speed of 60 mph according to the assigned Desired Acceleration Function for that given Vehicle Type. Therefore, it is important to account for the distance required for vehicles to accelerate or decelerate after receiving their new Desired Speeds, particularly in the case of on-ramps and off-ramps on freeway facilities.

2.1.5.2 Reduced Speed Areas

Reduced Speed Areas assign a temporary Desired Speed to a vehicle while that vehicle is within the defined Reduced Speed Area, after which the Desired Speed of the vehicle reverts to the Desired Speed from before it encountered the Reduced Speed Area. Reduced Speed Areas will only affect vehicles with a Desired Speed greater than the temporary Desired Speed assigned to it within the Reduced Speed Area. If the Desired Speed of a vehicle is less than the temporary Desired Speed that would be assigned to it within the Reduced Speed Area, then that vehicle will ignore the Reduce Speed Area and continue to operate at the Desired Speed.

In addition to applying a temporary rather than a permanent change to the Desired Speed of a vehicle, Reduced Speed Areas also differ from Desired Speed Decisions since a vehicle will decelerate in anticipation of a Reduced Speed Area to reach its temporarily reduced Desired Speed. The vehicle will then maintain that reduced Desired Speed through the length of the Reduced Speed Area. Upon exiting the Reduced Speed Area, the vehicle will begin to accelerate back to their previous Desired Speed. The deceleration and acceleration length on either end of the Reduced Speed Area is a function of Desired Acceleration/Deceleration Functions of that Vehicle Type and the difference between Desired Speed and the temporary Desired Speed within the Reduced Speed Area of that Vehicle Type.

Another important aspect of vehicle behavior in Reduced Speed Areas is that a vehicle must cross the start of a Reduced Speed Area for it to take effect. For example, in the link coding shown in **Figure 2-4** vehicles entering link 84 from connector 10068 will not adjust their speed according to the Reduced Speed Area shown since the vehicles are entering part way into a Reduced Speed Area.

2.1.5.3 Speed Control

Desired Speed Decisions and Reduced Speed Areas are two ways to regulate speeds in a Vissim model to supplement resulting traffic friction from roadway geometry (e.g., slope), traffic control, and vehicle interactions. Desired speed decisions and reduced speed areas are used for different purposes in the model (e.g., setting speed limits and reducing the speed on links with high curvature); however, these elements should not be used to replicate congestion within the study area. The only exception is for bottlenecks completely outside the study area in which congestion spills back into the study area. It is important to verify whether these downstream bottleneck and capacity constraints are expected to be resolved with planned projects in the future model year and adjust future models accordingly.

2.1.6 Conflict Areas Versus. Priority Rules

Conflict Areas and Priority Rules are different network objects that can be used to dictate the interaction between simulated vehicles at the intersection between two links (e.g., an at-grade intersection crossing). Conflict Areas are typically easier to apply to a network and are sufficient for dealing with most vehicle interactions, however in some cases geometric complexities can make Conflict Areas difficult or impossible to apply, in which cases Priority Rules may be required.

2.1.6.1 Conflict Areas

Conflict Areas allow the user to specify priority at any location where two or more link intersect. Vissim will automatically determine whether there is a potential conflict with yellow highlighted lines when the Conflict

Areas network object is selected. From there it is up to the user to specify which link, if any should have priority. **Figure 2-5** illustrates the potential conflict priorities that can be established at an intersection of two one-way streets.

Users should avoid over constraining their model and only apply conflict areas in locations where they are necessary for the network to operate correctly. For example, during initial network coding a signalized intersection will likely not need a Conflict Area for the northbound/southbound versus eastbound/westbound through movements because those pairs of movements will never have simultaneous green time (provided the signal is coded correctly). However, a Conflict Area may end up being required at these movements during calibration to account for certain intersection specific behaviors (e.g., headways, safety distance factor, avoid blocking percentage, etc.). Calibration is discussed in **Section 5**.

Figure 2-4. Connector Entering Link Part Way Through Reduced Speed Area

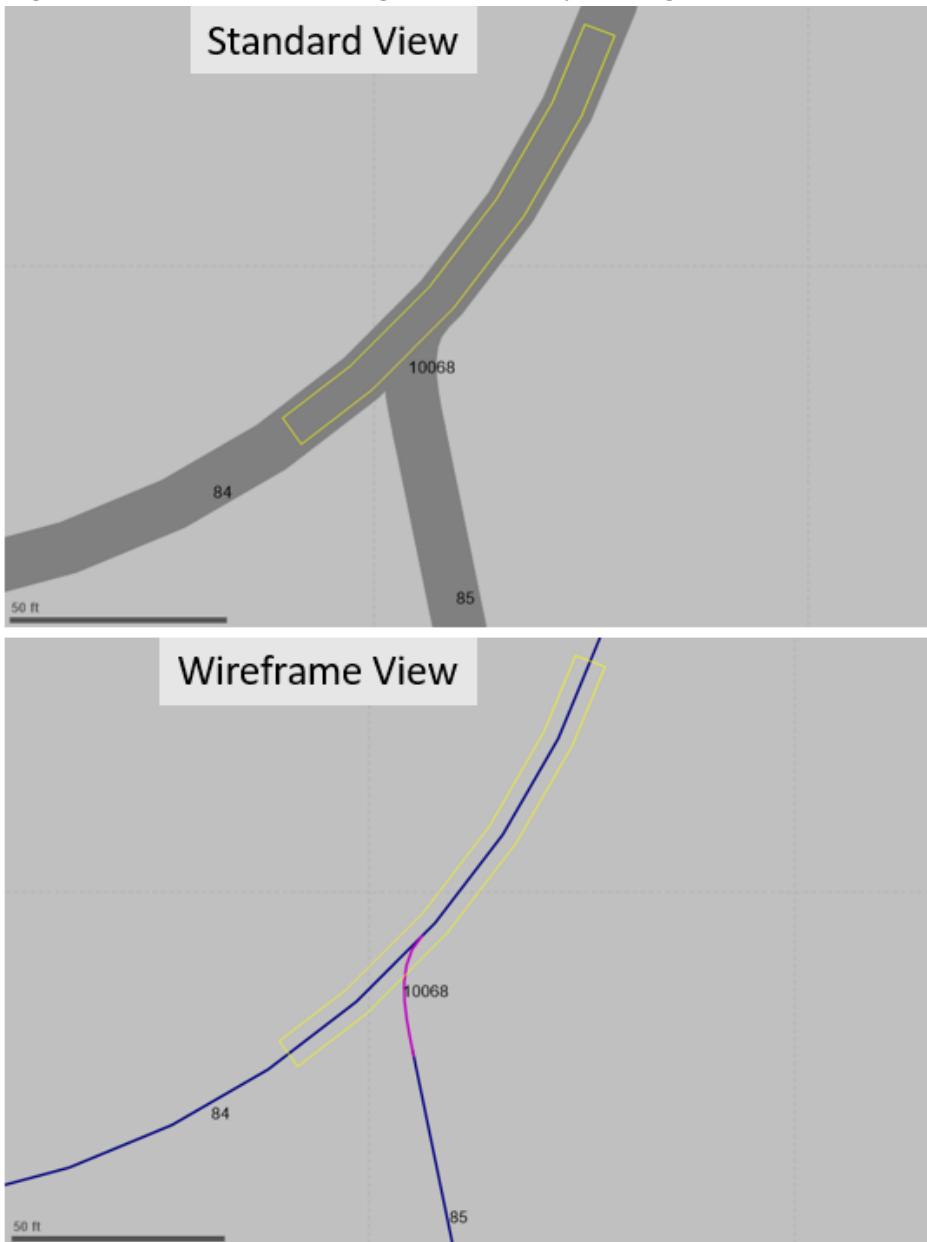
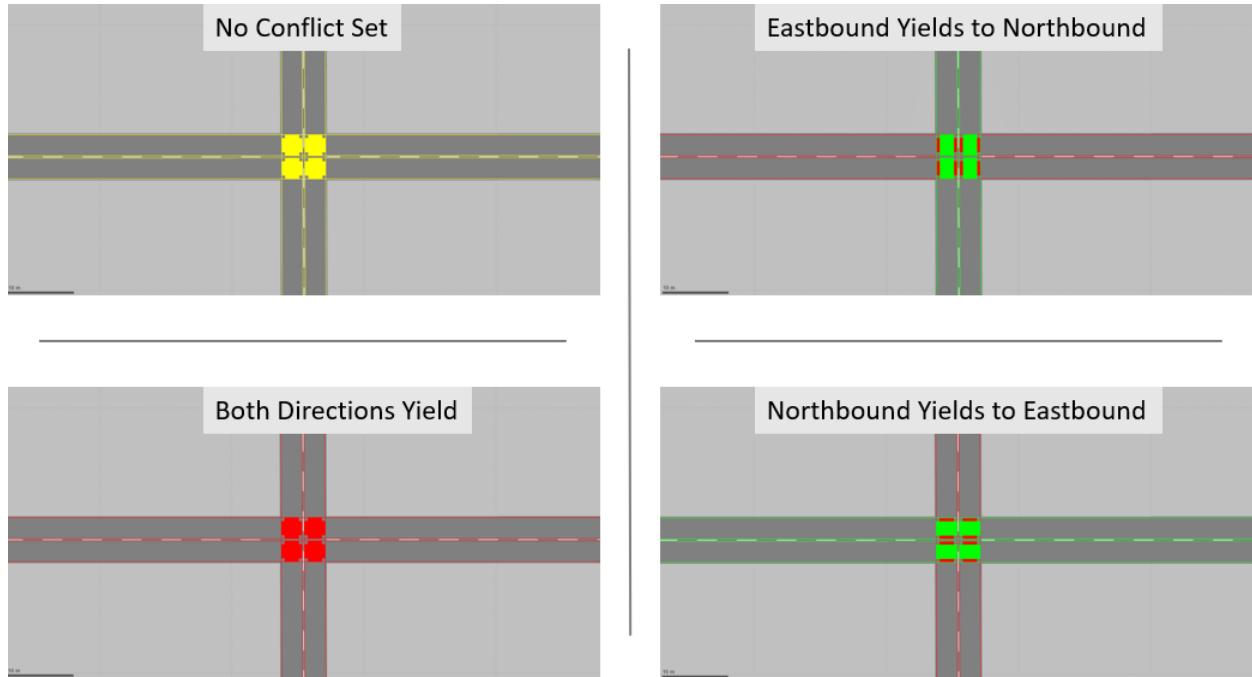


Figure 2-5. Conflict Area Options for Intersecting Facilities



2.1.6.2 Priority Rules

Priority Rules offer greater flexibility than Conflict Areas when determining how vehicle interactions should take place, however they are also more difficult to apply and should only be used when Conflict Areas cannot reasonably replicate the desired interaction. Two of the most common applications of Priority Rules are: multi-lane roundabouts and “do not block the box” situations (i.e., situations in which vehicles enter the box but downstream congestion does not allow them to clear the intersection, so they effectively get “run over” by opposing traffic streams during simulation). In both scenarios, a red bar is used to define the stop line of the yielding road, while green bars are used to define headways and time gaps that must be available for the yielding vehicle to move forward. Specific examples on how to code Priority Rules for roundabouts can be found in the PTV Vissim help manual. To apply a priority rule for “do not block the box” situations, the guidance provided in the PTV Vissim help manual should be referenced, and the following parameters should be set.

- The red marker should be set at the intersection stop bar (just upstream of the signal head).
- The green marker should be set at the downstream intersection departure.
- The “Link (all lanes)” should be selected for the Stop Line and Conflict Marker (additional coding may be needed if lane utilization is different between lanes).
- The minimum gap should be set to 0 seconds.
- The minimum headway should be set to at least half the intersection width and not to exceed the distance between the green and red marker.
- The maximum speed should be set between 8-13 mph, depending on traffic conditions.
- In the case of close intersection spacing, the downstream signal should be referenced with the associated signal phase.

2.1.7 Gradient

In Vissim, each link can be defined with a unique gradient coded to represent vehicle performance with specific roadway grades. Adjustments to the gradient influences the throughput, speed, density, and travel time of a corridor. As gradient increases from the default value of 0 percent, noticeable decreases in throughput and

speed are observed, while density and travel time increase. Due to the sensitivity of this parameter, it is important that it be coded accurately to reflect real-world conditions.

2.1.8 Simulation Setup

Simulation setup is defined in the simulation parameters window of a Vissim instance as shown in **Figure 2-6**. Simulation parameters are typically set during the development of the existing conditions model and kept consistent through all subsequent model scenarios for consistency of model results (model scenarios are discussed in detail in **Section 6**). Key Simulation Parameters include: Simulation Period, Simulation Resolution, Random Seed/Random Seed Increment, and Number of Runs. In addition to these key simulation parameters, which are discussed in detail, it is good practice to include the scenario name in the comment section provided in the simulation parameters window so that different files can be easily differentiated.

2.1.8.1 Simulation Period

The Simulation Period is the total simulation time per model run in simulation seconds. The Simulation Period is determined based on two different sub-periods: Seeding Period, and Analysis Period. The Seeding Period is the amount of time required for the network to reach saturated conditions. The standard convention is to use a Seeding Period equal to twice the free-flow travel time (travel time for vehicles operating at their desired speeds with no impedance/congestion) for the longest end to end route within the network. The Seeding Period may need to be further increased to replicate congestion formation in oversaturated networks.

The Analysis Period is the period over which network evaluation objects are actively recording network performance metrics (See **Section 4.1.1** on Evaluation Setup). The Analysis Period is determined based on the needs of the project and will typically range from one to several hours. Ideally, the Analysis Period will encompass the development and dissipation of congestion that occurs within the study area during the time period of interest under existing conditions.

2.1.8.2 Simulation Resolution

The Simulation Resolution determines how many time steps occur per simulation second. Higher Simulation Resolutions will create smoother and more realistic vehicle and pedestrian movements, but require a longer simulation run time. It is recommended that simulation resolution used in the production of final simulation results be kept to a minimum of 10 time steps per simulation second.

2.1.8.3 Random Seed/Random Seed Increment

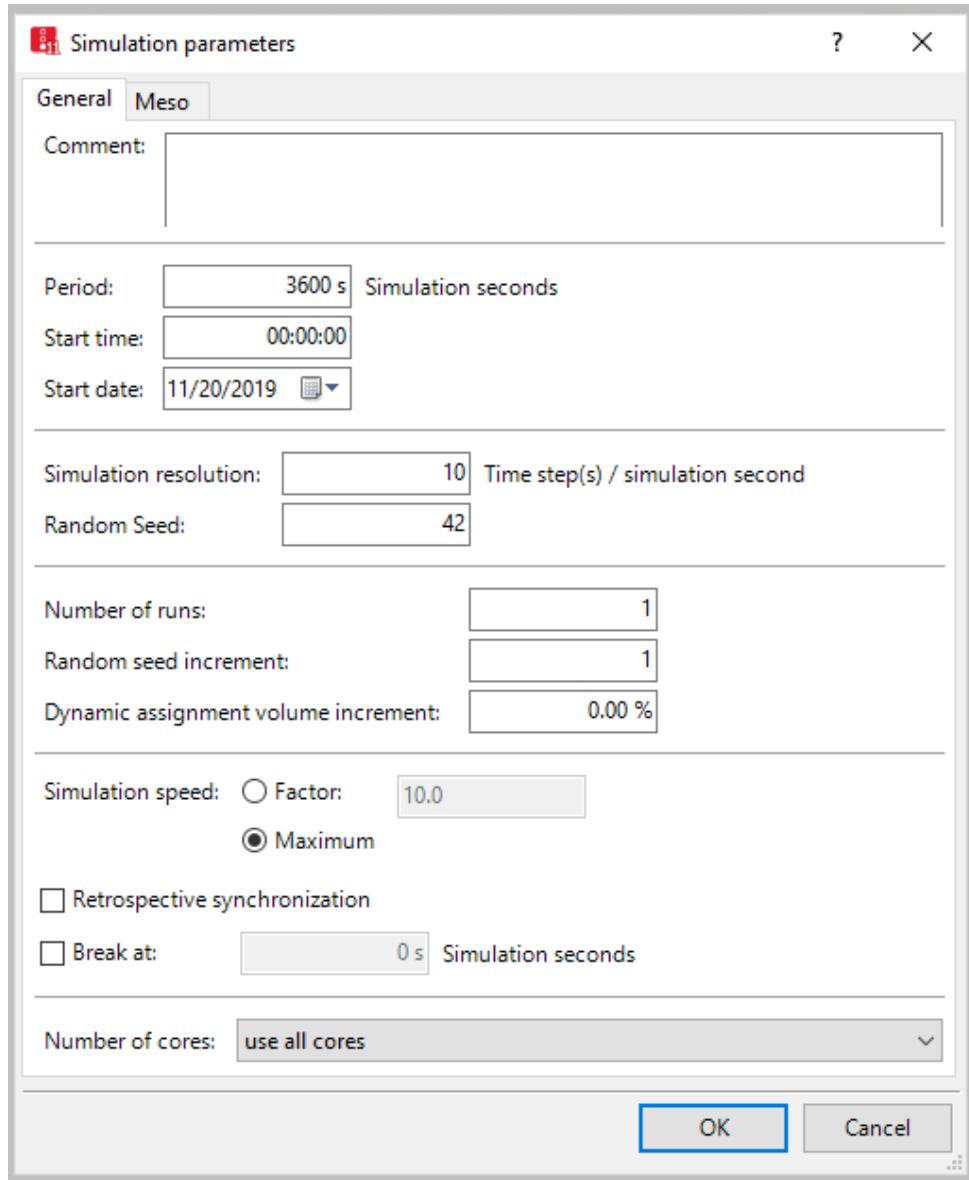
The Random Seed and Random Seed Increment parameter allow for stochastic variations of vehicle arrivals within the Vissim network, helping to account to variations in real world traffic conditions. The Random Seed value initializes a random number generator. Two simulation runs using the same network file and random start number will look the same; however, if the Random Seed is varied between two runs of the same network file, the stochastic functions are assigned a different value sequence thus changing traffic flow within the network.

The Random Seed Increment is the difference between random seed values when multiple simulation runs are performed (e.g., a network with a Random Seed Increment of 10 and a starting seed of 100 runs for 3 runs would have Random Seeds of 100, 110, and 120 for each of the respective simulation runs). Best practice is to use a consistent combination of Random Seeds/Random Seed Increment for existing and future year models.

2.1.8.4 Number of Runs

The Number of Runs parameter will determine how many Random Seeds the model will be run for, provided that the Random Seed Increment is set to a value greater than 0. The number of runs required for reporting of final simulation results should be determined using the VDOT *Sample Size Determination Tool*. At a minimum 10 runs, using different random seeds for each run (see **Section 2.1.8.3** Random Seed/Random Seed Increment for more information), are required to adequately account for variations in network operational performance.

Figure 2-6. Simulation Parameters Window



2.2 FREEWAY CODING

This section contains examples of best practices for common scenarios related to the coding of freeway corridors.

2.2.1 Merge/Diverge/Weave Segments

The coding of Merge, Diverge, and Weave Segments represent the most important aspect of most freeway corridor coding efforts since the vehicle interactions at these locations often dictate overall network performance. As such, it is very important that these locations are modeled correctly.

Because of the unique vehicle interactions that occur at Merge/Diverge/Weave Segments, custom driver behaviors are often necessary. As such, it is recommended that a “baseline” Merge/Diverge/Weave Segment

driver behavior be developed and applied to all Merge/Diverge/Weave locations within the network. Further discussion of the development of custom driving behaviors is included in **Section 5**.

2.2.1.1 Merge Segments

An example Merge Segment is shown in **Figure 2-7**. The on-ramp should connect to the mainline as a single link starting at the physical gore of the merge. This typically replicates field conditions wherein vehicles will begin to make lane changes before reaching the end of the solid lane markings. The on-ramp should continue as a separate link up to the end of the solid lane markings only if field data suggests that vehicles always respect the lane markings.

The other important thing to consider when coding merge segments is the Lane Change Distance for the connector downstream of the acceleration lane. If the acceleration lane is longer than the default lane change distance (656.2 feet) then the lane change distance may need to be adjusted to the length of the acceleration lane to allow vehicles to begin making lane changes once the on-ramp merges onto the mainline.

Further consideration should be given to the Lane Change Distance with respect to the through movement as well. In the case of acceleration lanes that are longer than the default Lane Change Distance (656.2 ft), the lane change distance for the through movement connector should be increased to be at least the length of the acceleration lane. Otherwise, vehicles intending to use the through movement may use the acceleration lane briefly until the Lane Change Distance condition is met. This can create additional artificial congestion, particularly in an already congested network.

In cases where the length of the acceleration lane is less than the default Lane Change Distance (656.2 ft), the Lane Change Distance for the connector downstream of the acceleration lane can be left at its default value unless field observations dictate otherwise.

2.2.1.2 Diverge Segments

An example Diverge Segment is shown in **Figure 2-8**. Like Merge Segments, the off-ramp should separate from the single mainline link at the physical gore of the diverge. This allows vehicles enough distance to make last minute lane changes, as often happens in the field. The off-ramp should only separate from the mainline at the beginning of the solid lane marking if field data suggests that vehicles always respect the lane markings.

Another key consideration for Diverge Segments is the Lane Change Distance coded on the connector leading to the off-ramp. The Lane Change Distance should typically reflect driver expectation and so should be coded according to the location of the first Guideway or Overhead sign informing drivers of the approaching off-ramp. This value may need to be adjusted during the debugging/calibration process to better reflect field conditions. If there are multiple lanes being traversed, this initial estimate can be set by activating the “per-lane” option. In doing so, vehicles will change lanes based on the input distance for each lane they need to traverse. For example, if a vehicle needs to change three lanes, and a lane change distance of 1,500 feet was assigned, the vehicle would make the first lane change 4,500 feet upstream, the second 3,000 feet upstream, and the third 1,500 feet upstream.

Further consideration should be given to the Lane Change Distance on the through connector as well. In the case of deceleration lanes that are longer than the default Lane Change Distance (656.2 ft), the lane change distance for the downstream through movement connector should be increased to be at least the length of the deceleration lane. Otherwise, vehicles intending to use the through movement may utilize the deceleration lane briefly until the Lane Change Distance condition is met. This can create additional artificial congestion, particularly in an already congested network.

In cases where the length of the deceleration lane is less than the default Lane Change Distance (656.2 ft), the Lane Change Distance for the connectors leading to the through movement downstream of the deceleration lane can be left at their default value unless field observations dictate otherwise.

Figure 2-7. Freeway Merge Segment Coding Example



2.2.1.3 Weave Segments

An example Weave Segment is shown in **Figure 2-9**. Weave Segments should follow the same rules as Merge and Diverge Segments to determine the locations where ramps tie into and from the mainline, and what the appropriate starting Lane Change Distance should be.

