Explanation of the Matlab/Simulink model

Ties Hoenselaar

February 27, 2018

1 Introduction

This report will give a explanation of the car following model which is build in Matlab in combination with Simulink. The Intelligent Driver Model (IDM) [Treiber et al., 2000], which is a time-continuous car-following model, is used to determine the acceleration of the following car. In this model only considers one following car, but can be extended very easily. The Matlab/Simulink model is available via GitHub via the following url: `https://github.com/thoenselaar/car-following-model`.

2 Specifications of the model

1. The model will be created in Matlab/Simulink
2. Only the longitudinal movement of the car is taken into account, because we assume that the platoon moves along the highway and changes direction slowly.
3. The first vehicle of the platoon, the platoon leader, is driven manually. All the following vehicles automatically adjust their speed and acceleration.
4. The duration of the simulation will be 60 seconds.
5. All the vehicles in the platoon are within each other's communication range. [Vinel et al., 2015]
6. First the IDM car following model will be implemented followed by a model which uses CAM messages for communication between the vehicles.
7. Later on some distortions are implemented in the model and their influcences are tested.

3 Matlab script

The Matlab script which is used in this model is shown in Listing 1.

Listing 1: Matlab script

```
1  clear all; warning off;
2
3  a_max = 1;       % maximum acceleration of the leading car
4  a_min = -0.5;    % minimum acceleration of the leading car
5
6  T_THRESHOLD = 1;   % time threshold
7  D_THRESHOLD = 4;   % distance threshold
8  S_THRESHOLD = 0.5; % speed threshold
9
10 N = 1000; % moving average parameter
11
12 sim('CAN_car_following_model_extended')
```

The first line of the script makes sure all the variables are cleared before the simulation is started and it also makes sure the warnings are turned off. At line 3 and 4 the minimum and maximum acceleration of the leading car is set. The acceleration profile of the leading car is shown in Figure 1 where also the meaning of a_{min} and a_{max} is indicated.

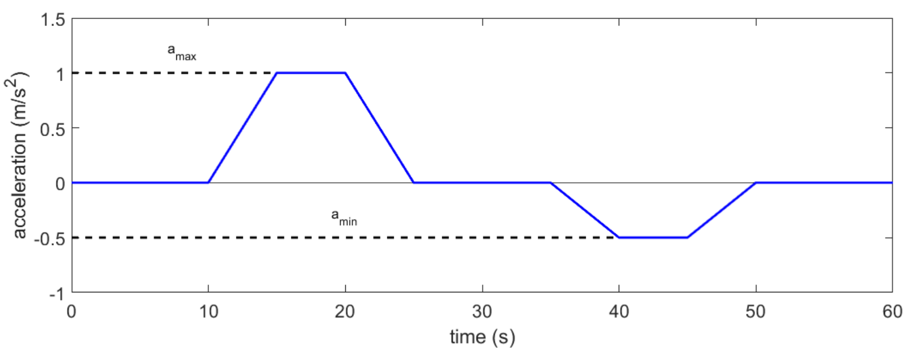


Figure 1: Prescribed acceleration of the leading car

Line 6-8 set some parameters which are used in the CAN message protocol. This protocol will later on be explained in Section 4.4. Line 10 sets a parameter used in the moving average subsystem which is explained in Section 4.6. Finally the Simulink file is ran at line 12.

4 Simulink model

The Simulink model of the leading and the following car are shown in Appendix 5.1 and 5.2 respectively. The checks used for the messaging protocol are shown in Appendix 5.3 and explained in depth in Section 4.4. Finally the moving average filter is shown in Appendix 5.4 and explained in Section 4.6.

4.1 Leading car model

The model of the leading car, shown in Appendix 5.1, has a repeating sequence as input. This repeating sequence creates the acceleration profile displayed in Figure 1. After that, the acceleration is integrated twice. First time to determine the velocity and the second time to determine the position of the leading car. The leading car starts with an initial speed of 10 m/s and at a position of 100 m.

Furthermore the clock is used as input for the model. The time is used in the CAN messages and is also used to check if a message must be triggered via the 3 different checking subsystems.

4.2 Following car model

The following car model, which can be found in Appendix 5.2, has as input the position and the velocity of the leading car. This information is received via a CAN Receive block and later on unpacked. This information together with the velocity and position of the following car is used as input for the IDM model. This model will than calculate the right acceleration for the following car, which after that is done will be averaged. After this step the acceleration is integrated twice just like the leading car to get the velocity and the position of the following car.

4.3 Data store blocks

Both the leading and the following car models use data store blocks[1] to store information about the last CAN message sent. The advantage of these blocks is that they can be used everywhere in the model.

