Deliverable 2.4

PTV Vissim extension – new features and improvements

Version: 1.0
Date: 02.07.18
Author: Peter Sukennik, PTV Group

The sole responsibility for the content of this document lies with the authors. It does not necessarily reflect the opinion of the European Union. Neither the EASME nor the European Commission are responsible for any use that may be made of the information contained therein.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No. 723281.
<table>
<thead>
<tr>
<th><strong>Title</strong></th>
<th>Default behavioural parameter sets for AVs</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Creator</strong></td>
<td>Peter Sukennik</td>
</tr>
<tr>
<td><strong>Editor</strong></td>
<td>Charlotte Fléchon</td>
</tr>
<tr>
<td><strong>Reviewer/s</strong></td>
<td>Charlotte Fléchon, Bernard Gyergyay, Syrus Gomari</td>
</tr>
<tr>
<td><strong>Brief Description</strong></td>
<td>Deliverable 2.4</td>
</tr>
<tr>
<td><strong>Contributors</strong></td>
<td>Lukas Kautzsch</td>
</tr>
<tr>
<td><strong>Type (Deliverable/Milestone)</strong></td>
<td>Deliverable</td>
</tr>
<tr>
<td><strong>Format</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Creation date</strong></td>
<td>5/2018</td>
</tr>
<tr>
<td><strong>Version number</strong></td>
<td>1.0</td>
</tr>
<tr>
<td><strong>Version date</strong></td>
<td>02.07.2018</td>
</tr>
<tr>
<td><strong>Last modified by</strong></td>
<td>Charlotte Fléchon</td>
</tr>
<tr>
<td><strong>Rights</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Audience</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>☑ Internal</td>
</tr>
<tr>
<td></td>
<td>☑ Public</td>
</tr>
<tr>
<td></td>
<td>☐ Restricted, access granted to: EU Commission</td>
</tr>
<tr>
<td><strong>Action requested</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>☐ To be revised by Partners involved in the preparation of the Deliverable</td>
</tr>
<tr>
<td></td>
<td>☐ For approval of the WP Manager</td>
</tr>
<tr>
<td></td>
<td>☐ For approval of the Internal Reviewer (if required)</td>
</tr>
<tr>
<td></td>
<td>☐ For approval of the Project Co-ordinator</td>
</tr>
<tr>
<td><strong>Deadline for approval</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Version</strong></td>
<td><strong>Date</strong></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table of contents

1 Introduction ................................................................................................................................... 4
  1.1 PTV Vissim version .................................................................................................................. 4
  1.2 Knowledge transfer .................................................................................................................. 4
  1.3 Driving logics .......................................................................................................................... 4
  1.4 Cooperation & communication functions of AVs ................................................................. 4

2 API improvements ....................................................................................................................... 4
  2.1 Drivermodel.dll ....................................................................................................................... 5
  2.2 Drivingsimulator.dll .................................................................................................................. 6

3 New features ................................................................................................................................ 6
  3.1 Enforce absolute braking distance .......................................................................................... 6
  3.2 Use implicit stochastics .......................................................................................................... 7
  3.3 Class dependent safety distance in following behaviour ..................................................... 8
  3.4 Number of interaction objects & vehicles ............................................................................ 9
  3.5 OpenDrive import ................................................................................................................... 9
  3.6 Consider vehicles in dynamic potential ................................................................................. 10
  3.7 Zero passengers ..................................................................................................................... 11
  3.8 Increased acceleration in following possible ......................................................................... 11
  3.9 Labels for vehicles ................................................................................................................. 11

4 Future outlook ............................................................................................................................ 12
  4.1 Platooning .............................................................................................................................. 12
  4.2 Simple ITS ............................................................................................................................. 12

5 Appendix: PTV Vissim autonomous vehicles new features and how to
1 Introduction

This document describes new automated vehicle (AV) related features and improvements specifically in PTV Vissim software developed within the CoEXist project.

1.1 PTV Vissim version

All parameters or features mentioned below are available since PTV Vissim 11 release version (planned release date: end of September 2018). Most of the API (application programming interface) improvements are available in PTV Vissim 10 version already. For consortium partners & CoEXist purposes, a development version is available since May 2018. A beta version, available for all users with a valid Vissim 10 license, will be released at the beginning of August 2018.

