Asynchronous Vehicle Control System Basing on Analytical Continuous-Time Functions

Damian Petrecki
Department of Computer Science and Management
Wroclaw University of Science and Technology
Wrocław, Poland
e-mail: damian.petrecki@pwr.edu.pl

Abstract **— The paper describes a proposition of a driver support system composed of multiple independent processes producing discrete outputs and consuming continuous inputs, with a shared interpolate process. The research rejects multiple controllers handling different areas with the same actuators in favor of single but both multi-criteria and asynchronous decision making system. This way, a decision making problem has been limited and a big data processing has been eliminated, keeping high vehicle performance and low physical system complexity. The solution presented in this paper offers very promising safety and comfort during the simulation-based experiments.**

Keywords – integrated driver support system; multi domain controller; continous-time controll; asynchronous algorithm.

I. Introduction

The paper describes a new way to decompose the driver support problem, not into stability control, anti-slip issue, extreme situation handling, etc., but into data acquisition, trajectory calculation and control execution, which provides comparable results: safe and comfortable ride. The solution is literally a heuristic algorithm performing the vehicle control task, basing on driver's reference input, exclusively producing actuator signals for all actuators in the systems. This way, a vehicle equipped with the proposed solution can uses a centralized computer system which eliminates multi-system interferences. Moreover, the vehicle can be easily maintain, including software based tuning, updating and introducing new features without adding new physical sensors and controllers or modifying existing ones. What is more, there is no ability to bypass the system by a driver, so it cannot be called a typical decision support system.

The paper structure is as follows. In section 2, the algorithm is compared to a classic approach. In section 3, the algorithm itself is described. In section 4, the simulation: in part A, the environment used during the research, in part B, both test end reference vehicles, in part C, performance evaluation, and in part D, dynamic experiments. In section 5, the results are evaluated. In section 6, a future development direction is shown.

II. State of the art

It is hard to say what the most modern, scientific approach is for supporting drivers. There are well-known common safety systems, like Anti-Lock Braking System (ABS) [1] or Electronic Stability Control (ESC) [2], but the mainstream is developed under non-public licenses or even as companies' secrets. On the other hand, the most popular, related conference topics are vision and perception [3][4], traffic models [5][6], accident preventions [7][8], and, of course, autonomous driving [9]-[11].

This research presents a different perspective – it is an integrated driver support system, and the main goal is not developing a better perception system, a more precise model, or a smarter autonomous driver-replacement, but presenting a new way to compose different, existing solutions to achieve high performance (in a way of vehicle safety and comfort) while lowering the computing power at the same time.

Let us consider a case study, a very common situation, well know from everyday driving – a driver wants to launch rapidly with front wheels turned, similar to when entering the flow of traffic. A modern car, equipped with typical safety systems would involve a lot of these to influence the same parameter – wheels' speed. Engine Management System (EMS) [12] uses the engine to raise it, Acceleration Slip Regulation (ASR) [13] reduces it, active differential differentiates it, ECS applies brakes to avoid slipping and ABS limits this brake action. The proposed solution is a very different one. It would calculate the proper speed for all wheels, taking into consideration all variables handled by mentioned classic systems and apply it, in this very case even without using brakes, but only an engine and a transmission system.

What is more, classic in-vehicle systems often use very similar sensors, e.g., one camera for tracking traffic and the second one for analyzing traffic lanes [14]. The proposed solution aggregates data regardless of source, type and frequency and uses an assembled model, so it does not need the duplication and it is more hardware-independent, especially in case of frequency.