2.2.2 Link Segmentation

For reporting purposes, freeway links should be segmented to reflect HCM influence areas defined for merge, diverge and weave locations. This allows for more consistent reporting of freeway results. Definitions of Influence Areas are as follows:

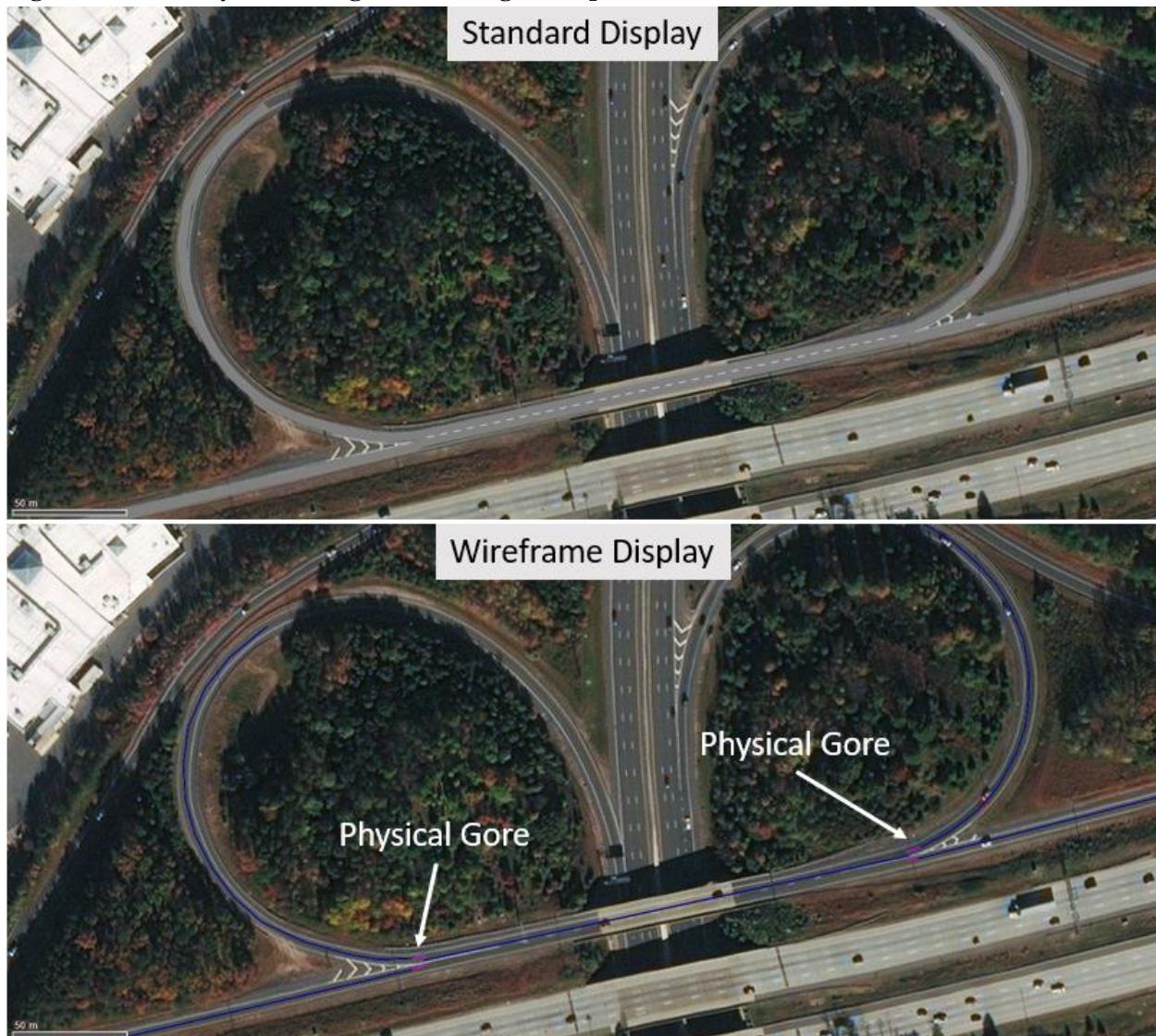
- Merge Influence Area: 1,500 feet downstream of the location where the on-ramp meets the mainline.
- Diverge Influence Area: 1,500 feet upstream of the location where the off-ramp separates from the mainline.
- Weave Influence Area: the mainline section between two ramps where the distance between on-ramp and a subsequent off-ramp is less than 3,000 feet.

Figure 2-8. Freeway Diverge Segment Coding Example



For example, in the merge represented in **Figure 2-10**, the link downstream of the end of the acceleration lane is broken at 700 feet so that the two links together equal approximately 1,500 feet. Results for these two links should be aggregated together to report results for this Merge Segment.

Figure 2-9. Freeway Weave Segment Coding Example



2.2.3 Parallel/Concurrent Facilities (e.g., HOT/HOV)

A common occurrence in coding freeways is the presence of Parallel/Concurrent Facilities. Parallel Facilities refers to facilities which run beside the general-purpose lanes, often in the median, but only allow access to and from the general-purpose facilities at specified locations. Concurrent Facilities refers to restricted facilities which are directly connected to the general-purpose facilities and allow traffic to move freely between the two facilities. In some situations, a facility may be intended to operate as a Parallel Facility, but the absence of any physical barrier allows vehicles to violate the boundary between the two facilities, causing it to behave as a Concurrent Facility. In these cases, the decision on how to code the interaction between the two facilities should be made based on observations made in the field.

2.2.3.1 Parallel Facilities

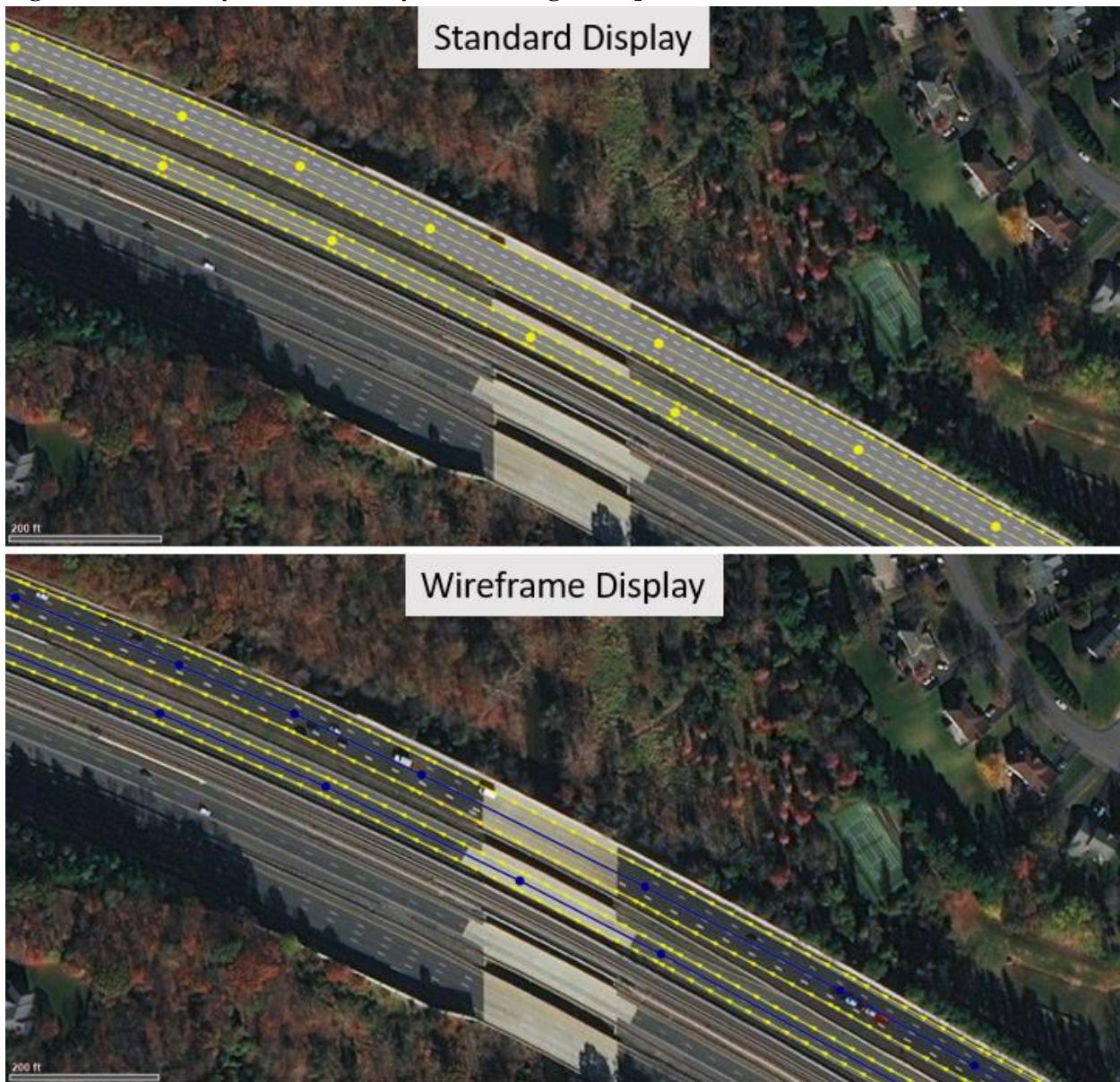
Parallel Facilities are coded as a separate link beside the general-purpose facility as shown in **Figure 2-11**. In this case, vehicle restrictions are not required as Vehicle Routings will determine which facilities vehicles use. Parallel Routings may be required for this type of facility to ensure that the parallel facility is utilized as intended.

Figure 2-10. Freeway Link Segmentation Example for Merge Influence Area



2.2.3.2 Concurrent Facilities

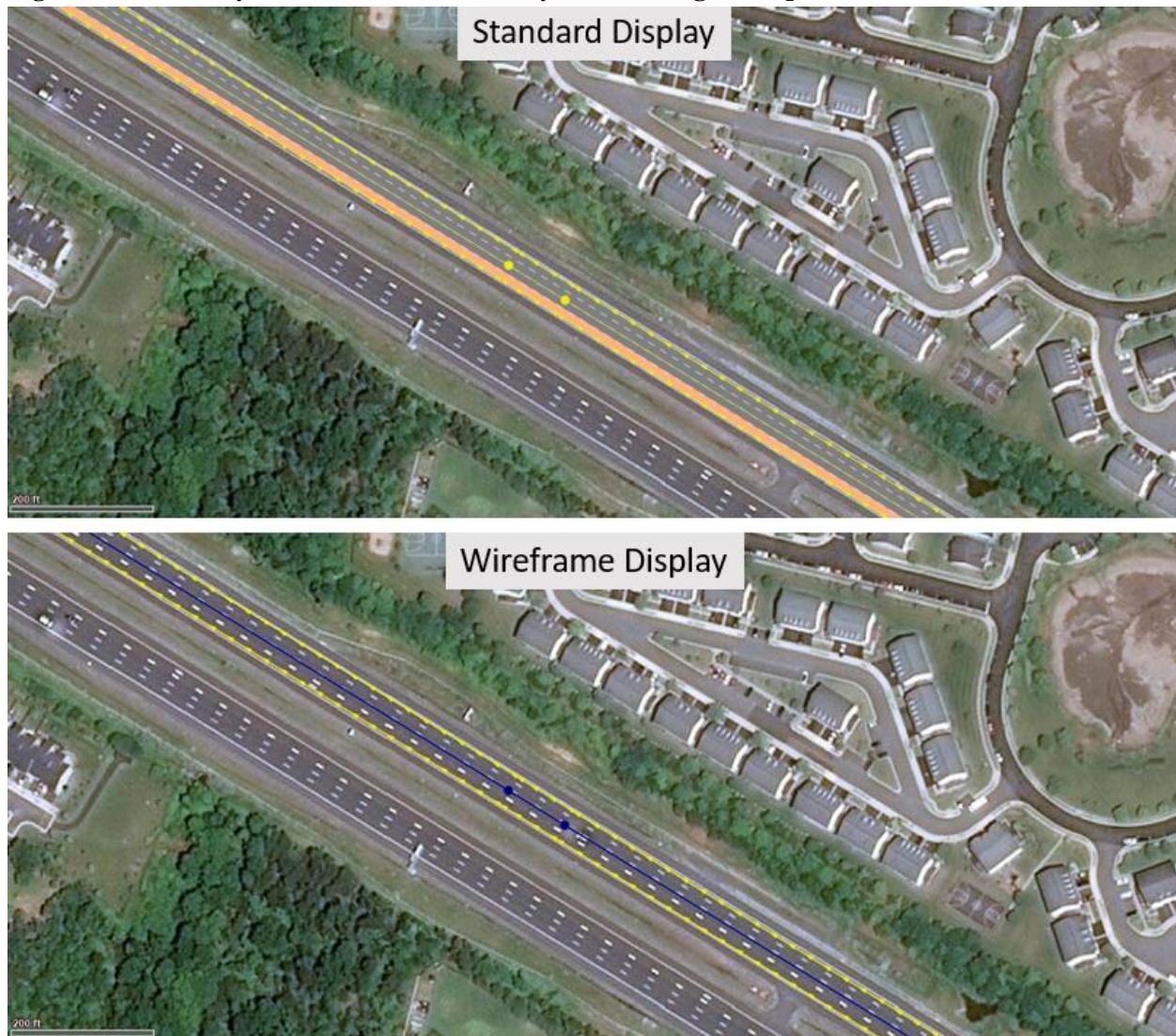
Concurrent Facilities are coded as lanes on the same link as the general-purpose facility. In this case, Vehicle Restrictions are required in the Concurrent Facility (e.g., HOV only in the left-most lane) to prevent certain Vehicle Classes from using the concurrent facility. An example of a concurrent HOV facility is shown in **Figure 2-12**, where all vehicle classes except the HOV Vehicle Class are restricted from using the left-most lane. A separate HOV Link Display type is also used in the left-most lane to clearly identify which lanes are restricted. HOT lanes, like HOV facilities, can be coded as concurrent facilities on the same link with General Purpose lanes or as an independent link if physically separated from General Purpose lanes.

Figure 2-11. Freeway Parallel Facility Link Coding Example

2.2.4 Ramp Meters

For Vissim models that employ ramp meter operations, the functionality of a ramp meter should match real-world conditions as closely as possible. For fixed rate ramp meters, a fixed-time signal controller can be used to replicate the meter. For ramp meters with more complicated logic, such that metering rates are dependent upon mainline speed/density or arterial queuing, VAP should be used to code the meter. The VAP logic needs to be calibrated to replicate field conditions; VDOT may request the VAP codes for review. Additional methods, such as using RBC, can be considered when coordinating with VDOT to determine a proper way to code ramp meters.

Figure 2-12. Freeway Concurrent HOV Facility Link Coding Example



2.3 ARTERIAL CODING

This section contains examples of best practices for common scenarios related to the coding of arterial corridors.

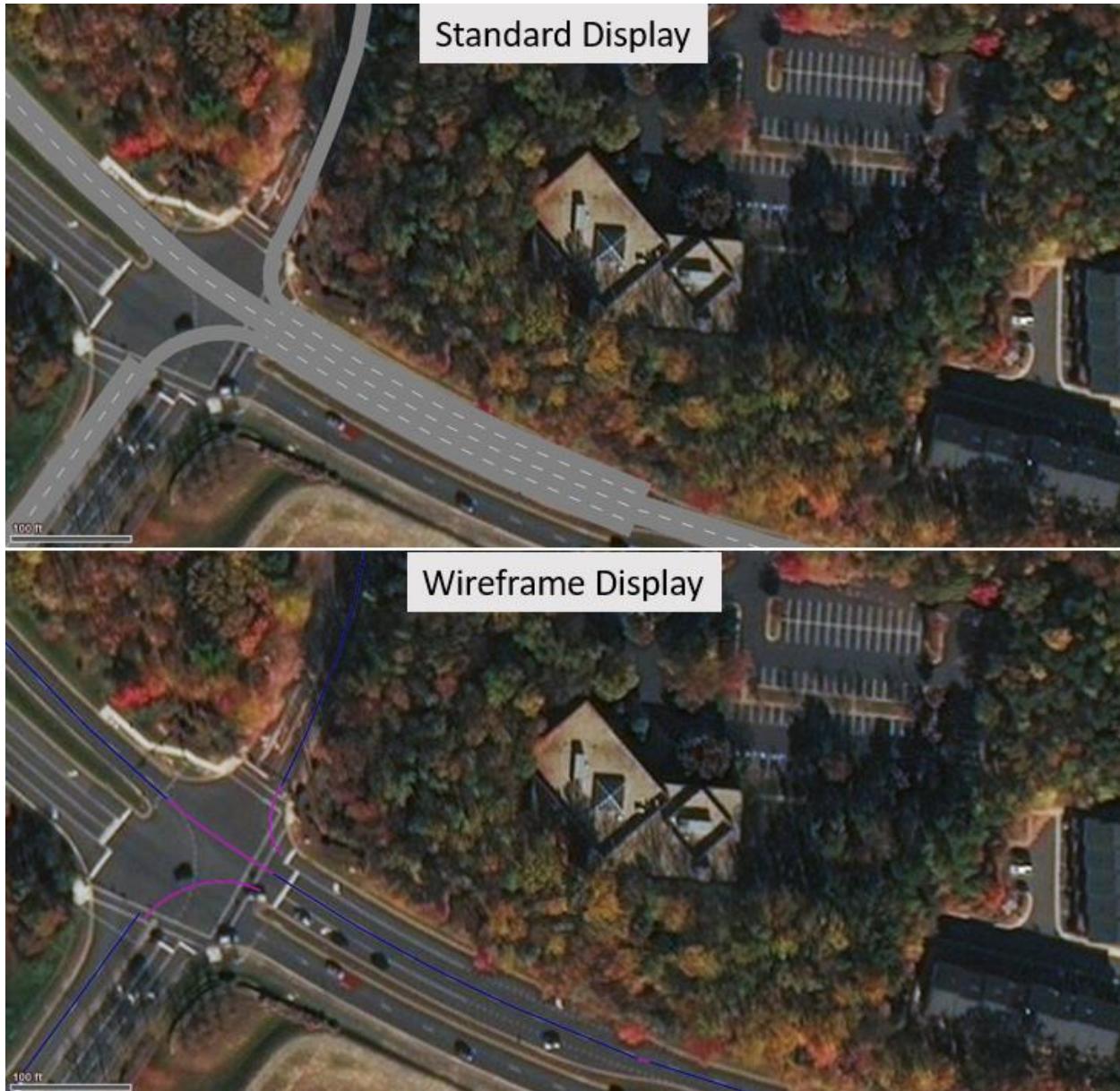
2.3.1 *Intersection Approaches*

Intersection Approaches can take many forms. Guidance on how to code three of the most common intersection approaches is provided in the following sections.

2.3.1.1 *Left Turn, Right Turn, and Through Movement with No Barrier Separation*

Figure 2-13 illustrates an intersection approach with one left-turn lane, one right-turn lane, and two through lanes. Since there is no barrier separation between either turn lane, the approach should be coded as a single link up to the point where the through lanes would intersect the perpendicular approach. Connectors should be used to cross the intersection for all three movements including the through movement.

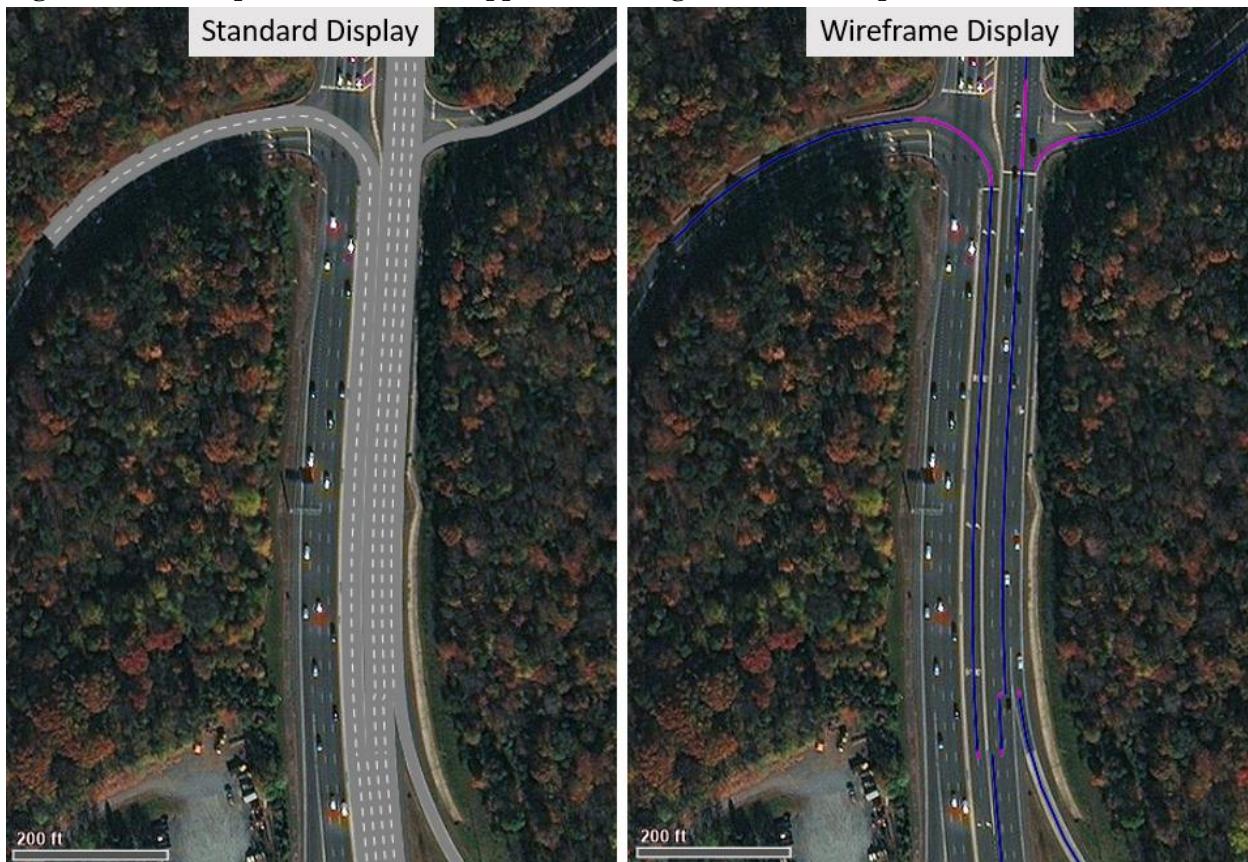
Figure 2-13. Example of Intersection Approach Coding with No Barrier Separation



2.3.1.2 Barrier-Separated Left Turn, Right Turn, and Through Movement

Figure 2-14 illustrates an intersection approach with two barrier-separated left-turn lanes, one right-turn lane, and four through lanes. Since there is barrier separation between the left lanes and the through lanes, the left-turn lanes should be coded as a separate link once barrier separation begins. The right-turn lane remains connected to the through lanes as a single link since there is no separation between the through and right-turn lanes. Once again, connectors should be used across the intersection for all three movements including the through movement.

Figure 2-14. Example of Intersection Approach Coding with Barrier Separation for Left-Turn Lane



2.3.1.3 Through Movement with A Channelized Right Turn

Figure 2-15 illustrates an intersection approach with one channelized right-turn lane, two left-turn lanes, and two through lanes. The channelized right turn should be coded as a separate link and should separate from general-purpose lanes at the beginning of the “pork chop”.

Also note at this intersection the left-turn lane connector is attached to the right two lanes of the downstream link (lanes one and two). This is because at this intersection the left two lanes of the downstream link are left-turn lanes as well, which can be seen partially in the wireframe display. Striping at this intersection dictates that vehicles making left turns from the northbound approach should enter the right two lanes of the downstream link, also shown in the wireframe display. In general field conditions and observations should dictate where on the downstream link the connector for each movement should attach.

2.3.2 Stop-Controlled Intersections

For Stop-Controlled Intersections, stop signs should be coded at the same locations as the stop bars in the field. Conflict areas/priority rules should be coded at the actual vehicle-vehicle conflict zone. For intersections with only yielding control, vehicle interactions should be controlled by just conflict areas and/or priority rules. Coding of unsignalized intersections should start with conflict areas and if it is necessary to better replicate real-world conditions, priority rules can be used as a supplement.

Figure 2-15. Example of Intersection Approach Coding with Channelized Right Turn



2.3.3 Signalized Intersections

For signalized intersections, Ring Barrier Controller (RBC) is to be used for coding signal control unless an alternative method is approved by VDOT. It contains all the standard parameters of a real-world controller to model free-running and coordinated signal operations, plus some advanced options, such as full actuation with volume density function and advanced pedestrian timing. The frequency of the RBC file is a factor of the simulation resolution. RBC also has its own preemption and transit priority module that is capable of emulating standard preemption and priority signal operations. RBC settings can be imported from Synchro. It is a good practice to maintain a Synchro model for coding and optimizing timing while simulating it in Vissim.

While signal files can be directly imported from Synchro and other signal optimization software, it is critical to perform a quality check of the imported RBC file to verify all elements of the signal timing plan were transferred correctly. The following list is not exhaustive; however, it contains the common elements of a signal timing plan that often require manual adjustment of the Vissim *.RBC file after importing from Synchro.

- Verify offset from the Synchro “TS2 – First Green”, which matches the “LeadGreen” reference point for signal control in Vissim.

- Verify signal phase overlaps were correctly imported.
- Verify all vehicle and pedestrian detectors were correctly imported.
- Verify pedestrian control settings (e.g., walk-in-rest, leading pedestrian intervals, pedestrian recall) were correctly imported. Sometimes these are not in the Synchro file and the original signal timing cards should be referenced.
- For coordinated signal operations, verify that one signal group in every ring is marked as coordinated or else the signal will be free-running.

The preferred method for coding future/proposed traffic signal timing is to optimize signal timing using Synchro or another accepted signal timing optimization tool, as Vissim itself is not an optimization tool. The RBC can be imported or the signal timings can be manually coded into the *.RBC file from existing conditions. If changes to the signal timing plan are relatively minor, it is recommended to perform manual adjustments to the RBC file directly, rather than re-importing the file from Synchro or another signal optimization program. When large signal-timing changes occur, and *.RBC files are re-generated from an optimization software, the verifications need to occur, and the Vissim model should be fully debugged to ensure all signal heads, detectors, stop signs, and priority rules associated with signal phases are still correct. In Vissim 11, when a signal head is referencing a signal group that no longer exists, the signal head will be deleted; thus, diligence is needed to prevent model errors.