4.4 Message protocol

In this model the ETSI Cooperative Awareness Basic Service is used to determine when a CAM message must be generated. A CAM message shall be triggered in one of the two cases:

- The time elapsed since the last CAM generation is equal or larger than 1000 ms.
- The time elapsed since the last CAM generation is equal or larger than 100 ms and any of the following events has occurred:
    1. the absolute difference between the current position of the vehicle and its position included in the previous CAM exceeds 4 m;
    2. the absolute difference between the current speed and the speed included in the previous CAM exceeds 0.5 m/s;

[1] https://uk.mathworks.com/help/simulink/data-stores.html

3. the absolute difference between the current direction of the vehicle and the direction included in the previous CAM exceeds 4°.

3 subsystems are used to check these triggers. The last trigger will not be implemented in the model, because it is assumed that the cars drive in longitudinal direction and therefore the direction changes very slowly. The first trigger is tested with the subsystem called "time_check" and is shown at the bottom of Appendix 5.3. The second and the third trigger are tested with the "distance_check" and "speed_check" respectively and can also be found in Appendix 5.3.

4.5 CAN messages

The structure of the CAM messages is derived from the a Master Degree's project by Joakim Kjellberg [Kjellberg, 2011]. Three different signals will be send within a CAM message, which are: position, velocity and time. These will be explained in detail below.

Position

The position of the leading vehicle will be send as a 16-bit signal with a resolution of 0.1. This means that the data can range from 0 to 6553.6 meter. In this model the leading vehicle will drive approximately 1000 meters so this range should be big enough.

Velocity

The velocity of the leading vehicle will also be send with a 16-bit signal but with a resolution of 0.0025, this means that the velocity can range from 0 to 163.84 m/s which should also be high enough.

Time

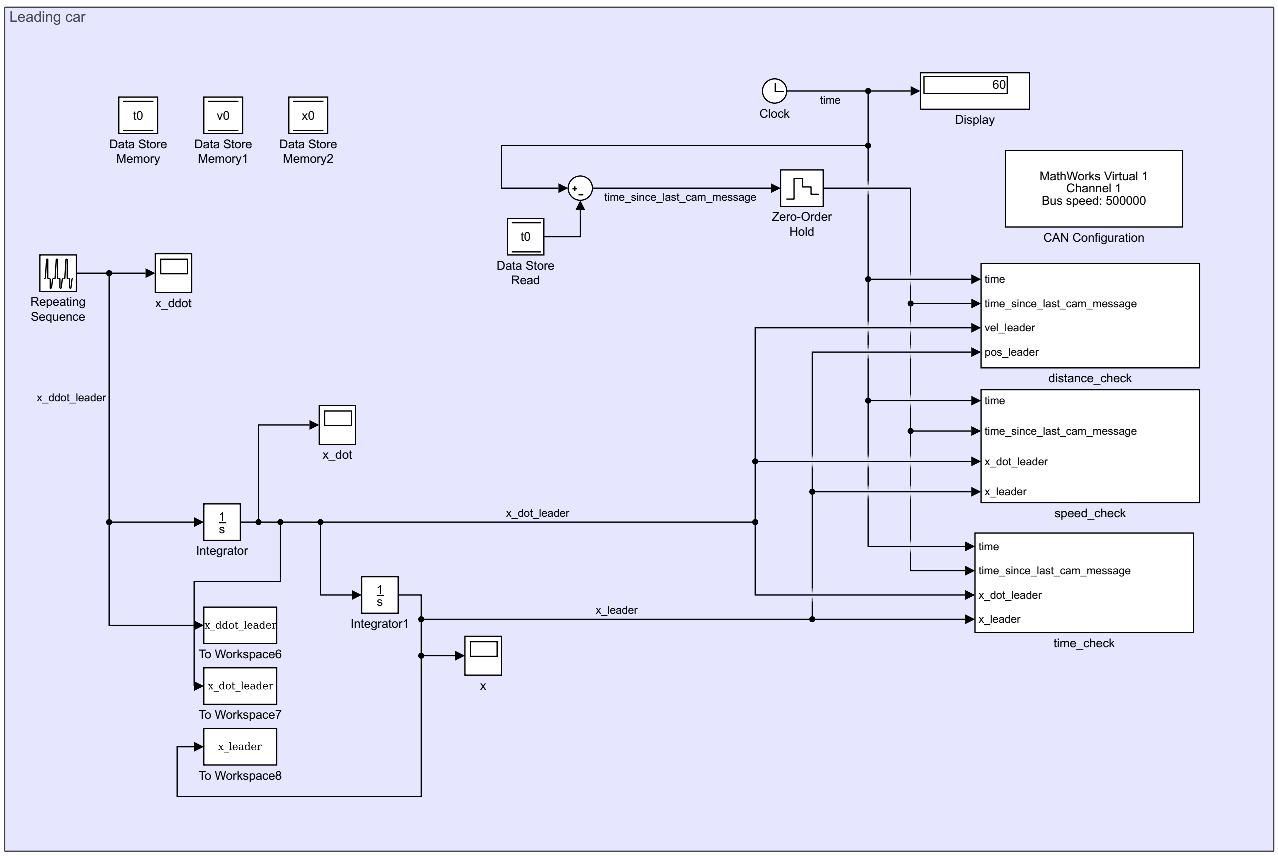
Lastly the time will be send within the CAM message. This information won't be used by the following car but is instead used to analyze when the CAN messages are sent. This information will also be sent with a 16-signal with a resolution of 0.01 which results in a range of 0 - 655.36 seconds.