1.2 Knowledge transfer

For an explanation of all new functionalities and AV-related how-to, a webinar was held on June 21, 2018. The slides of the webinar are in appendix A of this document. An AV-related PTV Vissim guide will be prepared in October 2018 (D2.5). An updated PTV Vissim manual will be available (most of it) until beta release (August 2018), a full update will be available with the release version of PTV Vissim 11 (end of September 2018).

1.3 Driving logics

How to use individual features in combination with the driving logics\(^1\) used within the CoEXist project is described in deliverable D2.3. This document describes what was developed and for what purpose.

1.4 Cooperation & communication functions of AVs

Some implications of simple cooperation & communication functions of AVs might be simulated by using new features and appropriate settings in PTV Vissim. For modelling of complex cooperation & communication functions additional actions might be necessary – using one of the available interfaces (COM, drivermodel.dll or drivingsimulator.dll) to mimic complex communication and cooperation strategies.

2 API improvements

PTV Vissim offers three APIs. Two of them, described below, have been improved significantly. APIs can be used for:

- Simulating vehicles with own algorithms within PTV Vissim

\(^1\) See D1.4 and its appendix A
• Connecting & exchanging information with an external simulator (e.g. a sensor simulator)
• Simulating communication & cooperation strategies or any other feature which is not directly available in the GUI of PTV Vissim

API are widely used by car manufacturers and researchers who are developing their own control algorithms. The use of API requires programming skills.

The full description of both mentioned PTV Vissim interfaces can be found in the documents delivered within PTV Vissim installation (see Appendix 1 and 2 of D2.2 “PTV Vissim API - Driver Model Interface.pdf” and “PTV Vissim API - Driving Simulator.pdf Interface”).

2.1 Drivermodel.dll

The External Driver Model DLL Interface of Vissim provides the option to replace the internal driving behaviour by a fully user-defined behaviour for some or all vehicles in a simulation run. The user-defined algorithm must be implemented in a DLL written in C/C++ which contains specific functions (as specified below). During a simulation run, Vissim calls the DLL code for each affected vehicle in each simulation time step to determine the behaviour of the vehicle. Vissim passes the current state of the vehicle and its surroundings to the DLL and the DLL computes the acceleration / deceleration of the vehicle and the lateral behaviour (mainly for lane changes) and passes the updated state of the vehicle back to Vissim.

The external driver model can be activated for each vehicle type separately in the dialog box “Vehicle Type” by checking the checkbox “Use external driver model” on the tab page “External Driver Model” and selecting a driver model DLL file and optionally a parameter file to be used. If this option is checked, the driving behaviour of all vehicles of this vehicle type will be calculated by the selected DLL. A subdirectory DriverModelData\ must exist in the directory of vissim.exe in order to avoid a warning message when Vissim is started.

The following improvements have been implemented:

• External driver model DLLs may be used now in multithreaded simulation runs if all DLLs confirm that they support multithreading (unless there is only one externally controlled vehicle or all externally controlled vehicles are on the same link, the DLL needs to be programmed accordingly, of course).
• If the DLL requests it, Vissim sends the data of all nearby vehicles that the ego vehicle sees according to the current driving behaviour (min./max. look ahead and look back distances, number of observed vehicles) instead of at most 2 for each upstream and downstream per lane.
• User-defined vehicle attribute values can be passed to the DLL and can be modified by the DLL.
• World coordinates of the front end and rear end of nearby vehicles are passed from Vissim to the DriverModel.DLL as well.
• The polyline of the current lane of the ego vehicle (along its route/path, within the visibility distance) is passed to the DLL as well.
2.2 Drivingsimulator.dll

The PTV Vissim add-on module "Driving Simulator Interface" (see Figure 1) allows to connect Vissim to a driving (cycling, walking) simulator (DS). That DS can either be a simulator hardware used by a human or a piece of software representing the algorithms of a CAV (or multiple CAVs).