Current research is not related to an autonomous driving at all. The proposed system, thanks to its model of a

 ISBN: 978-1-61208-720-7

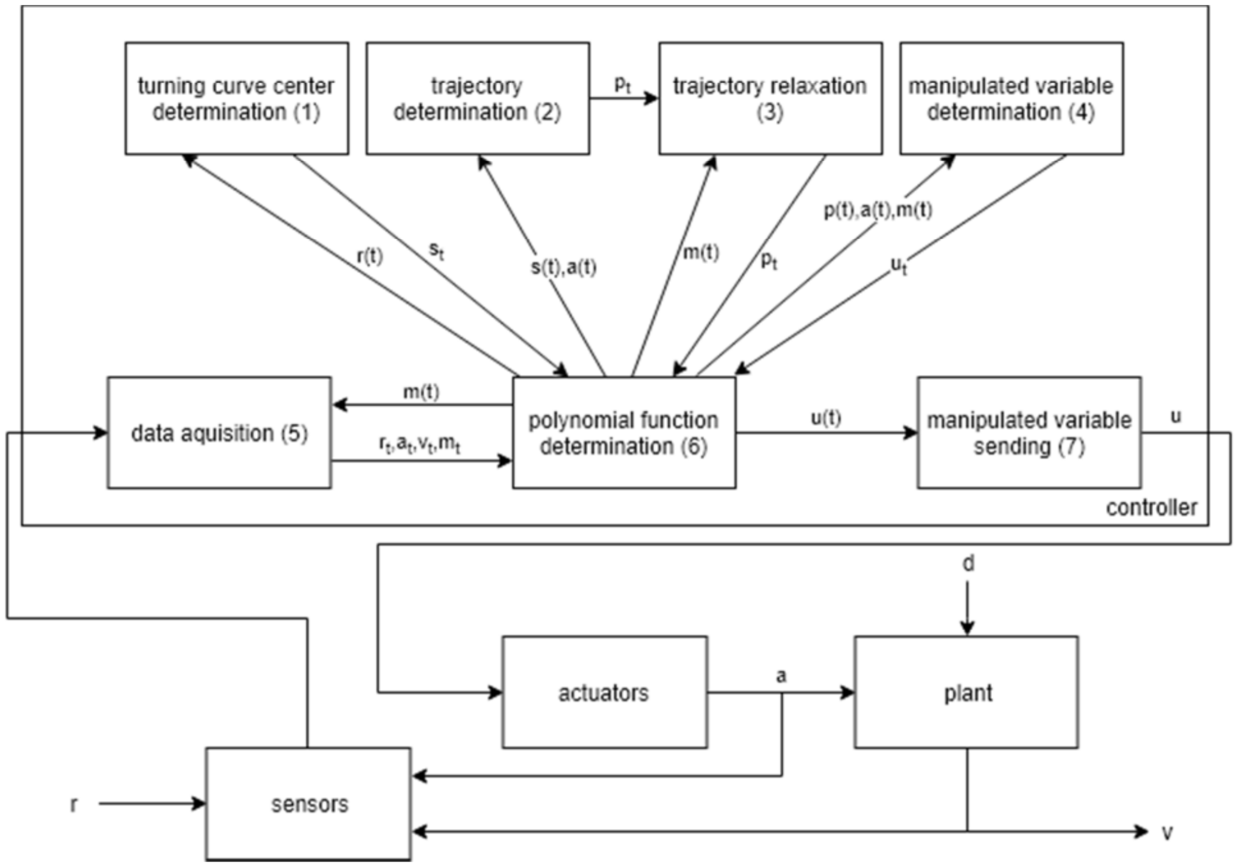

Figure 1. Control system schema

vehicle's environment could be a base of such solution, but for now this part is out of scope of the research.

III. ALGORITHM GENERAL DESCRIPTION

The algorithm is shown in Figure 1. When modeling using black-box method, ignoring the controller's structure, the presented solution seems to be very similar to typical control systems – it reads reference input *r* from the user (mainly steering wheel and pedals positions), the vehicle behavior in an environment (using cameras and radars) and vehicle-related data (using accelerometers, thermometers, etc.) *v* and also actuators state readers *a* and produces a control vector *u* consisting of all actuators manipulated variables: engine, linkage system, suspension, driveline, brakes. The only difference is the lack of time connection between inputs and the output.