2.3.4 Right-Turn-on-Red Coding

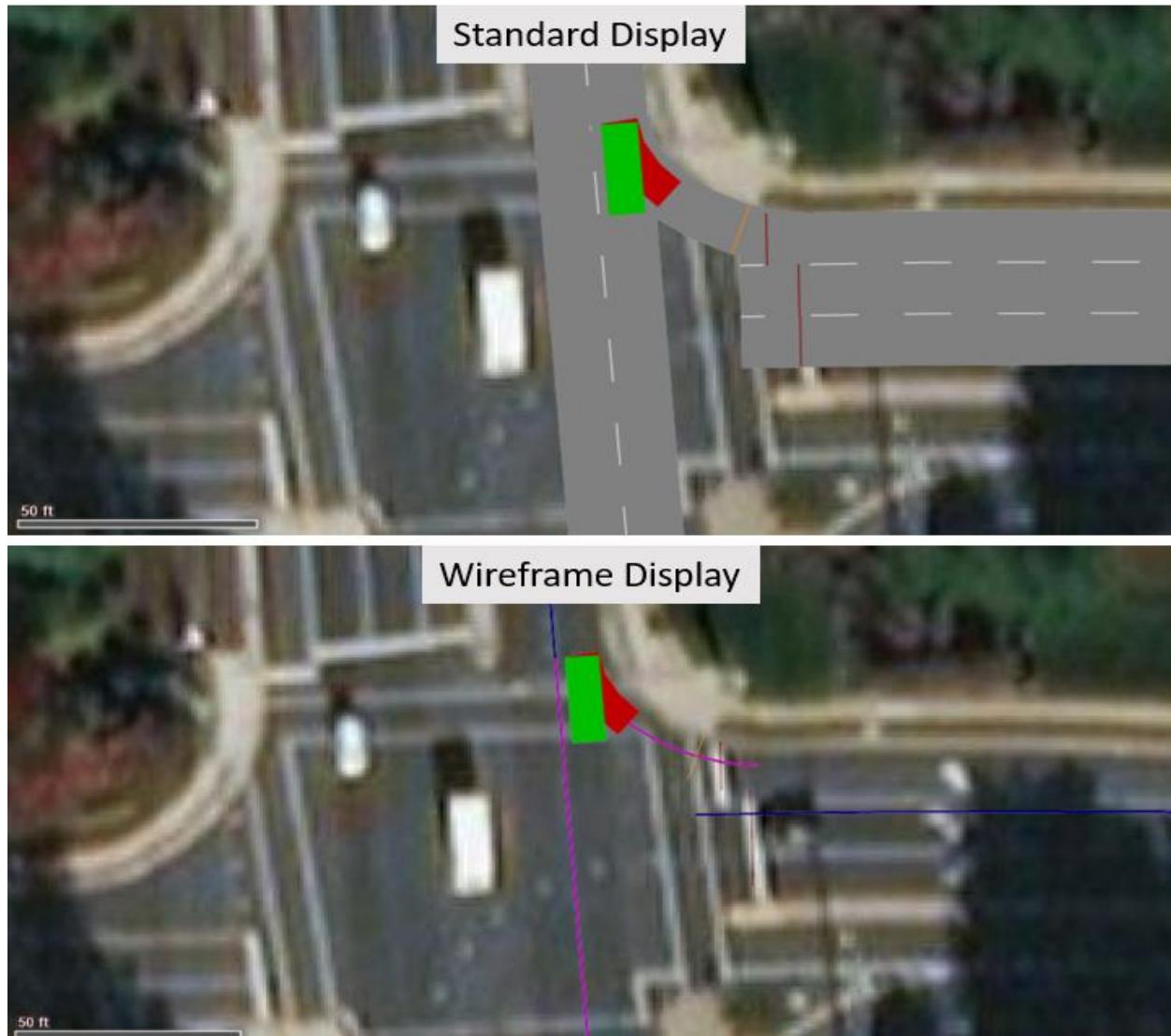
Often at signalized intersections, it is necessary to code right-turn lanes to allow for right turns on red. To code a right-turn-on-red, the signal head for the right-turn lane should be coded on the approach link and should be associated with the correct signal phase. The connector for the right turn movement should then be coded such that it overlaps the signal head for the right-turn lane, this will cause vehicles making right turns to ignore the signal head. A stop sign should then be coded on the connector, at the location where vehicles stop in the field when making right turns on red. The stop sign should be set as a “RTOR” stop sign and should be associated with the same signal phase as the right turn signal head. Lastly, a conflict area should be set for the interaction between the right-turn connector and the through movement connector, giving priority to the through movement. An example of coding for a right-turn-on-red is shown in **Figure 2-16**.

Software and hardware constraints should be clarified to model any unique signalization. For example, cycle length at some intersections in Virginia can run up to 6 minutes (360 seconds), while the cycle length or maximum split is limited to 255 seconds in RBC. Modelers need to be aware of such incompatibility between a real controller and RBC. It is recommended that Vehicle Actuated Programming (VAP) be used to model the long cycle and splits instead. PTV has adopted a list of real-world controller modules to Vissim in a virtual interface. When available, they may be used instead of RBC for certain cases with project manager approval. For adaptive traffic signal controllers, the source code should be used to model adaptive signal timings.

2.3.5 Pedestrian Crosswalks

Pedestrian Crosswalks can be coded similar to personal vehicles, with links representing the crosswalk where it crosses over each leg of the intersection, stop bars at the locations where pedestrians wait for the pedestrian phase of their movement to begin (and detectors if the signal uses pedestrian recall), and conflict areas coded between the vehicle and pedestrian links (in most cases pedestrians should have priority over vehicles). Compositions for the pedestrian Inputs should be coded using pedestrian Vehicles Types, making sure to apply a 2D/3D model distribution which includes only pedestrian models. In the simplified case shown in **Figure 2-17**, pedestrian routes are not necessary as the pedestrians simply leave the network after crossing their single link.

Figure 2-16. Right-Turn-on-Red Coding Example



2.3.6 Roundabouts

This section discusses the best practice of coding roundabouts. Roundabouts can be coded using two different processes. The first coding option is a more traditional approach: coding the roundabout from a blank file and drawing in all the links and connections. When using this approach, the modeler must draw continuous links through the roundabout, starting in Vissim 9 there is an option to create a circular link which is helpful when coding roundabouts from a blank file. Each lane should be modeled as its own link if modeling a two-lane roundabout. Connectors should be drawn as small as possible to ensure that no vehicles begin overlapping when the roundabout reaches capacity. This will limit the need for a conflict rule to be placed internally in the roundabout. An example coding of a dog bone interchange is shown in **Figure 2-18**. Detailed instruction for coding priority rules within roundabouts can be found in the *PTV Vissim User Manual*.

Figure 2-17. Example of Intersection Pedestrian Crosswalk Coding

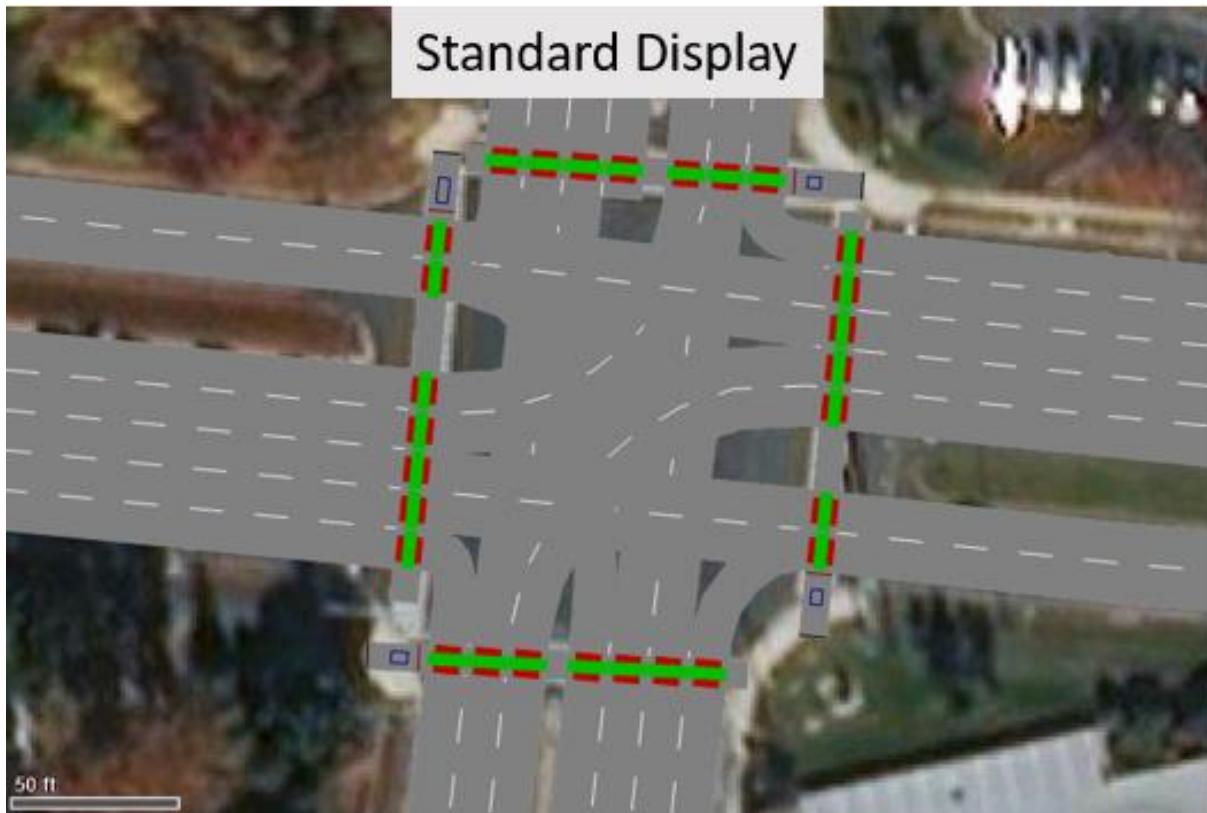
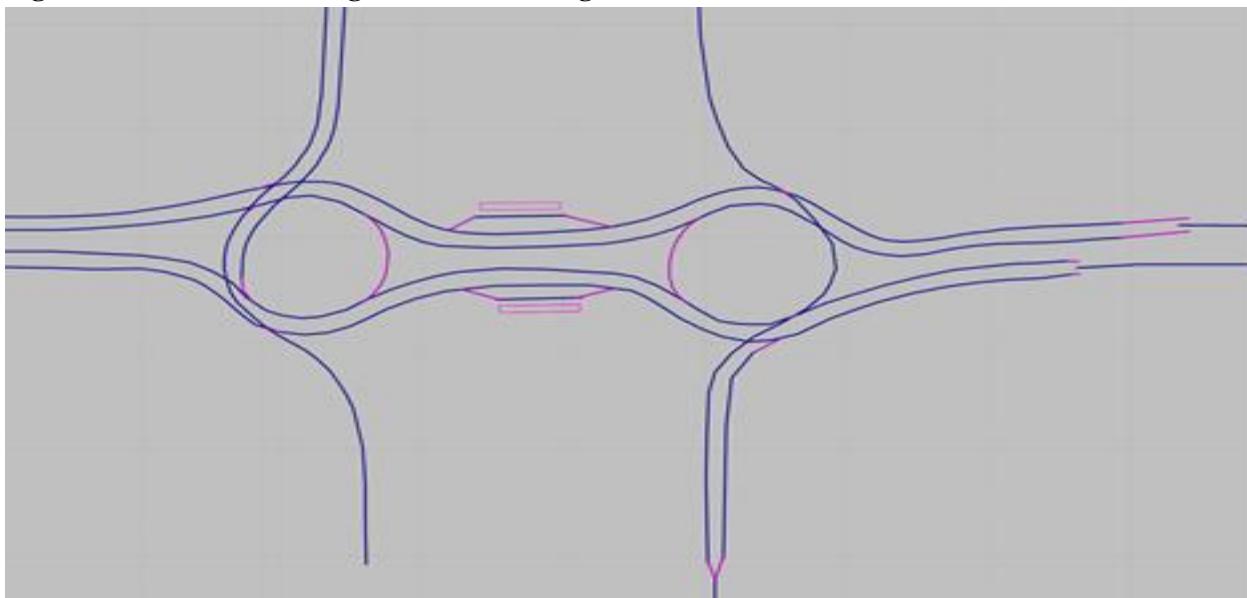


Figure 2-18. Two-Lane “Dog Bone” Interchange



3 Model Review and Debugging

This section contains examples of best practices for Review and Debugging of Vissim networks, once initial Model Development is complete. The Review and Debugging process typically consists of three steps:

1. Review of model input data
2. Animation review
3. Review of the model error log

Model Reviewing should be completed by a reviewer who is familiar with Vissim coding best practices, and who was uninvolves with the initial Model Development process. The reviewer should also be familiar with the approved methodology and scope of the project to ensure that the model will be able to serve its intended purpose.

For large models, multiple people may need to review simultaneously. In this case, division of responsibility and clear documentation of recommendations and corrections is crucial to ensure thorough model review and to prevent duplicative efforts. A master model file from the original Model Development process should be retained until Model Review has been completed at which point all agreed upon recommendations and corrections to the model coding should be applied.

3.1 MODEL DEBUGGING VERSUS CALIBRATION

To understand the purpose of the Model Review and Debugging process, it is important to distinguish between Model Debugging and Model Calibration.

Model Debugging is the process by which coding errors are identified and addressed. Error refers to any coding within the Vissim network that either does not conform to coding best practices, as discussed in **Section 2**, or that results in simulated conditions that deviate significantly from observed field conditions. Rectifying issues during the Model Debugging process typically would not include making driver behavior modifications (apart from ensuring that default driver behaviors are being used on the correct facility types). Instead Model Debugging focuses on larger issues such as: incorrect number of lanes, incorrect RBC coding, omission of Reduced Speed areas on ramps with large degrees of curvature etc. The Model Debugging process will not usually require reviewers to closely examine model results to identify issues. Instead debugging issues can typically be identified simply by: reviewing network coding before running the simulation, observing operations visually during simulation, or by reviewing the error log file produced by Vissim at the end of each run.

Model Calibration is the process by which the model is fine tuned to replicate observed field conditions within a relatively small margin of error. The VDOT *TOSAM* provides thresholds that must be met for a Vissim network to be considered calibrated. Model Calibration takes place after the Model Review and Debugging process has been completed. It is typically an iterative process in which adjustments are made to the network, several model runs are completed, and then results are reviewed and compared against data collected in the field. If simulated results do not meet the calibration thresholds established in the *TOSAM*, then further adjustments are made to network coding until simulated results meet the calibration thresholds and the model can be considered calibrated. Model Calibration is discussed in depth in **Section 5**.

Another key difference between Model Debugging and Model Calibration is that Model Calibration is only conducted for the existing conditions model, while a Model Review and Debugging process should be incorporated into each subsequent scenario in addition to the existing conditions model.

3.2 MODEL REVIEW CHECKLIST

A review checklist should be used to assist with the model inputs verification process. This list should cover the majority of Vissim input data from geometry data, traffic control and speeds, traffic demand data, and vehicle class/type, to driving behaviors. The reviewer should note any changes in assumptions that are

inconsistent with what was previously approved in the methodology and assumptions document. A quality control/review log form should be used by the reviewing team to document their findings. A template model review checklist can be found in [Appendix A](#).

3.3 IDENTIFYING MODEL CODING ERRORS

The Review and Debugging process consists of three steps:

1. Review of model input data
2. Animation review
3. Review of the model error log

These steps are discussed in detail in the following sections.

3.3.1 *Review of Model Input Data*

Review of Model Input Data is typically the first step in the Model Review and Model Debugging process. This step should include among other things: verifying that link/connector geometry matching existing conditions or the proposed design plans, ensuring the simulated vehicle mix is reflective of the vehicle mix observed in the field, checking that Compositions, Inputs, and Routings are coded such that link demand within the Vissim network is consistent with volumes observed in the field, and that Desired Speeds are correctly coded based on observed field data. For network objects such as Desired Speed Decisions, Reduced Speed Areas, signal heads, Vehicle Inputs, and others, reviewing the network object list can significantly improve the efficiency of the debugging process and can also allow for bulk editing of specific attributes.

A full list of Model Input Data which should be reviewed during the Model Review and Debugging process is included in the model review checklist in [Appendix A](#).

3.3.2 *Animation Review*

This step is to assure that model is free of obvious errors and fatal flaws. It is often overlooked if a thorough review has been given to the input data; however, model animation should be observed closely for the duration of the seeding and analysis period. The observations should start from a high level to check for gridlocks and unwarranted spillbacks and then zoom in to review key locations of the study area in greater detail. In the case of signalized intersections, multiple cycles should be observed at each intersection to ensure the signal is operating properly. Reviewers need to be familiar with field conditions because some “unexpected” issues may be explained by unique field conditions rather than errors. See [Section 4](#) for methods of reviewing and presenting simulation results.

It is anticipated that a large simulation model will involve multiple people observing the simulation; creating a plan to divide up the modeling area and a shared review log is crucial to identifying and resolving issues efficiently without overlapping efforts.

3.3.3 *Error Log*

Vissim provides an error log file (.err) at the end of a simulation run that describes the time and location of an error. Modelers should review the error log file to ensure that messages that indicate potential model coding errors are addressed. Important error messages include:

- An entry link did not generate all vehicles (too big a demand or downstream congestion coding error causes spillback)
- A vehicle did not react to its designated route because of the short distance between the first connector and the start of the routing decision
- Many vehicles were removed from the network because they have reached the maximum diffusion time before being able to make a lane change

Typically, the unfinished input and vehicle diffusion errors indicate congestions in the network. While the process of calibration described in **Section 5** may reduce the number of such errors, it is still the responsibility of the modeler to verify that such error messages did not result from coding issues. Unfinished demand should not occur in existing conditions models when demand is based on field counts as those field counts represent demand which is being served in the field. These errors should be rectified before moving onto the calibration phase for an existing conditions model.

3.3.4 Common Coding Errors

Vissim provides a large amount of flexibility to allow users to code any number of unique transportation facilities. However, this flexibility also allows for a wide range of potential errors to be made during model coding. As such an exhaustive list of potential coding errors would be difficult or impossible to provide. In some cases, the correctness of coding depends on the situation (e.g., a concurrent freeway facility coded as a parallel facility of vice versa).

The following list provides some coding errors common in Vissim network development, as well as how those errors can be identified during the debugging process.

- **Signal timing/signal head coding:** Signal timing is typically coded in a separate *.RBC file that is referenced in the given *.inp file and thus can be difficult to visualize. To assist with this Vissim provides a test function which allows the user to observe signal heads changing color in the network according to their referenced *.RBC file without running the full simulation. This function can act as a first step to check signal coding, however because the test function does not simulate vehicles, the test signal function will not account for vehicle actuation. RBC files should be reviewed when imported from a prior version of Vissim (e.g., RBC files originally developed in Vissim 8 and imported to Vissim 9) or from Synchro to ensure all signal timing parameter data are consistently transferred. Refer to Section 1.3 for general guidance on Vissim version control.
- **Vehicles driving over red signal heads:** When observing signal timings at intersections during simulation, pay careful attention to whether vehicles are stopping on lanes when signal head on that turns red. If a connector is coded over a signal head which is coded on a link, the simulated vehicles will be considered to be on the connector instead of the link and therefore they will not account for the status of the signal head.
- **Ensure adequate Conflict Areas are coded:** at any at grade crossing within the network, check that vehicles are not “running over each other” in locations where links overlap. If this behavior is observed, an additional Conflict Area is likely required so that vehicles on one link yield to vehicles on the other.
- **Check that Lane Change Distances are coded correctly:** A common issue, particularly in freeway coding, is the use of an insufficient Lane Change Distance for connectors. This error will typically manifest itself in Vissim networks in the form of long, unrealistic queues at decision points in the network (e.g., an off-ramp on a freeway).
- **Check that Desired Speed Decisions have been coded correctly:** Desired Speed Decisions dictate the speed vehicles will travel at within the network. If a Desired Speed Decision is omitted for example on an on-ramp leading to a freeway facility, vehicles from that on-ramp will continue to travel at the desired speed from the arterial facility they came from, which is typically lower than the Desired Speed on the Freeway. Users should check for simulated vehicles moving at significantly different speeds on the same facility.
- **Check that Reduced Speed Areas are coded at all intersections for right-turn and left-turn connectors:** Reduced Speed Areas are required on all intersection right and left turns to account for reduced vehicle speeds as they make those movements. *TOSAM* provides speed distributions that should be used for intersection right and left turns.
- **Check that correct vehicle types are coded on correct facilities:** During Animation Review, users should check to make sure the mix of vehicles is realistic for that facility. An obvious example would

be pedestrians being simulated on a freeway, which while technically possible within Vissim, is never a realistic condition. A less obvious example would be a very large percentage of trucks ($>20\%$) unless the user is modeling a unique facility where large percentages of trucks are common. In this case, it is important that the user be familiar with typical conditions on the facility being analyzed.

- **Ensure entry links are sufficiently long for downstream routing decisions:** Entry links, particularly those with two or more links, should be coded sufficiently long so that vehicles have space to make any necessary lane changes upon entering the network. If a user notes significant congestion at the beginning of their network, entry links may need to be extended further upstream.

4 Results and Presentation

4.1 EVALUATION

This section highlights different evaluation methods and the coding of evaluation parameters to measure model performance. The model simulation should be set up with a seeding period and an analysis period. Seeding period is necessary for initializing the model to match network conditions by the time the analysis period begins; therefore, MOEs are not collected during this period. Model performance is evaluated during the analysis period, which contains the peak hour defined for the project. The MOEs collected during the peak hour and full analysis period may vary on a project-by-project level. However, in general, speed heat maps, travel times, and queue lengths are generated from the full analysis period, whereas vehicle throughput and intersection delays are recorded just for the peak hour. These are based on the field data collected and balanced for model generation and calibration; therefore, field data collection intervals should be considered during the selection of simulation time periods.

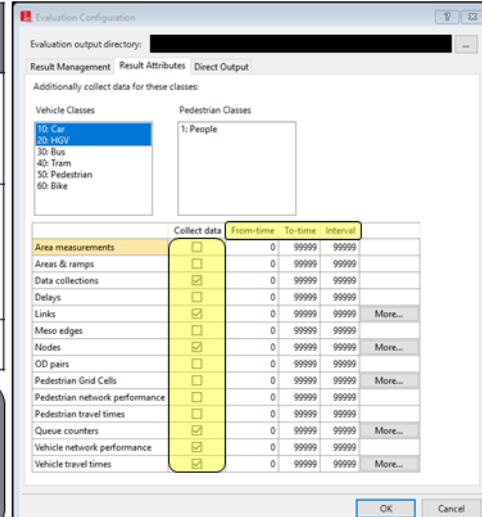
Simulation time periods can be adjusted from the Evaluation/Configuration window, under Result Attributes as shown in **Figure 4-1**. Within this window, the methods of model evaluation should be defined. Evaluation elements used in analysis should be selected under the “Collect Data” tab, and the time period and interval in which data are collected should be filled in subsequently. The following sections describe the most common Evaluation tools used to collect model results.