Vissim provides the surrounding traffic (vehicles, bicycles, pedestrians) to be visualized in the DS, and the DS passes back the current position and orientation of the simulator vehicle(s) (bicycle(s) / pedestrian(s)). The vehicles and pedestrians in the Vissim network react to this simulator data as to all other vehicles and pedestrians in the microscopic simulation model. In addition, Vissim passes traffic signal states to the DS for visualization, and the DS can set detectors in Vissim explicitly in order to affect the signalization.

The DS does not need to know the Vissim data model where the network is modeled from links, connectors, areas, ramps and obstacles. The DS needs to have its own world model (for simulation and visualization). As all vehicle and pedestrian positions are exchanged in cartesian world coordinates (x/y/z), the DS must be able to provide/use such coordinates, and the coordinates of the networks on both sides (Vissim / DS) must match precisely.

The following improvements have been implemented:

- Automatic interpolation of Vissim vehicle/pedestrian world coordinates, orientations and speeds between Vissim time steps if the simulator informs Vissim about a higher frame rate than the Vissim simulation resolution in the call of VISSIM_Connect().
- Optionally, simulator pedestrians can be passed to Vissim now. The pedestrian type for those needs to be selected in the network settings on the new tab page "Driving simulator". The checkbox for activation of the driving simulator interface and the selection box for the vehicle type for simulator vehicles have been moved there as well.
- The maximum number of objects of each type to be exchanged between the simulator and Vissim can be set by the simulator in the call of VISSIM_Connect().
- The simulator can pass a maximum visibility radius to Vissim in the call of VISSIM_Connect(). Vissim vehicles and pedestrians will be passed to the simulator only if they are inside of this radius from the center of the front end of a simulator vehicle or pedestrian.

3 New features

3.1 Enforce absolute braking distance

Idea: the vehicle can stop safely anytime (without a crash), even if the leading vehicle stops instantly (“turns into brick wall”). This ensures that the brick wall stop distance is always kept.

It has implications for:

- Following behaviour
- Lane change behaviour
• Gap acceptance at intersections (conflict areas only)

![Driving Behavior](image)

Figure 1 Enforce absolute braking distance dialog (left) and an illustration (right)

### 3.2 Use implicit stochastics

Idea: the stochastic imperfection of human driving is replaced by deterministic machines & computers. If this attribute is false, a deterministic instead of a stochastically distributed value is used. Affected are the values that:

- cannot be influenced by user e.g. via an adjustable distribution
- are expected to be unaffected by human perception in AVs.
3.3 Class dependent safety distance in following behaviour

Idea: the headway to the followed vehicle depends on the followed vehicle class.

This allows to set different following distances to conventional vehicles, automated vehicles, connected and automated vehicles, cyclists etc.
3.4 Number of interaction objects & vehicles

Idea: automated vehicles can see the signals ahead, but only one (or two…) vehicle in front of itself because the sensors cannot see through the leading vehicle.

Figure 4 Number of interaction objects & vehicles in the dialog (left) and an illustration (right)

3.5 OpenDrive import

OPEN DRIVE = open file format for the logical description of road networks. It was developed and is being maintained by a team of simulation professionals with large support from the simulation industry. Its first public appearance was on January 31, 2006.

This feature allows to read opendrive database and create a network in PTV Vissim.
3.6 Consider vehicles in dynamic potential

This feature improves the interaction between vehicles and pedestrians in crossing conflicts (e.g. on shared space areas). The pedestrians are able to find and use gaps between standing or slowly moving vehicles dynamically, so their behaviour is more intelligent and closer to the real behaviour.
3.7 Zero passengers

Idea: Simply to allow “empty trips” for future autonomous vehicles (SAE automation level 4 & 5).

![Figure 7 Zero passengers in the dialog](image)

3.8 Increased acceleration in following possible

This new parameter allows to set higher acceleration in following process in order to “stay in touch” when the speed of the leading vehicle increases significantly. The ability to keep close following distance without big losses when leading vehicle accelerates was observed on the test track in Helmond (described in D2.6 Technical report on data collection and validation process). To mimic such behaviour in Vissim, this parameter can be set to value above 100% for specific vehicle class and also in dependency on leading vehicle class.

