Vissim

(Version 11)

Module 7:
Start-Up & Replicate Trials
And
Verification, Validation, & Calibration

Georgia Institute of Technology

Latest Update: 8/10/2021

Contact:
Michael Hunter
Michael.Hunter@ce.gatech.edu

Disclaimer: The objective of these tutorials is to bring new users sufficiently (and quickly) up to speed so they can use the manual and other resources when they have questions on model development. These tutorials should not be considered as official guidance; users should always refer to official PTV Vissim or project sponsor documentation for the final word on a model feature question. Send any errors, issues, or comments to michael.hunter@ce.gatech.edu so any corrections can be made to the material.

PTV-Vissim (V11) Module 7 Overview

This is the seventh in a series of Vissim modules. Module 1 guides the user through the creation of a corridor which consists of three signalized, pre-time, four-approach intersections. Module 2 adds coordinated-actuated control to the corridor from Module 1. Module 3 adds an unsignalized intersection. Module 4 familiarizes the user with the basics of designing a freeway segment with a diamond interchange. Module 5 introduces additional concepts and tips for building and reviewing a Vissim model, exploring some of the Vissim modeling issues not covered in the first four modules (allowing a user to quickly get up to speed) but critical to model development. Module 6 demonstrated how to collect various performance measures: travel time, speed, delay, queues, etc. Module 7 will provide guidance on starting a model, on setting the length of a data collection period (e.g., accounting for over-capacity conditions), and on utilizing replicate trials; as well as a discussion on verification, calibration, and validation. Module 7 assumes the user is familiar with the concepts in the prior six modules and will build on the final .inpx developed in Module 6. The wireframe of this model is seen in *Figure 1*.

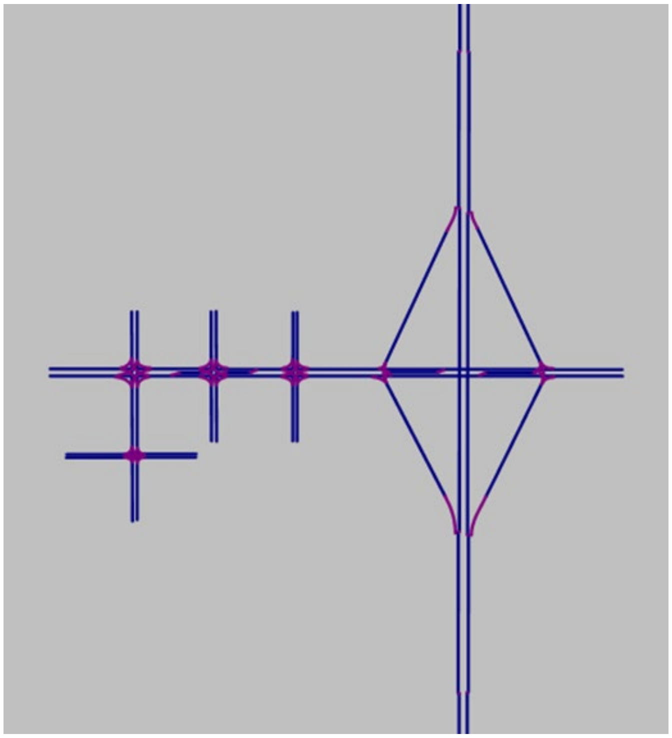


Figure 1: Module 6 final network configuration

Module 7 will step through six primary topics:

1. Start-Up Fill Period,
2. Data Collection Period Length,
3. Replicate Trials,
4. Model Verification,
5. Model Validation, and
6. Model Calibration.

Base Knowledge

For this module, it is assumed that you have a basic understanding of the concepts covered in Modules 1 through 6, including:

- Starting Vissim and Navigating the GUI (Graphical User Interface),
- *Links* and *Connectors,*
- *Signal Controllers, Signal Groups*, and *Signal Heads,*
- Ring Barrier Control,
- Alternate Intersection and Ramp *Link-Connector* layouts,
- *Emergency Stop Distance* and *Lane Change Distance,*
- *Conflict Areas, and*
- Performance Metric Data Collection:
    - *Data Collection Points,*
    - *Travel Time Measurements,*
    - *Delay Measurements,*
    - *Queue Measurements,*
    - *Link* and *Node* data collection, and
    - *Signal Control* data collection.

Create Module 7 Vissim File

To start open the Module 6 Vissim .inpx file. Save the .inpx under a new name, e.g. Module 7. Let's also update the RBC controllers to:

- RBC_Mod7_Buzz_and_Burdell_v1.rbc
- RBC_Mod7_Buzz_and_Ramblin_v1.rbc
- RBC_Mod7_Buzz_and_Pi_v1.rbc
- Mod 7 Buzz Blvd and Peach Freeway West Int.rbc
- Mod 7 Buzz Blvd and Peach Freeway East Int.rbc

IMPORTANT: SAVE FREQUENTLY AS YOU WORK!

Simulation start-up, data collection periods, and replicate trials.

Thus far we have become familiar with how to build and fine tune a model, as well as collect numerous data elements. However, in each of our model runs we have collected data from the start of the run and considered the performance metrics based on a single run. In the following sections we explore means to improve the robustness of simulation data collection and analysis.

Simulation Start-Up

At the very beginning of a model run, i.e., simulation time t = 0, there are no vehicles in model. Thus, the first vehicles to enter the model will experience empty roads and high levels of service. The performance data for these initial vehicles will likely be better, potentially much better, than the conditions under study. To account for this, we utilize a "fill" period (also often referred to as a "warm-up period") at the beginning of a simulation run. Intuitively, the fill time allows for the distribution of vehicles into the model, such that traffic conditions at the start of data collection reflect those that would be expected at the beginning of an analogous field data collection effort. For example, if you perform a travel time field data collection effort from 5:00 PM to 6:00 PM, at 5:00 PM when you start driving a probe vehicle on the roadway significant traffic will already be present, i.e., the road does not start empty. The same must also be true when collecting simulated travel time data. At the start of the data collection traffic should be present throughout the model. The fill period seeks to address this issue. Any data collected during the fill period is not used in the scenario simulation analysis.

Fill period duration

What is the appropriate fill period duration? As stated, the over-arching concept is that the fill period needs to allow the model to reach conditions at the beginning of the data collection period that reflect what would be expected in the field. There are a number of rules-of-thumb for the fill period length, such as twice the travel time of the longest path through the model, or until the number of vehicles entering the network is approximately equal to the number exiting, or visual inspection of model performance measures. There are also statistical methods to set the fill period, such as Welch's procedure (Law et. al, 2013). (More formally, what the rule-of-thumb and statistical methods are seeking to do is insure that the model data collection does not include the initial transient resulting from model start-up.)

For example, let's consider the travel time for each vehicle on a given path through a model. *Figure 2* shows the travel time results, by simulation time, for every vehicle that completes the path.

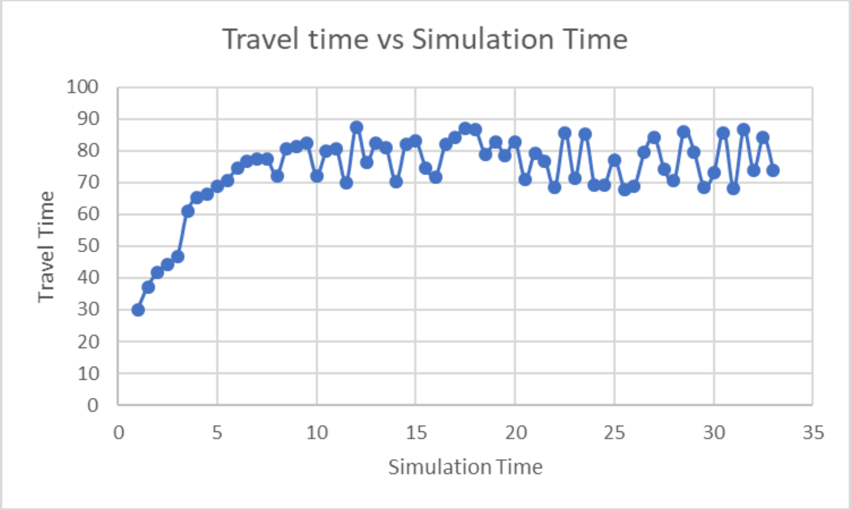


Figure 2: Example travel time for a path relative to simulation time

In *Figure 2* we can see through visual inspection of the data that the travel time, while variable, achieves a general "steady-state" somewhere between 10 and 15 minutes into the model. For the first 10 to 15 minutes the travel times are lower (but increasing) as vehicles initially populate the model. Thus, data from the first 15 minutes of the run should not be used for analysis as these travel times are influenced by the initial model start-up conditions. Ideally, the fill period length should not be based on a single path through the network, instead considering several paths (typically the longest, as measured in time) and other performance metrics, such as queue length, at other key locations. (Speed generally does not work well for bounding a fill period.) If utilizing an ad-hoc rule, such as those mentioned previously, it is recommended to generate some travel time and queue plots as a secondary check of the adequacy of chosen fill time. This also has the added benefit of being another check on the model. For example, if uncongested conditions are expected but the travel time or queue length in a model increases throughout the model duration, there is likely a modeling error (or an incorrect expectation!). While this guidance document does not recommend specific fill period rules it is observed that most transportation microscopic simulations (e.g., Vissim) will have a start-up period of at least 15 minutes on smaller models (say up to 10 to 15 intersections or a few freeway miles) with longer periods as the model size increases. As general guidance, it is better to error on the side of a fill period longer than required rather than one that is too short.

Steady state vs non-steady state

The proceeding discussion makes a very strong assumption, that is, that steady state conditions exist. Steady state implies that a model reaches some average condition and remains at that condition for some extended period. For steady state to be reached in a transportation model the demand (i.e., input volumes) must be relatively constant and no location in the model should exceed capacity. (While these are not formal rules and exceptions exist, they are pretty good rules-of-thumb). For instance, for *Figure 2*, we see that from approximately simulation time 15 minutes to the end of the run the travel time varies around approximately 77 seconds.

However, in many of the more complicated scenarios (where we are more likely to employ simulation!) volumes will change throughout the analysis period and at times demand may exceed capacity. For example, consider *Figure 3*.

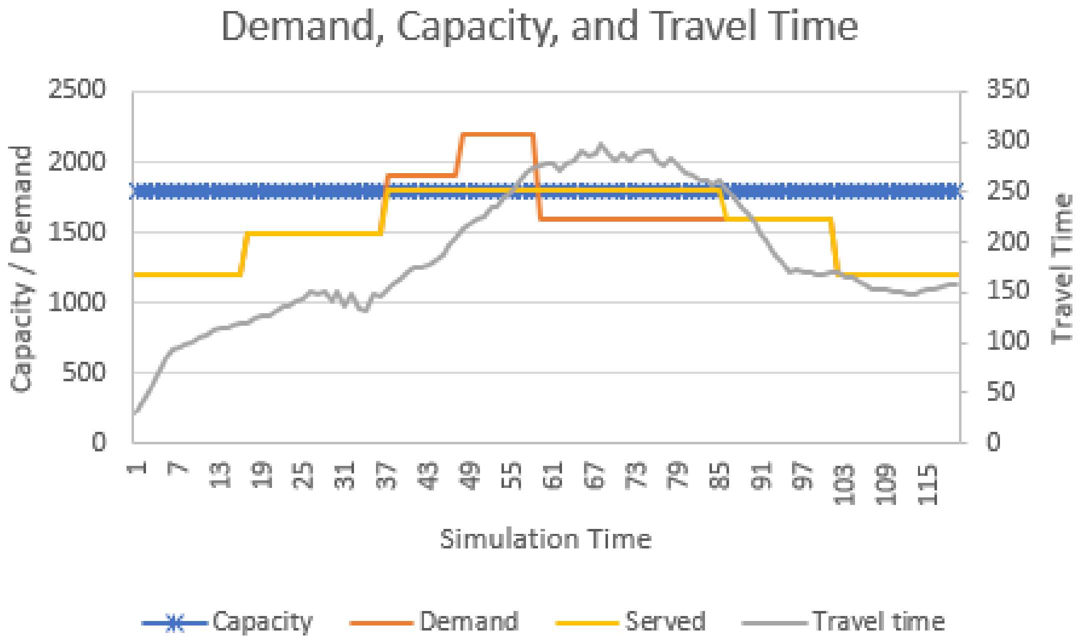


Figure 3: Analysis period with changing demands including exceeding capacity

Figure 3 presents the demand (orange), capacity (blue), served (or processed) vehicles (yellow), and simulated travel time (grey), for an example freeway section. In this example the demand (vehicle input) increases over time and then decreases. The demand exceeds the capacity for a short period of time. The processed vehicles are the number of vehicles that depart the roadway segment. As seen, the number of vehicles processed is capped by the capacity, whereas demand is not. Critically, the processed vehicle count remains at capacity after the demand drops below capacity. This results from the need to process the remaining unserved vehicles, i.e., clear out standing queues on arterials or break-down on the freeway.

From a visual inspection of the travel time data it is seen that this model never reaches a steady state. That is, the travel time never settles around an average, instead continually increasing or decreasing, reflecting the changing demand and congestion conditions throughout the run. However, while the model does not reach a steady-state, a fill period is still required prior to starting data collection for analysis. In such non-steady situations, we tend to rely on the rules-of-thumb and the general expertise of the modeler to ensure that the model has reached an appropriate state prior to data collection. The desire remains to ensure that the model reaches an acceptable traffic state prior to collecting data.

Helpful Hints

Even for non-steady state conditions it is possible to utilize steady state concepts to determine the fill time. To accomplish this the model needs to reach a short-term steady-state condition, representative of the conditions at the (uncongested) desired start of data collection. That is, set the fill period to achieve steady state for a short time period, say 15 to 30 minutes, for the starting demand. At the end of this short steady state period performance metric collection may begin and additional demands added or subtracted as needed. For example, if it is desired to simulate traffic demands from 4:00 PM to 6:00 PM, and 4:00 PM is uncongested, model the 4:00 PM traffic demand until the model reaches a steady state condition for a short time period. The model is now "filled". You may now begin the 2-hr simulation of 4:00 PM to 6:00 PM, updating traffic inputs throughout the time period and collecting performance data. As will be seen in the following section, the end of the data collection period should also be uncongested.

Aside: It is acknowledged that this approach is approximate. For instance, in the given example, it is arguable that this approach may place more traffic on the roadway at the start of the data collection than would exist in the field, if the field demands are lower from say 3:00 PM to 4:00 PM than at 4:00 PM. A fine-tuning of the method would be to determine the necessary fill time using the 4:00 PM traffic as described, then rerun the model using the field data for the fill period.

