22nd International Conference
ENGINEERING MECHANICS 2016
Svratka, Czech Republic, 9 – 12 May 2016

AUTOMATIC DIFFERENTIAL LOCK CONTROL IN A TRUCK – STRAIGHT DRIVE

P. Kučera[*], V. Píštěk[**], P. Porteš[***]

Abstract: *The paper deals with the creation of control algorithm for locking differential of the truck. This algorithm is based on assumptions of the ZF ADM system and it is created using software Simulink-Stateflow. The article describes the function of algorithm and its application to the computational model of vehicle with 4x4 drive. In conclusion, a simulation of driving manoeuvres is done to verify the functionality of the assembled control algorithm of the differential.*

Keywords: Control Algorithm, Simulink, Stateflow, Computational Model of Vehicle, Differential

1. Introduction

Mechatronic systems aimed at controlling of the vehicle differential locking can be used for more efficient use of a vehicle torque. The available systems are very sophisticated. For example Torque Vectoring System works with a continuous change of the torque control. In case of trucks, simple systems of a torque step change are used. This principle is used by ZF Company which has developed an algorithm for the control of the dog clutch with special teeth for the differential lock. This system is known as ZF ADM system. According to the literature (Stelzeneder, F. H. & Aitzetmüller, H., 2000), the system is intended for a vehicle up to 6x6 drive. Basic philosophy of this control is monitoring of each wheel speed and a slip evaluation then the algorithm determines a lock and open of the differential.

The article deals with the creation of a partially independent control algorithm for the differential lock which is based on assumptions of the ZF ADM system. This means that there is not a sequential lock of the differential in all cases as in the case of the ZF ADM system. The algorithm is also extended for a vehicle up to 8x8 drive. The algorithm was created in the Simulink-Stateflow software. In order to verify the algorithm function, the vehicle computational model was assembled. This vehicle model was built with the use of own libraries with blocks containing computational models of basic vehicle powertrain elements. A more detailed description can be found in (Kučera, P. 2015; Kučera, P. & Píštěk, V., 2014; Kučera, P. & Píštěk, V., 2013).The article describes the application of the control algorithm on the vehicle computational model, as well as simulation of the drive in muddy conditions. The aim was to develop a control algorithm which could be easily compiled into an appropriate language and used on the National Instruments hardware and software. This will allow initial testing of the control algorithm on a real vehicle.

2. Control algorithm of the differential lock

Knowing of the ZF ADM system (Stelzeneder, F. H. & Aitzetmüller, H., 2000) is the basis required to generate the algorithm. The created control algorithm for the control needs a speed monitoring of each wheel. The slip between the left and right wheel of a single axle is calculated from this wheel speed. The slip between the drive shafts of the respective axles is calculated, too. However, a problem may occur when turning because the individual wheels rotate at different angular velocity. Then the control system

[*] Ing. Pavel Kučera, Ph.D.: Institute of Automotive Engineering, University of Technology, Technická 2896/2; 616 69, Brno; CZ, kucera@fme.vutbr.cz

[**] Prof. Ing. Václav Píštěk, DrSc.: Institute of Automotive Engineering, University of Technology, Technická 2896/2; 616 69, Brno; CZ, pistek.v@fme.vutbr.cz

[***] Doc. Ing. Petr Porteš, Ph.D.: Institute of Automotive Engineering, University of Technology, Technická 2896/2; 616 69, Brno; CZ, portes@fme.vutbr.cz

would evaluate a condition as a slip. For this reason, the steering angle sensor is used, and a slip correction is carried out. For example, the slip of the front axle is described by the equation in the created algorithm

$$s=\frac{\max(\omega_L,\omega_R)-\min(\omega_L,\omega_R)}{\max(\omega_L,\omega_R)}-s_{cor},\ \text{if}\ \max(\omega_L,\omega_R)=0,\ s=0, \tag{1}$$

where s is the slip between the right and left wheel, s_{cor} – slip correction, ω_L – angular velocity of the left wheel and ω_R – angular velocity of the right wheel. The speed difference is controlled to prevent the destruction of powertrain parts. The brake pedal is another control element. In case of braking, axle differentials must be opened because the vehicle stability must be preferred. The braking state of the engine is a similar case. The throttle is used to control a state of the engine braking. In terms of vehicle velocity, a function of an axis differential lock is limited. The last of the main sensors is the clutch sensor, which is used for the changing gear. The control algorithm is created in the Simulink software using the Stateflow library, which is intended primarily for creating logic algorithms. The algorithm is divided into two main parts and the user has four control buttons. The first button sets a manual or automatic control of the differential lock. The diagram of the automatic control is shown in Fig. 1. In case of manual control, another three buttons are used for locking.