![Figure 8 Increased acceleration in the list & coupled list](image)

3.9 Labels for vehicles

This feature allows to show any vehicle attribute as a vehicle label which is moving with the vehicle during 2D visualisation. This is useful for debugging or analysing the model, showing results etc.
4 Future outlook

From many possible ideas, two major issues have been identified as interesting features in relation to automated driving: platooning and simple ITS. Both features are not within the scope of CoEXist. The realisation of the development is dependent also on existence/availability of data and standards allowing reasonable parametrisation of the process. Important criterion is also the flexibility of the feature (e.g., it is not ineffective to develop features covering just a single use case).

4.1 Platooning

Now (5/2018), complex platooning can be simulated using APIs only. Platooning functionality configurable through PTV Vissim GUI might be developed in the future. Outputs of the MAVEN\(^2\) project (EU project running parallel) might help (and will be considered).

4.2 Simple ITS

Application of ITS (intelligent transport systems) can be simulated using VAP module (vehicle actuated programming) or COM interface. For the future, PTV is considering a simplification of the process and to offer a direct way through GUI.

5 Appendix: PTV Vissim autonomous vehicles
new features and how to

\(^2\) http://www.maven-its.eu/
Welcome to the Webinar

PTV VISSIM: AUTONOMOUS VEHICLES
NEW FEATURES AND HOW-TO

Presenter:

Peter Sukennik
PTV Vissim Product Management
PTV Group, Karlsruhe
peter.sukennik@ptvgroup.com
OVERVIEW

1. Explicit vs. implicit stochastics
2. New features related to AV
3. Driving behavior parameters for AV
4. How to deal with cooperation & communication
Presenter:
Peter Sukennik
PTV Vissim Product Management
PTV Group, Karlsruhe

Answering your questions:
Lukas Kautzsch
PTV Vissim Product Management
PTV Group, Karlsruhe
EXPLICIT STOCHASTICS - FUNCTIONS

Assumption: AVs accelerate/decelerate the same/very similar way

- Desired acceleration/deceleration
- Maximum acceleration/deceleration
EXPLICIT STOCHASTICS - DISTRIBUTIONS

Assumption: AVs will respect the speed limit and the spread will be minimized

- Desired speed
- Time
- Distance
- Occupancy
AV DRIVING LOGIC

- Drive control logic
  - Conflict resolution
  - Following behavior
  - Signal control
  - Lane change behavior
  - Lateral behavior
### Definition under CoExist project:

<table>
<thead>
<tr>
<th>Rail safe</th>
<th>Cautious</th>
<th>Normal</th>
<th>„all knowing“</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Brick wall stop distance</td>
<td>• Gaps similar to human drivers but with higher safety</td>
<td>• Smaller gaps but still safe</td>
<td></td>
</tr>
<tr>
<td>• Big gaps</td>
<td>• Big gaps</td>
<td>• Cooperative behavior</td>
<td></td>
</tr>
<tr>
<td>• Predefined route</td>
<td>• Cautious behavior</td>
<td>• Communication is a precondition</td>
<td></td>
</tr>
<tr>
<td>• No lane change</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• No unprotected signal phase</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Higher lateral distance or physical separation</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Mostly closed environment</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
ENFORCE ABSOLUTE BRAKING DISTANCE

Idea: the vehicle can stop safely anytime (without a crash), even if the leading vehicle stops instantly (“turns into brick wall”)

Implications for:

- Following behavior

- Lane change behavior

- Gap acceptance at intersections (conflict areas only)
  Rear gap = time to brake to full stop 1 m in front of the conflict area
NUMBER OF INTERACTION VEHICLES

Vissim 10:

- Number of observed vehicles

The following network objects are modeled as vehicles in Vissim. Vehicles treat these network objects as a preceding vehicle:
  - Red signal heads
  - Reduced Speed Areas
  - Priority rules (if the minimum time gap or minimum headway condition is not met)

Vehicles also treat the following network objects as a preceding vehicle if they have to stop there:
  - Stop Signs
  - Public transport stops
  - Parking Lots

Conflict areas behavior: A vehicle takes into consideration all conflict areas up to the n-th preceding vehicle (for n observed interaction objects (up to max. look-ahead distance if there is no object).