Duration of Data Collection Period

Next, we consider the length of time data should be collected for analysis. When a model remains uncongested throughout the run data collection will be based on an agreed upon (agreement between the model builder and agency or sponsor) time frame, typically one to two hours per analysis period. However, more flexibility in selecting the data collection period may be required where congestion exists. For instance, we may reasonably expect congestion to begin as demand approaches and exceeds capacity. However, as seen in *Figure 3* congestion does not ease until after demand falls under capacity AND the unserved demand is able to be processed. To model such scenarios, it is critical to simulate from *before* any breakdown or congestion occurs to *after* the traffic returns to uncongested conditions. If the model data collection fails to capture this entire time period than delays, emissions, queuing, or other critical performance metrics impacted by the high demands will not be fully captured in the model performance metrics.

Over-Capacity Conditions in VISSIM.

In the above example it is stated that when the demand drops back under capacity that the built-up unserved demand may then be processed. Where are these unserved vehicles while the demand exceeds capacity? If the model is sufficiently large, vehicles will generally be queued within the model, either in jam density conditions on the freeways or in long queues on the arterial. However, if there is not sufficient space for the vehicles in the model they will essentially queue outside the model. For instance, if you watch a model animation what you will see is the vehicle queue extend to the beginning of the input link. Vissim will then input vehicles as the space becomes available on the link, essentially holding the excess vehicles outside the model until space exists. The vehicles being held outside the model will NOT contribute to any of the model performance metrics. Thus, if a vehicle desired to enter the model at 4:05 PM but did not enter the model until 4:15, those ten minutes of delay will NOT be reflected in any performance metrics. To capture this delay in the performance metrics it is necessary to extend the underlying model input links until vehicle queues no longer reach the end of the link. Similarly, even within a model, if a queue extends upstream of a travel time measurement starting point then the travel time (and delay) upstream of the start of the measurement will not be included in the measurement statistics. It is critical that measurements be set to capture the full extent of all queuing and delay, across all model scenarios. Thus, the minimum length of your entry links and travel time measurement zones will be determined by the worst-case model, to allow for scenario-to-scenario comparisons.

A note on Vissim holding vehicles outside of the model

In general, Vissim will hold vehicles outside of a model until they are able to enter at the given link. (Aside, this is not necessarily true of other simulation packages. It is important to check these behaviors with any modeling platform you may use.) However, there are certain circumstances where vehicles being held outside of the model may be lost. Another useful check with any modeling effort is the placement of *Data Collection Points* on the critical model entry and exit points. The counted entering and departing vehicles should then be compared to the expected values. Any significant differences are indicative of an underlying model error, such as not all vehicles entering the model, vehicles trapped in the model at a bottleneck, vehicles not properly routing, etc.

Replicate Trials

As a final point, we discuss replicate trials. In the modules thus far, all results have been based on a single model run. However, Vissim is a stochastic model, meaning it seeks to incorporate randomness. For instance, we saw in Module 5 that *driver behaviors* such as *acceleration, desired speed,* and *route* choice are based on distributions. For each individual vehicle, when an acceleration, desired speed, or route decision is needed the model must draw a single value from the parameter's distribution. By changing the random draws for the vehicles (but not the underlying distributions) variability in the model results should occur. This is analogous to seeing day-to-day variability in a corridors operation, even if the demands and total turn movement counts remain similar. We seek to reflect this variability due to randomness using replicate trials.

Replicate trials allow us to understand the variability that may be expected for a given design and demand condition. Without getting too deep into the formal details, most simulations utilize a random number generator to differentiate vehicle behaviors between replicate trials. Whenever a decision needs to be made, such as selecting the *desired speed* for a given vehicle, the random number generator "generates" a random number, which is then used to select a value from the parameter's distribution. However, you may have noticed that, thus far, when you run a Vissim simulation multiple times (making no changes to the model), you get the exact same answer every time! This is because all of our Vissim runs have started with the same *Random Seed*. The random seed determines the sequence of random numbers from the random number generator. If you change the random seed for a run you will get different results! We will see how to do this in Vissim in the next section. (In the preceding discussion we avoid the formal details on generating random numbers, using a random number to select a value from a distribution, etc. For readers interested in these details I highly recommend Simulation Modeling and Analysis, 5th edition, by Averill Law, 2013.)

Once the replicate trials are generated the average value of the performance metric(s) of interest are typically reported. For instance, in our travel time example above we would report the average value across all replicates, for the time period(s) of interest. The question becomes, how may replicate trials are necessary to determine a meaningful (or statistically significant) average? The general concept is to complete a sufficient number of replicate trials such that an acceptable confidence interval (C.I.) may be placed on the average performance metric value. For example, ensuring that the 95% C.I. on path travel times is within ± two-minutes. The choice of these targets, i.e. 95% or ± two-minute range, must be determined, in advance, by the model developers and users (e.g., DOT) based on the needed accuracy for the project at hand.

For reference, the number of replicate runs required to achieve a given C.I. is typically found using (Florida, 2014):

$$n = \left(\frac{s * t_{\propto/2}}{\mu * \varepsilon}\right)$$

Where:

n is the required number of simulations runs

s is the standard deviation of the system performance measure

$t_{\propto/2}$ is the critical value of the two-sided Student's t-statistic at the confidence level of α and n-1 degrees of freedom.

μ is the mean of the system performance measure

ε is the tolerable error, specified as a fraction of μ

Standard deviation will likely need to be estimate and updated during the study as replicates are generated, although often an "educated" guess can be made based on previous simulation results. The tolerable error is user determined, typically on the order of 5% or 10%. This value should be set in coordination with the end users of the model, accounting for the final model application. As a rule-of-thumb, ten replicate trials is usually sufficient for most Vissim models, although the value calculated using the equation above should be checked for several key parameters, such a critical travel time routes or average queues at key locations. The performance metrics that will be used to determine the replicate trials should be agreed upon *before* the simulation study is undertaken.

A few thoughts on replicate trials

Replicate trials are a critical component of a robust simulation analysis. Loosely, in an uncongested model the number of required replicates tends to be low as the slack in the traffic stream is able to absorb the perturbations resulting from randomness, leading to minimal differences between replicates. Highly congested systems also tend to require fewer replicates as the model will show failing service regardless of the randomness. However, for models with conditions near capacity, random influence can significantly impact simulation results. In these scenarios it is critical to obtain a sufficient number of replicates to adequately estimate the average of the desired performance metrics as well as understand the potential variability in the system performance. As Vissim will automatically generate replicate trials and average statistics, the resource cost of increasing the number of replicate trials tends to be minimal. Thus, similar to fill time, it is better to error on the side of more replicates than required than too few.

Finally, DO NOT THROW OUT REPLICATE TRIALS. There will be a temptation when conducting replicate runs to throw out the results of a replicate trial with the justification that "it must be an error." For instance, if nine out of ten replicate trials show acceptable performance and one fails. This outcome is why we conduct replicate trials! If the modeler can demonstrate that the replicate trial has uncovered a simulation error, it is necessary to find and fix the error, we cannot just assume an error exists and ignore the replicate run. If an error is found and fixed then all replicate runs must be repeated.

Simulation Start-Up, Data Collection, and Replicate Trial Take-Aways

Key simulation start-up, data collection periods, and replicate trials take-aways when building or reviewing a model include:

- The model must be allowed to fill for a sufficiently long period that at the start of data collection for scenario analysis the traffic reflects expected conditions.
- Travel time, data collection points, etc. should be used to confirm an acceptable fill time has been reached as well as being indicators for potential underlying model issues / errors.
- The data collection period should be set prior to modeling and based on project needs and analysis requirements.
- When congested conditions are being modeled, the data collection period, at a minimum, should begin from before the onset of congested conditions and end after the return to uncongested conditions.
- Vehicles queues should not be allowed to extend off the network as these vehicles will not be included in performance metrics.
- To allow for the comparison of performance metrics between scenarios the setting of travel time measurement zones, the length of input link lengths, etc., must be adequate for all scenarios.
- It is necessary to conduct replicate runs of the simulation to reflect the stochastic variability in transportation system. The allowable confidence interval bounds in reported metrics should be set before the simulation study is conducted.

A model developer must be able to demonstrate that each of the above has been accounted for in the model, similarly a reviewer should confirm that each condition above is satisfied.

Implementing Replicate Trials in Vissim

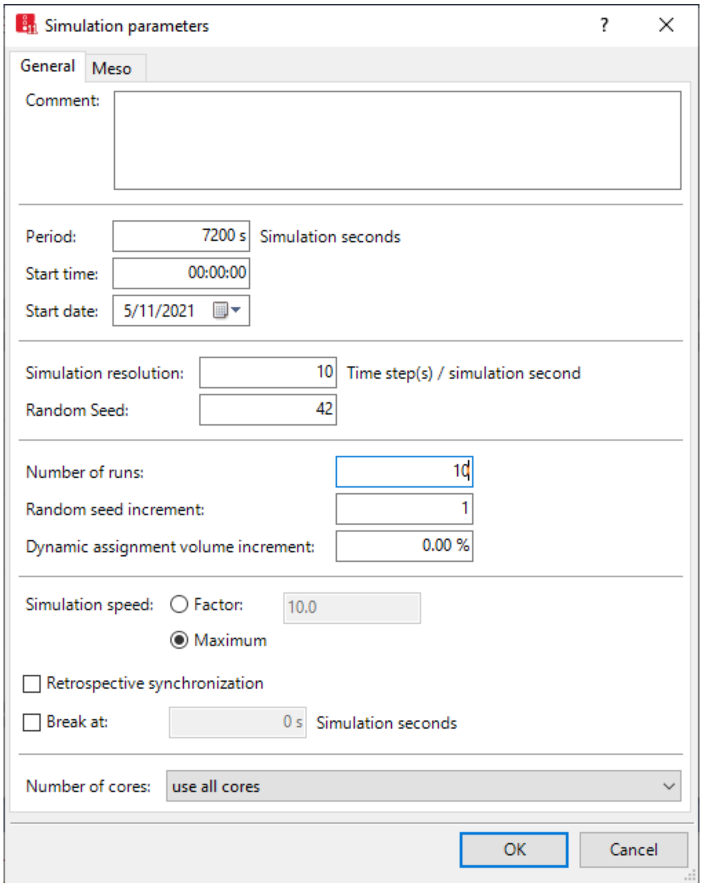
Next, we will step through the implementation of replicate trials and determine the fill period for our demonstration model.

step i. Go to *Simulation* → *Parameters* → *General*, Figure 4:

- Set the *Period* to 7200s.
- The *Random Seed* should default to some value, typical 42. That is fine.
- Set the *Number of runs* to 10. This is the number of replicate runs. You can leave the *Random seed increment* at 1.

step ii. Select *OK*.

step iii. You may receive a message regarding evaluation results that says previous results will be lost. This is ok. Select *Continue*.



Simulation parameters

General | Meso

Comment:

Period: 7200 s Simulation seconds

Start time: 00:00:00

Start date: 5/11/2021

Simulation resolution: 10 Time step(s) / simulation second

Random Seed: 42

Number of runs: 10

Random seed increment: 1

Dynamic assignment volume increment: 0.00 %

Simulation speed: ○ Factor: 10.0

● Maximum

□ Retrospective synchronization

□ Break at: 0 s Simulation seconds

Number of cores: use all cores

Figure 4: Simulation Parameters

step iv. For the analysis let's use the *Vehicle Travel Time measurements* and *Data Collection Points* that we set up previously. Go to *Evaluation* → *Configuration* → *Results Attributes*.

step v. Be sure only *Data Collections* and *Vehicle Travel Times* are selected. Set the interval to 120 for both. (A shorter interval will let us explore the fill period length.)

step vi. Go to the *Results Management* tab (you should still be in *Evaluation Configuration*). Under *Keep result of previous simulation runs* select *of current (muli-)run only*.

step vii. Go to the *Direct Attributes* tab. Uncheck all of the *Write to file* boxes that may be highlighted from our previous work.

step viii. Open the *Vehicle Travel Time Results* list. Be sure the disk with the red arrow (💾) is selected to allow for automatic saving of the results, the summation sign (Σ) is selected so Vissim will calculate simulation run aggregates, and the summation within the circle is selected (Σ) so Vissim calculates time interval aggregates.

Save and run the file. The simulation should run 10 times, that is, ten replicates. You may wish to click the continuous mode (⏲) icon to speed up the runs (continuous model turns off the animation, significantly speeding up the model run time).

At the end of the runs check the *warnings messages*. If you are receiving warnings you should address those before performing any analysis. While you may not address all warnings, you should seek to address as many as possible such that the remaining warnings have minimal potential impact.

When the run is complete there will be a *Data Collection Results* and a *Travel Time Results* *.att* file for each replicate trial. By selecting to keep the previous run results each *.att* file will contain the results of the previous replicate trials. While this makes the files a little larger it is more convenient for analysis.

In Excel (or other software) open the *.att* travel time file iteration ten, i.e. 010. The data should be available for all replicate trials.

Figure 5 below shows the Buzz Blvd EB – End-to-End travel time for each 120 second interval for each replicate trial and the average across all runs.

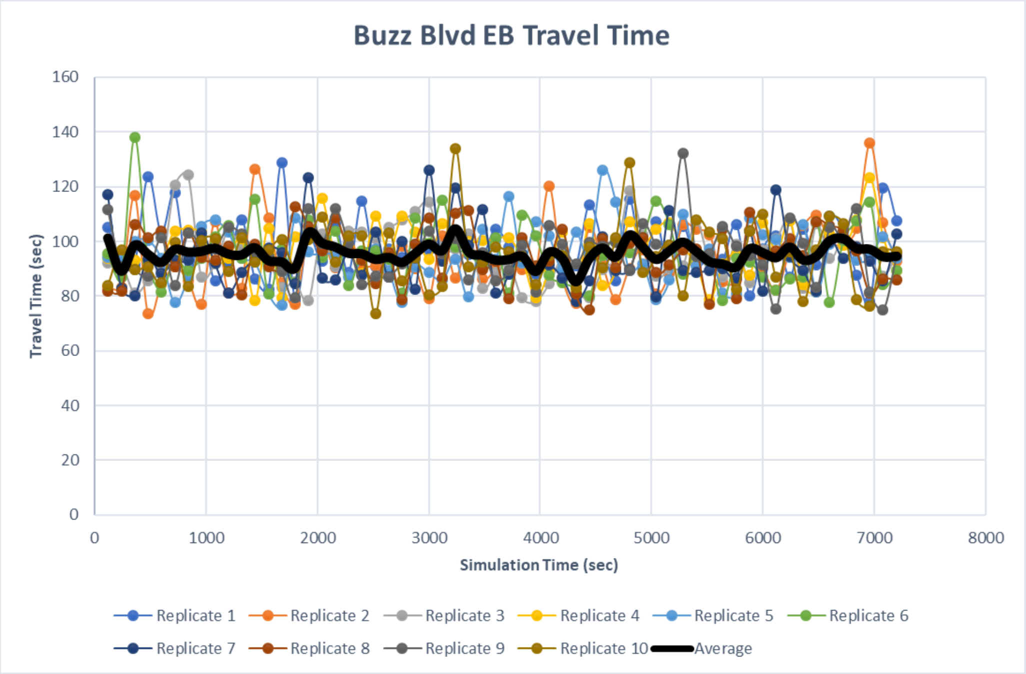


Figure 5: Replicate Trial Travel Time Results for Buzz Blvd EB.