3. Computational model of the vehicle

The available literature (Dabney, J. B. & Harman T. L., 2004; Grepl, R., 2007) was used to create computational models. The vehicle computational model with 4x4 drive was used for the differential lock simulation of the truck. The basic parameters of the vehicle are the weight 20000 kg, front axle track 1.994 m, rear axle track 1.774 m, wheelbase 3.7 m, engine power 265 kW at 1500-1900 rpm and engine torque 1775 Nm at 1000-1410 rpm. To assemble a computational model of a commercial vehicle, blocks from own library are used and its powertrain is assembled from an engine block (Paccar MX Euro 5), clutch, transmissions (ZF 16S 2530), auxiliary transmission with the option of turning on and off of the front drive, two axles with axial differentials, drum brakes, computational models of tyres (315/80 R22.5) and vehicle block. The control unit block, differential block, tyre block and road block are the most important elements in the computational model. The computational model of the differential allows a simulation where the differential is locked. It is an important aspect for testing the control algorithm. In this case, the differential can be closed or opened. The tyre block (Pacejka, H. B., 2006) has been extended by a possibility to change the value of the scale factor for the peak friction coefficient $\lambda_{\mu x}$, $\lambda_{\mu y}$ and the rolling resistance coefficient q_{sy1}. It is important for simulation of the slip.

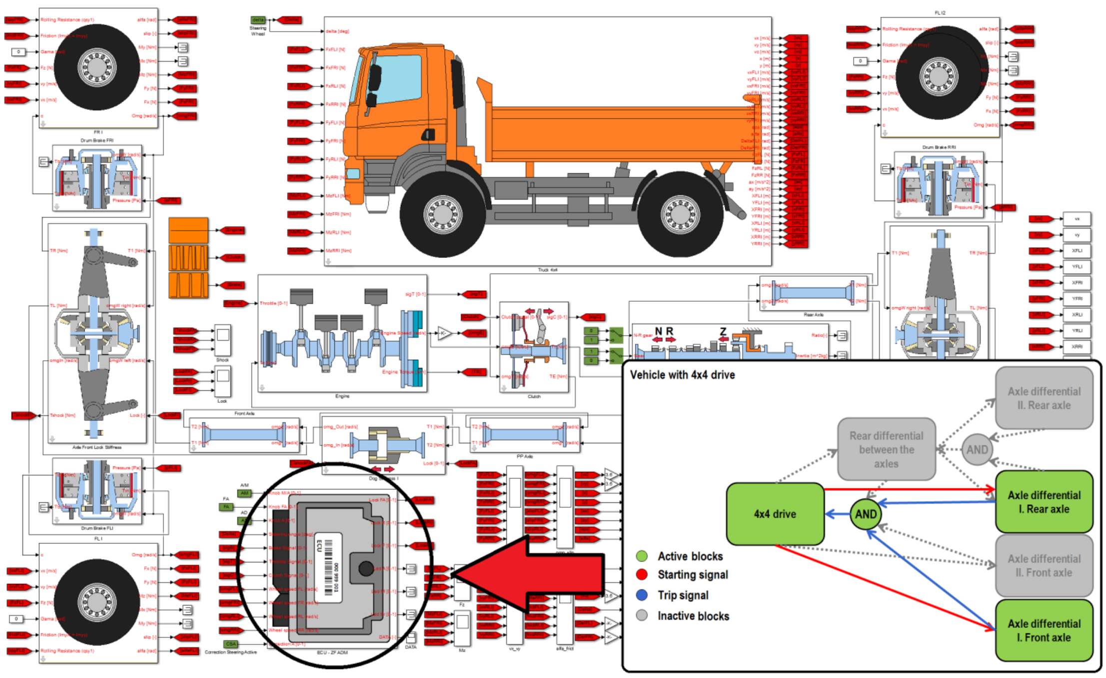

Fig. 1: The vehicle computational model with the differential lock control algorithm.