Vissim 11:

- Number of interaction objects
- Number of interaction vehicles

Idea: automated vehicle can see the signals ahead, but only one (or two…) vehicle(s) in front of itself because the sensors cannot see through the leading vehicle

<table>
<thead>
<tr>
<th>Settings</th>
<th>Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>#obj &lt; #veh</td>
<td>As Vissim 10</td>
</tr>
<tr>
<td>#obj &gt;= #veh</td>
<td>New limited sight and interaction behavior</td>
</tr>
</tbody>
</table>

limits the number of visible vehs within the min. look-ahead dist.
NUMBER OF INTERACTION VEHICLES – CONFIGURATION EXAMPLES

- # interaction vehicles = 4
  - # interaction objects = 4
  - min. look ahead distance = 0

- # interaction vehicles = 1
  - # interaction objects = 1
  - min. look ahead distance = 50

- # interaction vehicles = 5
  - # interaction objects = 3
  - min. look ahead distance = 0

- # interaction vehicles = 3
  - # interaction objects = 5
  - min. look ahead distance = 0

visible vehicles  visible objects
USE IMPLICIT STOCHASTICS

Idea: the stochastic imperfection of human driving is replaced by deterministic machines.

In the internal behavior model (for humans), there are several stochastic values indicating the spread of human behavior:

- the risk acceptance,
- the ability to estimate distance and speed difference,
- the precision when operating the throttle and braking pedals.

For AVs, deterministic values for these parameters can be assumed.

If the attribute “use implicit stochastics” is false, a deterministic average value is used instead of such a stochastically distributed value whenever the distribution cannot be set by the Vissim user.

This option affects:

- desired safety distance,
- desired acceleration,
- desired deceleration,
- decision points (when to start braking / accelerating).

The user may also specify a distribution or function but these are not adjusted automatically, only an implicit stochastic term is suppressed.
Idea: AVs, especially if using C2C communication, can use a tight coupling with small headways.

The normal “human” acceleration behavior cannot use reliable information about the future behavior of the leading vehicle.

- Normal Vissim vehicles tend to fall behind when the leading vehicle is accelerating.

In order to allow vehicles to keep a small headway even during an acceleration process, there is the new parameter “Increased acceleration”.

This value defines a percentage (usually > 100%) of the normal acceleration to be used when the leading vehicle is accelerating. The vehicle cannot exceed its maximum acceleration (which defines the technical limit) but it can exceed its desired acceleration in this situation.

### INCREASED ACCELERATION

<table>
<thead>
<tr>
<th>Count</th>
<th>No</th>
<th>Name</th>
<th>NumInteractObj</th>
<th>StandDistFix</th>
<th>IncrsAccel</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>Urban (motorized)</td>
<td>4</td>
<td></td>
<td>100.0 %</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>Right-side rule (motorized)</td>
<td>2</td>
<td></td>
<td>100.0 %</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>Freeway (free lane selection)</td>
<td>2</td>
<td></td>
<td>100.0 %</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>Footpath (no interaction)</td>
<td>2</td>
<td></td>
<td>100.0 %</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>Cycle-Track (free overtaking)</td>
<td>2</td>
<td></td>
<td>100.0 %</td>
</tr>
</tbody>
</table>

[Driving Behaviors / Vehicle Class Following Behaviors](#)
# NEW FEATURES & DRIVING LOGICS

<table>
<thead>
<tr>
<th>driving logic</th>
<th>recommended setting for new features</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>enforce absolute breaking distance (EABK)</td>
</tr>
<tr>
<td>rail safe</td>
<td>ON</td>
</tr>
<tr>
<td>cautious</td>
<td>ON</td>
</tr>
<tr>
<td>normal</td>
<td>OFF</td>
</tr>
<tr>
<td>all knowing</td>
<td>OFF</td>
</tr>
</tbody>
</table>