4.6 Moving average subsystem

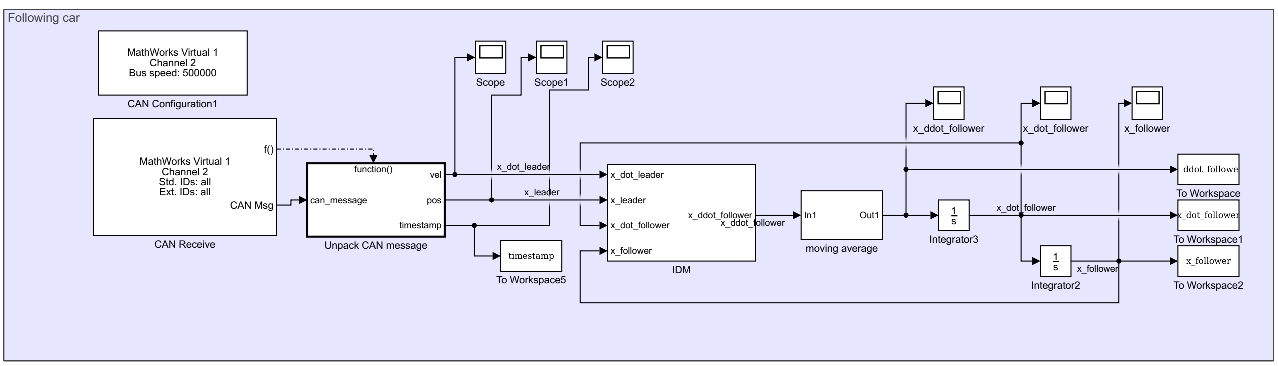
To smooth out the output of the IDM model (the acceleration of the following car) a moving average filter is used. A custom subsystem is designed in Simulink which is visible in Appendix 4.6. This subsystem will store the last N values of the acceleration, which is set at line 10 of the Matlab script, and than calculate the mean.