Figure 4-1. Evaluation/Configuration Window

Simulation Time Periods	Length	Vissim Evaluation for Calibration
Seeding Period	Dependent on network size. Recommended minimum of 30-minutes, preferred 60-minutes	None – No MOEs should be collected during seeding period
Analysis Period	Dependent on project scope. Recommended minimum of 2-hours , representing time before and after the Peak Hour.	Speed Heat Maps (15 minute increments), Travel Time, Queue Lengths
Peak Hour	1 hour	Vehicle Throughput, Intersection Delay

Seeding Period
Analysis Period

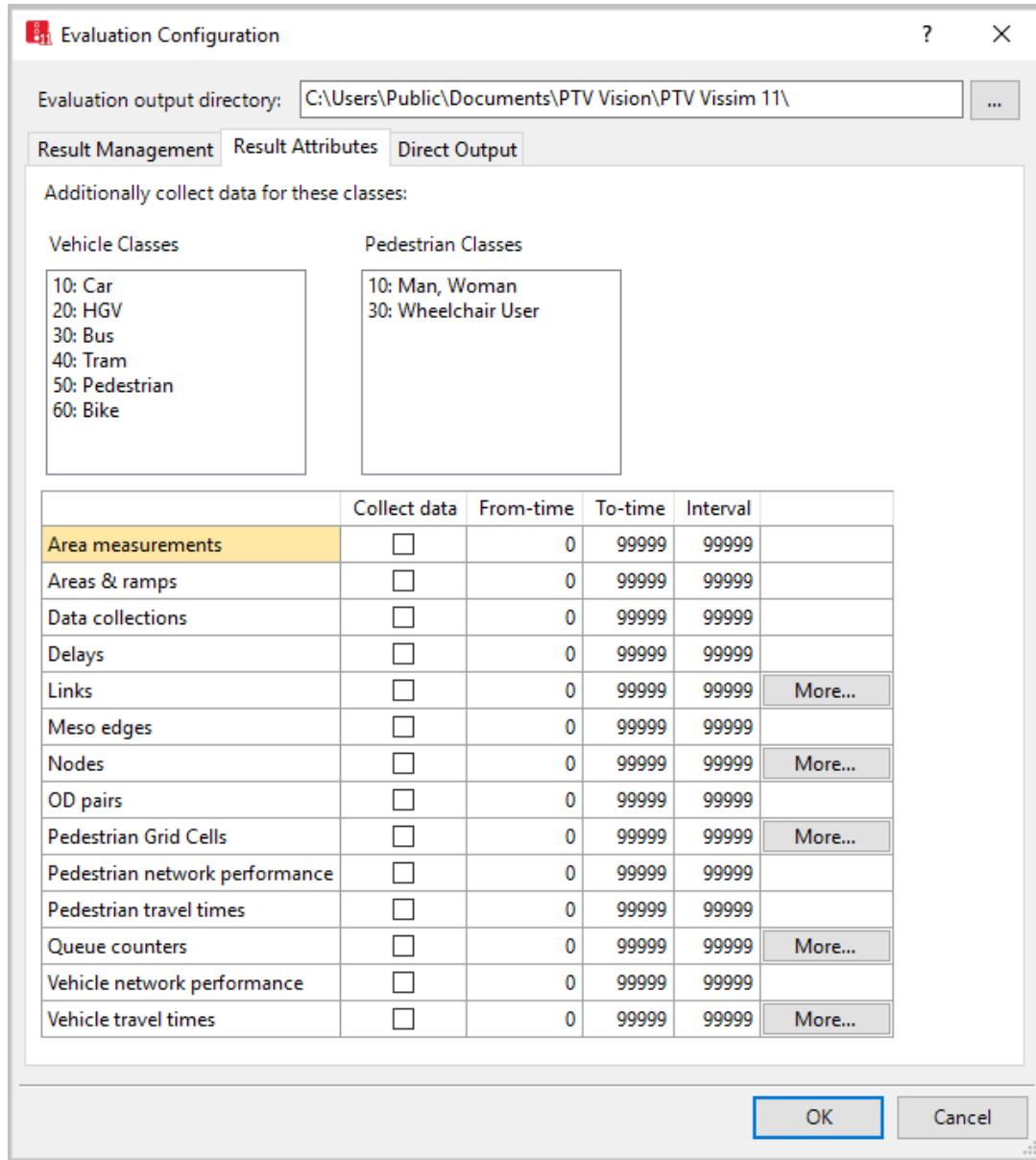
Peak Hour



The screenshot shows the 'Evaluation Configuration' dialog box. At the top, tabs for 'Evaluation output directory', 'Result Management', 'Result Attributes', and 'Direct Output' are visible. The 'Result Attributes' tab is active. Below it, a section titled 'Additionally collect data for these classes:' lists 'Vehicle Classes' (e.g., 10: Car, 20: Motor) and 'Pedestrian Classes' (e.g., 1: People). The 'Collect data' table below has columns for 'Area measurements', 'Areas & ramps', 'Data collections', 'Delays', 'Links', 'Meso edges', 'Nodes', 'OD pairs', 'Pedestrian Grid Cells', 'Pedestrian network performance', 'Pedestrian travel times', 'Queue counters', 'Vehicle network performance', and 'Vehicle travel times'. Each row in the table includes 'From-time', 'To-time', and 'Interval' fields. Most rows have 'From-time' and 'To-time' set to 0 and 'Interval' set to 99999, with 'More...' buttons for each column. Buttons for 'OK' and 'Cancel' are at the bottom right.

4.1.1 Evaluation Setup

Evaluation Setup is defined in the Evaluation Configuration window of a Vissim instance as shown in **Figure 4-2**. Evaluation Configuration parameters are typically set during the development of the existing conditions model and kept consistent through all subsequent model scenarios for consistency of model results (model scenarios are discussed in detail in **Section 6**). Key Model Configuration parameters are included in the Result Attributes and Result Management tabs of the Evaluation Configuration window.

Figure 4-2. Evaluation Configuration Window


4.1.1.1 Result Attributes

The Result Attributes tab allows the user to determine which attributes will be recorded during simulation, including the total simulation time over which data will be collected (“From time” and “To time” attributes) and the interval over which data will be aggregated (“Interval” attribute). For example, if the user wishes to collect results for “Links” in a model with a total Simulation Period of 5400 seconds (Seeding Period of 1800 seconds and Analysis Period of 3600 Seconds) with Link results aggregated for the entire Analysis Period, the user would check “Collect Data”, set “From time” to 1800, set “To time” to 5400, and set “Interval” to 3600. Some results attributes also have additional parameters that can be set under the “More...” control. For example, results for “Links” can be collected either per lane or per link segment depending on the project needs.

4.1.1.2 Result Management

The Result Management tab allows the user to determine how evaluation results are previous after each new run is initiated. Vissim provides the following three options.

- **None:** No results are recorded, this should only be used during the Debugging phase of network coding or if the user wishes to simply record an animation file with reporting results.
- **Of current (multi -) run only:** Results from only the current run, or set of multi-runs, are preserved. If this option is checked any results from previous runs or multi-runs are deleted when the simulation starts.
- **Of all simulation runs:** All results from any runs conducted for that model are retained.

Best practice is to use the “of Current (multi-) runs” option to maintain version control.

4.1.2 Node Evaluation

Node evaluation is primarily used to evaluate intersections, as it offers flexibility in defining boundaries for an intersection while providing a variety of MOEs for vehicles traversing the intersection (e.g., throughput, delay, queue, and number of stops). Typically, a node boundary is drawn as a square box encompassing an intersection. However, for intersections that overlap other movements (e.g., grade-separated, free-flow movements) or were designed with complicated geometries, nodes can be constructed to match individual turning movement to ensure precise outputs. Examples of two such intersections are shown in **Figure 4-3**.

Node settings can be adjusted within the Evaluation/Configuration window. As shown in **Figure 4-4**, the distance downstream where the delay segment begins can be adjusted and queue counter settings can be set. The *Vissim Intersection Data Processing Tool (Version 2.0)* can be used to process node results. More details on this tool can be found in *TOSAM Appendix F*. The default maximum queue length should be increased to a large value if long queues are known or anticipated to ensure the end of queue is correctly captured. The “Consider adjacent lanes” parameter is selected by default and results in queue length calculations that encompass vehicles in adjacent lanes (e.g., uneven lane utilization, queue spillover from turn lane).

4.1.3 Data Collection Points

Data Collection Points (DCPs) are commonly used to retrieve throughput volumes and spot speeds. DCPs should be set up on each lane at the same location of a link. See **Figure 4-5** for an example of DCP coding. The placement of DCPs should mimic the location where field data were collected. An individual DCP is inserted along each lane of a link. Each DCP should be named to match the index from the collected field data (i.e., in the example, count location A and count location B). Once all DCPs are coded and named, Data Collection Measurements (DCMs) can be generated manually or automatically. Vissim reports vehicular counts and spot speeds from DCMs—not DCPs—therefore, it is important to set up DCMs accordingly. An individual DCM can be coded for each DCP to capture lane-by-lane data, or DCPs can be grouped to capture full roadway cross-sectional data.

Figure 4-3. Node Defined for Intersection Movements

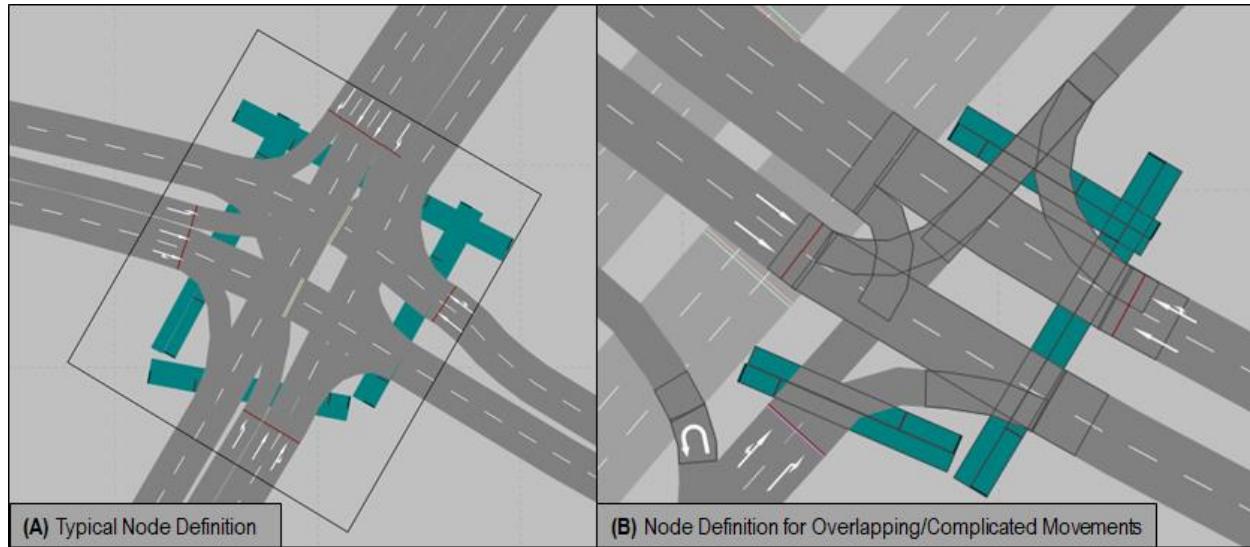


Figure 4-4. Node Evaluation Configuration and Settings

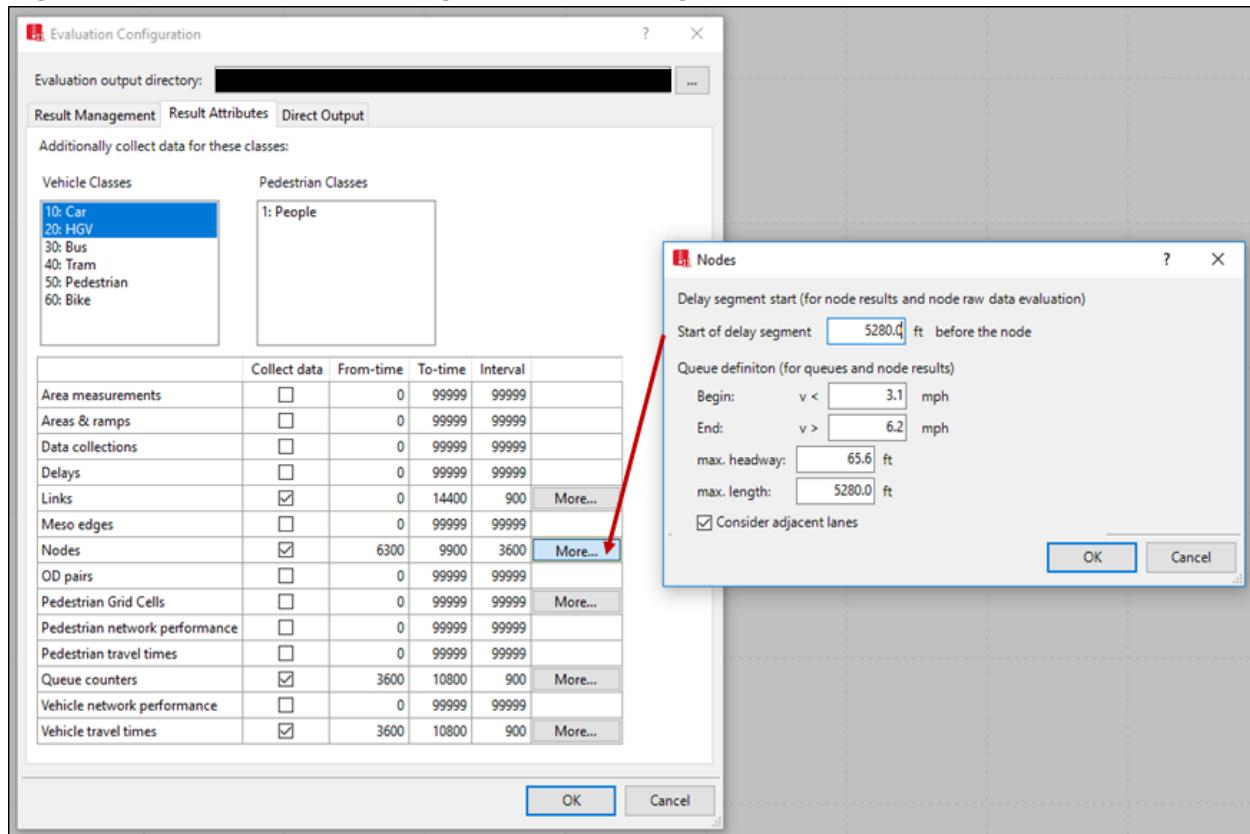
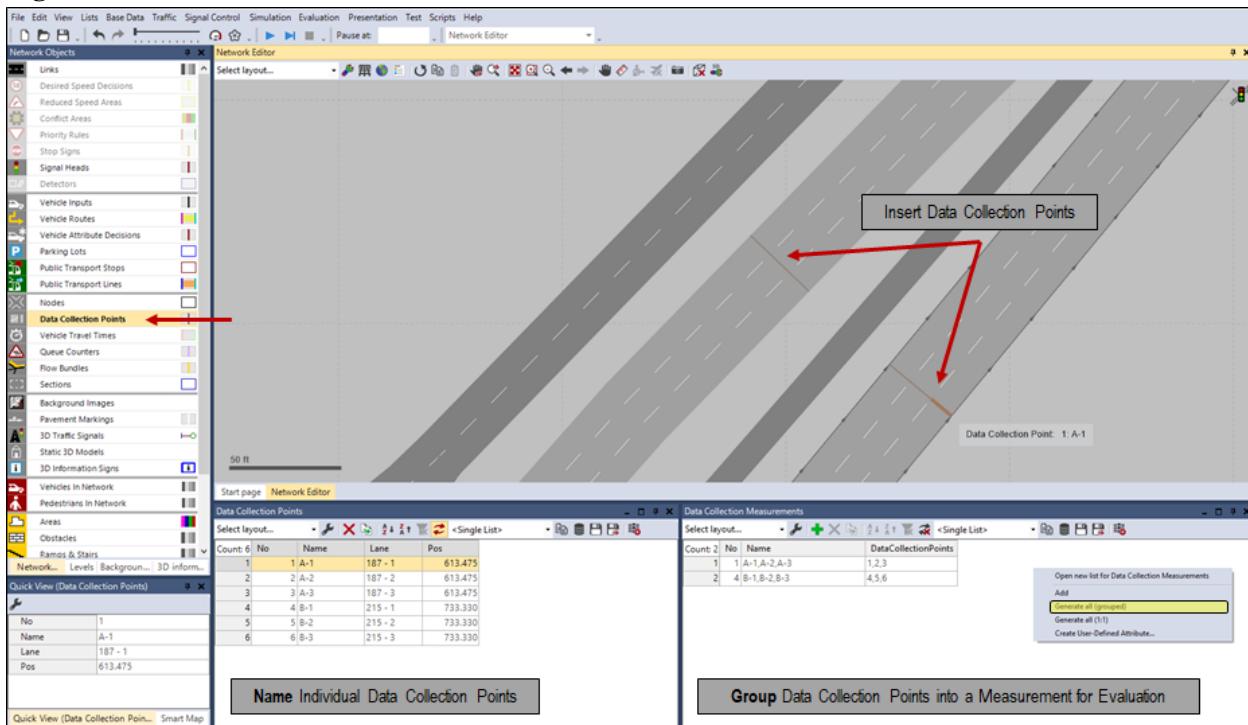


Figure 4-5. Data Collection Points and Data Collection Measurements Definition


4.1.4 Queue Counter

Queue counters are another coding element that can be used to measure queue length on freeway ramps and at intersections. Queues are measured from the downstream position of the queue counter until to the furthest upstream vehicle that has entered queueing conditions. Queue outputs are provided in terms of length, rather than the number of vehicles. A queue counter reports the average queue length, maximum queue length, and the average number of stops for the defined time intervals. The maximum queue length can extend as far upstream as the next queue counter or as long as specified in the attribute Queue Maximum Length. To properly capture Queues using Queue Counters, entry links (links with no upstream connectors) should be long enough to account for the longest queue length that might occur during simulation. Queue Counters will only record a queue length up to the end of the most upstream link, so any queue which extends outside the network will not be fully accounted for when reporting results.

As previously mentioned, Nodes are commonly used to detect queues at intersections. These embedded queue counters are automatically generated and can report a maximum queue length up to the attribute Queue Maximum Length or the distance to the nearest upstream Node.

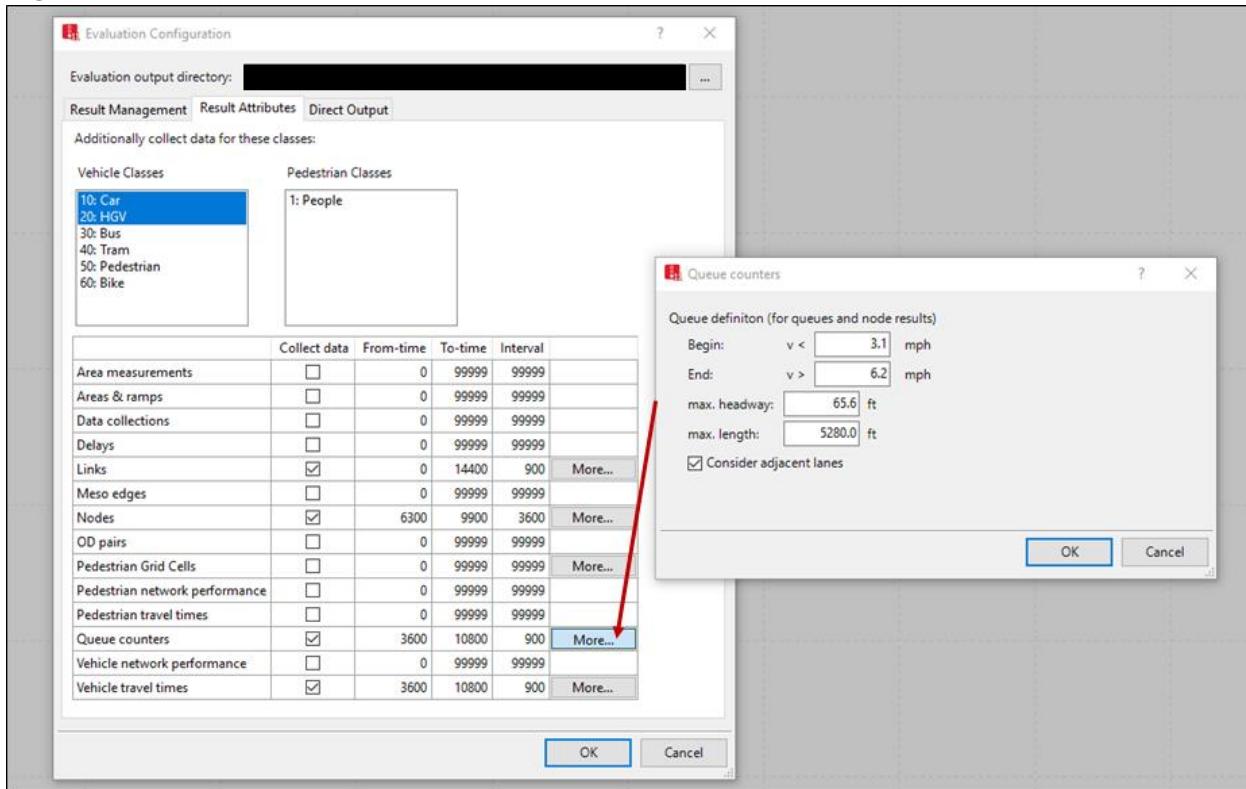
Queue counter settings can be found under Evaluation/Configuration, as shown in **Figure 4-6**. Like the Node result options, the default maximum queue length should be increased to a large value if long queues are known or anticipated to ensure the end of queue is correctly captured. The “Consider adjacent lanes” parameter is selected by default and enables the consideration of vehicles in adjacent lanes of previous links within the queue length calculation.

4.1.5 Travel Times

Travel time can be collected with Travel Time Collection (TTC) segments. TTC segments are coded with a starting point and stopping point, as shown in **Figure 4-7**. Evaluation of TTC segments reports the average time it takes for vehicles to traverse the defined segment during the selected time interval. Travel time segments cannot be directly developed to represent specific routes through the network (i.e., only vehicles passing both

the starting and stopping point are considered). Ultimately, these segments should be coded to match field-collected travel time segments and/or probe data segments (e.g., INRIX data) for direct comparison during calibration.

Figure 4-6. Queue Counter Definition



When placing start and end bars to record TTC segments it is important to place them carefully to ensure that the desired path and vehicles are being captured. If a TTC segments is set such that there are two paths from the start bar to the end bar, Vissim will record the average of all travel times for vehicles on both paths 1 and 2. To differentiate between the travel times on the two different paths, TTC segments would have to be coded in such a way that they filter out the unwanted path. One way to do this would be to code sequential TTC segments as shown in **Figure 4-8**, where TTC segments A and B would give the total travel time for path 1, while TTC segments C and D would give the total travel time for path 2.

The most common outputs from TTC segments are the (a) travel time, (b) total distance traveled, and (c) number of vehicles. Thus, from this information, the space mean speed (i.e., the average speed over the segment) can be collected and used in Speed Heat Maps for direct comparison with INRIX speed data during calibration. Typically, output results are collected as an average for all vehicle classes; however, detailed results on a per-vehicle class basis can be set up to calibrate against high resolution field data that differentiates vehicle class in speed data collection (i.e., average speed for single occupancy vehicles versus average speed for heavy vehicles).

Like the combination of Data Collection Points into Data Collection Measurements, TTC segments can be combined into Delay Measurements. Delay measurements capture the number of vehicles, average delays, and average stop delays for one or several TTC segments. Thus, the total delay along a corridor can be captured by combining consecutive TTC segments into a single Delay Measurement.