As is readily seen the travel time reaches a steady state condition very quickly (within the first couple of minutes!), which is not unexpected for a relatively small model with uncongested conditions. In addition to checking the Buzz Blvd EB travel path other routes should be checked as well, such as Buzz Blvd WB, and Peach Freeway. As this is a relatively simple model, and assuming that the animation also seems reasonable and no warning errors are showing, a 15-minute fill time is sufficient. Thus, if an hour of data is needed for analysis, the data from 900 second to 4500 seconds should be used. However, additional data could be explored to confirm model warm-up has been achieved, such as queues and delays. The measures to be used for warm-up should be determined as part of the design of the study. For the performance metrics of interest, the average (and standard deviation if desired) across the replicate trials should be reported at minimum.

Lastly, are ten replicate trials sufficient? Let’s take our EB Travel Time for the example.

For the ten runs the average travel time is 95.58 seconds. The standard deviation is 1.21 (this is the standard deviation of the ten replicate averages). The t value for 5% confidence level would be 2.26 (using a student t-test table for 9 (n-1) data points). Assume an allowable error of approximately of 5% or 0.05. The number of replicates as determined by the C.I. equation above would be less than 1. Why, because the standard deviation is very small relative to the travel time. However, exercise caution, this is based on a measure over a long distance, which is likely to have a lower standard deviation relative to the average. If we check the travel time for the EB left at Buzz Blvd at Ramblin St. we would see an average travel time of 59.9 seconds with a standard deviation of 4.3 seconds. This would result in a need for 4 replicate trials. Thus, it is important to confirm the number of replicate trials for the various critical measures of interest, with the more variable measures governing the number of trials. Finally, we generally will set the minimum number of replicates at 10, which is what should be used in this case.

Verification, Validation, and Calibration

Nearly all modeling efforts will include some level of calibration. However, what is meant by calibration, and what calibration should include, is not always clear. To help clarify, there are actually three separate, but related, processes that should be undertaken for each model: verification, validation, and calibration. Each is critical to the successful design, construction, and use of a model. Briefly, verification is the confirmation that a model has been constructed as intended (e.g., a roadway that is 3 lanes in the field is 3 lanes in the model), validation confirms that the performance of the model satisfies expectations (e.g., the model approximately matches field conditions), and calibration is the process of adjusting the parameters of a verified model to achieve a valid model. Additional detail is offered on each process in the following.

Verification

Verification is the process of confirming that the model is built and operates as intended. (This differs from validation, verification only means the model is doing what you think it is doing, not that the model is necessarily correct.) For example, at the intersection of Buzz Blvd and Ramblin street there should be an EB left turn bay. Verification is the process of confirming that the bay is actually in the model. Other examples include: are the number of lanes on a link correct, is the signal timing programmed as intended, are the volume inputs as intended, are the desired vehicle routes all placed, are conflict zones appropriately placed and defined, etc. Verification requires carefully stepping through a model and ensuring that each model element is constructed and operating as intended. Some verification issues will show up as an error or warning message; however, this is not always (or even often) the case. For example, if a link should be three lanes and you entered two, this would not trigger a warning or error message.

When reviewing a model or after the initial model construction a **deliberate verification of each model element is critical**. Unfortunately, and all too often, model developers will seek to use calibration techniques to account for what is actually an underlying issue in the model development. While it may be possible to calibrate a model with an error to give the expected results for an existing condition, the likelihood is that when then using the model to conduct scenario analysis that the results will be unreliable. The importance of a verifying a model cannot be understated! A sample model verification list is provided in Module 8.

Validation

Validation is the process whereby we seek to confirm that a verified model is valid. Valid is typically defined according to some set of performance metrics, e.g., the travel time on selected routes are within some acceptable percentage of field travel times, the vehicle counts at critical locations approximately match field (or expected) conditions, etc. (In the next section a set of potential validation measures will be given.) Essentially, validation seeks to confirm that a model matches (approximately) the real world. Where real world conditions do not exist (e.g., the simulated roadway has not yet been constructed) validation may seek to confirm that aspects of model performance match reasonable expectations. That is, while you cannot field validate the travel time for a corridor that does not yet exist, it is possible to confirm that that the free-flow speed, saturation flow, etc., are reasonable.

For clarification, when considering verification vs. validation, generally speaking, input data is verified and output data and performance metrics are validated. For example, if the model developer sought to set the vehicle inputs to match field data counts, this must be verified in the vehicle input list during the model review. Whereas, data collection points may be placed at the entry points and other locations throughout the model to confirm that the number of vehicles crossing these points match expectations. Similarly, the entered desired speed is verified during model review, whereas travel time is validated against field travel times, that is, desired speed is an input and travel time is an output.

As good practice in model development, the data used to construct and calibrate a model should not be used to validate the model. That is, if we have a set of vehicle inputs and associated travel times that we have used during the verification and calibration processes a separate set of vehicle inputs and travel times should be utilized for validation. For instance, ideally a model developer could utilize the input data from one week for verification and calibration then validate the model using data from a subsequent week. (The interested reader is referred to Law (2013) for a detailed discussion on validation.)

However, it is recognized that transportation simulation modeling efforts often do not have multiple available data sets, or data set that can be readily split into multiple data sets. The same data set is often used for validation that is used for calibration. While this will often need to be the accepted process it should be understood that this is not as robust as utilizing separate data sets for validation and calibration. For instance, if there is an underlying verification issue that is "resolved" through calibration, the use of a secondary data set for validation may help uncover the issue.

(ASIDE: The use of the same data set for validation and calibration is often not raised as a significant issue in the development of Vissim models as the undertaken effort is not a simulation development in the purest sense. That is, the traffic flow algorithms, weaving logic, etc., are not being developed and coded as part of the modeling effort. In constructing a Vissim simulation the modeler is leveraging pre-existing model elements that have already been validated in many other models. Thus, Vissim model construction is primarily an application of existing simulation elements rather than a simulation development effort, eliminating many of the pitfalls that separate validation data sets seek to address. However, we expect the PTV developers of the Vissim elements are utilizing multiple data sets!)

Calibration

Calibration is the adjustment of the underlying parameters of a **verified** model to achieve a valid model. In Vissim this is often taken to be a calibration of the underlying *Wiedemann car following parameters*. The next section will explore the impact of adjusting these parameters.

However, it must be stressed that **these parameters represent only one part of calibration**. DO NOT ASSUME THAT VISSIM CALIBRATION INVOLVES ONLY THE ADJUSTMENT OF *WIEDMANN* PARAMETERS!!! Significant time has been spent in the previous modules considering model aspects such as *Lane Change Distance*, *Emergency Stop Distance*, etc., as these are critical to a valid model. Prior to any *driver behavior* parameter (i.e., *Wiedemann 74* and *99*) calibration these elements should be reviewed and adjusted. Often the desired (i.e., valid) model performance can be achieved through calibration of these model elements. For instance, adjusting the *Emergency Stop* and *Lane Change Distances,* or using the alternative *link-connector* layout, to alter the performance at an on-ramp (see Module 5). Other calibration efforts may involve adjusting *vehicle routes*. For instance, when intersections are closely spaced it is often challenging to properly capturing weaving and turn movements, resulting in invalid congestion estimates and diffused (i.e., removed) vehicles. Improved performance may be obtained by treating these intersections as a single model element, using a set of *routing decisions* that accounts for both intersections. This will allow vehicles to pre-position for a turn at the downstream intersection while in the upstream intersection through lanes. When we use calibration of *driver behavior* parameters to resolve the issues between these intersections we tend to create more “aggressive” drivers. However, a need for increased aggressiveness is not the underlying issue; rather, drivers knowing which lane they want to use in advance of the combined intersections. Other examples may include adjusting the length of turn bays to reflect actual use rather than the painted layout, ensuring that the *desired speed distribution* reflects free-flow conditions, and so one.

Vissim offers are great deal of flexibility and ability to adjust *driver behaviors* using model elements and features. A calibration effort must include the thoughtful adjustment of these elements before any parameter adjustments to the underlying *driver behavior* model parameters! You will find that a thorough calibration of these elements will minimize or even eliminate the need to adjust the underlying traffic flow models. In addition, remember, the adjustment of underlying *driver behavior* parameters affects **every** link that uses that driving behavior. The use of model parameters such as *Lane Change Distance* and *Emergency Stop Distance* allows for a targeted calibration of specific areas of the model.

Calibration Guidance

This module will not recommend specific methods for calibration; however, numerous examples of simulation and Vissim calibration methods exist, such as Gomes et. al (2004), Law (2013), Park et. al (2006), VDOT (2020), and Wunderlick (2019). Approaches to calibration range from iterative manual approaches to the use of genetic algorithms to search for optimal parameter values. While this module will not recommend a method, the following provides high-level guidance for the general application and review of a model calibration.

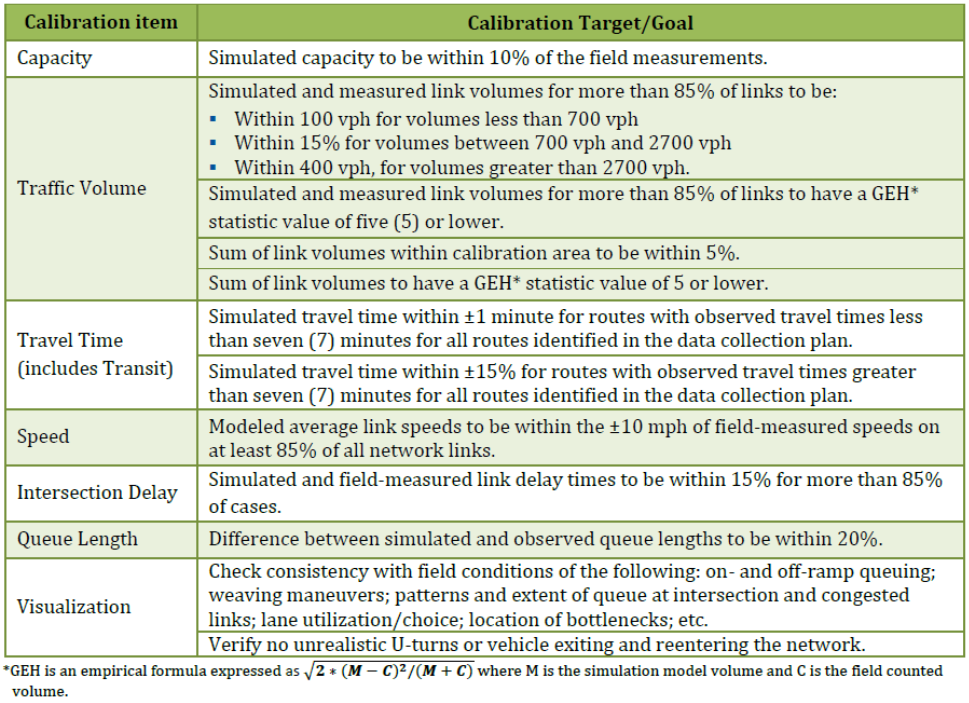
Critical to successful calibration is that both the model developer and model reviewer are familiar with field operations for the network being simulated. A calibration effort has a much higher likelihood of being successful (i.e., resulting in valid model) if the developer and end user can quickly judge through visual inspection the reasonableness of model operations.

The simulation developer and the supervising client (e.g., GDOT) should agree, prior to model development, on each of the following:

- Calibration objectives, including:
    - Performance measures of interest (clearly defined)
    - Allowable deviation from the field or expected conditions
- Base or existing network to be calibrated:
    - Calibration conditions to be considered, e.g., congested, uncongested, etc.
    - Network boundaries
- The calibration method
- Locations in the model identified as critical
    - bottlenecks
- Site field data to be collected
- Data that may be utilized from other locations
- Graphical representation of the calibration effectiveness
- Statistical analyzes of the calibration effectiveness
- Sensitivity analysis to be undertaken
- Constraints on parameters sets (i.e., min or max parameter values)
- Flexibility in calibration
- Items specific to the calibration of the given network
- Documentation

Calibration objectives should be clearly defined.

- Calibration objectives will help focus the field and simulation data collection and analysis. *Figure 6* contains some of the more common calibration targets.



Calibration item	Calibration Target/Goal
Capacity	Simulated capacity to be within 10% of the field measurements.
Traffic Volume	Simulated and measured link volumes for more than 85% of links to be: ▪ Within 100 vph for volumes less than 700 vph ▪ Within 15% for volumes between 700 vph and 2700 vph ▪ Within 400 vph, for volumes greater than 2700 vph.
	Simulated and measured link volumes for more than 85% of links to have a GEH* statistic value of five (5) or lower.
	Sum of link volumes within calibration area to be within 5%.
	Sum of link volumes to have a GEH* statistic value of 5 or lower.
Travel Time (includes Transit)	Simulated travel time within ±1 minute for routes with observed travel times less than seven (7) minutes for all routes identified in the data collection plan.
	Simulated travel time within ±15% for routes with observed travel times greater than seven (7) minutes for all routes identified in the data collection plan.
Speed	Modeled average link speeds to be within the ±10 mph of field-measured speeds on at least 85% of all network links.
Intersection Delay	Simulated and field-measured link delay times to be within 15% for more than 85% of cases.
Queue Length	Difference between simulated and observed queue lengths to be within 20%.
Visualization	Check consistency with field conditions of the following: on- and off-ramp queuing; weaving maneuvers; patterns and extent of queue at intersection and congested links; lane utilization/choice; location of bottlenecks; etc.
	Verify no unrealistic U-turns or vehicle exiting and reentering the network.

*GEH is an empirical formula expressed as $\sqrt{2 * (M - C)^2/(M + C)}$ where M is the simulation model volume and C is the field counted volume.

Figure 6: Classical Model Calibration Targets

Source: Florida (2014) Table 7-7

- These should only be considered as guidance. The model developer and reviewing agency should determine which of these are most relevant for current project, determine the appropriateness of stated accuracy, as well as determine if other calibration targets should be considered.
- At a minimum, calibration efforts should include traffic volumes, speeds, and travel times.