The algorithm consists of several processes. Each of them can be scheduled (triggered by time) or started by data incoming from a sensor. The outputs of all processes are discrete values (in one or more dimensions) shown as sequence elements with bottom index *t*, e.g., s_t. Most input data (both starting processes and read during them) come from continuous-time functions stored in analytical, polynomial forms, shown as functions with *t*-argument, e.g., *s(t)*. It means a single, distinguished process (6) is introduced to build continuous-time functions from discrete sequences, which allows data interpolation and extrapolation. The process uses polynomial curve fitter method [15], accepts discrete values and timestamps, and produces a vector of polynomial coefficients. This way, a very specific storage is introduced that stores discrete variables and provides analytical functions as its output.

A single dimensional data acquisition process is proposed (5). It reads and stores input values from input devices r_t and in-vehicle sensors, it reads actuators' states a_t, like suspension status, accelerations, engine status, etc., and

TABLE I. SYMBOLS USED IN FIGURE 1

symbol	description
r	reference input form an user
a	actuators state
d	external distortions
v	behavior measured via sensors (cameras, radars, accelerometers, etc.)
m	environment map – list of measured objects with its position and classification and parametrization results
u	control vector
s	center of turn
p	vehicle position p_t/trajectory *p(t)*

simple (non-matrix) measurements from v_t. Each variable is handled by a separated thread.

The process of a second type calculates the desired trajectory using environment knowledge, vehicle-geometry model, and input data. It is split into several sub-processes, without any time-synchronization:

- The first sub-process referred to (1) in Figure 1 calculates the center of the turning curve (if any) in the vehicle-centered coordinate system, using speed, steering wheel position, and vehicle geometry.
- Sub-process (2) calculates the desired vehicle positions p_t in the future, which means the desired vehicle trajectory.
- Sub-process (3) uses genetic relaxation algorithm [16] and environment knowledge v_t to improve the trajectory to avoid accidents, lowering external objects hit possibility. Please note this process can change the trajectory in any way, e.g., by increasing speed or changing the turn, and its behavior is unpredictable. This is the only process that reads its input directly from the other process, not the storage.

The next process (4) uses the trajectory *p(t)* to calculate control values for all executors u_t, e.g., calculates each wheel speed and turn and then engine power, braking force, linkage system, and differential parameters. Calculated manipulated variables are being sent to the vehicle by own sender processes (7) (one process per variable), which read data from storage, not from the processes that actually generated them.

The process of the last type (5) handles the environment data v_t and is the most complex one. This is the complex part of the data acquisition process. This is the only case when the matrix data (distances from radars or bitmaps from cameras) have to be handled. The process is triggered for each input from each signal separately. The result is a 3D model of the vehicle's environment consisting of a set of classified objects, in the form of objects' shapes (3D line segments), class and positions, so data size is significantly decreased. Due to long processing time, the output of this process is stored with the input data appearance timestamps.

The environment analysis process is the biggest challenge related to the research. It needs a separate algorithm that accepts various formats of input data arriving at unpredictable time (cameras, radars) with, optionally, the already known 3D environment model *v(t)* to update the model as its output. The Long Short-Term Memory (LSTM) neural networks [17] are considered as the most promising way to solve this problem so far.

The most important idea in this algorithm is the absolute lack of time synchronization between input and