* for advanced sensors and/or communicating vehicles choose more than 1 if information from more than one vehicle ahead is available
HEADWAY BASED ON LEADING VEHICLE CLASS

Idea: the headway to followed vehicle depends on the followed vehicle type

Implementation:
- New field in the dialog
- New coupled list for following analogical to „lateral“
CONSIDER VEHICLES IN DYNAMIC POTENTIAL

Idea: pedestrians find gaps between standing vehicles to cross the road
ZERO PASSENGERS

- Occupancy distribution (empirical) with value 0 possible now
- Simply to allow empty trips for future autonomous vehicles
OPENDRIVE IMPORT (INFO)

http://www.opendrive.org

OpenDRIVE = open file format for the logical description of road networks. It has been developed and is being maintained by a team of simulation professionals with large support from the simulation industry. Its first public appearance was on January 31, 2006.

- Import restricted to links & connectors

Usage

The following figure shows the typical incorporation of an OpenDRIVE® file into a simulation tool-chain:
DRIVING BEHAVIOR PARAMETERS
### DRIVING BEHAVIOR PARAMETERS: FOLLOWING

PTV Vissim uses the Wiedemann models. Parameters control the driving behavior during the following process.

*if EABK is on, brick wall stop distance is guaranteed** see PTV Vissim manual for detailed description

<table>
<thead>
<tr>
<th>Model</th>
<th>Parameter **</th>
<th>Rail Safe</th>
<th>Cautious</th>
<th>Normal</th>
<th>All Knowing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wiedemann 99</td>
<td>cc0</td>
<td>def</td>
<td>def</td>
<td>def</td>
<td>smaller</td>
</tr>
<tr>
<td></td>
<td>cc1</td>
<td>def/higher*</td>
<td>def/higher*</td>
<td>def</td>
<td>smaller</td>
</tr>
<tr>
<td></td>
<td>cc2</td>
<td>def/smaller</td>
<td>def/smaller</td>
<td>smaller</td>
<td>smaller</td>
</tr>
<tr>
<td></td>
<td>cc3</td>
<td>def/higher</td>
<td>def/higher</td>
<td>def</td>
<td>def</td>
</tr>
<tr>
<td></td>
<td>cc4</td>
<td>Smaller</td>
<td>def/smaller</td>
<td>def/smaller</td>
<td>smaller</td>
</tr>
<tr>
<td></td>
<td>cc5</td>
<td>Smaller</td>
<td>def/smaller</td>
<td>def/smaller</td>
<td>smaller</td>
</tr>
<tr>
<td></td>
<td>cc6</td>
<td>def/smaller</td>
<td>def/smaller</td>
<td>def</td>
<td>smaller</td>
</tr>
<tr>
<td></td>
<td>cc7</td>
<td>def/smaller</td>
<td>def/smaller</td>
<td>def/smaller</td>
<td>smaller</td>
</tr>
<tr>
<td></td>
<td>cc8</td>
<td>Smaller</td>
<td>smaller</td>
<td>def</td>
<td>def</td>
</tr>
<tr>
<td></td>
<td>cc9</td>
<td>Smaller</td>
<td>smaller</td>
<td>def</td>
<td>def</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Model</th>
<th>Parameter **</th>
<th>Rail Safe</th>
<th>Cautious</th>
<th>Normal</th>
<th>All Knowing</th>
</tr>
</thead>
<tbody>
<tr>
<td>W74</td>
<td>ax</td>
<td>def</td>
<td>def</td>
<td>def</td>
<td>smaller</td>
</tr>
<tr>
<td></td>
<td>bxadd</td>
<td>def/higher*</td>
<td>def/higher*</td>
<td>def</td>
<td>smaller</td>
</tr>
<tr>
<td></td>
<td>bxmult</td>
<td>def/higher*</td>
<td>def/higher*</td>
<td>def</td>
<td>smaller</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Model</th>
<th>Parameter **</th>
<th>Rail Safe</th>
<th>Cautious</th>
<th>Normal</th>
<th>All Knowing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wiedemann 99</td>
<td>cc0</td>
<td>1.5</td>
<td>1.5</td>
<td>1.5</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>cc1</td>
<td>1.5</td>
<td>1.5</td>
<td>0.9</td>
<td>0.6</td>
</tr>
<tr>
<td></td>
<td>cc2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>cc3</td>
<td>-10</td>
<td>-10</td>
<td>-8</td>
<td>-6</td>
</tr>
<tr>
<td></td>
<td>cc4</td>
<td>-0.1</td>
<td>-0.1</td>
<td>-0.1</td>
<td>-0.1</td>
</tr>
<tr>
<td></td>
<td>cc5</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td></td>
<td>cc6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>cc7</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td></td>
<td>cc8</td>
<td>2</td>
<td>3</td>
<td>3.5</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>cc9</td>
<td>1.2</td>
<td>1.2</td>
<td>1.5</td>
<td>2</td>
</tr>
</tbody>
</table>