Base or existing conditions model(s) calibration

- Model calibration should be undertaken for the base or existing condition model(s).
- Once the existing conditions model is calibrated all build or future scenario models should use the same calibration as the existing conditions. While some arguments (such as drivers will be more aggressive in the future) may occasionally be made to change calibration parameters, such adjustments should be avoided as it becomes difficult to distinguish if changes in performance metrics are due to build conditions or changes in the calibration.
- Often several base models will be developed, such as: peak period, off-peak, weekend, etc. Each of these models must be validated; however, they will generally utilize the same calibration across all models. Typically, the peak period calibration will be applied to the off-peak and weekend. Detailed justification is needed for altering calibration between no-build models.
- The analyst or agency may wish to consider varying environmental, seasonal, or other conditions. For instance, it may be desired to calibrate a separate model for inclement

conditions, for in-season and out-of-season in a high tourist area, nighttime vs daylight (or glare conditions at dusk), weekend (majority non-familiar drivers) vs weekday (majority familiar drivers), event traffic (i.e., the driving population near 50,000-seat stadium may have very different characteristics on game days than non-game days), etc. These varied conditions can merit unique calibration efforts and separate analyze. However, each calibration will require data for the given conditions. The impact of varying conditions should not just be "assumed". The decision to calibrate separate models for different conditions should be determined prior to model development. The FHWA Traffic Analysis Toolbox Part III, Guidelines for Applying Traffic Microsimulation Modeling Software (Wunderlick et. al, 2019) provides extensive discussion and recommendations for these calibration issues.

- As discussed previously queues should not spillback off of the network. Where this occurs the model area may need to be adjusted. However, it is recognized that in some situations it may not be possible to sufficiently expand the model. In such situations any undertaken analysis and **the interpretation of the model performance metrics** must account for the inability to capture the entire congested area.

Calibration method.

- Numerous calibration methods exist, from entirely manual to full blown genetic algorithms. While a method is not recommended here it is stressed that it is critical that the model developer understand the approach selected, i.e., that it is not a "black box".
- Model calibration starts with Vissim default parameters. Where parameters are changed the modeler should provide a justification for the change.
- The adjustments to any underlying parameters should be intuitively reasonable and not a result of a random combination of parameters.
- Calibration should be based on replicate trials as discussed earlier in this module. It is not reasonable to assume that a model successfully calibrated for a single random seed is appropriate under multiple random seeds.
- A stepwise approach through performance measures is often an efficient calibration technique. For instance, Mai et al. (2011) focuses calibration efforts first on volume density, then speed, then travel time, queuing, weaving, and lane utilization. Attempting to calibrate multiple performance metrics simultaneously, particularly if utilizing a manual method, can lead to bouncing between solutions.

Critical calibration locations

- Much of a calibration effort will focus on a combination of network sections and critical locations. These locations should be identified prior to model development.
- Critical locations include major intersections, freeway interchanges, etc. Critical locations should include any bottlenecks within the model. **At a minimum, bottlenecks should be calibrated to reflect field measured capacity**.

- Network sections typically include corridor travel times and throughput, critical routes through the corridor, etc. Network wide measures may also be incorporated, such as the link conditions found in *Figure 6* above.
- Calibration targets, such as those found in *Figure 6* should be clearly identified prior to the calibration of the model.

Calibration Data

- To the greatest extent possible calibration should utilized data from the existing site.
- For example, when validating travel time site data must be used. Similarly, volume counts and queue length data should be drawn from the site.
- In some instances, it may be acceptable to use regional data. For example, the capacity of a ramp junction may be based on the known capacity of similar sites.

Model animation and Graphical Representations Performance metrics

- Viewing the model animation and the development of graphic representations of the model performance are critical to any calibration effort.
- Scatter plots, heat maps of speed over time, travel time histograms over time, graphical displays of queuing, etc. quickly provide insight into the reasonableness of a model's operations. *Figure 7* and *Figure 8* show example field versus simulation speed comparisons. Similar graphics could be created for travel time, queues, etc., allowing for a visual confirmation of the model accuracy.

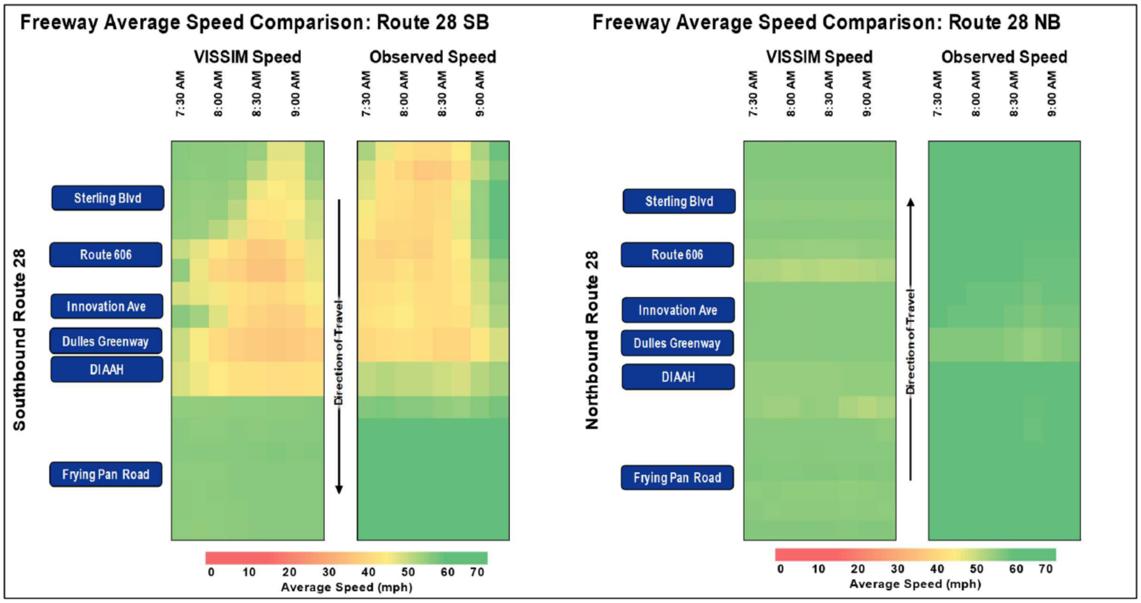


Figure 7: Example Comparison of Field Speed (INRIX) and Simulated Speeds.
Source (Punty, et al.,, 2019)

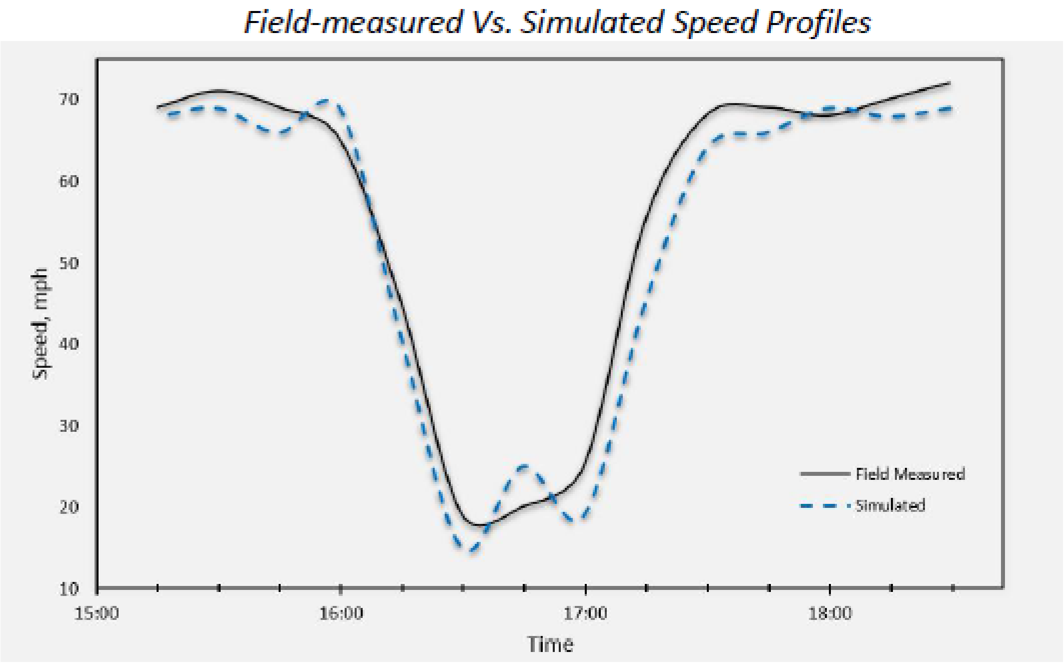
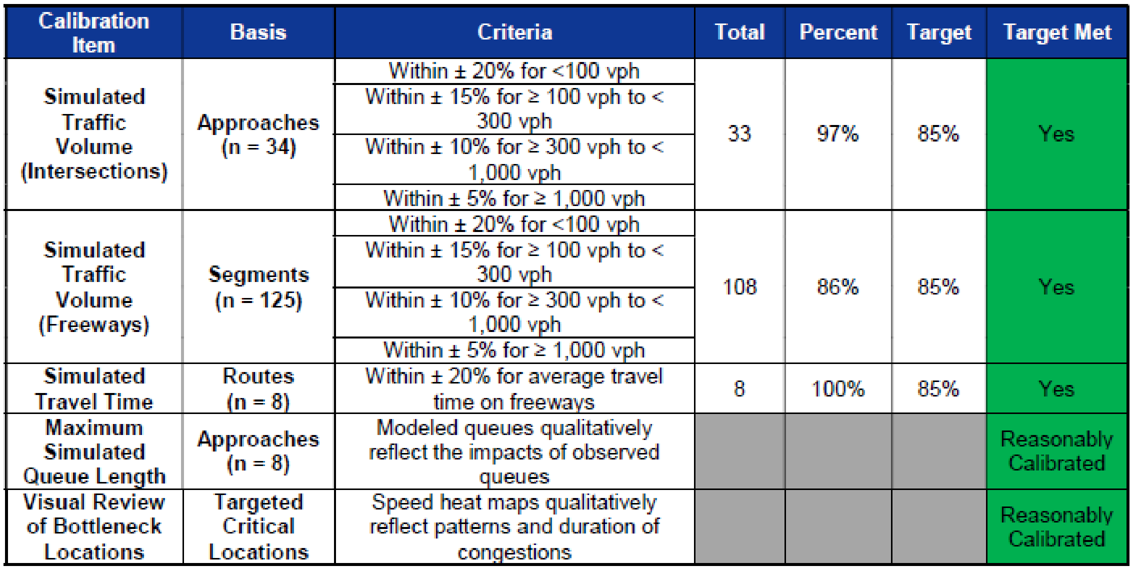


Figure 8: Field vs Simulation Speed Profile
(Source: Florida 2014)

Statistical Analysis

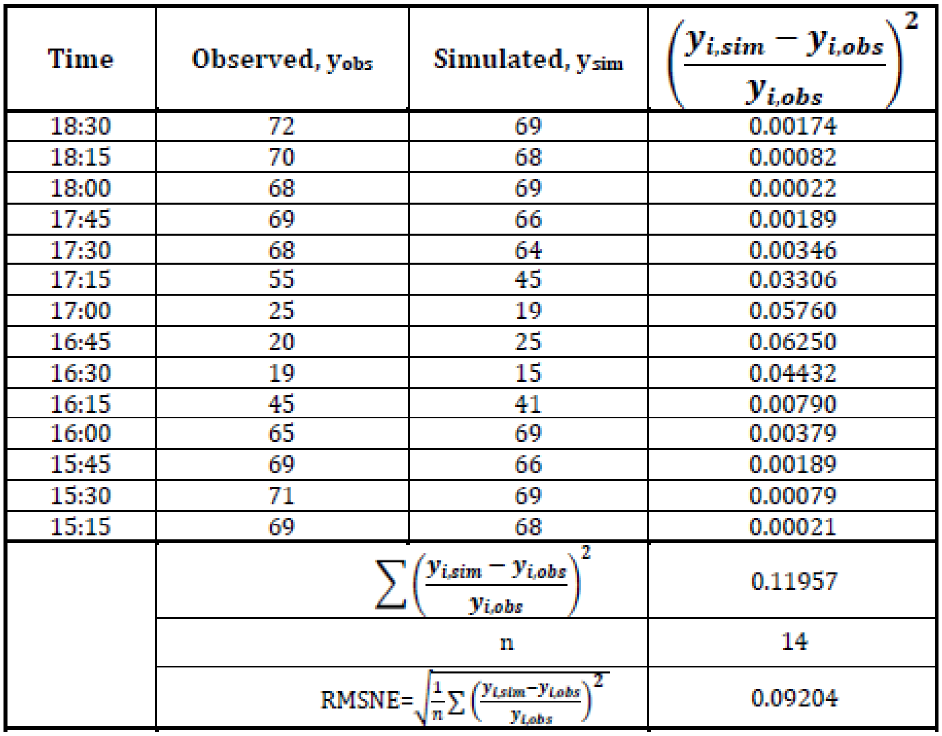
- After the use of a graphical method, a statistical analysis of the calibration should be provided.
- Analysis may be based on percentage criteria as found in *Figure 6* or more formalized goodness of fit tests.
- Typical goodness of fit methods include Root Mean Square Normalized Error (RMSNE), Correlation Coefficient (CC), Mean Absolute Percentage Error (MAPE), and GEH. For additional detail on these methods please see Florida (2014), Mai et al. (2011), Law (2013), or any of the many guides on statistical tests.
- *Figure 9* provides an example summary of calibration results based on guidelines similar to those found in *Figure 6*. *Figure 10* provides example statistical results (RMSNE).
- Even when using guidelines similar as those found in *Figure 6*, users and reviewers must confirm that locations that fail to meet the guidance will not significantly impact model results. For example, if the speeds meet or exceed the required criteria at 85% of the links but one of the links not satisfied is a critical bottleneck ramp junction additional calibration may be necessary.



Calibration Item	Basis	Criteria	Total	Percent	Target	Target Met
Simulated Traffic Volume (Intersections)	Approaches (n = 34)	Within ± 20% for <100 vph	33	97%	85%	Yes
		Within ± 15% for ≥ 100 vph to < 300 vph				
		Within ± 10% for ≥ 300 vph to < 1,000 vph				
		Within ± 5% for ≥ 1,000 vph				
Simulated Traffic Volume (Freeways)	Segments (n = 125)	Within ± 20% for <100 vph	108	86%	85%	Yes
		Within ± 15% for ≥ 100 vph to < 300 vph				
		Within ± 10% for ≥ 300 vph to < 1,000 vph				
		Within ± 5% for ≥ 1,000 vph				
Simulated Travel Time	Routes (n = 8)	Within ± 20% for average travel time on freeways	8	100%	85%	Yes
Maximum Simulated Queue Length	Approaches (n = 8)	Modeled queues qualitatively reflect the impacts of observed queues				Reasonably Calibrated
Visual Review of Bottleneck Locations	Targeted Critical Locations	Speed heat maps qualitatively reflect patterns and duration of congestions				Reasonably Calibrated

Figure 9: Example Summary of Calibration Results.