Figure 4-7. Travel Time Collection Segment Definition

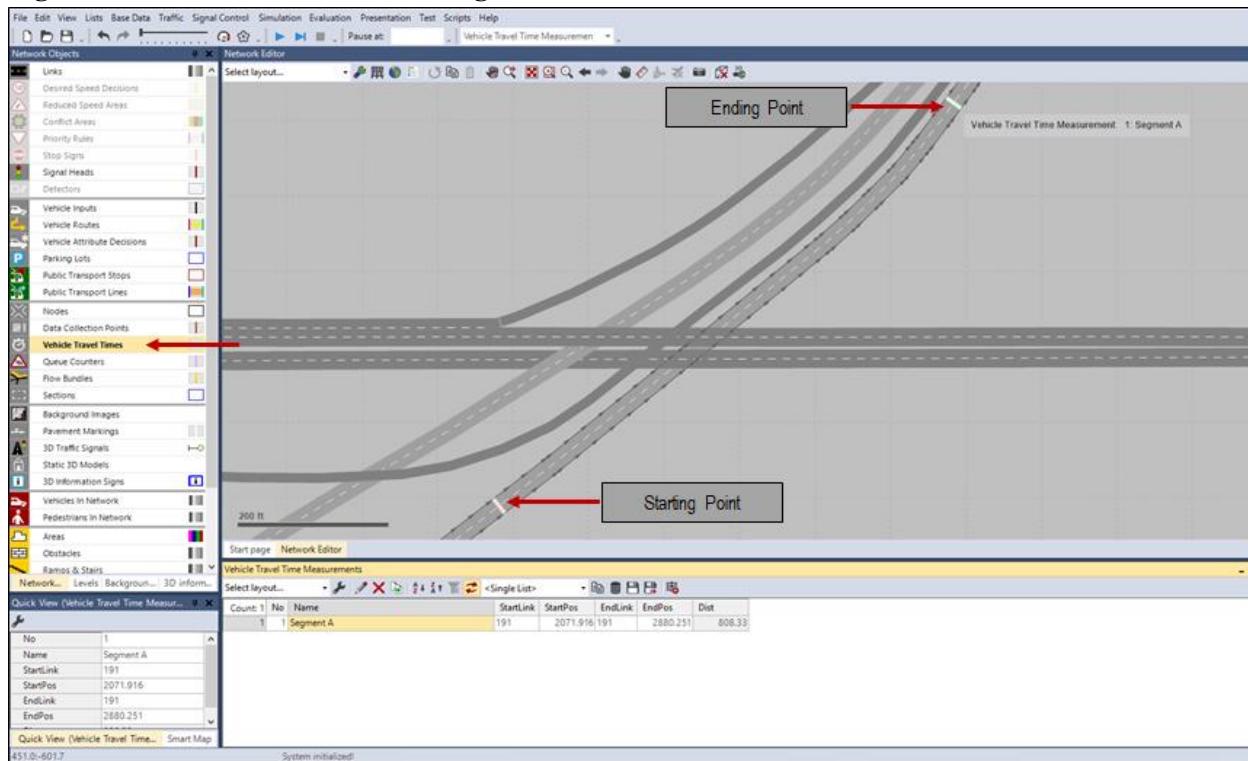
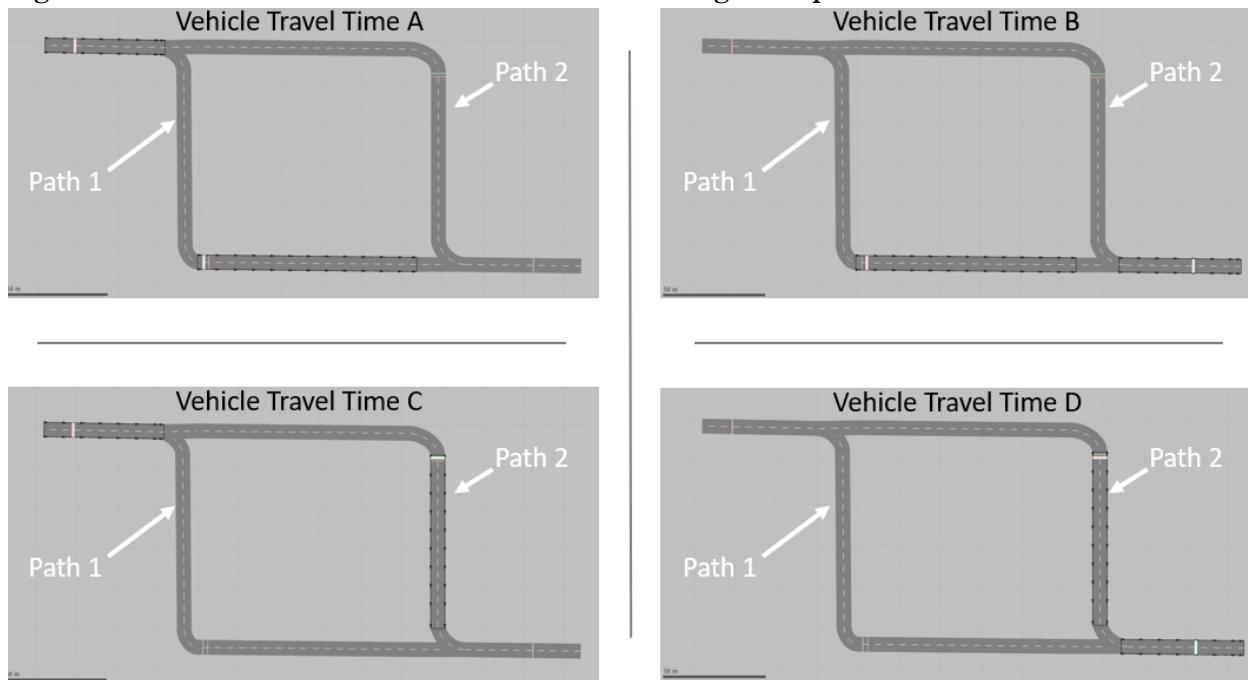


Figure 4-8. Travel Time Collection Parallel Routes Coding Example

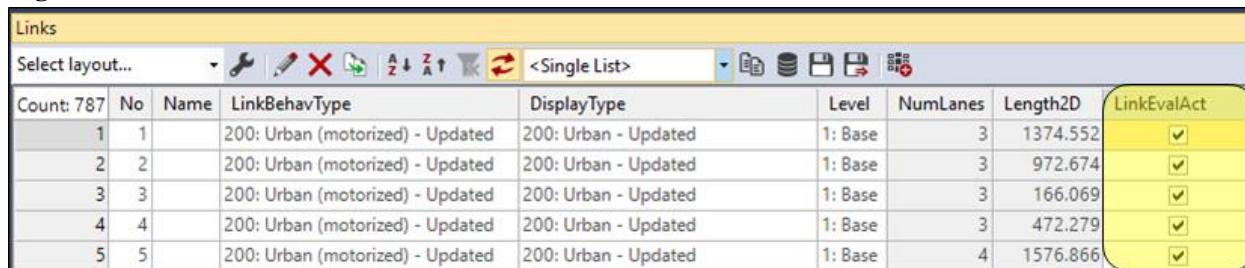


4.1.6 Link Evaluation

Evaluation data can also be collected for each individual link and connector; examples of these data elements include throughput, average speed, and density. Link Evaluation can be collected on a per-lane or per-link basis. When setting up Link Evaluation, the modeler must activate this setting in the Link List to capture data for the desired links; this is shown in **Figure 4-9**.

Link evaluation outputs are primarily used to evaluate freeway networks and are the input for the *Vissim Freeway Segment Processing Tool Version 2.2* to create freeway schematic charts with per-lane speed and density measurements. This tool is described in the *TOSAM Appendix F*. To ensure one set of results are reported for the entire link, the link evaluation segment length should be set to a value larger than the link length.

Figure 4-9. Link Evaluation Activation



Links								
Select layout... <Single List>								
Count: 787	No	Name	LinkBehavType	DisplayType	Level	NumLanes	Length2D	LinkEvalAct
1	1	200: Urban (motorized) - Updated	200: Urban - Updated	1: Base	3	1374.552	<input checked="" type="checkbox"/>	
2	2	200: Urban (motorized) - Updated	200: Urban - Updated	1: Base	3	972.674	<input checked="" type="checkbox"/>	
3	3	200: Urban (motorized) - Updated	200: Urban - Updated	1: Base	3	166.069	<input checked="" type="checkbox"/>	
4	4	200: Urban (motorized) - Updated	200: Urban - Updated	1: Base	3	472.279	<input checked="" type="checkbox"/>	
5	5	200: Urban (motorized) - Updated	200: Urban - Updated	1: Base	4	1576.866	<input checked="" type="checkbox"/>	

4.1.7 Vehicle Network Performance

Comprehensive network performance can be evaluated by capturing average delay, number of stops, distance travelled, vehicular throughput, latent demand, and fuel consumption. This feature is typically used to compare different network scenarios at a high level.

4.2 OUTPUT RESULTS

Once model runs have been completed, each of the evaluated parameter results can be displayed in tabular form. A Node Result attribute table is provided as an example in **Figure 4-10**. Vissim model outputs can be displayed individually for each run and a statistical summary of multiple runs can be computed automatically using the “summation” button on the top. Attribute result tables can be configured as desired for efficient data extraction and manipulation using the “wrench” button on the top left corner of the table. These tables can be saved into text files that are stored with a *.att file extension using the “save” button at the top. Automated saving of results during the simulation can be done using the “save with arrow” button at the top. Additional filters and sorting mechanisms are available for results interpretation. Vissim stores model results in *.db files and references back to these database files when generating the attribute tables within the user interface. These database files are saved in a results folder and can be queried outside of Vissim. If different simulation seeds are run on different computers, the database files can be merged into a single Vissim model for results presentation and *.att file generation under “File”/“Read Additionally”.

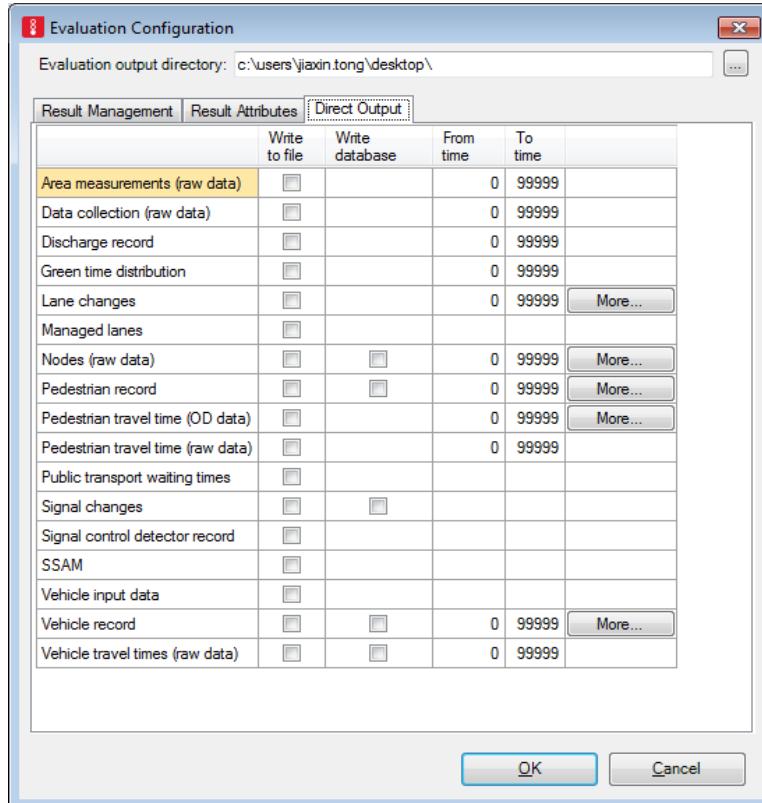
4.2.1 Direct Output (Raw Data)

Vissim offers a wide range of direct outputs in addition to the aggregated results for those MOEs mentioned above. **Figure 4-11** shows the list of direct outputs, many of which are raw data. Vehicle record data can be used to create vehicle trajectory and other information pertaining to individual vehicles in the network. Signal change and detector data are useful in examining and analyzing signal operations in detail.

Figure 4-10. Attribute Result Tables

Count: 774	SimRun	TimeInt	Movement\Node	Movement\Direction	Movement\FromLink	Movement\ToLink	Vehs(All)	VehDelay(All)	QLen	QLenMax
1	1	6300-9900	101	W-SE	5	19	22	7.72	0.08	86.33
2	1	6300-9900	101	W-E	5	20	990	4.12	9.31	209.81
3	1	6300-9900	101	W-N	5	23	8	32.05	9.31	209.81
4	1	6300-9900	101	S-W	18	6	124	57.66	50.41	383.63
5	1	6300-9900	101	S-E	18	20	84	16.64	48.95	387.25
6	1	6300-9900	101	S-N	18	23	0		50.41	383.63
7	1	6300-9900	101	E-W	21	6	2795	7.56	48.89	476.04
8	1	6300-9900	101	E-SE	21	19	17	24.62	48.89	476.04
9	1	6300-9900	101	E-N	21	23	12	8.51	47.61	498.06
10	1	6300-9900	101	N-W	22	6	10	51.78	8.69	102.85
11	1	6300-9900	101	N-SE	22	19	0		8.69	102.85
12	1	6300-9900	101	N-E	22	20	25	47.13	8.69	102.85
13	1	6300-9900	101	Total			4087	8.91	30.56	498.06
14	2	6300-9900	101	W-SE	5	19	32	10.08	0.13	86.86
15	2	6300-9900	101	W-E	5	20	1021	4.69	11.23	256.49
16	2	6300-9900	101	W-N	5	23	18	41.70	11.23	256.49
17	2	6300-9900	101	S-W	18	6	121	50.26	41.34	347.62
18	2	6300-9900	101	S-E	18	20	87	12.89	39.31	351.24
19	2	6300-9900	101	S-N	18	23	0		41.34	347.62
20	2	6300-9900	101	E-W	21	6	2738	6.86	46.79	451.35
21	2	6300-9900	101	E-SE	21	19	34	24.08	46.79	451.35
22	2	6300-9900	101	E-N	21	23	23	9.29	41.81	473.37
23	2	6300-9900	101	N-W	22	6	7	40.93	5.92	42.27
24	2	6300-9900	101	N-SE	22	19	0		5.92	42.27
25	2	6300-9900	101	N-E	22	20	21	40.38	5.92	42.27
26	2	6300-9900	101	Total			4102	8.29	26.65	473.37
27	3	6300-9900	101	W-SE	5	19	37	3.84	0.08	48.61
28	3	6300-9900	101	W-E	5	20	1043	4.11	9.39	202.91
29	3	6300-9900	101	W-N	5	23	11	47.01	9.39	202.91
30	3	6300-9900	101	S-W	18	6	118	57.27	39.98	215.63
31	3	6300-9900	101	S-E	18	20	72	11.72	37.91	219.25

Simulation Runs Node Results Link Segment Results Vehicle Travel Time Results Queue Results

Figure 4-11. Direct Outputs


4.3 VISUALIZATION OF SIMULATION RESULTS

In most cases, modelers export simulation evaluation results and process them in custom tools (i.e., spreadsheets and other data analytic software programs). However, Vissim also offers numerous methods for displaying and interpreting simulation results within the program itself. A high-level overview of these features are provided in the following sections. These features are only available after one or more simulation runs have been completed and are saved within the Vissim file.

4.3.1 Output Plots

Vissim offers a few different methods for visualizing simulation results. The first method is shown in **Figure 4-12**, where the travel time for different travel time segments are shown in a line graph. Plots such as this one can be generated for numerous different output parameters, including but not limited to, Node outputs, Link Segment outputs, Vehicle Travel Time results, and Queue results. These plots can be generated as line graphs or bar charts depending on the intended interpretation and desired illustration of the data. **Figure 4-13** demonstrates the procedure for producing these plots and the user interface for selecting different plot features.

4.3.2 Network Heat Maps

Performance measures can also be illustrated geographically on the modeled roadway network. For example, heat maps representing average speed, density, and volume can be shown on the links within the Vissim graphical user interface. Pre-defined color schemes are available; however custom color classifications can be defined as desired. **Figure 4-14** demonstrates the process for setting up link-based heat maps, while **Figure 4-15**, **Figure 4-16**, and **Figure 4-17** show examples of heat maps for speed, volume, and density, respectively.

Figure 4-12. Output Plots

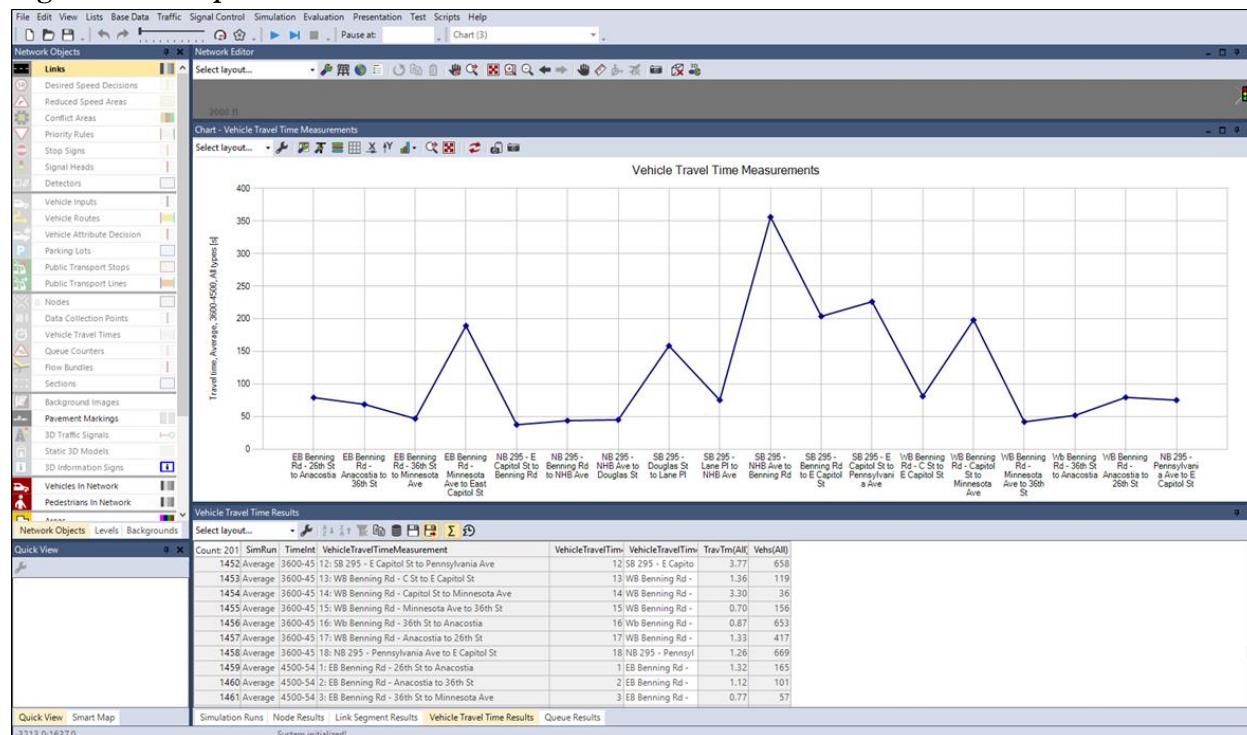


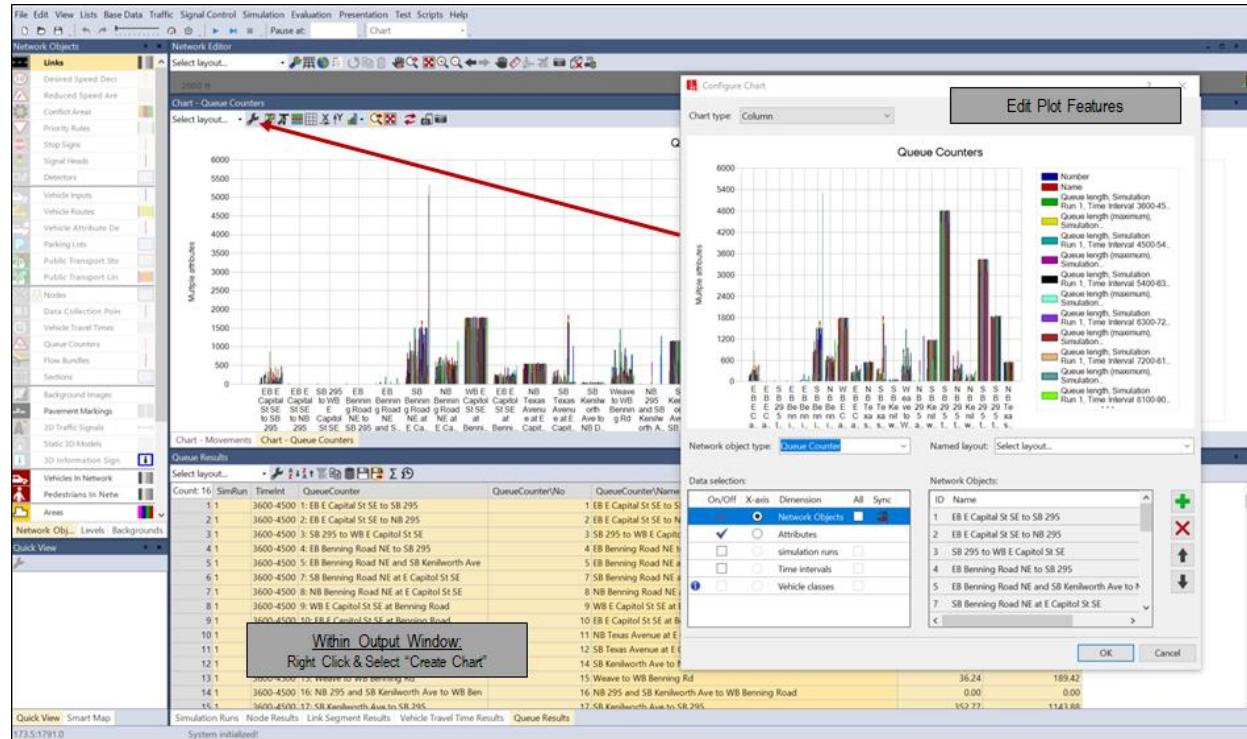
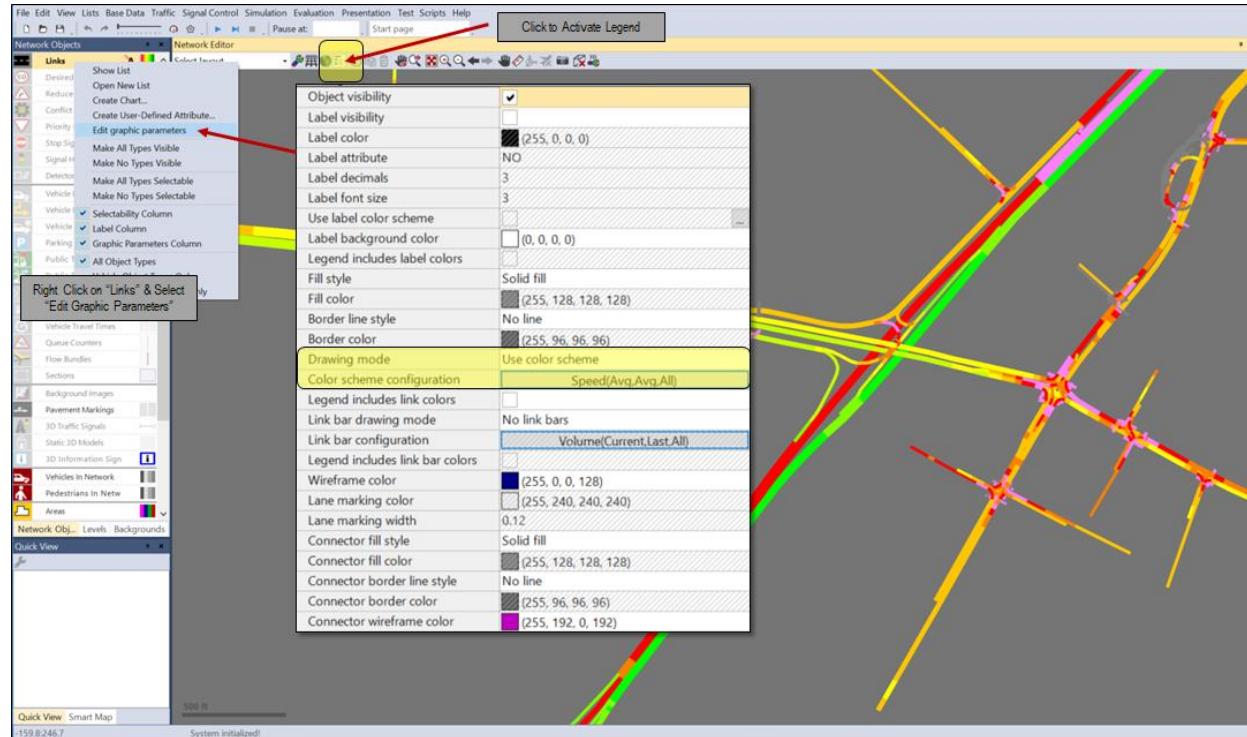
Figure 4-13. Generating Output Plots