| Model | Parameter ** | Rail Safe | Cautious | Normal | All Knowing | Def |
|-------|--------------|-----------|----------|--------|-------------|
| W74  | ax | 2 | 2 | 2 | 1 | 2 |
| | bxadd | 2 | 2 | 2 | 1.5 | 2 |
| | bxmult | 3 | 3 | 3 | 2 | 3 |
**DRIVING BEHAVIOR PARAMETERS: LANE CHANGE**

Parameters controlling the necessary lane change behavior (because of the route):

<table>
<thead>
<tr>
<th>parameter for necessary lane change*</th>
<th>own</th>
<th>trailing vehicle</th>
<th>own</th>
<th>trailing vehicle</th>
<th>own</th>
<th>trailing vehicle</th>
<th>own</th>
<th>trailing vehicle</th>
</tr>
</thead>
<tbody>
<tr>
<td>maximum deceleration</td>
<td>n.a.</td>
<td>n.a.</td>
<td>smaller/def</td>
<td>smaller/def</td>
<td>def</td>
<td>smaller/def</td>
<td>def</td>
<td>higher/def</td>
</tr>
<tr>
<td>- 1 m/s per distance</td>
<td>n.a.</td>
<td>n.a.</td>
<td>smaller/def</td>
<td>smaller/def</td>
<td>def</td>
<td>def</td>
<td>def</td>
<td>smaller/def</td>
</tr>
<tr>
<td>accepted deceleration</td>
<td>n.a.</td>
<td>n.a.</td>
<td>smaller/def</td>
<td>smaller/def</td>
<td>def</td>
<td>def</td>
<td>def</td>
<td>higher/def</td>
</tr>
</tbody>
</table>

*necessary lane change means a lane change which is necessary in order to follow a defined route (it is not overtaking because of higher own desired speed)

** EABD (enforce absolute breaking distance) must be on

n.a. = not applicable
DRIVING BEHAVIOR PARAMETERS: LANE CHANGE

Parameters controlling the necessary (because of the route) & free (overtaking) lane change behavior:

<table>
<thead>
<tr>
<th>Behavioral Functionality</th>
<th>Rail Safe</th>
<th>Cautious**</th>
<th>Normal</th>
<th>All Knowing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Advanced merging*</td>
<td>n.a.</td>
<td>on***/off</td>
<td>on***</td>
<td>on</td>
</tr>
<tr>
<td>Cooperative lane change*</td>
<td>n.a.</td>
<td>on***/off</td>
<td>on***</td>
<td>on</td>
</tr>
<tr>
<td>Safety distance reduction factor</td>
<td>n.a.</td>
<td>higher+EABD</td>
<td>def/smaller</td>
<td>def/smaller</td>
</tr>
<tr>
<td>Min. headway (front/rear)</td>
<td>n.a.</td>
<td>higher</td>
<td>def</td>
<td>def</td>
</tr>
<tr>
<td>Max. deceleration for cooperative braking</td>
<td>n.a.</td>
<td>smaller***</td>
<td>smaller***/def</td>
<td>def</td>
</tr>
</tbody>
</table>

*depends on technical equipment and implemented connectivity & cooperation functions
** EABD (enforce absolute breaking distance) must be on
*** If the AV cannot detect that the other vehicle wants to change lanes, the value should be off/zero
n.a. = not applicable
**DRIVING BEHAVIOR PARAMETERS: SIGNAL CONTROL**

Parameters control the reactions on signals.