Source (Punty, et al.,, 2019)



Time	Observed, y_{obs}	Simulated, y_{sim}	$\left(\frac{y_{i,sim} - y_{i,obs}}{y_{i,obs}}\right)^2$
18:30	72	69	0.00174
18:15	70	68	0.00082
18:00	68	69	0.00022
17:45	69	66	0.00189
17:30	68	64	0.00346
17:15	55	45	0.03306
17:00	25	19	0.05760
16:45	20	25	0.06250
16:30	19	15	0.04432
16:15	45	41	0.00790
16:00	65	69	0.00379
15:45	69	66	0.00189
15:30	71	69	0.00079
15:15	69	68	0.00021
		$\sum\left(\frac{y_{i,sim} - y_{i,obs}}{y_{i,obs}}\right)^2$	0.11957
		n	14
		RMSNE=$\sqrt{\frac{1}{n}\sum\left(\frac{y_{i,sim}-y_{i,obs}}{y_{i,obs}}\right)^2}$	0.09204

Figure 10: Field vs Simulation Statistical Analysis

(Source: Florida 2014)

- Statistical tests provide a quantifiable measure of the calibration. Often a statistical test is treated as a numerical goal for calibration. However, regardless of the statistical test undertaken it is critical to utilize visual inspection as the foundation of the assessment of the calibration accuracy. Should a statistical test show a calibration as "passed" but the graphical check raises concern, it is the graphical check that should drive the continuation of the calibration. (See aside at the end of this section for additional comments.)

Sensitivity analysis.

- Model calibration may be improved through the use of a sensitivity analysis that covers uncongested and congested conditions.
- Certain model elements tend to influence stable flow operations while others have higher influence on unstable flow conditions. Low volume testing highlights issues regarding free flow speed, high speed weaves, actuated signal control, etc. High demand testing highlights issues regarding queue blockage (spillback out of turn bays, etc.), merge behavior under congested conditions, bottleneck capacity, etc.
- Regardless of the decision to utilize sensitivity testing bottlenecks are locations that should receive additional attention during calibration if future conditions may experience congested or near congested conditions. Calibration should include increasing the demand through these areas to ensure adequate capacity calibration.

Driver Behavior Calibration

- With Vissim, traffic flow (i.e., *Wiedemann* models) and lane change calibration parameters as are specific to a driver behavior. Thus, different roadway sections may have unique calibrations. This was seen in the creation of a Freeway Taper driver behavior that was different from the default Freeway driver behavior in Module 5. Such facility specific calibration can be highly useful but the calibration of each driver behavior type must be documented.

Flexibility in Calibration

- While calibration objectives, targets, critical ranges, etc. should be determined prior to the calibration effort some flexibility must be recognized.
- In some circumstance it may prove difficult or impossible to satisfy proposed calibration conditions. In such circumstances the model developer and reviewing agency should consider the potential impact of the higher deviations from the field on results and recommendations. It should be determined if this is acceptable or if additional effort, or possibly a different analysis approach, is required. For instance, while a statement of absolute LOS may not be possible, will the model allow for a reasonable estimate of directionality of traffic performance changes in the build conditions. That is, will proposed changes improve or worsen traffic conditions.

Documentation

- Any calibration effort should include detailed documents of the calibration.
- A documentation check list is provided in Module 8.

Aside

Often people are uncomfortable with the use of the graphical methods over the statistical analysis, or question why even bother use statistical analysis. Some explanation is merited. While statistical analysis is highly useful it often measures only a single aspect of a performance measure, typically the mean or variance. While it important that the mean value in the simulation match the field, this is not sufficient to state a model is calibrated. For example, it is possible for the vehicles in the field and those in the simulation to have very different travel time experiences, yet exhibit the same mean. Consider a corridor where in the field one group of vehicles has low travel times as they fall within the green band, while a second group of vehicles have much higher travel times. However, in the simulation, most of the vehicles have the same travel time. The field and the model are actually quite different, although have the same mean. A scatter plot of travel times will quickly show this difference. Thus, the graphical method shows the experience across the population. Once we are satisfied with the general experience we can then use the statistical test to determine if we are "close enough" to the average.

(It is readily acknowledged that this is a loose intuitive discussion. Significant important details are skipped, such as the underlying distributional assumptions of most parametric statistical tests would mean the above application is incorrect. However, unfortunately, most model end users are not well versed in the ins-and-outs of parametric and non-parametric tests and such violations are common place. The use of a graphical evaluation will help those without an in-depth statistical background avoid such errors. For the reader interested in simulation statistical analysis I highly recommend Law (2013).)

Driver behavior parameters Car-Following and Lane Change parameters

As part of a calibration effort it may be necessary to adjust the underlying driver behavior. Driver behavior is generally considered in two parts, car-following models and lane changing.

Car-Following

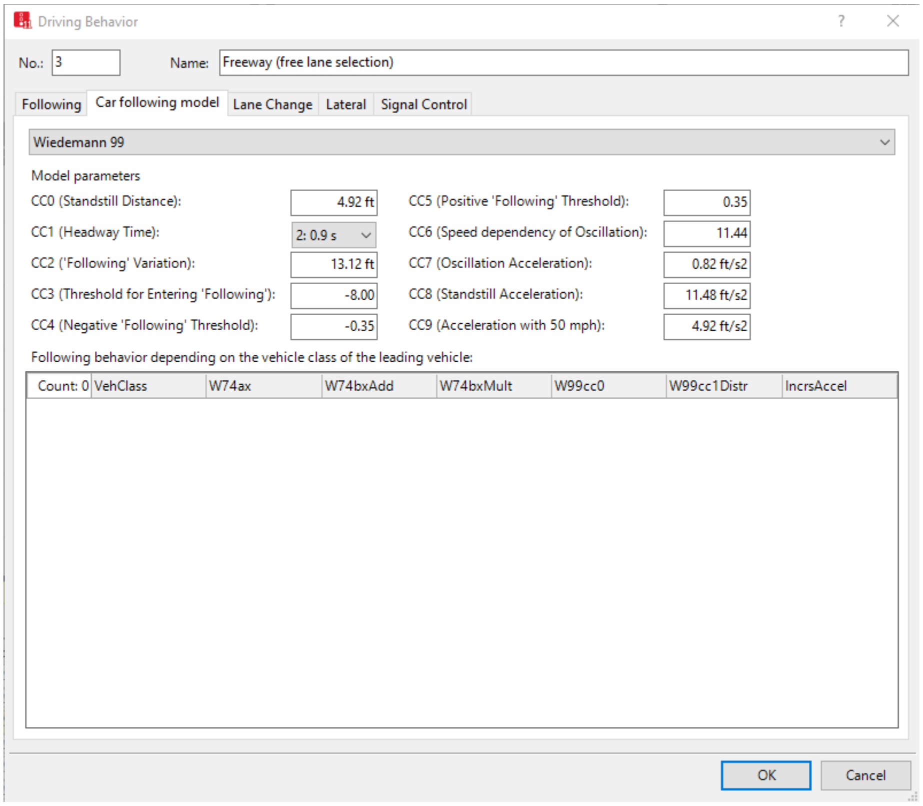
Vissim utilizes a psycho-physical perception models to capture car-following behavior. These are the *Weideman 74* and *Wiedemann 99* models. The Wiedemann models categorize the car-following behavior of a vehicle into different states. These states are: 1) free driving, where the following vehicle behavior is not influenced by the lead vehicle (i.e., the lead vehicle, if any, is sufficiently far away that the following vehicle ignores it); 2) approaching, where a faster following vehicle must decelerate and adapt its speed to a slower leading vehicle; 3) following, where the following vehicle maintains a safe distance from the leading vehicle (allowing for some variability in the following behavior); and 4) braking, where the following vehicle decelerates to maintain the safe following distance, typically in response to the lead vehicle slowing or stopping (PTV Vissim, 2019). Additional detail may be found in the Vissim Manual (PTV Vissim, 2019).

As general guidance *Wiedemann 74* is used for lower speed roadway sections and *Wiedemann 99* is utilized for higher speed roadway sections. In practice, *Wiedemann 74* is typically applied to signalized and low speed corridors, while *Wiedemann 99* is utilized for freeways. However, the application of Wiedemann models by facility type is not a hard and fast rule. The expected speeds should guide which model is selected. For instance, when modeling on-ramp junctions, if the general condition being modeled is congested than the *Wiedemann 74* model may be preferred over the *Wiedemann 99*, even though it is a freeway facility.

The Wiedemann models directly influence a roadway segment capacity, saturation flow, density, etc. Thus, it may be necessary to calibrate the model parameters, to reflect field conditions. However, as stated previously, prior to calibrating the Wiedemann parameters a thorough calibration should be made of other location specific model elements.

Wiedemann 99

The parameters for the Wiedemann models for a driving behavior may be found at *Base Data → Driving Behaviors*. Then select the driving behavior of interest and go to the *Car following model* tab. *Figure 11* show the default parameters for the *Freeway (free lane selection)* Vissim *driving behavior*. The utilized model is the *Wiedemann 99*. The *Wiedemann 99* model has ten parameters, *CC0* to *CC9*. However, most calibration efforts will focus on the first three: *CC0, CC1,* and *CC2*.



No.: 3

Name: Freeway (free lane selection)

Wiedemann 99

Model parameters

CC0 (Standstill Distance):	4.92 ft	CC5 (Positive 'Following' Threshold):	0.35
CC1 (Headway Time):	2: 0.9 s	CC6 (Speed dependency of Oscillation):	11.44
CC2 ('Following' Variation):	13.12 ft	CC7 (Oscillation Acceleration):	0.82 ft/s2
CC3 (Threshold for Entering 'Following'):	-8.00	CC8 (Standstill Acceleration):	11.48 ft/s2
CC4 (Negative 'Following' Threshold):	-0.35	CC9 (Acceleration with 50 mph):	4.92 ft/s2

Following behavior depending on the vehicle class of the leading vehicle:

Count: 0	VehClass	W74ax	W74bxAdd	W74bxMult	W99cc0	W99cc1Distr	IncrsAccel

Figure 11: Driving Behavior, Wiedemann 99 Default Parameter Values

Prior to discussing *CC0 (Standstill Distance)* and *CC1 (Following Distance)* we define the *Desired Safety Distance* between vehicles, where:

$$\textit{Desired Safety Distance} = \textit{CC0} + (\textit{CC1} \text{ x speed})$$

The *Desired Safety Distance* represents the gap that the lagging vehicle will seek to maintain from the leading vehicle. The gap is measured from the front of the lagging vehicle to the rear of the leading vehicle. Notice, this is different from the typical definition of headway, which is from the front of the lagging vehicle to the front of the leading vehicle. Be careful that you do not use field headway measurements to calibrate *Desired Safety Distance* without accounting for the lead vehicle length! (Unfortunately, the Vissim interface uses the term "headway" for *CC1*, really *CC1* is only part of headway. As just defined headway (in distance, defined from the same position of the lagging and leading vehicles) would be *CC0* + (*CC1* x Speed (f/s)) + Length of Lead Vehicle.

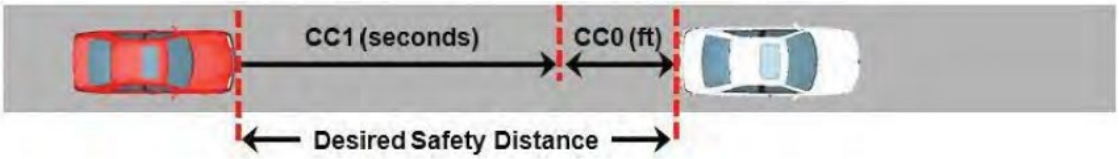


Figure 12: Desired Safety Distance Components, CC0 + CC1 x Speed
Figure Source: Schiloperoot et al., 2014

CC0 - Standstill distance. *CC0* is the desired distance that will exist between vehicles when stopped on a link. *Standstill distance* is a fixed value, having no stochastic variation. This parameter has several primary impacts: 1) it will directly impact queue length, i.e., length in terms of distance, not the number of vehicles; 2) it will directly influence density, particularly at lower speeds where *Standstill distance* is a higher fraction of *the Desired Safety Distance* between vehicles, i.e., under stopped conditions (queues or jam density) the *Desired Safety Distance* equals the *Standstill Distance*; 3) it will influence capacity as *Standstill distance* is a portion of the *Desired Safety Distance* between vehicles, thus a higher *Standstill distance* may reduce capacity; and 4) it may influence merging as it effects the distance between vehicles.

CC1 - Following distance. The following distance is the speed dependent portion of the *Desired Safety Distance* and can have a significant influence on capacity. As such it is one of the most commonly calibrated parameters. For instance, Figure 13 is the capacity at the south bound ramp junction, using the alternative design, for our example network from the prior modules. Three *CC1* values are tested, 0.5, 0.9, and 1.5, with all other parameters at their Vissim default value. The results for each *CC1* value is based on three replicate trials, with each data point representing a 5-minute flow rate. Each three-hour run varied the traffic volume from under-capacity, to over-capacity, and then returned to under-capacity. As seen in Figure 13, *CC1* values may significantly influence capacity, with lower values CC1 resulting in higher capacities. This is as expected, as lower values result in reduced spacing between vehicles.

While not shown, *CC0* may be expected to demonstrated similar capacity trends, although to a much lesser extent. At capacity speeds (i.e. 30 to 50 mph) a change in *CC0* will represent a relatively small change in the headway. Thus, *CC1* is most commonly calibrated when considering capacity.