Figure 4-14. Generating Link-Based Network Heat Maps


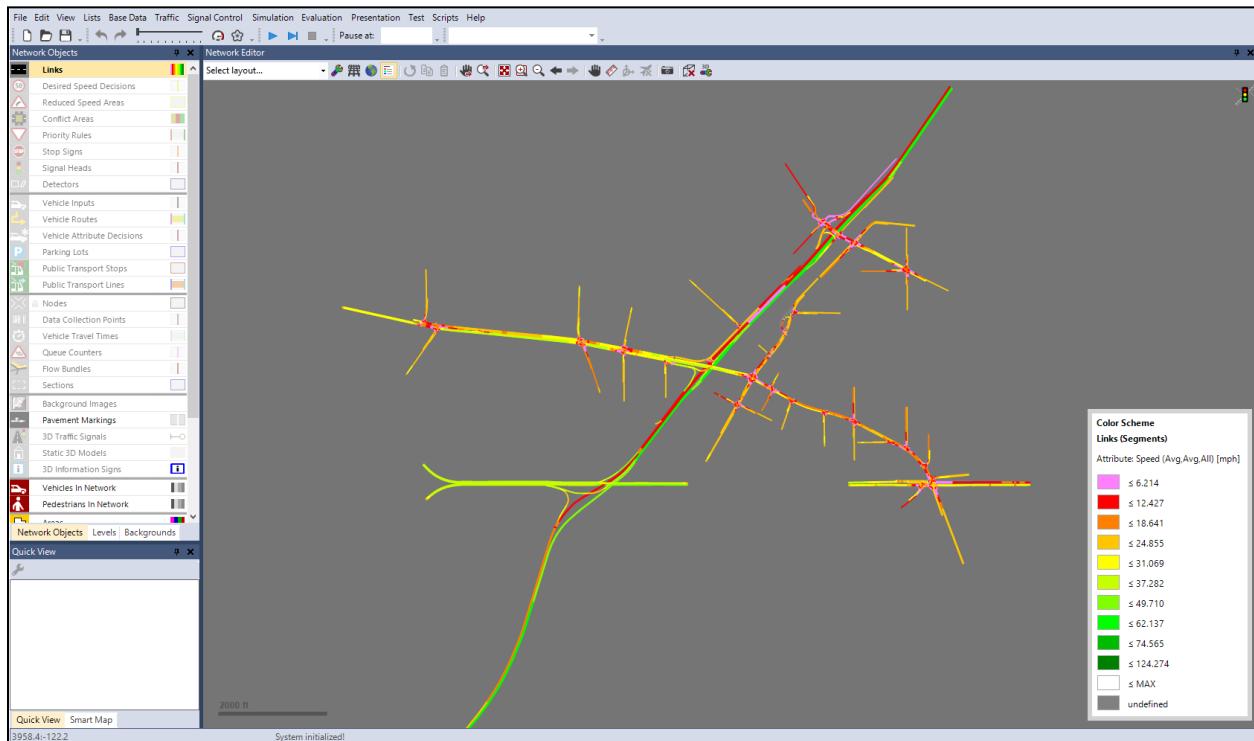
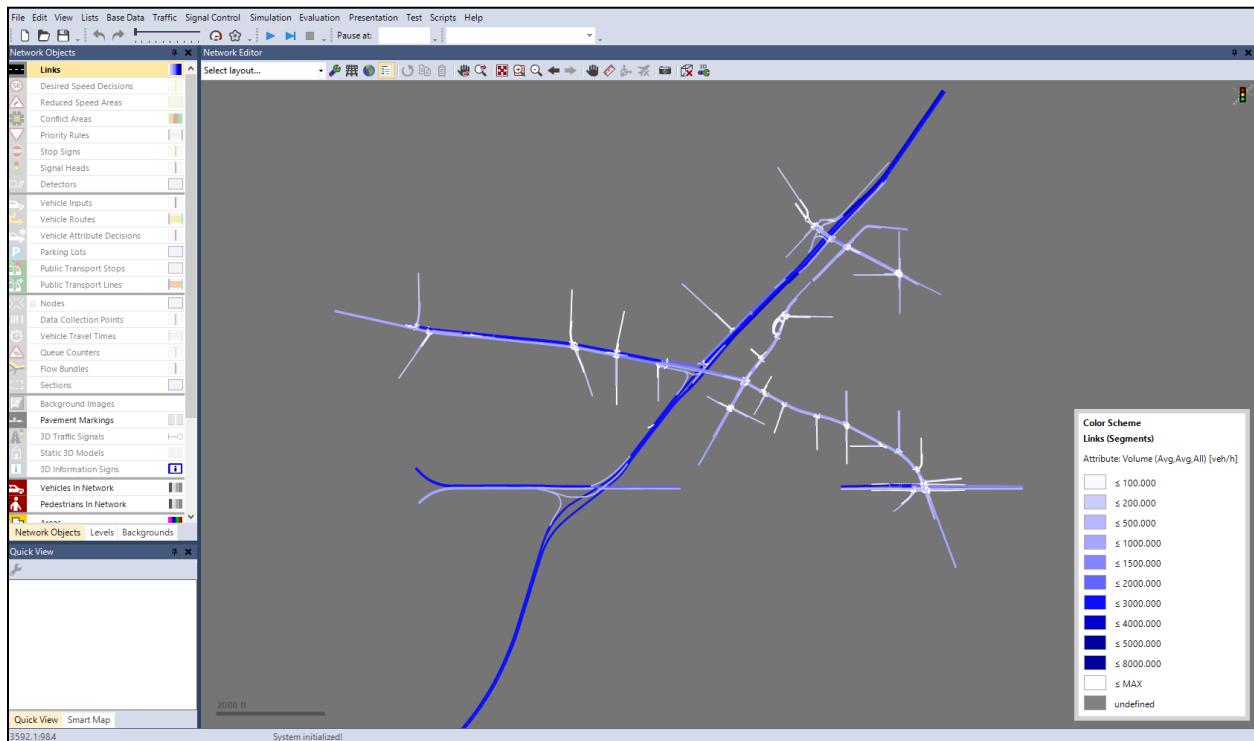
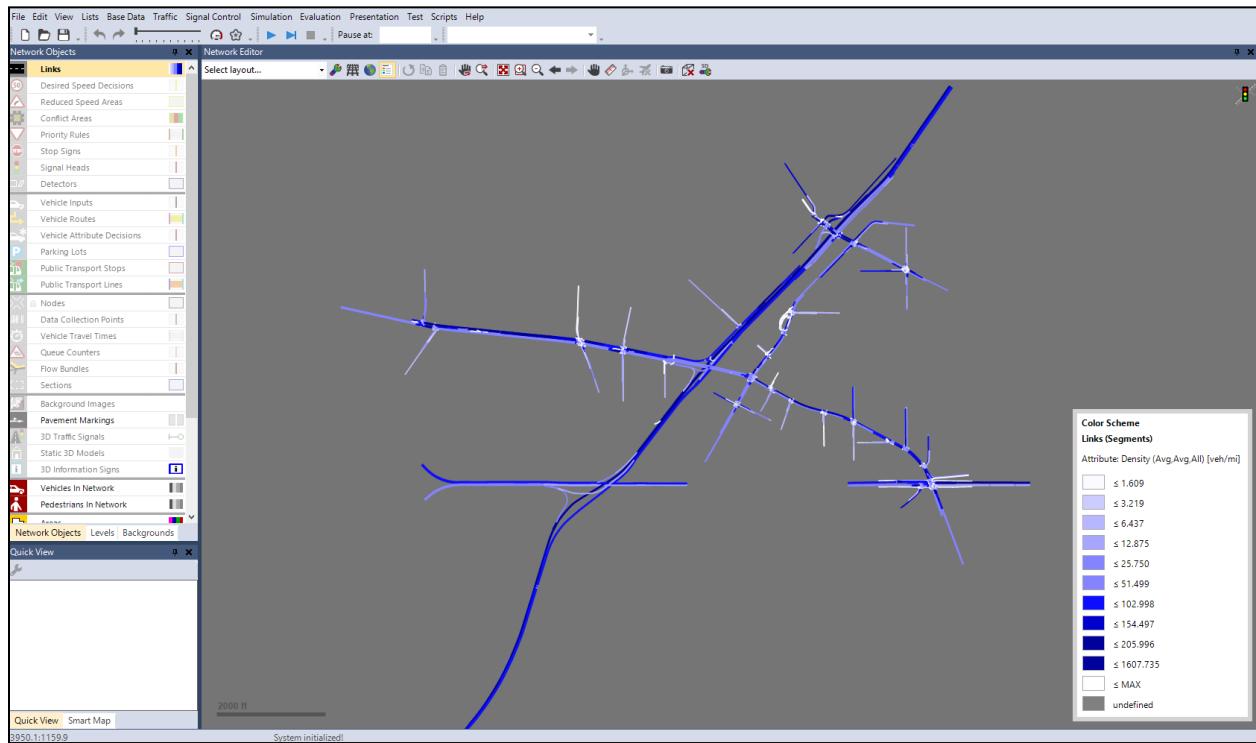
Figure 4-15. Speed Heat Map

Figure 4-16. Volume Heat Map


Figure 4-17. Density Heat Map


4.3.3 Ancillary Performance Visualization

Like the generation of heat maps displaying link performance, performance of other network features can also be represented graphically on the network. For example, node performance in terms of Vissim calculated Level of Service (LOS) can be used to color-code intersection nodes, and queue lengths can be graphically illustrated on their respective approaches; this is demonstrated in **Figure 4-18**. In addition, throughput volumes can be shown at intersections down to the individual turning movement level. Vissim offers a feature that graphically shows these volumes with labels and scaled bars; this is demonstrated in **Figure 4-19**. All these described visualizations are useful for sanity checking results and developing strategies for reporting findings.

4.4 VIDEO PRESENTATION

In many cases, it may be useful to generate an illustration of model performance to validate model performance and communicate findings effectively. Video animations of model performance can be conducted following numerous methods. Traditional visualization within the Vissim user interface during simulation runs shows vehicles traversing the network in two-dimensions. In addition, real-time speeds, densities, throughput volumes, and queues can be displayed graphically and recorded for presentation and demonstration of results. Three-dimensional videos of vehicles traversing the network can also be developed if necessary; however, their use is far less common than the two-dimensional view. The following sections briefly describe each of these presentation methods.

4.4.1 2D Vehicle Animations

When running a Vissim simulation, the default layout view is a 2D depiction of vehicles traversing the network. This viewpoint is critical for model debugging and validation; however, it is not as effective in communicating results and outputs as the other methods. **Figure 4-20** illustrates two screen captures of the sample model running in 2D mode.

Figure 4-18. Node Performance LOS and Queue Length

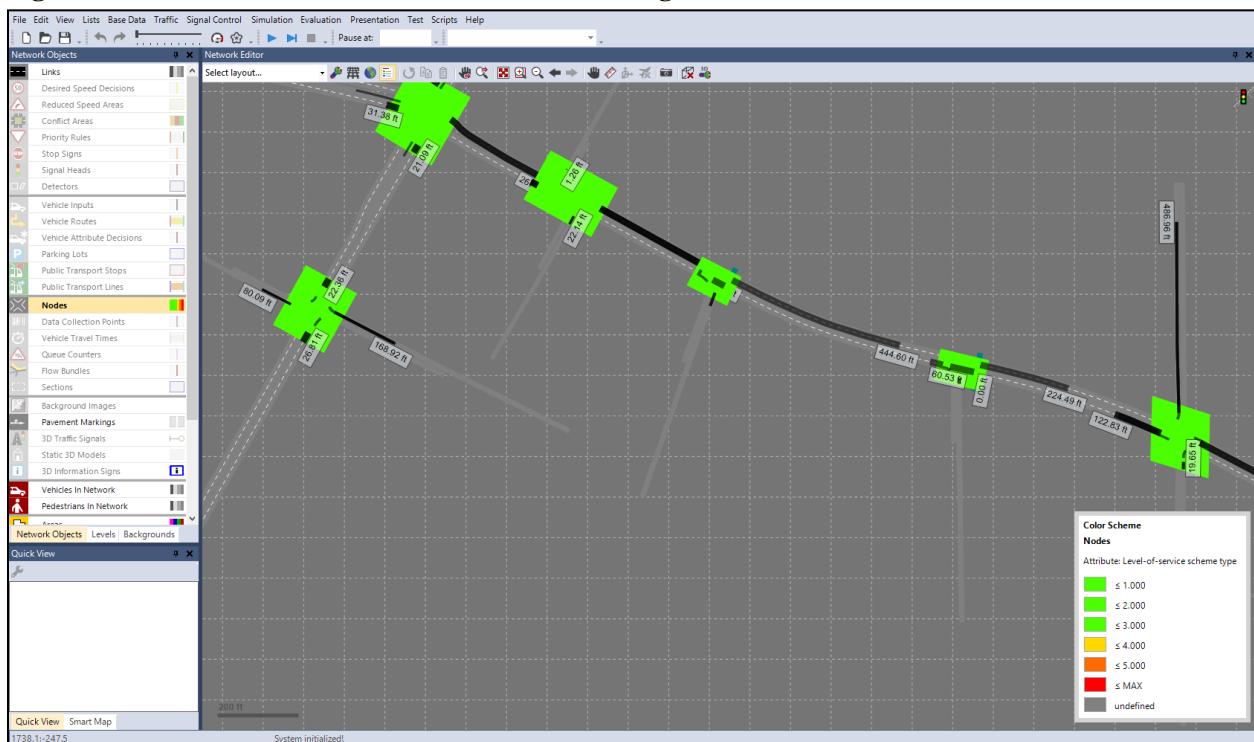


Figure 4-19. Node Performance Turning Movement Volumes

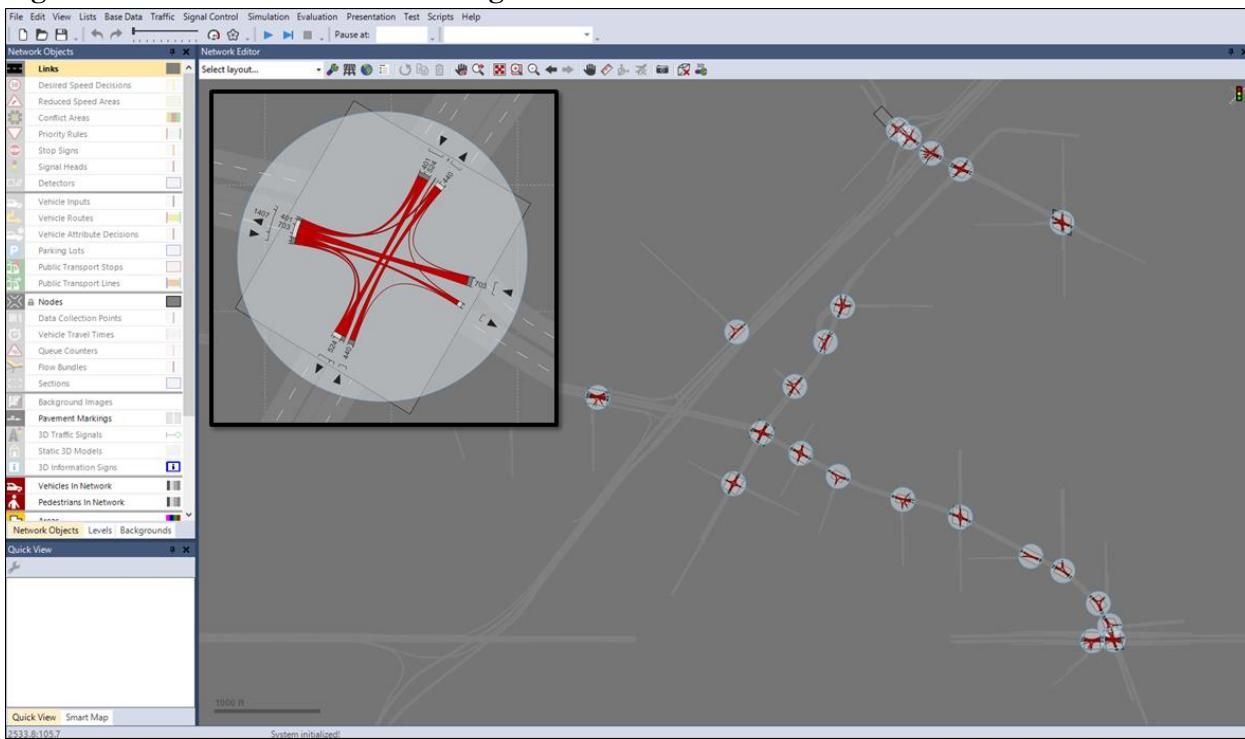
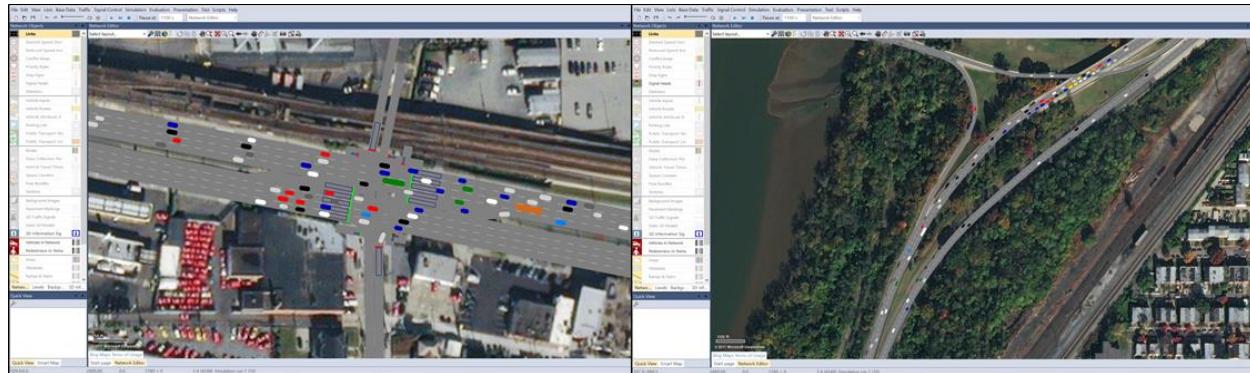


Figure 4-20. Screen Captures of 2D Model Simulation



4.4.2 Real-Time Network Performance Results

Real-time network performance can be graphically displayed during simulation runs using the same methods described in **Sections 4.3.2** and **4.3.3**. As previously described, illustrations of speed, volume, density, queue length, intersection performance, and many other metrics for model performance can be imposed on the model. This feature is particularly useful for illustrating the formation and dissipation of congestion through the peak hour. By recording the variance of these measures of performance over time, traffic operations can be communicated more clearly.

Figure 4-21 demonstrates the average link speeds and queue lengths during the peak hour of an example model. As shown in this example, queues from the intersections are spilling back onto freeway links. This finding could be corroborated by field data, in which the visualization would be useful in communicating the state of the network, or this could be an indication of a model-coding error that requires modeler attention. **Figure 4-22** shows four snapshots of average link speeds and queue lengths throughout the peak hour of an example model. For report documentation, stationary screen captures can be used to generate figures to illustrate model performance, whereas video screen captures could be used to create videos of the changes throughout the analysis period for presentations. **Figure 4-23** also shows four snapshots of average link speeds with the default Bing background map activated to demonstrate these animations and screen captures can be developed with or without background imagery based on project needs.

4.4.3 3D Animation

Three-dimensional animation provides a more vivid presentation of traffic movements and roadway alternative concepts, which are particularly effective for presenting to non-technical audiences. Modelers can produce movie-style animations in 3D mode by recording the simulation under defined camera positions and sequences developed in a storyboard. Fine tuning to the video can be made in external video-editing software programs. A 3D animation can be developed using a standard aerial background or can incorporate other 3D elements from the Google Sketchup online warehouse or custom made in Google Sketchup. While these video features can be effective communication tools, their development is very resource-intensive and should only be used when other methods for communicating model outputs and design concepts are ineffective. Two examples of screen captures for 3D model simulation are shown in **Figure 4-24**.

Figure 4-21. Screenshot of Real-Time Link Speed and Queue Length

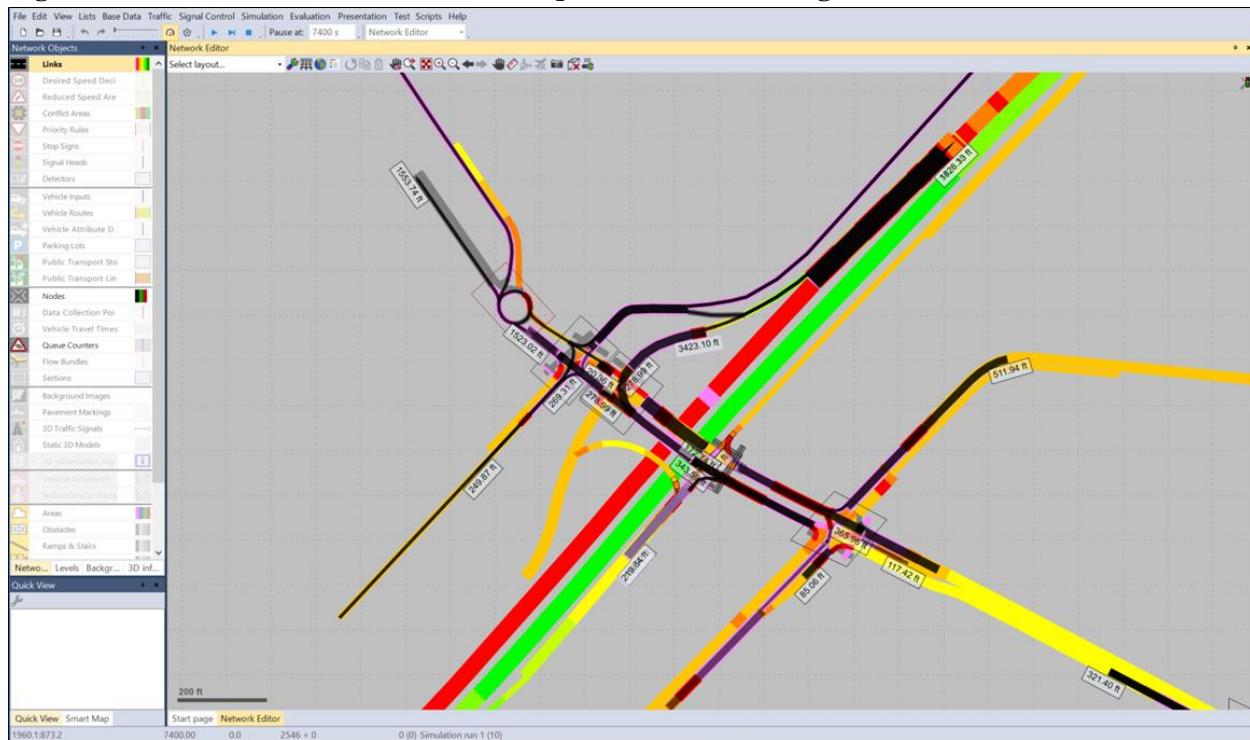


Figure 4-22. Storyboard Screenshots of Vehicle Speed and Queueing During Simulation Period

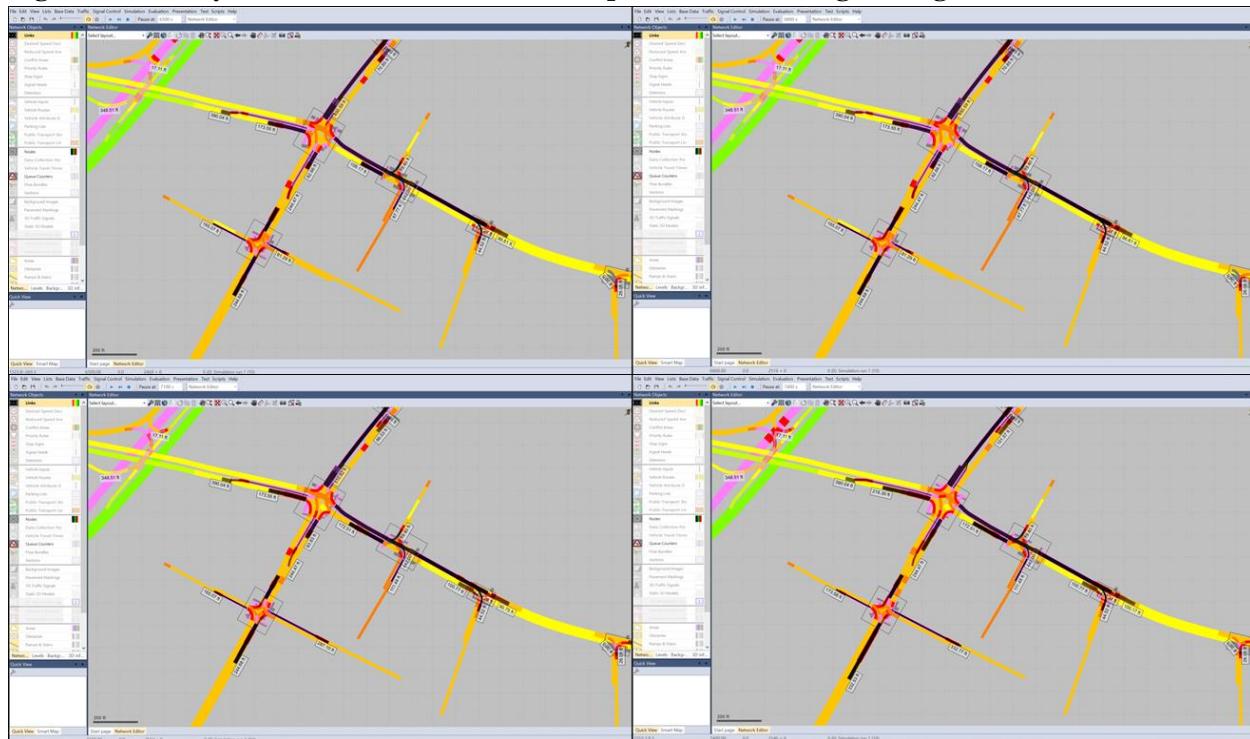


Figure 4-23. Storyboard Screenshots of Vehicle Speed During Simulation Period

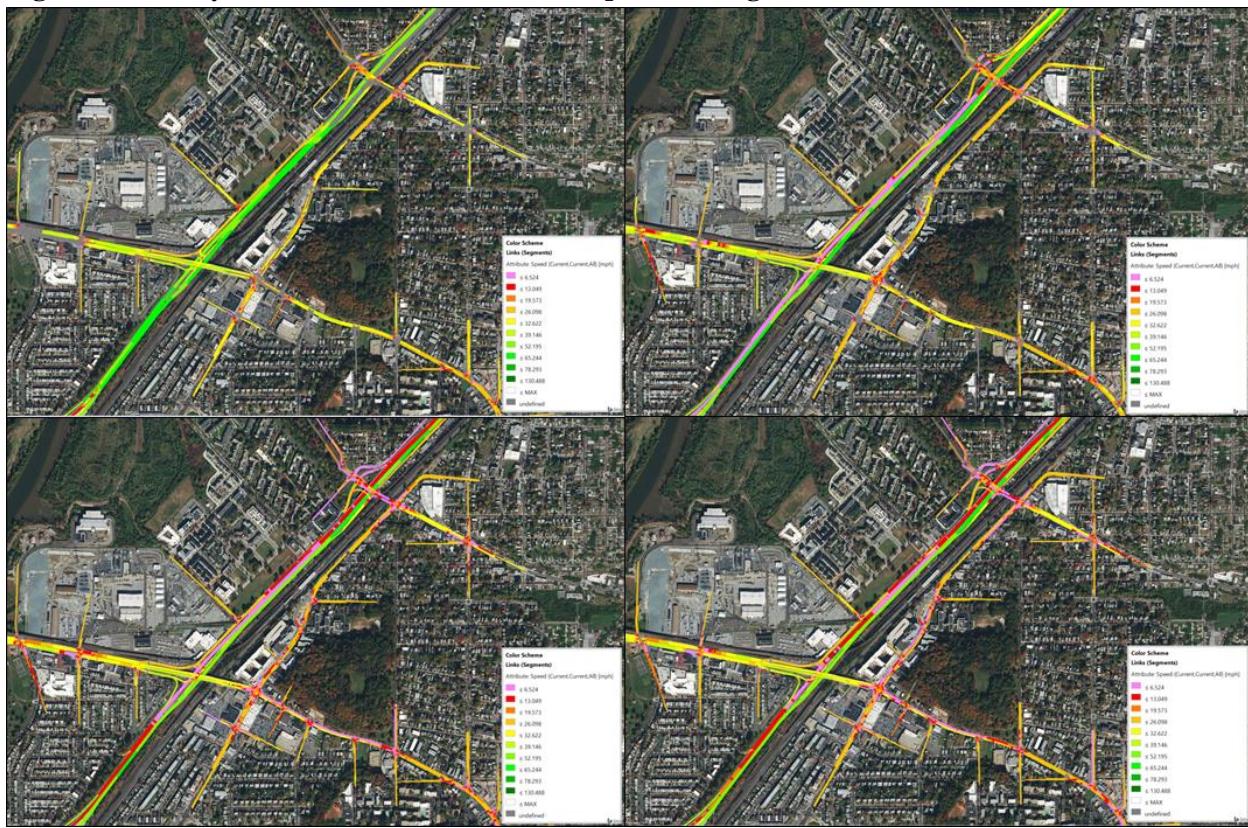
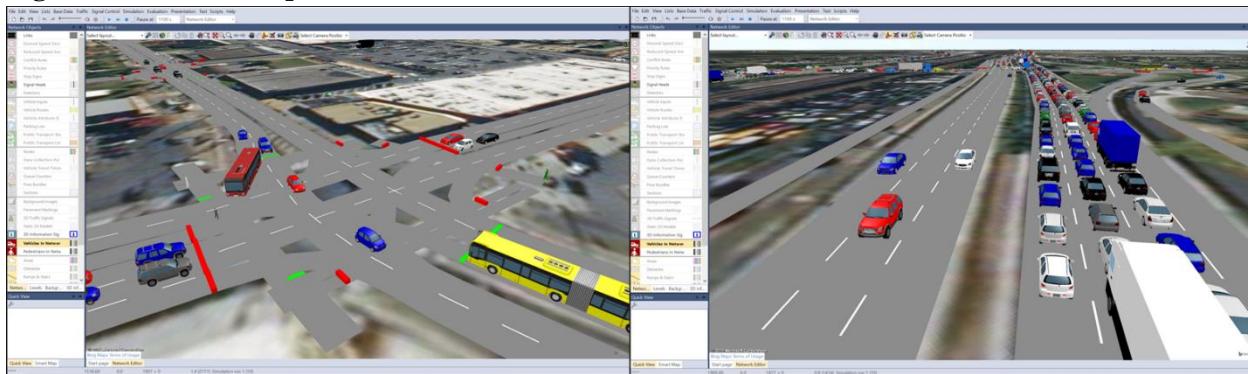


Figure 4-24. Screen Captures of 3D Model Simulation



5 Calibration Guidance

This section is intended to provide guidance and structure to the Vissim calibration procedure.