<table>
<thead>
<tr>
<th>attribute</th>
<th>rail safe*</th>
<th>cautious**</th>
<th>normal</th>
<th>all knowing</th>
</tr>
</thead>
<tbody>
<tr>
<td>behavior at amber signal</td>
<td>continuous check</td>
<td>continuous check</td>
<td>one decision***</td>
<td>one decision</td>
</tr>
<tr>
<td>behavior at red/amber signal</td>
<td>stop</td>
<td>stop</td>
<td>stop/go</td>
<td>stop/go</td>
</tr>
<tr>
<td>reaction time distribution</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>reduced safety distance factor</td>
<td>higher+EABD</td>
<td>higher+EABD</td>
<td>def</td>
<td>def/lower</td>
</tr>
<tr>
<td>reduced safety start upstream of stop line</td>
<td>lower/def</td>
<td>lower/def</td>
<td>def</td>
<td>def/higher</td>
</tr>
<tr>
<td>reduced safety end upstream of stop line</td>
<td>lower/def</td>
<td>lower/def</td>
<td>Def</td>
<td>def/higher</td>
</tr>
</tbody>
</table>

*only protected phase possible in AV-certified environment  
** probably protected phases only, EABD must be on  
*** requires that the AV can remember that it has made a decision for that particular signal head already, else: continuous check

---

<table>
<thead>
<tr>
<th>attribute</th>
<th>rail safe*</th>
<th>cautious**</th>
<th>normal</th>
<th>all knowing</th>
<th>def</th>
</tr>
</thead>
<tbody>
<tr>
<td>behavior at amber signal</td>
<td>continuous check</td>
<td>continuous check</td>
<td>one decision***</td>
<td>one decision</td>
<td>cont. Check</td>
</tr>
<tr>
<td>behavior at red/amber signal</td>
<td>stop</td>
<td>stop</td>
<td>stop</td>
<td>stop</td>
<td>go</td>
</tr>
<tr>
<td>reaction time distribution</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>reduced safety distance factor</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0.6</td>
</tr>
<tr>
<td>reduced safety start upstream of stop line</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>reduced safety end upstream of stop line</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
</tr>
</tbody>
</table>
IMPORT OF PREDEFINED PARAMETERS INTO YOUR NETWORK

PTV Vissim file “base network” available – contains recommended values for cautious, normal & all knowing driving logic – of course these can be adjusted if needed.
COOPERATION & COMMUNICATION

Questions to answer:

- What is the impact of a communication/cooperation?
- Can I replicate expected behavior with standard PTV Vissim or do I need an interface?

Available interfaces:

- **COM** => allows to read & set attributes of Vissim objects or to manipulate them
  - Usable for V2V or V2I communication, e.g. for platooning or time slot based intersection control
  - Examples available in ...\Documents\PTV Vision\PTV Vissim 10\Examples Training\Autonomous Vehicles (AV)
- **DriverModel.dll** => replace internal car following behavior model of Vissim by own algorithm + optionally:
  - Own algorithm for lane changing
  - Own algorithm for reaction on signals
- **DrivingSimulator.dll** => couple own control algorithm (full behavior) with Vissim + optionally integration of:
  - Vehicle dynamics
  - Sensors
  - nanosimulation

Third party software (PreScan, CarMaker...) can be coupled with Vissim => co-simulations.
MATERIALS FOR INTERFACES

- Interface descriptions & example files – see the installation directory
  - C:\Program Files\PTV Vision\PTV Vissim 10\API\DriverModel_DLL
  - C:\Program Files\PTV Vision\PTV Vissim 10\API\DrivingSimulator_DLL

- COM Help & Examples – see the examples directory
  - ...\Documents\PTV Vision\PTV Vissim 10\Examples Training\COM
  - Recommended: “...COM\Basic Commands” for instruction & tips in different programming languages
  - ...\Documents\PTV Vision\PTV Vissim 10\Examples Training\Autonomous Vehicles (AV)