5 Appendix

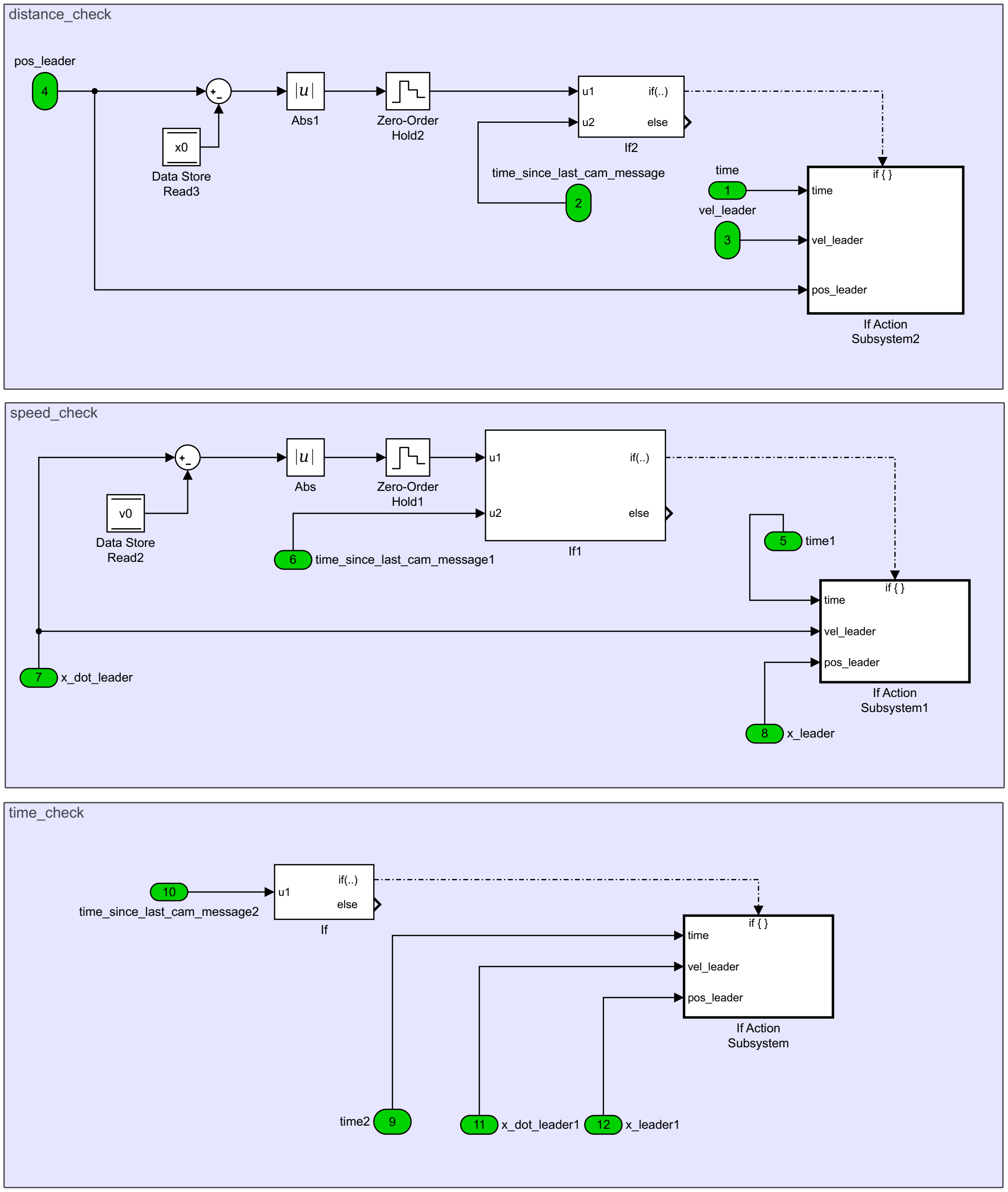
5.1 Leading car



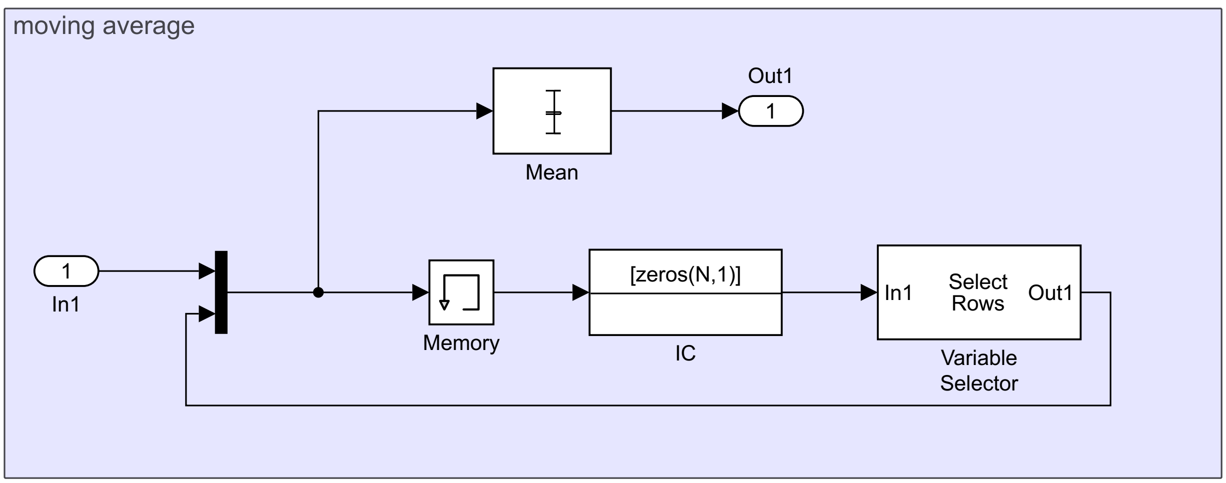
5.2 Following car



5.3 Checks for the messaging protocol



5.4 Moving average



References

[Kjellberg, 2011] Kjellberg, J. (2011). Implementing control algorithms for platooning based on v2v communication.

[Treiber et al., 2000] Treiber, M., Hennecke, A., and Helbing, D. (2000). Congested traffic states in empirical observations and microscopic simulations. *Phys. Rev.*, E 62:1805—-1824. doi:10.1103/PhysRevE.62.1805.

[Vinel et al., 2015] Vinel, A., Lan, L., and Lyamin, N. (2015). Vehicle-to-vehicle communication in c-acc/platooning scenarios. 53:192–197.