5.1 BACKGROUND INFORMATION

Calibration can be defined as an optimization problem in which model parameters are iteratively adjusted to align model output data with observed field data. The state-of-the-practice for microsimulation model calibration provides limited guidance on preferred approaches, or “styles”, toward achieving a calibrated model. Available Vissim-specific calibration guidance primarily consists of acceptable value ranges for commonly used parameters, as noted in documents published by several state DOTs, including the Wisconsin Department of Transportation (WisDOT), Florida Department of Transportation (FDOT), and Oregon Department of Transportation (ODOT). Experience indicates that there is no single set of parameters that will result in a calibrated model; rather, by adjusting different parameters in various sequences to opposing limits, similar model outputs can be produced. Thus, while working within the recommended parameter boundaries, modelers have developed different ways for adjusting model parameters to satisfy applicable calibration thresholds (i.e., traffic volumes, travel speeds, travel times, and queues).

Findings from the *VDOT Vissim 11 Evaluation Report* suggest that there may exist an optimal methodology for correctly calibrating driving behavior models within Vissim. In other words, one “calibration style” may be preferred over others and may allow for the development of more robust models. The following sections describe this preferred “calibration style” and provide guidance for calibrating Vissim models to existing conditions.

5.2 DATA COLLECTION FOR CALIBRATION

Calibration of a Vissim model is only as accurate as the field data that is used to calibrate the model against. Therefore, the following recommendations are provided for setting up the data collection plan.

Identify limits of the study area: Clearly define the study area, including listing all intersections, ramps, and mainline segments that will be included. Remember to think about potential future improvements that may result in an expansion of the study area in existing conditions. Including these areas will reduce duplicate efforts later in the process.

Traffic data counts: Continuous 48- or 72-hour freeways and ramp counts and peak period turning movement counts at intersections should be conducted. These counts should be collected in the desired model conditions; for example, typical week day during the school year.

Travel time runs: Travel time data should be collected during the time period count data are being collected. Travel time data can be collected using field runs, probe data, or a combination of both. A sufficient quantity of data should be collected and validated against historic or typical corridor travel times.

Speed data: Numerous sources of speed data are available. Depending on the equipment used to collect traffic counts and travel times, speed data may be available by those vendors. In addition, probe vehicle data can be purchased to provide average speeds for the study area.

Queue lengths: Queue length data can be collected manually in the field or using video footage. Specific details related to queue length data collection can be found in Chapter 6 of the *TOSAM*.

Average standstill distance: Average gaps between vehicles stopped at intersections can be collected to assist in calibration. This distance is directly applied in the Wiedemann 1974 car-following model used on arterial links; therefore, when these field data are collected, driving behavior can be more realistically replicated.

Field verification/visits: It is critical to get an in-person understanding of the corridor. Analysts working on model development and calibration should see the study area under peak conditions, preferably while recording video. Field visits also may be used to verify provided signal timings and/or inspect queues and other operational characteristics.

Table 5-1 summarizes the various calibration measures and suggestions for potential data sources. For guidance on specific targets and criteria, please see *TOSAM Section 5.3*.

Table 5-1. Various Calibration Measures and Potential Data Sources

Calibration Measure	Segments Used for Calibration	Potential Field Data Sources for Calibration	Potential Vissim Output Files
Volume (Throughput)	Freeway/Ramps	Machine (tube) counts or counts from Video recordings	Link evaluation Data collection points
	Arterials (by approach or movement)	Manual or Video collection of turning movement counts	Node evaluation Data collection points
Travel Times	Freeways/Arterials	Field Travel Time Runs or Probe Vehicle Data	Travel time segments
Speed/Congestion	Freeways/Arterials	Spot-Speed Data collection or Probe Vehicle Data	Data collection points Travel time segments
Bottleneck Locations	Freeways/Arterials	Field photographs/videos/notes or Peak period aerial imagery (if available)	Visual inspection of model
Queue Lengths	Freeways/Arterials/ Ramps	Field photographs/videos/notes or Peak period aerial imagery (if available)	Node evaluation Queue counters Visual inspection of model

5.2.1 Importance of Reliable Data

The importance of data quality in model calibration cannot be overstated. Before starting the calibration process, it is important to thoroughly quality check all data being used for model calibration. Not only should these data sources be vetted for erroneous inputs, but they should also be evaluated for consistency among one another. For example, if INRIX data is being used for speed heat map comparison of congestion intensity and duration and field-collected travel time runs are being used for travel time calibration, these two data sources should be compared to check if they represent the same conditions.

5.3 CALIBRATION PARAMETERS

The following parameters are commonly adjusted during Vissim model calibration. While additional parameters in Vissim can be adjusted, their impacts have not been studied as thoroughly and their adjustment should only be undertaken when justified.

5.3.1 Lane Change Parameters

Lane Change Distance (LCD) refers to the point at which a vehicle in the model will start attempting to make a lane change prior to a decision point (i.e., a turning movement at an intersection or exiting freeway). The default setting for this distance is 656.2 feet. In many cases, especially in congested arterial and most freeway networks, this distance is not realistic; causing vehicles to change lanes too late on freeways and arterials. The Emergency Stop Distance (ESD) can also be updated to more realistically match the location vehicles ultimately stop to wait for a lane change. Accurate calibration of this parameter requires field observations to determine lane change behavior along the network. It is important, however, to evaluate the network-wide impact of large LCDs, as overlapping lane change maneuvers can have adverse effects on upstream traffic. Moreover, these findings clearly show that modifications to the LCD greatly impacts network performance—within and outside the influence area—and should be considered in the initial steps of the calibration process.

5.3.2 Speed Distributions

During coding, speed decisions are entered in the model according to posted speeds. Reduced Speed Areas are added to links with significant curvature to temporarily reduce the speed of a vehicle without changing their desired speed assigned by the speed decisions. Speed decisions and reduced speed areas are assigned to defined speed distributions.

Speed distributions representing posted speed limits should be established such that 85 percent of vehicles would travel at or above the posted speed limit, and the maximum speed should be capped to 10 mph above the posted speed limit. During calibration, it may be necessary to modify speed distributions, increasing the percentage the 85 percent or the 10 mph limit described above. This is consistent with the generally accepted concept that drivers will drive over the speed limit in free-flow conditions. Baseline speed distributions for intersection turning movements are provided below (per *TOSAM Guidelines*):

- **Right Turn:** Linear distribution [7.5 mph, 15.5 mph]
- **Left Turn:** Linear Distribution [12.4 mph, 18.6 mph]

5.3.3 Driving Behavior

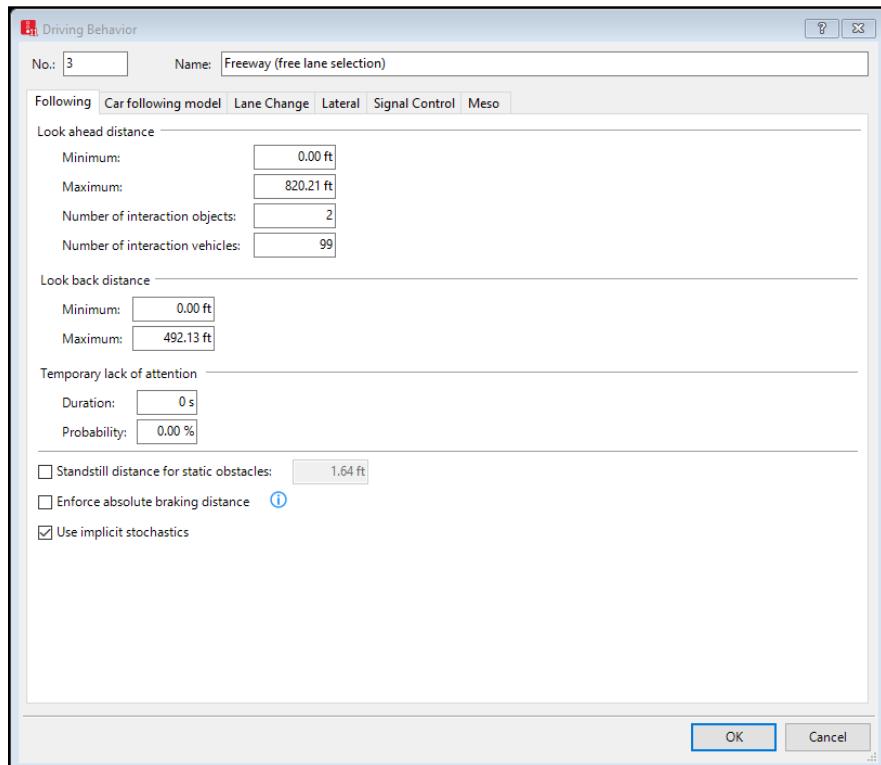
Vissim software represents vehicular interactions using the Wiedemann 1974 and 1999 car-following models alongside additional proprietary lane changing models. These models are controlled by a set of user-defined parameters located in a driving behavior container, shown in **Figure 5-1**. Driving behavior containers are assigned to each link in a network to dictate how vehicles traverse the link and interact with one another. Vissim 11 provides the following default driving behavior containers.

- **Urban (motorized):** based on the Wiedemann 1974 car-following model, which PTV recommends for arterials. This container is commonly the default for urban streets and arterials.
- **Right-side rule (motorized):** based on the Wiedemann 1999 car-following model, which PTV recommends for freeways. This container enforces the “slow lane rule” where vehicles only use the left lane to pass. This is a behavior common on the German autobahn and throughout Europe, but not enforced in the United States.
- **Freeway (free lane selection):** based on the Wiedemann 1999 car-following model, which PTV recommends for freeways. This container is commonly the default for freeways.
- **Footpath (no interaction):** does not monitor interaction between entities modeled on these links. This container is used to control pedestrian movements on pedestrian walkways and crosswalks.
- **Cycle-Track (free overtaking):** based on the Wiedemann 1999 car-following model, with updated parameters to more realistically mimic bicycle behavior. This container is used to control bicycle movements on dedicated cycle tracks.

- **AV_cautious (CoEXIST):** based on the Wiedemann 1999 car-following model, with updated parameters to more represent cautious behavior of an Automated Vehicle as determined by the CoEXIST research project.
- **AV_normal (CoEXIST):** based on the Wiedemann 1999 car-following model, with updated parameters to more represent normal behavior of an Automated Vehicle as determined by the CoEXIST research project.
- **AV_allknowing (CoEXIST):** based on the Wiedemann 1999 car-following model, with updated parameters to more represent all knowing (i.e., fully connected) behavior of an Automated Vehicle as determined by the CoEXIST research project.

These default driving behavior containers are provided as a starting point; however, are not intended to be used directly without calibration with field data. The parameters for the Urban (motorized) and the Freeway (free lane selection) containers, for example, were calibrated from observed driving behaviors in Germany. Therefore, their direct application for modeled roadways in Virginia are infeasible. Rather, parameter calibration is used to produce an output that matches field data. This practice is supported in the 2019 *FHWA Traffic Analysis Toolbox Volume III: Guidelines for Applying Traffic Microsimulation Modeling Software*. In Chapter five of this guidance document, the use of default model parameters is discussed: “For the convenience of the analyst, the software developers provide suggested default values for the model parameters. These default parameters do not represent a calibrated model. The analyst should always perform model calibration and review the calibration criteria to ensure that the model accurately reproduces system performance by travel condition.”

Figure 5-1 Driving Behavior Container



5.3.3.1 Car-Following Model Parameters

The definition and impact of the most common Wiedemann 1974 (W74) and Wiedemann 1999 (W99) parameters adjusted in calibration are summarized in this section.

W99 – CC0, Standstill Distance: The CC0 parameter represents the minimum allowable gap between vehicles at a complete stop. This parameter is used in the Wiedemann 1999 car-following model for the calculation of the desired following distance. It serves as the bare minimum following distance, and remains a baseline as the following distance increases with speed.

W99 - CC1, Time Headway: The CC1 parameter is used in the Wiedemann 1999 car-following model as an input that reflects the average time headway maintained by vehicles and is used to calculate the desired following distance of each vehicle from the leading vehicle. The CC1 parameter should remain consistent throughout the entire freeway network to avoid differences in the values between consecutive links. This is recommended to prevent the creation of unrealistic and artificially imposed shockwaves caused by changes to this parameter in consecutive links. Thus, changes to the car-following model between different types of freeway segments should be kept to the CC2 parameter. The CC0 and CC1 parameters should be calibrated to match average headways observed or expected corridor-wide.

W99 - CC2, Following Variation: The CC2 parameter represents “following variation” in the Wiedemann 1999 car-following model. Rather than directly contributing to the computation of the desired following distance—as the CC0 and CC1 parameters do—this parameter is used to set the boundary or range of acceptable following distances a vehicle can maintain before corrective action (i.e., acceleration or deceleration) is taken. Essentially, the CC0 and CC1 parameters define the “average desired following distance”, and the CC2 parameter defines the amount of allowable oscillation around that mean. A larger CC2 value provides a wider range in allowable following distance, while a smaller CC2 value gives a narrower range. The CC2 parameter can be calibrated to match localized conditions, as well as to differentiate behavior along different types of roadway segments. While CC0 and CC1 values should stay consistent among continuous freeway segments, CC2 values can be updated to more realistically model differences in observed behavior. In general, the value for CC2 along freeway basic segments should be less than the CC2 value at weave, diverge, and merge segments. CC2 can also be used to represent behavior at special freeway segments, such as work zones or narrow lanes.

W74 - AX, Standstill Distance: Standstill distance, or AX in the arterial model, refers to the desired distance between two stopped vehicles, measured from the front bumper of one vehicle to the rear bumper of the next vehicle. Research suggests that the AX parameter could be adjusted during calibration when throughput thresholds are met, but delay and queueing are not meeting calibration thresholds. Field observations should be used to assign the AX value, when available.

W74 - BX, Safety Distance: The Wiedemann 1974 model calculates BX, or the safety distance, in the underlying model logic to calculate the average following distance a vehicle maintains while in motion. This parameter is calculated with two non-dimensional parameters: BX ADD and BX MULT. Due to the strong correlation in these parameters, parallel changes to both parameter values were considered. Overall findings suggest that a lower BX parameter increases delay and queueing along arterials approaching an intersection, while throughput remains relatively unaffected.

5.3.3.2 Lane Change Model Parameters

The definition and impact of the most common Wiedemann 1974 (W74) and Wiedemann 1999 (W99) parameters adjusted in calibration are summarized in this section.

Advanced Merge: This parameter is shown to improve the fluidity of traffic flow through heavy weave segments. If this option is selected, more vehicles can change lanes earlier, increasing the operational capacity of the roadway and reduce the probability that vehicles will come to a stop waiting for a gap. These benefits are identified in the actual influence area as well as upstream segments. Advanced Merge is shown to have a greater influence on speed, density, and travel time, while limited impact on throughput.

Safety Distance Reduction Factor (SDRF): The SDRF is applied to the lane-change algorithm as a multiplier that reduces the minimum safety distance required between the trailing and proceeding vehicles to initiate a lane change. A reduction in the SDRF prevents “gridlock” along the network and by contrast the relatively large reduction in speed and throughput can be observed at higher SDRF values. However, the MOE differences observed for each parameter value is dependent on the operational conditions within the influence area. A reduction in SDRF can also exhibit a reduction in speed at the influence area if more lane changes ensue, yet a speed benefit is often observed in upstream links. SDRF is shown to have a greater influence on speed, density, and travel time, while limited impact on throughput.

Cooperative Lane Change: The cooperative lane change (CLC) parameter is a binary parameter that is deactivated by default in Vissim. By activating this parameter, vehicles are conditioned to identify opportunities to assist other vehicles in making lane change maneuvers. This feature allows a vehicle traveling on the freeway to make a non-necessary lane change to create space for a merging vehicle to enter the mainline. Activation of CLC has greater influence on link speeds and densities compared to vehicular throughput. Activation of CLC in heavy weave and merge areas is beneficial to freeway operations and can be used to prevent unrealistic wait-times as a vehicle merges into the network. The impact of activating CLC at a weave, merge, or diverge segments can be seen further upstream as the increased number of lane changes produces upstream traffic flow turbulence.

Maximum Deceleration of Own Vehicle (MDOV) and Maximum Deceleration of Trailing Vehicle (MDTV): The MDOV represents the maximum deceleration rate used to change lanes based on the specified routes for own vehicle overtaking. The MDTV represents the perceived maximum rate of deceleration for a trailing vehicle in the adjacent lane where a lane change is desired. A higher maximum deceleration for a trailing vehicle will allow more opportunities for lane changes. Intuitively, this can be interpreted as a driver being more comfortable making a lane change when the new trailing vehicle is perceived to have high deceleration capabilities. Traffic flow is smoother with lower levels of deceleration; shockwaves are formed when dramatic changes in speed propagate upstream through the network. At higher levels of deceleration in the influence area, increases in speed are observed within the influence area; however, this turbulent behavior causes negative side effects for upstream segments that experience speed reductions. Due to the combined use of the deceleration of a vehicle and its perception of the deceleration of other vehicles in the network in car-following and lane change algorithms, calibration of the various deceleration parameters should not be considered together as a set of parameters during calibration.

5.3.4 Conflict Areas and Priority Rules

The Safety Distance Factor (SDF) for conflict areas in Vissim applies to merging conflicts; it is a factor multiplied by the normal desired safety distance of a vehicle to determine the minimum safe distance a yielding vehicle must maintain within a conflict area. Thus, the smaller the SDF, the more aggressive the merge behavior, whereas the larger the SDF, the more conservative the merge behavior. Calibration of the SDF at merging conflict areas—such as right-turn-on-red or channelized right turning movements—should be done on a case-by-case basis at locations that are not matching observed field conditions. The sensitivity analysis results show this adjustment is not likely to have a great impact on throughput; however, could be used to impact delay and queueing. The sensitivity of this parameter is dependent on network conditions, specifically congestion levels and the ratio of volume between major and minor movements at the conflict area.

5.4 CALIBRATION PLAN

At the beginning of a project, a calibration plan should be developed and agreed upon by all parties before substantial Vissim coding begins. This calibration plan should contain all critical assumptions to the modeling task, calibration criteria, data used for calibration, and any other project-specific details that could influence the validity of the calibrated model. A list of assumptions to include in the calibration plan are provided below. This calibration plan should be submitted to the VDOT project manager alongside the analysis scenarios that will be built from the calibrated existing conditions models.

- **Study Area Boundary:** provide a map showing the extents of the study area, and a list of all intersections and interchanges that will be included in the simulation model.
- **Description of Traffic Conditions:** provide a description of typical traffic conditions through the study corridor, as well as a comparison of the typical conditions with the conditions prevalent on the day of data collection.
- **Project-Specific Assumptions:** provide a list of relevant project characteristics that will influence the Vissim model development and calibration. These could include upstream bottlenecks located outside the study area and unique construction or road work conditions.
- **Simulation Time Periods:** provide a list of the peak hour(s), peak period(s), analysis period(s), and seeding time(s) that will be used for model simulation.
- **Calibration Protocol:** provide a summary of the procedure that will be used to calibrate the model. This protocol should be developed from **Sections 5.5 and 5.6** of this user guide.
- **Calibration Criteria:** provide a list or table of the measures of performance that will be evaluated throughout the iterative calibration process. These calibration criteria should be taken from the *TOSAM Chapter 5*. In addition, a specified list of locations where bottlenecks and queues will be evaluated should be provided.
- **Sample Size Determination:** provide a list of the locations and performance measures that will be evaluated in the *VDOT Sample Size Evaluation Tool version 2.0* to determine the required number of simulation model runs. For example, four locations of throughput, four locations of speed, and four locations of travel times.

5.5 PRE-CALIBRATION CONSIDERATIONS

5.5.1 Verify Model is Ready for Calibration

Before starting the calibration process, ensure your model is adequately reviewed and debugged. Prior to adjusting model parameters, it is important to ensure a formal quality check of the basic functionality of the model has been completed. Errors in model coding can cause significant setbacks to calibration efforts. A detailed discussion of the model debugging process is provided in **Section 3**, and a checklist for model debugging is provided in **Appendix A**.

5.5.2 Driving Behavior Containers

In preparation for calibration, pre-defined driving behavior containers (i.e., sets of model parameters) and corresponding Display Types to represent different types of roadway segments should be coded. For freeways, this includes (1) basic segments and (2) merge, diverge, and weaving segments. For arterials, this includes (1) basic arterial segments and (2) oversaturated arterial segments. Following guidance from PTV, freeway segment containers should be set up to follow the Wiedemann 1999 car-following model with the “Freeway (free lane selection)” parameter values as default, while the arterial segment containers should be set up to follow the Wiedemann 1974 car-following model with the “Urban (motorized)” parameter values as default.

Once these containers have been created with the default driving behaviors, every link and connector in the network should be assigned to their appropriate driving behavior container and Display Type; note, the Display Type is a feature that allows the modeler to change the appearance of a link to clearly indicate the driving behavior container it is associated with. It is recommended that a specific display type be associated with each driving behavior container to improve model coding visualization and expedite model debugging. Freeway links should be split and grouped appropriately based on Highway Capacity Manual definitions for basic, merge, diverge, and weaving segments. From those definitions, freeway links should be assigned to either “basic segments” or “merge, diverge, weaving segments” behaviors. Whereas, all arterial links should be characterized with the “basic arterial segments” behavior to begin.

Additional driving behavior containers and display types may be added during calibration as deemed necessary by the modeler. Please reference the Calibration Guidance and Calibration Steps sections for details.