4. Simulation of the automatic control

The simulation was performed on the computational model of a vehicle with 4x4 drive. In order to conduct a test of the differential lock, it was necessary to simulate the slip of the individual wheels of the vehicle. For this purpose, the road block has been created with the matrix describing the road adhesion between the tyre and the road, and with the rolling resistance coefficient. These values are input parameters of the tyre block depending on a tyre position on the road. This principle can simulate various road surfaces and simulates a crossing between them.

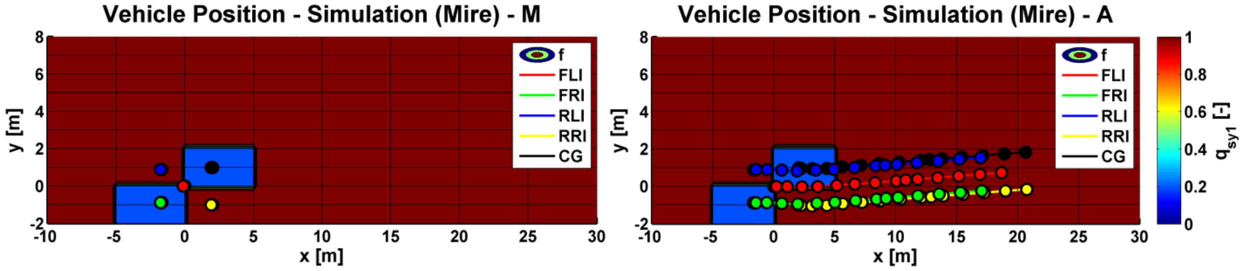

Fig. 2: Vehicle position (manual control and automatic control)

The simulation describes starting of the vehicle from zero velocity on an even road. This simulation is divided into automatic and manual control for driving without a change of the steering angle. Both combinations of simulation have three input signals which are accelerator, clutch and brake signal.

The first simulation can be described in more detail as follows: the vehicle began to move without changing of the steering angle. The manual control of the differential was set, and no differential or the front-wheel drive were activated. The dependence of the scale factor of the peak friction coefficient on a vehicle position is shown in Fig. 2. The blue region describes a value of 0.3. Other areas have the scale factor of the peak friction coefficient of 1. The rolling resistance of these blue areas reaches 0.4 which corresponds to the muddy surface of the road. The position of each wheel (FLI – left front wheel, FRI – front right wheel, RLI – left rear wheel, RRI – right rear wheel) and the centre of gravity of the vehicle in the simulation drive are displayed, too.