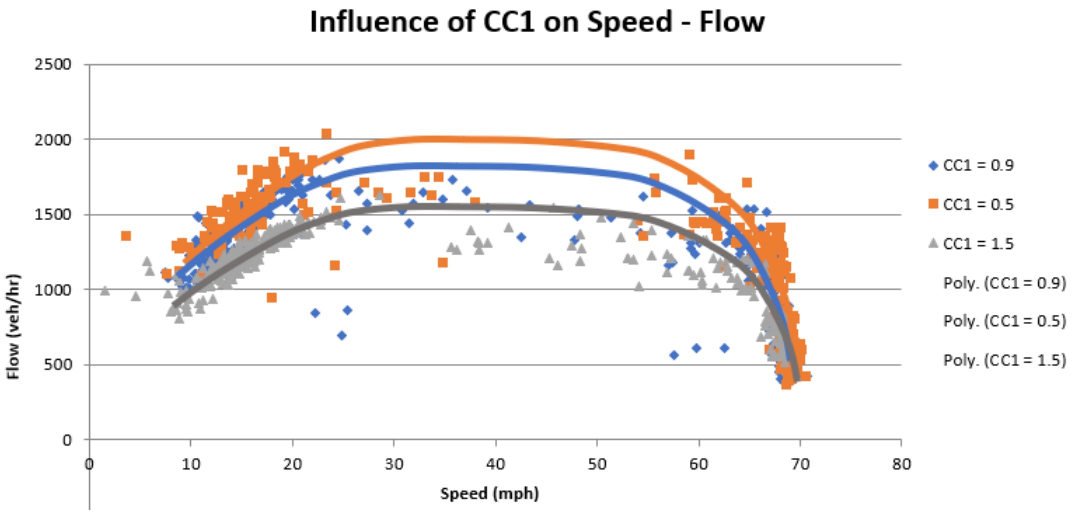
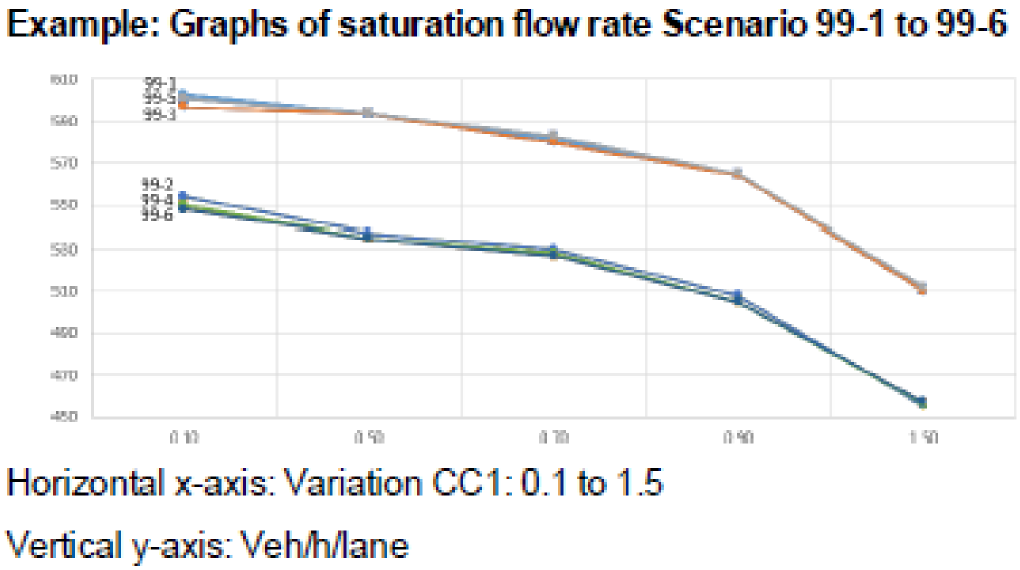


Figure 13: Influence of *CC1* on Speed – Flow, Capacity.

Figure 14 shows the impact of *CC1* on saturation flow for several speed, percent truck, and lane change rule alternatives (additional details may be found in PTV Vissim, 2019). In all cases higher *CC1* values decrease saturation flow. However, this graph also highlights that other factors, in this instance speed and percent trucks, are critical as well.


Example: Graphs of saturation flow rate Scenario 99-1 to 99-6

Horizontal x-axis: Variation CC1: 0.1 to 1.5

Vertical y-axis: Veh/h/lane

Figure 14: CC1 Impact on Saturation Flow
Source: PTV Vissim, 2019

Figure 15 explores the impact of *CC1* on travel time for the same example as Figure 13. *Figure 15* shows the travel time for each vehicle along the SB section of the Freeway. Even though all scenarios utilized the same input volume there is a clear increase in the length of the overcapacity period as witnessed by the high travel times. This is because the higher capacity of the lower *CC1* values is able to more quickly serve the unmet demand during the over-capacity conditions. We see that *CC1* may also have a significant impact on the length of a congested period. For the *CC1* value of 1.5 we can see that the model did not yet return to under-capacity conditions, thus a longer simulation run should be undertaken. (Also, this is a nice example of why the analysis period for comparison of scenarios must be set to capture the congestion of the worst-case scenario.)

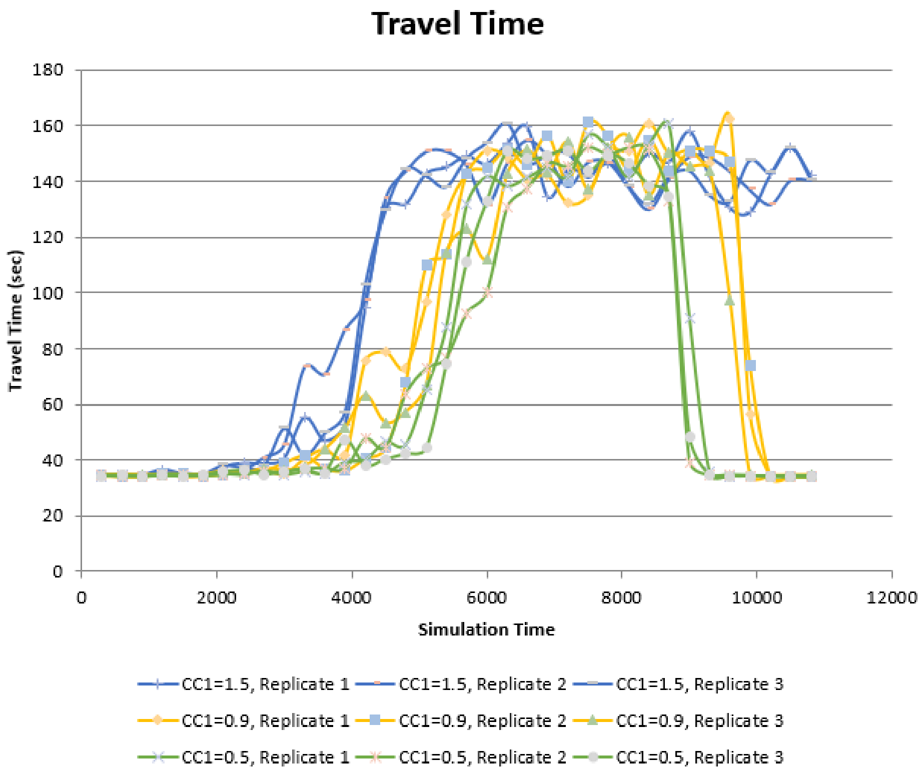


Figure 15: CC1 Influence on Travel Time

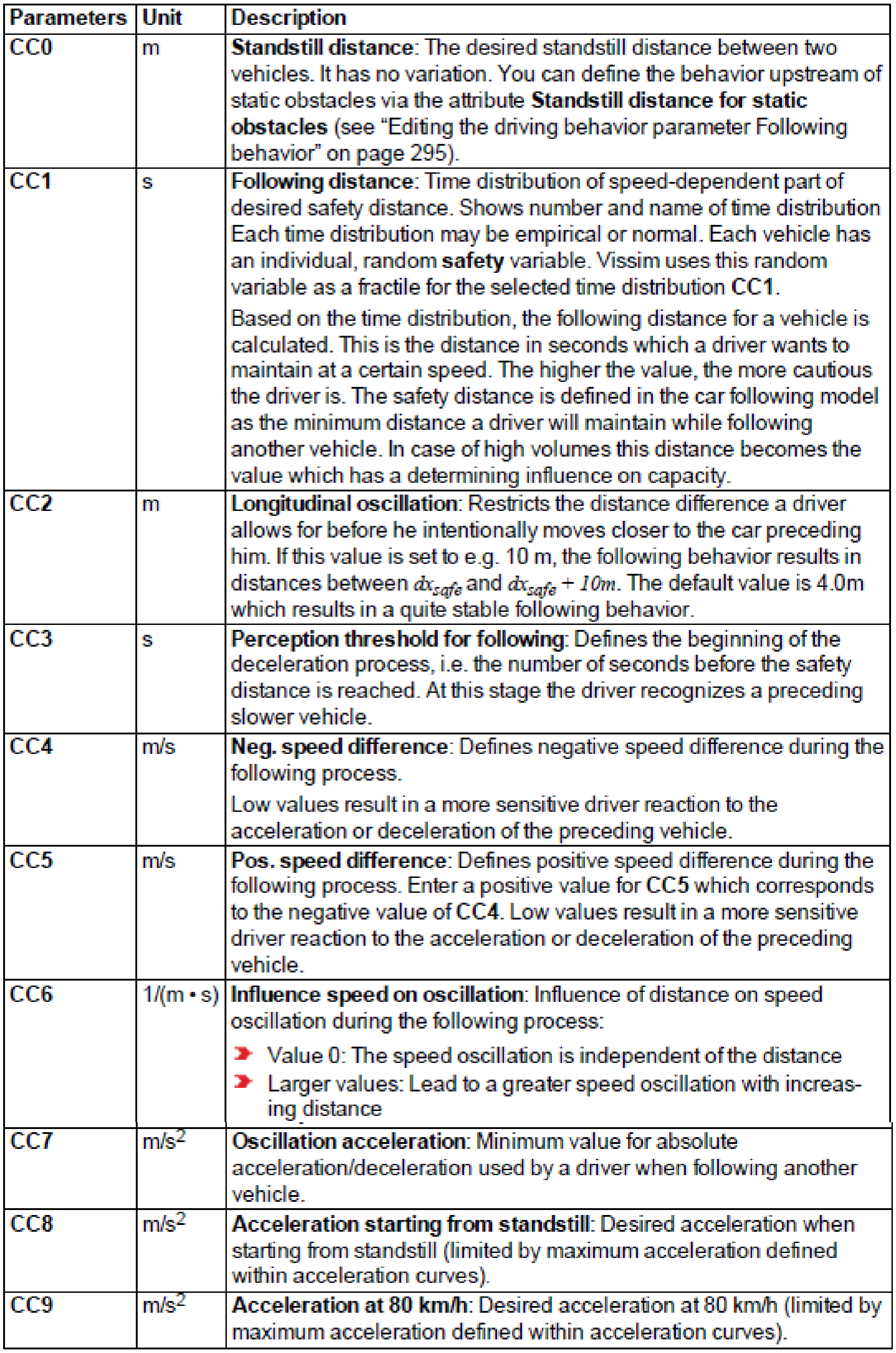
Aside

There is a puzzling aspect to the travel time results in Figure 15. We see that during congestion each *CC1* value results in approximately the same travel time. However, we have shown that the lower *CC1* values have a higher capacity; therefore, the queues created by the unmet demand should be shorter for the lower *CC1* values, resulting in lower travel times. Why is this not the case?? (Think about this before reading the discussion in the next paragraph!)

Why? In the model the traffic queue reached the beginning of the SB freeway lanes and traffic was queued off the model. The travel time only reflects the time to travel from the start of the SB lanes to past the ramp. It does not capture the full extent of the queuing! This is a good example of utilizing various graphical representation of performance metrics as part of a review or debugging process. In this case, the model should be extended to capture the maximum length of queue for *CC1* = 1.5, and the run duration lengthened to ensure the model returns to uncongested conditions, then all scenarios should be re-run.

CC2 – Longitudinal oscillation. In modeling car-following behavior Vissim seeks to improve realism by allow some oscillation in the car-following distance. This oscillation is capped by *CC2*. If the space between the lead and the following vehicle exceeds the *Desired Safety Distance* + *CC2* then following vehicle will intentionally move closer. Shorter *CC2* values may increase capacity. It is noted in the Vissim manual that a value of 4 meters has been found to provide stable flow.

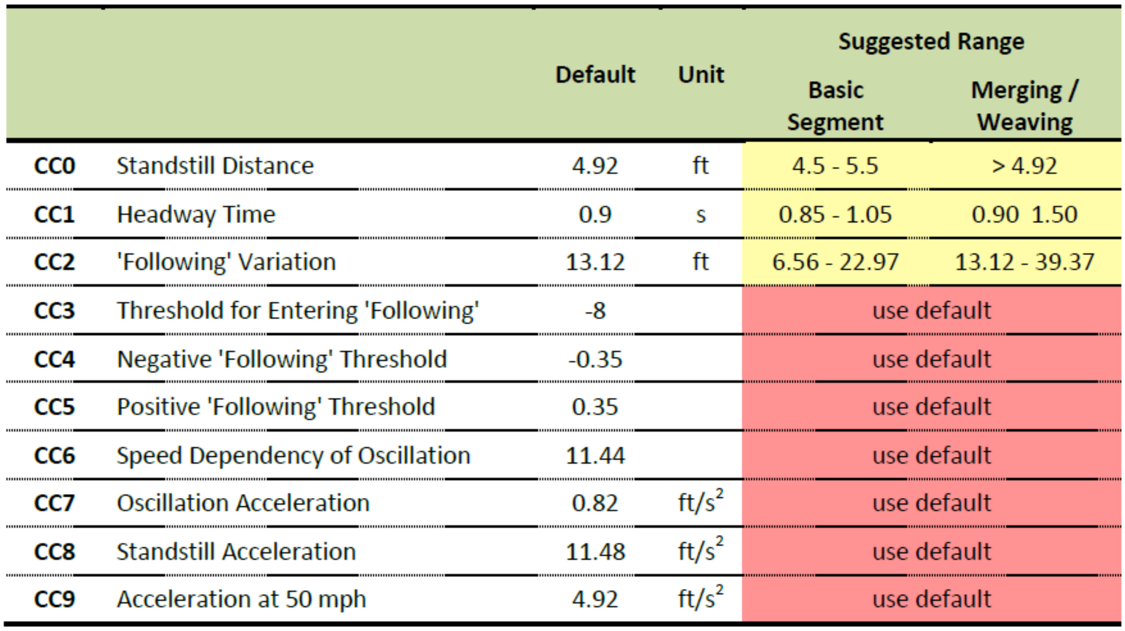
CC3 through *CC9* tend to have minimal impact on car-following behavior and performance metrics and are commonly not included in calibration. Where a developer deems that calibration of these parameters is needed it may be acceptable. However, calibration parameters should not be based on a solely “black box” approach, updated values should be justifiable based on the purpose of the parameter. *Figure 16* below from the Vissim manual provides the definition for each of the remaining parameters.