5.5.3 Lane Change Distances

Lane change distances should be modified based on roadway signage and field observations as research shows this parameter has a large influence on model performance. This parameter should be initialized prior to calibration with a realistic starting value as discussed in **Section 2.2.1**. While it is less common to modify this parameter on arterials, lane change distance for right and left turn connectors at arterial intersections may also require changes in dense urban networks. This need can be identified during debugging and model review and increased values can be applied to specific intersection approaches that are processing traffic flow successfully. In either case, the lane change distance should be initialized prior to calibration and refined during the calibration process.

5.5.4 Terminal Conditions

In some situations, terminal conditions may be required at the edge of the network footprint to account for downstream congestion or traffic patterns that influence the study area. These areas should be identified in project scoping and supported by collected field data. Typically, terminal conditions are modeled with assigned based on speed data observed in the field at those locations. In other situations, the inclusion of downstream lane change maneuvers can be represented by Partial Routes and dummy off-ramps. Regardless of the method used to represent terminal conditions, their impact on the network should be supported by field data.

5.5.5 Lane-By-Lane Traffic Conditions

The calibration criteria outlined in *TOSAM* focus on MOEs at a road link/segment or intersection approach/movement level. Traffic conditions could also vary per lane, particularly on a multi-lane limited access facility. One example is that a three-lane freeway facility sees queue spillback from the off-ramp to the right-most lane(s) of the freeway mainline due to congestion at the ramp junction with an arterial. The queue impact causes a stop-and-go condition on the right-most lane and affects the speeds on the adjacent through lane on the freeway mainline, but the left-most lane may still operate at free-flow conditions. Such conditions should be noted during the field review and addressed in the calibration process both qualitatively and quantitatively. The qualitative part of the calibration would focus on the cause and formation of the bottleneck (e.g., queue spillback) while the quantitative part of the calibration addresses the speeds on the mainline. It may be necessary to collect additional per-lane speed or travel time data if deemed necessary by the VDOT project manager. The calibration parameters and techniques should be applied to accurately model the formation and dissipation (e.g., duration) of the queues or slow-down on the specific lanes of the freeway mainline as well as the intensity (e.g., speed) of the congestions.

5.6 CALIBRATION CONSIDERATIONS

Model calibration will always be an iterative, trial-and-error procedure with numerous uncertainties; however, this user guide introduces new guidelines and steps that can be used to structure this iterative procedure to improve consistency, efficiency, and model validity.

5.6.1 Calibration Checkpoints

At every step of the calibration process, it is important to evaluate both the model outputs (i.e., adopted Measures of Effectiveness [MOEs] for calibration and reporting) and watch the simulation to verify realistic behaviors are being represented in the model. This process should include the use of the visualization tools described in **Sections 4.3** and **4.4**, and a comparison of the aggregated calibration results against the calibration thresholds defined in the Calibration Plan and provided in the *TOSAM Chapter 5*. The model is considered

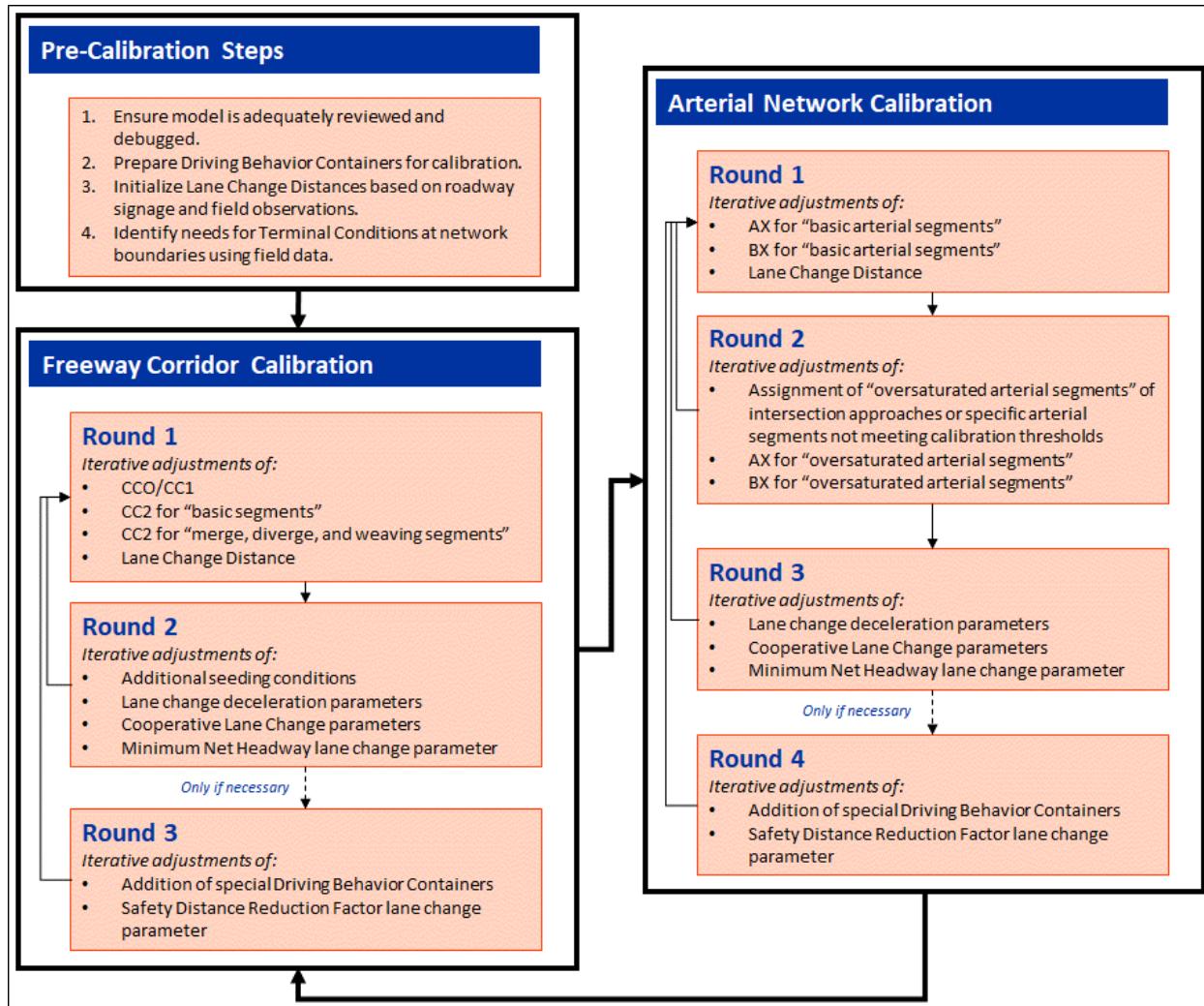
calibrated when all calibration thresholds are exceeded and visual inspection of the model during the analysis period depicts real-world conditions.

The number of required simulation runs should be determined from the *VDOT Sample Size Evaluation Tool version 2.0*. To use this tool, four simulation runs should be completed, then the results from the previously-agreed upon performance measures (i.e., those listed in the Calibration Plan discussed in **Section 5.4**) should be input into the tool to identify the number of recommended simulation runs. In most cases, ten simulation runs or less will be sufficient. In instances where the tool recommends more than ten simulation runs, considerable debugging and evaluation of model performance is recommended. Additional guidance on the use of the *VDOT Sample Size Evaluation Tool version 2.0* can be found in the **VDOT Macro User Guide**.

5.6.2 Calibration Steps

The calibration process is an optimization procedure that requires an unpredictable number of iterations. The flow chart in **Figure 5-2** illustrates an outline of the recommended procedure for model calibration. This flow chart is intended to provide guidance that can be applied to a wide variety of Vissim modeling projects.

Figure 5-2. Flow Chart of Calibration Steps



In models for Interchange Modification Reports, Interchange Justification Reports, or similar projects where the arterial network is centered around one or more freeway corridors, it is recommended to focus on freeway calibration before arterial calibration. Once freeway calibration thresholds are met, then focus can shift to arterials, which often occurs at an intersection-by-intersection basis. This is important as centralized freeway corridors feed into the surrounding arterial networks impacting arterial performance measures. Performance on arterials can also impact freeways; therefore, in situations when substantial congestion is observed on the arterial network preventing traffic from entering the freeway, or backing up onto the freeway, these specific locations should be addressed in conjunction with freeway calibration.

For this reason, Freeway Corridor Calibration is set as the starting point. In some circumstances, updates to arterial networks will be necessary during the freeway calibration effort (i.e., if congestion along arterials are preventing vehicles from entering the freeway network or causing spill-back onto the freeway ramps and mainline). Thus, in these circumstances, deviations from the recommended steps are necessary.

Within both the Freeway Corridor Calibration and the Arterial Network Calibration steps, distinct rounds of iterative adjustments with specific model parameters are defined. The parameter groupings within each round of calibration were defined based on the findings within this report and the previous calibration experience of the project team. Within each of these rounds, iterative adjustments are recommended testing different parameters and cases both in isolation and in conjunction with one another. Calibration checkpoints—evaluating both the MOE outputs and visual inspection of the model—should occur with each iteration. Engineering judgment is required to dictate when enough iterations have been attempted before moving on to the next round.

5.6.3 Calibration Guidelines

The following guiding principles should be followed to improve calibration consistency, ultimate model robustness, and simplify the calibration procedure. These principles were developed in the Vissim 11 Evaluation Project and more details can be referenced in the *Vissim 11 Evaluation Report*.

1. Activate Cooperative Lane Change at default values for Maximum Speed Difference and Maximum Collision Time in the “merge, diverge, weaving segments” driving behavior container.
2. Activate Advanced Merge in the “merge, diverge, weaving segments” driving behavior container. In most recent versions of Vissim, this parameter is activated by default for all containers.
3. Freeway car-following model parameters CC0 and CC1 should stay consistent between **all** freeway driving behavior containers for a specific vehicle type. In Vissim 11, the option exists to assign car-following model parameters based on vehicle class. Therefore, in the “basic segments” and “merge, diverge, weaving segments” for a single occupancy vehicle, the CC0 and CC1 should remain the same. Similarly, the CC0 and CC1 in both “basic segments” and “merge, diverge, and weaving segments” for heavy vehicles should stay the same. However, the CC0 and CC1 for heavy vehicles and single occupancy vehicles within “basic segments” do not need to be the same. In Existing Conditions models where the AM and PM peak periods are modeled separately, it is recommended to maintain as much consistency between models as possible; however, minor differences in the calibration of these models is expected as driving behaviors and traffic conditions are different throughout the day.
4. Freeway car-following model parameter CC2 should be greater in “merge, diverge, weaving segments” than in “basic segments”. A good starting point is to leave CC2 at default for “basic segments” and increase the parameter value for “merge, diverge, weaving segments”.
5. The Maximum Deceleration for “Own” Vehicle and Maximum Deceleration for “Trailing” Vehicle should be adjusted simultaneously considering the correlation between these parameters.
6. The Safety Distance Reduction Factor should remain at default unless all other parameter adjustments have been attempted and model calibration cannot be reached. In this case, the changes to Safety Distance Reduction Factor should still be kept to a minimum.

7. The Wait Time before Diffusion parameter in the lane change model should not be decreased from the default of 60 seconds.
8. In cases where field data shows congestion throughout the model simulation period—including the seeding period—it will be necessary to add additional time to the seeding period or add additional constraints to traffic flow during this period to initialize the network. It is preferred to set the start of a model simulation period while the network is observed to be in free-flow conditions, thus allowing congestion to form naturally in the simulation. However, resource constraints and the availability of data limit the extension of the simulation period.

In these cases, travel speed can be artificially reduced using Reduced Speed Areas during the seeding period. However, since this method is not preferred; justification should be provided for the addition and approval should be obtained by the VDOT project manager. When applied, this artificial reduction in travel speed is only recommended for the first quarter to half of the seeding period, leaving sufficient time to allow traffic flow to reach a natural equilibrium before the analysis period begins. It is critical to ensure that all remnants of these artificial network elements are gone by the time the analysis period begins. This is especially important for very long Reduced Speed Areas, as a vehicle that enters into a Reduced Speed Area while it is active will keep that newly assigned desired speed until it exits, even if the Reduced Speed Area is deactivated during that time.

9. When modeling terminal or seeding conditions using Reduced Speed Areas, it is recommended to activate the Overtake Reduced Speed Areas parameter to allow realistic lane change behavior through these segments.
10. Arterial car-following model parameters AX and BX for “basic arterial segments” and “oversaturated arterial segments” should be assigned based on field data and observations. In general, the AX parameter impacts intersection delay and queuing, as it represents the Standstill Distance when vehicles are at a complete stop. Whereas, BX represents the Safety Distance while the vehicles are in motion. Therefore, it has a greater influence on arterial corridor throughput and travel time. The Wiedemann 1999 and 1974 car-following models are very different models; however, intuitively AX can be associated with CC0 and BX can be associated with CC1. No evidence has been found to suggest that the same rules for applying a consistent AX or BX throughout the corridor is necessary. This could be attributed to the different characteristics of uninterrupted and interrupted flow facilities: (a) abrupt changes in driving behavior may be more common on arterial networks and (b) shockwaves caused by changes in driving behavior may be dampened by the consistent stop-and-go conditions inherent to arterial traffic control.
11. During the arterial network calibration process, once the “basic arterial segments” parameters are set, intersection approaches or specific arterial segments that do not meet calibration thresholds can be changed to the “oversaturated arterial segments” or another specialized driving behavior container when justified. Then, the parameters from these containers can be adjusted.
12. The same lane change model parameters are available for arterial driving behavior containers as they are for freeway driving behavior containers. For arterial driving behavior containers, these parameters should be initialized as default and may not require substantial adjustments. However, when necessary, these lane change parameters can be adjusted in arterial driving behavior containers for model calibration.
13. Additional driving behavior containers may be added on a case-by-case basis if driving behavior on a specific roadway segment is observed to operate significantly differently than the corresponding freeway “basic segments”, “merge, diverge, weaving segments”, “basic arterial segments”, or “oversaturated arterial segments”. Examples include driving behavior through a work zone or innovative intersection. The previously described rules related to specific parameters remain true for these “special segment” driving behavior containers and should still be followed. In these situations, justification should be provided explaining why driving behavior along this segment does not operate consistently with the rest of the network.

Appendix B contains a range of driving behavior parameter values used on past VDOT projects. This range is not provided as steadfast boundaries, but for comparison as parameters are updated during calibration. The guidelines take precedent over these parameter ranges.

5.7 CALIBRATION REPORT

Once the calibration has been completed, the procedure, assumptions, and results should be documented. This report should include the following elements.

- Description of the study area
- Description of traffic flow through the study area (i.e., bottlenecks, traffic patterns, other challenges)
- Calibration results alongside calibration criteria for previously defined MOEs
- Number of model runs required based on the **VDOT Sample Size Tool**
- Summary of parameter changes made during calibration
- Justification that the model is sufficiently calibrated
- Justification of any deviation from the calibration criteria

6 Model Scenarios

Scenario management is a critical consideration for Vissim projects. It is imperative that model assumptions and coding remain consistent between different scenarios and changes only occur when justified to match the conditions of the new scenario. Documentation is required to record updates and revisions made between the different scenarios.

6.1 EXISTING CONDITIONS MODEL DEVELOPMENT

In most cases, projects will begin with the development of one or more existing conditions models. These models are created to represent existing conditions for which field data can be collected for model calibration. Typically, each model is calibrated to match a single peak period—AM peak, Mid-Day peak, or PM peak—therefore, when multiple peak periods are modeled, they are often represented in separate models.

Once the existing conditions models are calibrated according to the calibration guidance in **Section 5** to adequately match field conditions, they are to be used as the baseline for generating future no-build models that are used to assess traffic conditions under future travel patterns and demands.

6.2 FUTURE NO-BUILD MODEL DEVELOPMENT

As the name implies, future no-build models are created to represent future traffic conditions if the considered alternative(s) are not implemented. These models are developed for the opening year and design year; the specific years are contingent on the travel demand models available from travel demand models selected in the traffic forecasting scoping process (e.g., MPO models or locality models).

The calibrated elements from the existing conditions model should be maintained in all future models wherever roadway conditions and/or geometry remain consistent between existing and future scenarios; this is the reason it is critical to have fully vetted and well-calibrated baseline models to ensure their transferability into new scenarios. These calibrated elements include link and vehicle-assigned driving behaviors, lane change distances, and seeding conditions. Engineering judgment should be used to determine if traffic conditions are unrealistic given the new travel demand and traffic patterns. In these cases, all model changes should be made consistent with the calibration guidelines and surrounding calibrated elements. Each change to the network should be justifiable and well-documented.

The no-build scenario for the opening and design years should be updated with planned and approved projects found in the long-range transportation plan and agreed upon in the project framework document. In this way, true no-build conditions for the analyzed alternatives will be represented.

In making geometric adjustments to the Vissim model to account for planned improvements along the corridor, engineering judgment should be used to update model parameters. In general, parameters that would have been updated during calibration—had those improvements existed in existing conditions—can be adjusted in accordance with the calibration guidance. These adjustments should mirror the selected parameters from the calibrated existing network features. For example, if a new intersection is planned for a future scenario along a corridor of intersections, the driving behavior approaching and departing the new intersection should match closely with those of adjacent intersections.

Once no-build models are developed, they are used as the baseline for generating and comparing alternative build scenarios. Therefore, their robust nature is just as important as the existing conditions model.

6.3 FUTURE BUILD MODEL DEVELOPMENT

Future build models are developed to represent one or more design alternatives that the project intends to evaluate. Build models are generated for the same opening and design year as the no-build models. The no-build models for each future year are used as the baseline model, and the proposed alternatives are coded into the model. Therefore, the calibrated elements of the no-build model should be maintained throughout the model, except for adjustments related to the alternative being evaluated. When coding the new alternative designs into the model, engineering judgment should be used to update model parameters. All adjustments should be made in accordance with the calibration guidance and should mirror the selected parameters for adjacent network features in the no-build model.

Disclaimer

This user guide was written to supplement the update to the VDOT TOSAM in response to the increased need for PTV Vissim software on VDOT projects. The purpose of this user guide is to inform Vissim users of best practices for model development, debugging, calibration, post-processing, and result reporting. The requirements and guidance stated herein should be followed in all VDOT projects using Vissim as an analytic tool. This user guide is intended to be used by VDOT staff, localities, and the consultant community for VDOT projects.

This page is intentionally left blank.

APPENDIX A

Vissim Model QC Checklist

Note: Microsoft Excel version of Appendix A is available.

VISSIM Modeling Review - [project or study name]

[date] [year, conditions, other]
 Model: [model or file name]



Model and Documentation Inventory			
	Acceptance AM	PM	Comments/Notes
Model Available?	yes		
Assumptions Documentation?	no		
Calibration Document?	n/a		
Field Data			
Volumes			
Travel Times			
Other			
Model Outputs			
Raw output			
Post-processed			
MOE graphics			

Model Verification and Review		
	Acceptance	
Level 1	Geometry, Speed and Controls	AM PM Comments/Notes
1.1 Network Coverage (adequately covers study area)		
1.2 Channelization (link/connector)		
1.3 Link geometry (length, lanes, link types)		
1.4 Free flow speed coding (location of desired speeds and reduced speed areas)		
1.5 Desired speed distributions (posted-speed or other data source)		
Intersection controls (see signal timing checklist for timing parameters)		
1.6 Signal head placement		
1.7 Detector placement		
1.8 Prohibited turns		
1.9 RTOR		
1.10 Permissive movement conflicts (LT, RTOR, ped)		
1.11 Pedestrian crossings		
1.12 Lane closures and restrictions		
1.13 Freeway Segmentation (HCM compatible)		
1.14 Ramp Meters		
1.15 Parking Zones (if used)		
1.16 Lane Change Distance		
Level 2	Vehicular Demand	AM PM Comments/Notes
2.1 Input volumes		
2.2 Traffic compositions		
2.3 Routing (volumes and route placement)		
2.4 Link assigned volumes		
2.5 Vehicle classes		
2.6 Vehicle types		
2.7 Vehicle occupancy (if HOV is used)		
2.8 O-D Matrices		
2.9 Transit routes and stops		
2.10 Transit dwell times		
Level 3	Simulation and Evaluation Settings	AM PM Comments/Notes
3.1 Seeding time		
3.2 Time steps per second		
3.3 Data collection period		
3.4 Data collection interval		
3.5 Number of runs		
3.6 Results Management (keeping only multi-runs) Delay and Queue evaluation distance (adjust default to 2 miles)		
3.7 Queues		
3.8 Link evaluation segment (adjust default to 1 mile)		
Level 4	Driver Behavior and Animation Review	AM PM Comments/Notes
4.1 Review Simulation (running the model) for reasonableness (gridlock, recurring conflicts, signal phasing etc.)		
4.2 Lane change distance (lane utilization and queuing) Driver Behavior Parameters (car-following and lane change models)		
4.3 Weaving/merging behavior		
4.4 Reduced capacity		
4.5 Queuing behavior		
4.6 Free-flow speeds		
4.7 Network termini traffic flow conditions		
4.8 Keep clear at intersection (priority rules)		

VISSIM Modeling Review - [project or study name]

[date] [year, conditions, other]
 Model: [model or file name]



Output Review			
	AM	PM	Comments/Notes
5.1 Err files checked			
Calibration and summary spreadsheet (missing targets -> check in the order of demand, signal operation, and traffic behavior at specific locations)			
5.3 Number of runs			
Identification of Critical results/comparison with animation/observation; document (narratives)			
5.5 Results consistency (observations, between scenarios)			
5.6 Variability among seeds (outliers)			
Model Comments			
- [Provide additional comments here]			

VISSIM Modeling Review - [project or study name]

[date]

Model: [model or file name]

[year, conditions, other]



Signal Timing Verification

Acceptance

Level 1 Signal Layout		AM	PM	Comments/Notes
1.1	VISSIM signal heads/detectors associated with the correct controller (RBC or VAP) number	yes		
1.2	VISSIM signal heads phase assignments match phasing plan	no		
1.3	VISSIM signal heads protected/permissive	n/a		
1.4	RTOR			

Level 2 Basic Signal Timing Parameters		AM	PM	Comments/Notes
2.1	Phase number			
2.2	Min, max1, veh extension, yellow/red clearance			
2.3	Pedestrian phase (Walk + FDW)			
2.4	Min/max/ped recall			
2.5	Pedestrian phase (Walk + FDW)			
2.6	Dual entry			
2.7	Overlaps			
2.8	Vehicle and pedestrian detector number			

Level 3 Free/Coordinated Signals		AM	PM	Comments/Notes
3.1	Phase sequence (lead/lag, permissive, overlap)			
3.2	Pattern/time of day plan			
	<i>Coordinated Pattern</i>			
3.3	Cycle length			
3.4	Split			
3.5	Offset			
3.6	MaxGreenMode (use MaxInhibit if coordinated)			
	<i>Global Values</i>			
3.7	Offset Reference* (match Synchro or field controller)			

Level 4 Error checking		AM	PM	Comments/Notes
4.1	Check error and warning message			
4.2	Observe signal operation in simulation to match field operation			

Note VISSIM LeadGreen = Synchro start of leading green (coordinated phase) = TS2 Controller

VISSIM LagFO = Synchro start of yellow (leading coordinated phase) = 170 Controller when coordinated phase lags (NOT lead/lag)

Model: [model or file name] [year, conditions, other]
File Name: [file name]
QC'd by: [name]
QC date: [date]
Backchecked By: [initials]
Backchecked Date: [date]
Updated By: [name]



APPENDIX B

Vissim Parameter Ranges

The following parameter ranges represent the high and low bounds of parameter values that have been used to represent driving behavior in VDOT projects.

Freeway Car Following Model (Wiedemann 99) – Calibration Parameters					
Parameter	Default Value	Unit	Suggested Range		
			Basic Segment	Weave/Merge/Diverge Segment	
CC0	Standstill distance	4.92	feet (ft)	4.5 to 5.5	>4.92
CC1	Headway time	0.9	seconds (s)	0.85 to 1.05	0.90 to 1.50
CC2	‘Following’ variation	13.12	ft	6.56 to 22.97	13.12 to 39.37
CC3	Threshold for entering ‘following’	-8	--	Use default	
CC4	Negative ‘following’ threshold	-0.35	--	Use default	
CC5	Positive ‘following’ threshold	0.35	--	Use default	
CC6	Speed dependency of oscillation	11.44	--	Use default	
CC7	Oscillation acceleration	0.82	ft/s ²	Use default	
CC8	Standstill acceleration	11.48	ft/s ²	Use default	
CC9	Acceleration at 50 mph	4.92	ft/s ²	Use default	

Arterial Car Following Model (Wiedemann 74) – Calibration Parameters			
Parameter	Default Value	Unit	Suggested Range
Average standstill distance	6.56	feet (ft)	3.28 to 6.56
Additive part of safety distance	2.00	--	2.0 to 2.2
Multiplicative part of safety distance	3.00	--	2.8 to 3.3