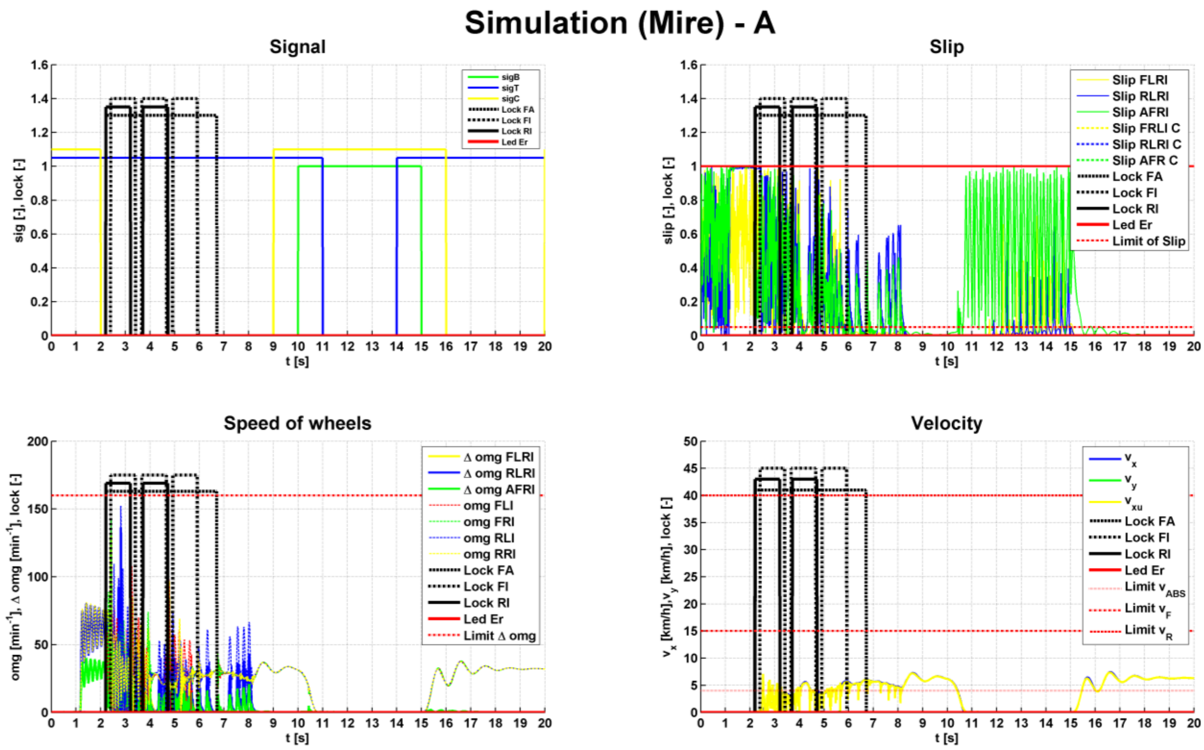

Fig. 3: Input and output signals of the control algorithm.

In the simulation process of a vehicle move, there was the rear wheel slip caused by a different condition of the rear wheels and the vehicle remained standing in one place. In that case, the driver would have to use buttons to control differential locking. The second simulation is shown in Fig. 4 where the control system is set to automatic control. The vehicle began to move and the slip of the right rear wheel occurred again. In this case, the control algorithm evaluated the slip by Eq. 1 and other control parameters. There are four graphs in Fig. 3 describing all parameters. The first graph describes a throttle, brake and clutch signal. The second graph shows the slip signals and the slip correction (not applied in this simulation). The third graph displays the differences of the wheel speed. The fourth graph represents the vehicle velocity and the velocity limit for controlling. The activated front wheel drive or differential lock can be found in each of these graphs of simulation. In case of all differential locks, the algorithm does not have the slip value used for evaluation. Therefore, a 1s test was performed. Axial differentials were opened and input parameters for locking of the differential were checked again. The test status is visible in the results of the second simulation. The algorithm was tested three times before the vehicle drove from a muddy field to the asphalt road.

5. Conclusions

The aim was to create a control algorithm for the differential lock of the truck. This algorithm was created in the software Simulink-Stateflow. The created algorithm was based on the ZF ADM system.

The created algorithm can be described as a mechatronic system which evaluates the input wheel speed signal, steering angle, throttle, clutch and the brake signal. The evaluation of algorithm determines locking or opening of the differential. In order to verify the created algorithm, the simulation was carried out and described in the relevant chapter. The computational model of the vehicle with 4x4 drive was made to test the algorithm. The results achieved in the simulation and the theoretical predictions suggest that the response behaviour of the algorithm is correct. Errors occurred in the algorithm before the initial testing on a real vehicle. This simulation could save costs if the algorithm was applied incorrectly to a real control unit.

The algorithm will be expanded and improved to be applicable to the vehicles with more axles. It will be adapted to different types of vehicles as well. It would also be appropriate to test the algorithm on a real vehicle using NI 3110 RT hardware. This hardware would serve as a control unit. But for this testing on a real vehicle, it is necessary to create an appropriate dog clutch for the differential lock. This work has already started in cooperation with one a truck manufacturer.

Acknowledgement

This work is an output of the internal BUT research project Reg. No. FSI-S-14-2334 and was also supported through NETME CENTRE PLUS (LO1202) by financial means from the Ministry of Education, Youth and Sports under the „National Sustainability Programme I".

References

Dabney, J. B. & Harman T. L. (2004) Mastering Simulink. Pearson Prentice Hall, Upper Saddle River.

Grepl, R. (2007) Modelování mechatronických systémů v Matlab SimMechanics. BEN, Praha.

Kučera, P. (2015) Mechatronic Approach to Vehicle Dynamics. Brno University of Technology, Faculty of Mechanical engineering, Brno

Kučera, P. & Píštěk, V. (2013) Transmission Computational Model in Simulink. Perner's Contacts, 8, 4, pp. 37-48.

Kučera, P. & Píštěk, V. (2014) Virtual prototype of a heavy duty off-road truck driveline in Simulink software, in: Transport means 2014 Proceedings of 18th International Conference, Technologija, Kaunas, pp. 5-8.

Pacejka, H. B. (2006) Tyre and vehicle dynamics. Butterworth-Heinemann, Oxford.

Stelzeneder, F. H. & Aitzetmüller, H. (2000) ADM Drive-Train Management, in: FISITA World Automotive Congress, Korea Society of Automotive Engineers, Seoul, pp. 1-7.