Parameters	Unit	Description
CC0	m	**Standstill distance**: The desired standstill distance between two vehicles. It has no variation. You can define the behavior upstream of static obstacles via the attribute **Standstill distance for static obstacles** (see "Editing the driving behavior parameter Following behavior" on page 295).
CC1	s	**Following distance**: Time distribution of speed-dependent part of desired safety distance. Shows number and name of time distribution Each time distribution may be empirical or normal. Each vehicle has an individual, random **safety** variable. Vissim uses this random variable as a fractile for the selected time distribution **CC1**. Based on the time distribution, the following distance for a vehicle is calculated. This is the distance in seconds which a driver wants to maintain at a certain speed. The higher the value, the more cautious the driver is. The safety distance is defined in the car following model as the minimum distance a driver will maintain while following another vehicle. In case of high volumes this distance becomes the value which has a determining influence on capacity.
CC2	m	**Longitudinal oscillation**: Restricts the distance difference a driver allows for before he intentionally moves closer to the car preceding him. If this value is set to e.g. 10 m, the following behavior results in distances between dx_{safe} and $dx_{safe} + 10m$. The default value is 4.0m which results in a quite stable following behavior.
CC3	s	**Perception threshold for following**: Defines the beginning of the deceleration process, i.e. the number of seconds before the safety distance is reached. At this stage the driver recognizes a preceding slower vehicle.
CC4	m/s	**Neg. speed difference**: Defines negative speed difference during the following process. Low values result in a more sensitive driver reaction to the acceleration or deceleration of the preceding vehicle.
CC5	m/s	**Pos. speed difference**: Defines positive speed difference during the following process. Enter a positive value for **CC5** which corresponds to the negative value of **CC4**. Low values result in a more sensitive driver reaction to the acceleration or deceleration of the preceding vehicle.
CC6	1/(m • s)	**Influence speed on oscillation**: Influence of distance on speed oscillation during the following process: ➤ Value 0: The speed oscillation is independent of the distance ➤ Larger values: Lead to a greater speed oscillation with increasing distance
CC7	m/s^2	**Oscillation acceleration**: Minimum value for absolute acceleration/deceleration used by a driver when following another vehicle.
CC8	m/s^2	**Acceleration starting from standstill**: Desired acceleration when starting from standstill (limited by maximum acceleration defined within acceleration curves).
CC9	m/s^2	**Acceleration at 80 km/h**: Desired acceleration at 80 km/h (limited by maximum acceleration defined within acceleration curves).

Figure 16: Wiedemann 99 model parameters and definitions.
Source: PTV Vissim, 2019

Figure 17 provides example allowable ranges for CCO, CC1, and CC2 from the Oregon Department of Transportation Protocol for Vissim Simulation. (Mai et al., 2011).

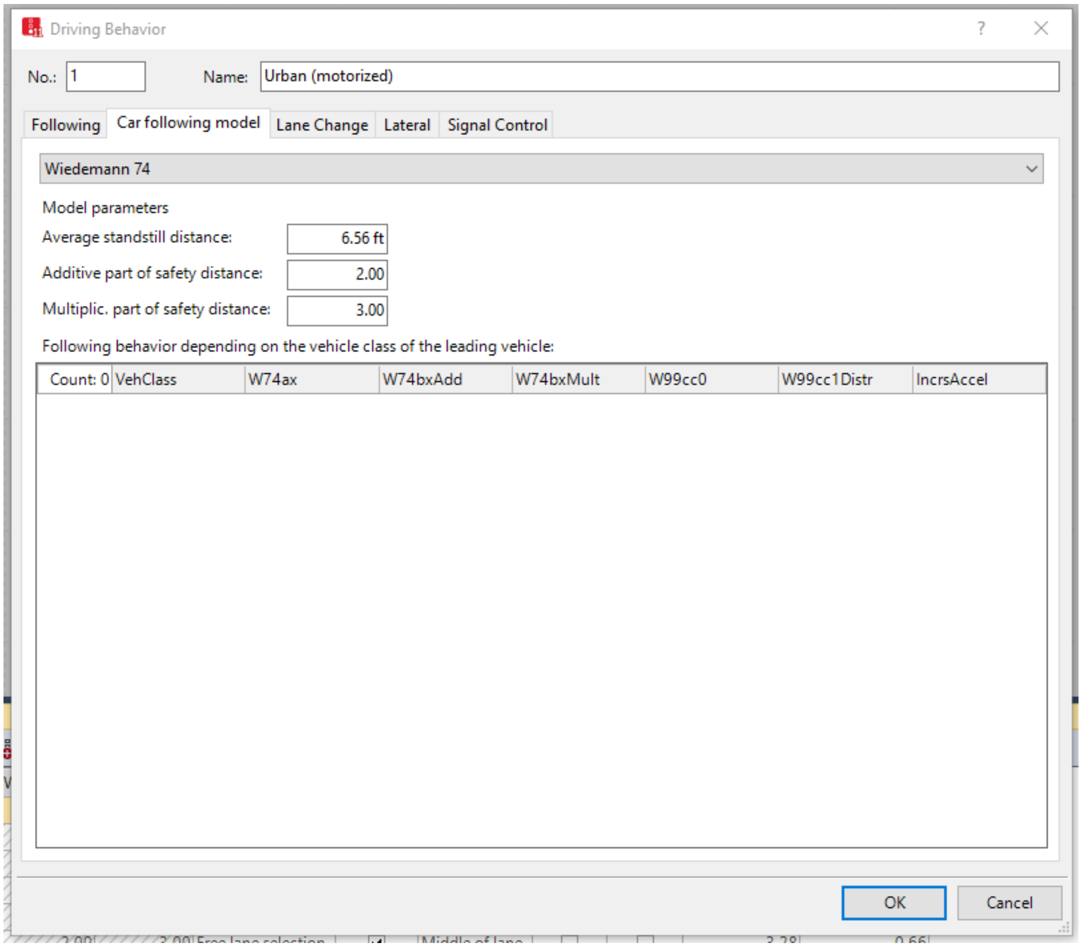


		Default	Unit	Suggested Range	
				Basic Segment	Merging / Weaving
CC0	Standstill Distance	4.92	ft	4.5 - 5.5	> 4.92
CC1	Headway Time	0.9	s	0.85 - 1.05	0.90 1.50
CC2	'Following' Variation	13.12	ft	6.56 - 22.97	13.12 - 39.37
CC3	Threshold for Entering 'Following'	-8		use default	
CC4	Negative 'Following' 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	

Figure 17: Example Wiedemann Car-Following Parameter Ranges.
Mai et al., 2011

Wiedemann 74

The *Wiedemann 74* model is primarily for lower speed roads, most commonly utilized on non-freeway corridors (i.e., arterials, collectors, etc.) and on-ramp junctions. *Wiedemann 74* is less complex than *Wiedemann 99*, consisting of three main parameters: *Average Standstill Distance*, *Additive Part of Safety Distance*, and *Multiplicative Part of Safety Distance*. As with *Wiedemann 99*, the parameters for *Wiedemann 74* may be found under *Driving Behaviors → Car following Model*, *Figure 18*.



Driving Behavior

No.: 1 Name: Urban (motorized)

Wiedemann 74

Model parameters

Average standstill distance: 6.56 ft

Additive part of safety distance: 2.00

Multiplic. part of safety distance: 3.00

Following behavior depending on the vehicle class of the leading vehicle:

Count: 0	VehClass	W74ax	W74bxAdd	W74bxMult	W99cc0	W99cc1Distr	IncrsAccel

Figure 18: Wiedemann 74 Driving Behavior Model Parameter Window

Prior to defining these three parameters we define the *Desired Safety Distance* for *Wiedemann 74* as (PTV Vissim 2019):

$$d = ax + (bx_{add} + bz_{mult} \times z) \times \sqrt{v}$$

Where:

ax is the *standstill distance*

bx_{add} is the *additive part* of *desired safety distance*

bx_{mult} is the *multiplicative part* of *desired safety distance*

v is the vehicle speed (m/s)

z ranges from 0 to 1, normally distributed with a mean of .5 and standard deviation of .15

Additive standstill distance is average distance between stopped vehicles. The default value is 2 meters, with a range of plus/minus one meter and a standard deviation of 0.3 meters. *Additive Standstill Distance* will directly impact queue lengths, measured in distance. *Additive Standstill Distance* will also influence the headway between vehicles, with a diminishing influence as speed increases.

The *additive part (bx_{add}) of desired safety distance* allows for a non-stochastic increase in the desired safety distance as the speed increase, at a rate equal to the square root of the speed. Increasing the additive part of *desired safety distance* will decrease the saturation flow.

The *multiplicative part (bx_{mult})* of *desired safety distance* introduces stochasticity into the desired safety distance across vehicles, with each vehicle having a unique random draw for the value of z. Increasing the multiplicative part of *desired safety distance* will decrease saturation flow.

Figure 19 provides an example of the impact of increasing the additive and multiplicative parts of safety distance on saturation flow. However, in addition to these parameters saturation flow in influenced by percent heavy vehicles, desired speed distribution, etc.

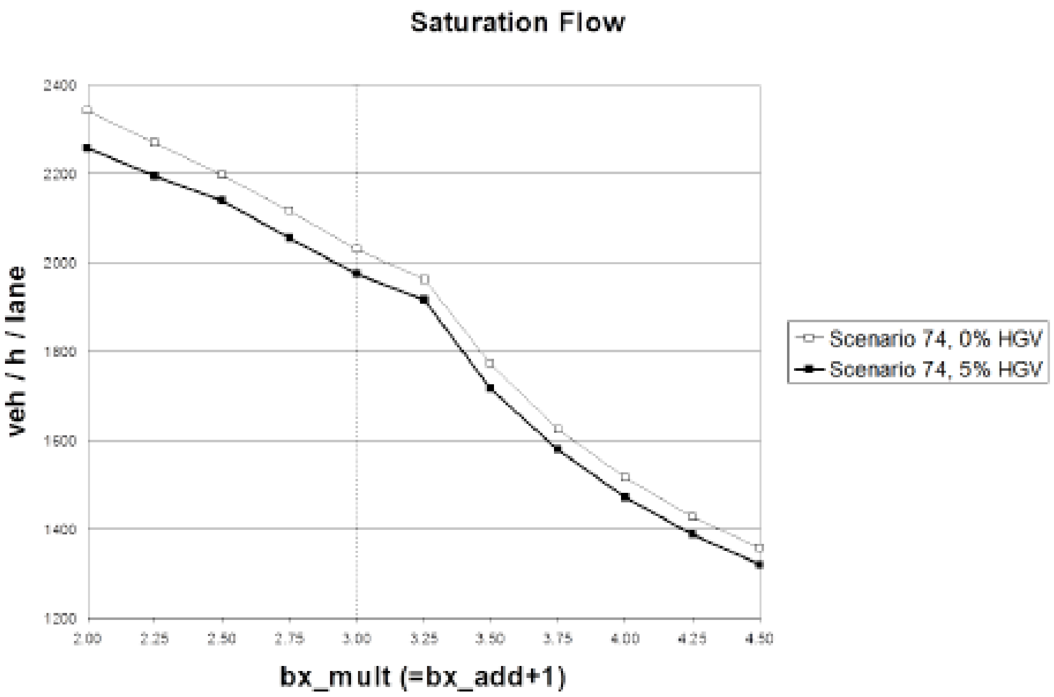
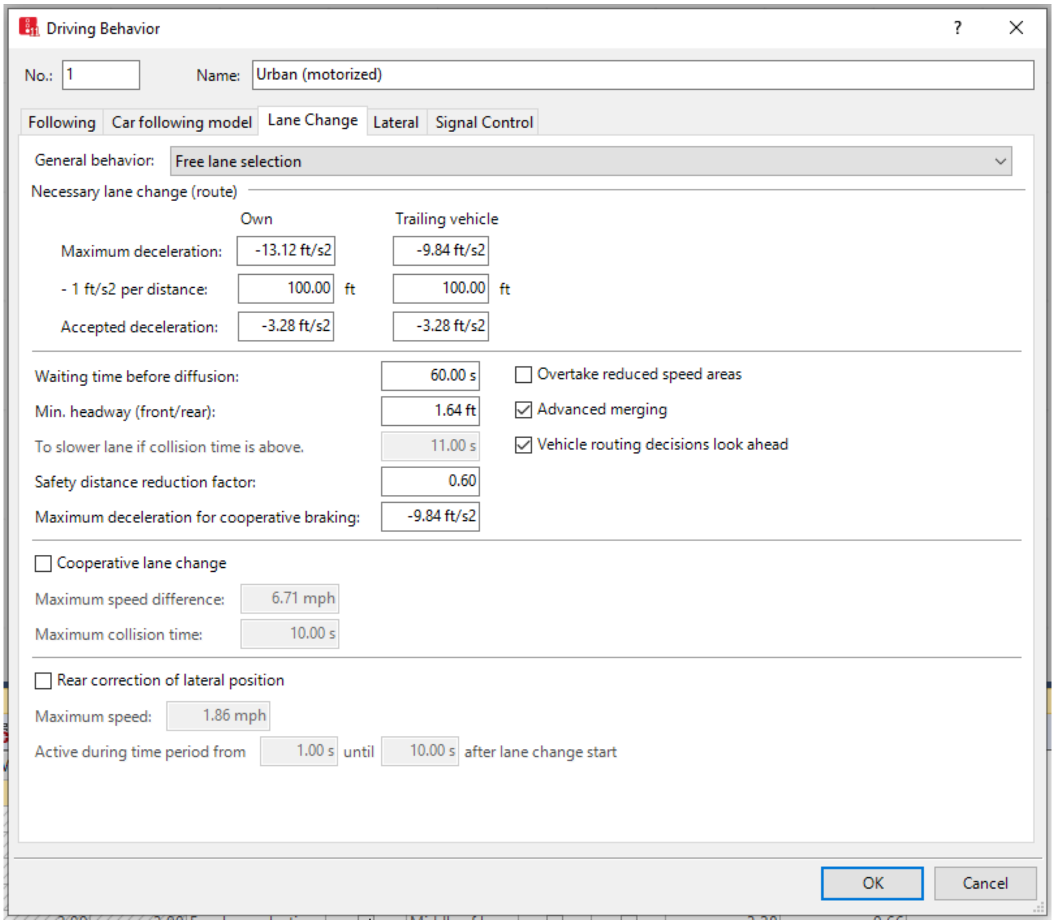


Figure 19: Influence of Additive and Multiplicative Parts of Safety Distance on Saturation Flow
Source: PTV Vissim, 2019

Lane Change Parameters

In addition to the Wiedemann car-following parameters Vissim also enables users to calibrate lane-changing parameters. The lane change parameters may be found in *Driving Behaviors → Lane Change, Figure 20.* These parameters are not discussed in detail within this module as the Vissim manual (PTV Vissim 2019, Section 5.8.5) provides a detailed table explaining each parameter. However, common parameters to calibrate are the *maximum* and *accepted deceleration*, allowing of more rapid deceleration as part of the lane change process; and *Safety Distance Reduction Factor*, allowing vehicles to merge into smaller gaps. In addition, generally all newer models will use *Advanced Merging* which allows vehicles to change lanes earlier.

In most scenarios model developers will seek to calibrate these parameters when there is a need to alter the "aggressiveness" of vehicles in merging, diverging, or weaving areas to match field conditions, either through more aggressive lane changes or increasing cooperative behavior from vehicles in adjacent lanes. Typically, the need for such calibration is identified through vehicles stopping in merge, diverge, or weave areas. For example, one challenge often faced is vehicles stopping on an on-ramp when they are not able to merge before reaching the *emergency stop distance*. It may even be seen that these vehicles are diffused, i.e., removed from the model. Calibration of lane change models is one technique to seek to address this challenge. *Figure 21* provides an example of recommended ranges for these parameters. However, it is once again stressed, that prior to calibrating *Lane Change* parameters the modeler should seek to fine tune other elements, such as *lane change distance, emergency stop distance, link* and *connector* configuration (e.g., the alternative on-ramp configuration given in Module 5), and so on. Also, as a final point, it should not always be assumed that vehicles stopping on a ramp or in a weave area is incorrect. While the behavior is often not reflective or real-world operations it certainly can be. It is important that the modeler and reviewer be familiar with the modeled area and have a sense of realistic operations.



No.: 1

Name: Urban (motorized)

General behavior: Free lane selection

Necessary lane change (route)

	Own		Trailing vehicle	
Maximum deceleration:	-13.12 ft/s2		-9.84 ft/s2	
- 1 ft/s2 per distance:	100.00	ft	100.00	ft
Accepted deceleration:	-3.28 ft/s2		-3.28 ft/s2	

Waiting time before diffusion: 60.00 s

Min. headway (front/rear): 1.64 ft

To slower lane if collision time is above. 11.00 s

Safety distance reduction factor: 0.60

Maximum deceleration for cooperative braking: -9.84 ft/s2

□ Overtake reduced speed areas

☑ Advanced merging

☑ Vehicle routing decisions look ahead

□ Cooperative lane change

Maximum speed difference: 6.71 mph

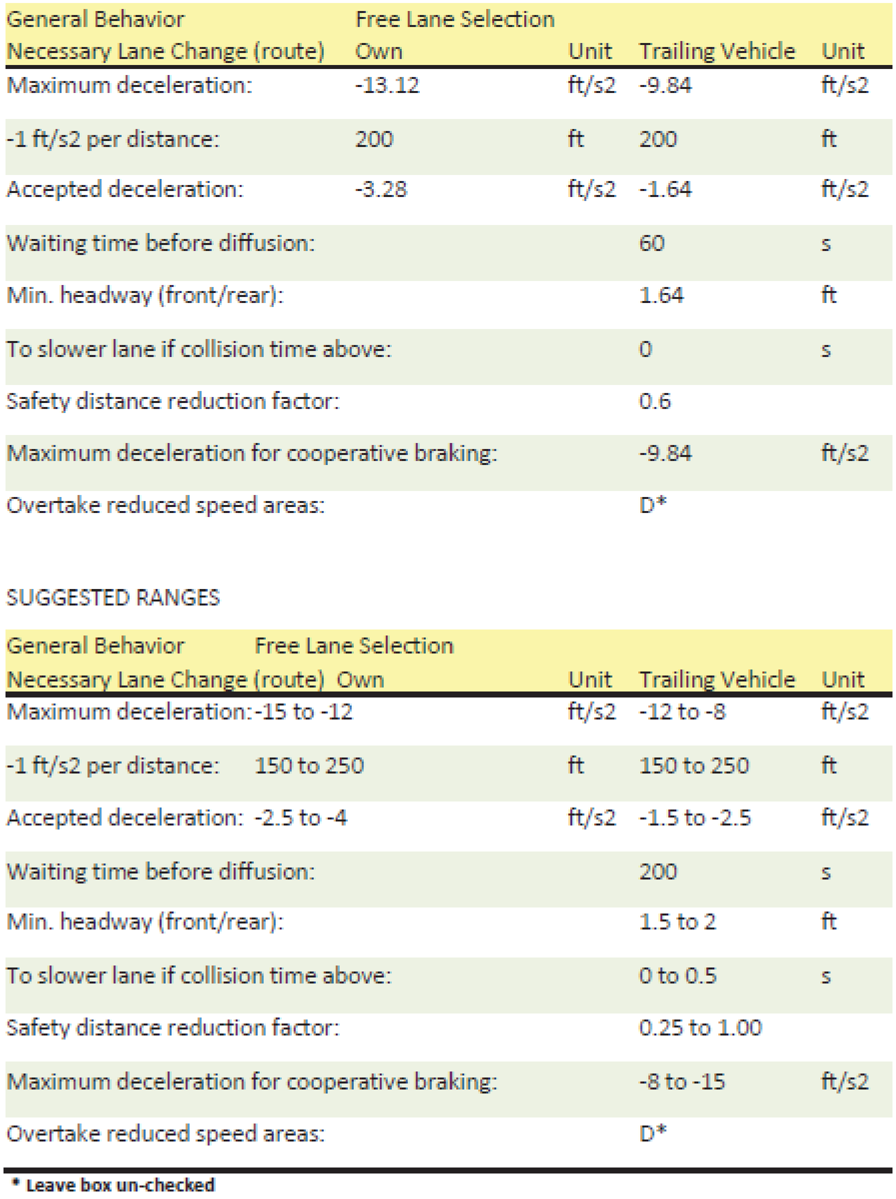
Maximum collision time: 10.00 s

□ Rear correction of lateral position

Maximum speed: 1.86 mph

Active during time period from 1.00 s until 10.00 s after lane change start

Figure 20: Driving Behavior, Lane Change Parameters



General Behavior	Free Lane Selection			
Necessary Lane Change (route)	Own	Unit	Trailing Vehicle	Unit
Maximum deceleration:	-13.12	ft/s2	-9.84	ft/s2
-1 ft/s2 per distance:	200	ft	200	ft
Accepted deceleration:	-3.28	ft/s2	-1.64	ft/s2
Waiting time before diffusion:			60	s
Min. headway (front/rear):			1.64	ft
To slower lane if collision time above:			0	s
Safety distance reduction factor:			0.6	
Maximum deceleration for cooperative braking:			-9.84	ft/s2
Overtake reduced speed areas:			D*	

SUGGESTED RANGES

General Behavior	Free Lane Selection			
Necessary Lane Change (route)	Own	Unit	Trailing Vehicle	Unit
Maximum deceleration:	-15 to -12	ft/s2	-12 to -8	ft/s2
-1 ft/s2 per distance:	150 to 250	ft	150 to 250	ft
Accepted deceleration:	-2.5 to -4	ft/s2	-1.5 to -2.5	ft/s2
Waiting time before diffusion:			200	s
Min. headway (front/rear):			1.5 to 2	ft
To slower lane if collision time above:			0 to 0.5	s
Safety distance reduction factor:			0.25 to 1.00	
Maximum deceleration for cooperative braking:			-8 to -15	ft/s2
Overtake reduced speed areas:			D*	

* **Leave box un-checked**

Figure 21: Suggested Lane Change Parameter Ranges, top figure represents Vissim defaults and bottom table Vissim suggested ranges.

Source: Schilperoot, et al., 2014

Wiedemann Parameter Simulation Exercise

Figure 13 showed the impact of *Wiedemann* Model Parameter Calibration. A detail step-by-step guidance for this test in not given within this module. However, the below provides a rough outline of what needs to be done. With the skills gained in Modules 1 through 6 a user should be able to undertake this study!

We will focus on the SB on-ramp.

step i. For this analysis we will focus on *Wiedemann 99*, for the SB on-ramp junction. Confirm that the On-ramp Freeway links are set to the *Wiedemann 99* model. (Recall, in the previous model we had set a Freeway Taper *Driving Behavior* for these links. We could also easily set the *Driver Behavior* model to *Wiedemann 74* and calibrate parameters for that model. It is up to you!)

step ii. Place *data collection points* immediately upstream, midway through the taper, and downstream of the ramp. These will allow for the measure of volume throughput (e.g., capacity) and speeds in different zones of the ramp junction. Be sure you are clearly labeling points as this will be very helpful in the later analysis.

step iii. For the data collection points set *data collection measurements*, include all individual points as well as a set for each group of mainline lanes.

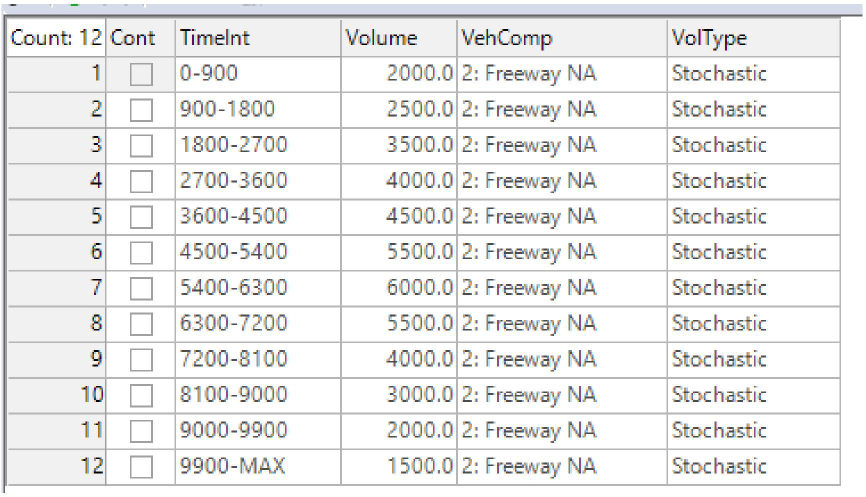
step iv. Add *travel time measurement* zones from the entrance point on the SB lanes to just after the end of the ramp junction. Similarly set a *travel time measurement* zone from the beginning of the on-ramp (just after the intersection with Buzz Blvd) to just after the end of the ramp junction.

step v. Set up *delay measurements* for travel time zones.

step vi. Set up the *Results Attributes lists* for the *Data Collection Points, Travel Time,* and *Delays*. Set the time *interval* to 5 minutes for each. At a minimum collect the default performance metrics for each.

step vii. Set the run time to 10800 seconds.

step viii. Set the Southbound vehicle input as in Figure 22.



Count: 12	Cont	TimeInt	Volume	VehComp	VolType
1	□	0-900	2000.0	2: Freeway NA	Stochastic
2	□	900-1800	2500.0	2: Freeway NA	Stochastic
3	□	1800-2700	3500.0	2: Freeway NA	Stochastic
4	□	2700-3600	4000.0	2: Freeway NA	Stochastic
5	□	3600-4500	4500.0	2: Freeway NA	Stochastic
6	□	4500-5400	5500.0	2: Freeway NA	Stochastic
7	□	5400-6300	6000.0	2: Freeway NA	Stochastic
8	□	6300-7200	5500.0	2: Freeway NA	Stochastic
9	□	7200-8100	4000.0	2: Freeway NA	Stochastic
10	□	8100-9000	3000.0	2: Freeway NA	Stochastic
11	□	9000-9900	2000.0	2: Freeway NA	Stochastic
12	□	9900-MAX	1500.0	2: Freeway NA	Stochastic

Figure 22: Southbound Vehicle Input

step ix. Run the model. Carefully review the animation for errors or unexpected behaviors. Adjust the model as needed (i.e., *Lane Change Distance*, *Emergency Stop Distance*, starting and ending points of routes, etc.)

step x. Check the warnings eliminating as many as possible.

(It is assumed you have already been through this process with the original volume set. If not, you may wish to use lower volumes, first addressing as many issues as possible. Then increase the volumes and address any new issues that arise. Fine-tuning and calibrating a model is an iterative process, take the time to work through the issues you find.)

step xi. Set the number of replicate trials to three. (You may certainly use more; however, the three replicates will provide sufficient data for understanding the process. For a published report you likely would need on the order of 10 replicates. Using three limits the amount of data you will need to analyze.)

step xii. Check the replicate run’s warning messages. Repeat steps ix and x as needed.

step xiii. You now have your test model. Pick the parameter you wish to test, for instance, *CC1*. Go to the *CC1* value and set it to the desired value, for instance 0.5.

step xiv. Save all generated *.att* file under new names that reflect the *CC1* value of 0.5. (This is critical, if you do not update the file names they will be overwritten with your next runs!)

step xv. Repeat the runs for *CC1* values of 0.9 and 1.5.

step xvi. You now have all of the data you need to explore the impact *CC1*. For instance, try to create the Speed – Flow and Travel Time graphs. (While I did this in Excel it was a bit cumbersome. I wrote scripts to automate the process. Feel free to use other data processing tools!)

CONGRATULATIONS!!!

Start-Up & Replicate Trials
And
Verification, Validation, & Calibration

Way to go!!!

https://commons.wikimedia.org/wiki/File:Mr._Smiley_Face.svg#file

References

Florida Department of Transportation, Traffic Analysis Handbook, A Reference for Planning and Operations. Systems Planning Office. March 2014

Gomes, G., A. May, and R. Horowitz. Calibration of VISSIM for a Congested Freeway. California PATH Research Report. California Path Program, Institute of Transportation Studies, University of California, Berkeley. UCB-ITS-PRR-2004-4. ISSN 1055-1425. March 2004

Law, A. Simulation Modeling and Analysis, 5th Edition. McGraw Hill Education. Tucson, Arizona, USA. ISBN 978-0-07-340132-4. 2013

Mai, C., C. McDaniel-Wilson, D. Norval, D. Upton, J, Auth, P. Schuytema, S. Abbott, R. Delahanty, C. Maciejewski, M. Wobken, M. Wells, X. Zhang, T. Bauer, J. Dale, K. Giese, and J. Won. Protocol for Vissim Simulation. Oregon Department of Transportation. June 2011

Maryland Department of Transportation, State Highway Administration. Vissim Modeling Guidance, August 2017.

Park, B. and J. Won, Microscopic Simulation Model Calibration and Validation Handbook. Traffic Operation Laboratory, Center for Transportation Studies, University of Virginia. Virginia Transportation Research Council Report VTRC 07-CR6. October 2006

Prunty, R., K. Boddu, A. Gallo, and, B. Hammit. Route 28 and Dulles Toll Road/Dulles Greenway Study. Traffic Analysis Microsimulation Calibration Results Memorandum. Kimley-Horn and Associates, Inc. May 2019

PTV Vissim 11 Users Manual. PTV AG, Karlsruhe, Germany. March 2019.

VDOT Vissim User Guide, Version 2.0. Virginia Department of Transportation. VDOT Traffic Engineering Division. Jan 2020.

Schilperoot, L, D. McClanahan, R. Shank, M. Bjordahl, T. Woody, J. Johnson, B. Love, M. Wells, R. Johnson. Protocol for Vissim Simulation, WSDOT, Washington State Department of Transportation. Sept, 2014.

Wunderlick, K., M. Vasudevan, and P. Wang. Traffic Analysis Toolbox Volume III: Guidelines for Applying Traffic Microsimulation Modeling Software, 2019 Update to the 2004 Version. U.S. Department of Transportation, Federal Highway Administration. April 2019.