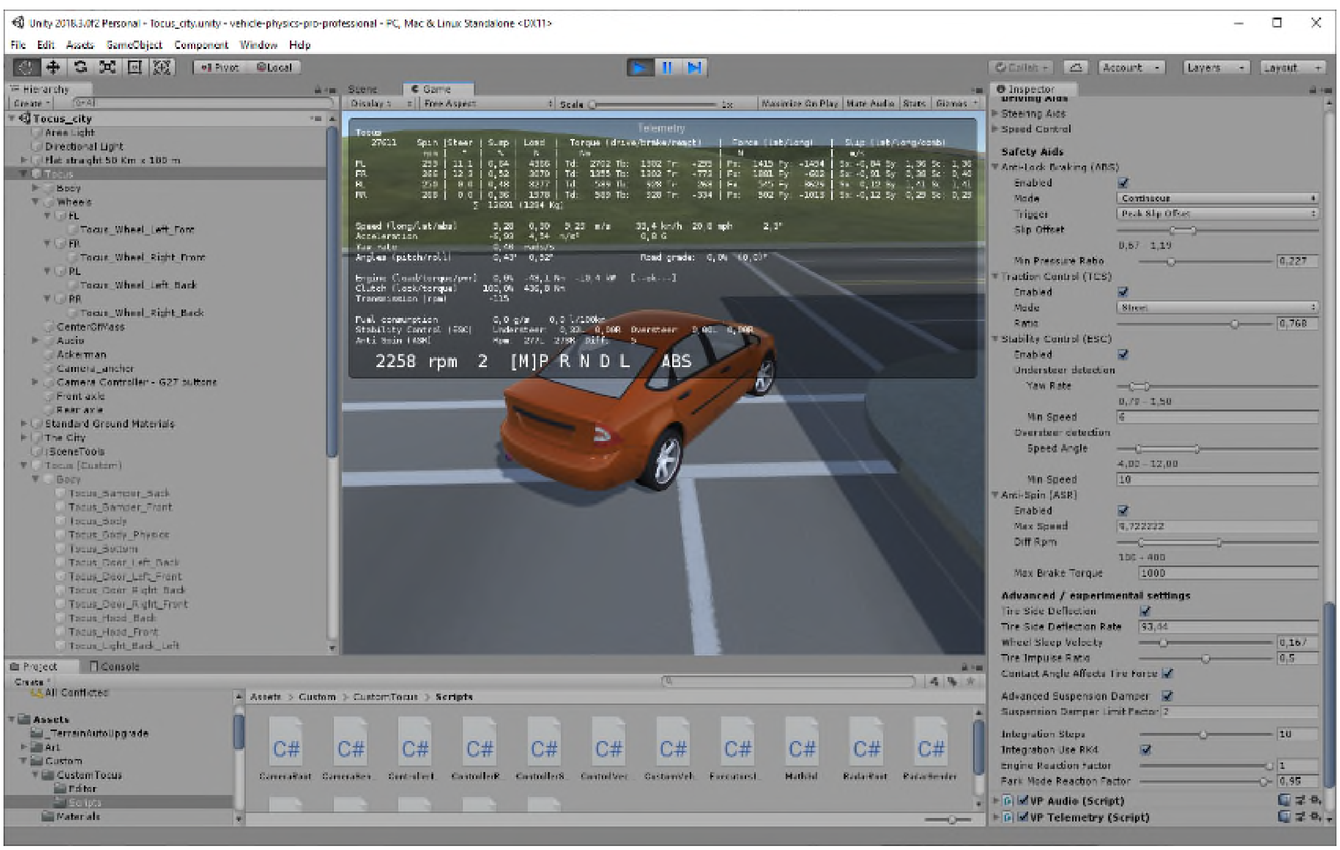

Figure 2. Simulator view (90-degree test)

output. Each process is run separately and uses functions extrapolated from other processes, no matter the age of the source values of the last polynomial calculation. It must be noted that all functions calculated during the process can be used by external processes, not related to vehicle control, like headlight control, climate control, comfort features, etc..

IV. CURRENT RESULTS

The presented solution has been tested in different scenarios and the current results are presented.

A. Simulation environment

All experiments are conducted in Unity3D [18] environment with Vehicle Physics Pro (VPP) [19]. Unity3D is responsible for communication with an operating system driver of Logitech G29 steering wheel [20], rendering visual interpretation of simulated rides and basic, Newton physics. VPP is responsible for vehicle simulation, including dependencies between in-vehicle physical subsystems, tires and suspension behavior, and standard active safety systems. VPP is in general a pre-compiled library, so a lot of its mechanisms are unknown. Its realism is not verified, just considered to be sufficient to compare two vehicles in the same conditions. The base assumption of the research is that the experiments results from a comparable, simplified environment should be applicable to the comparable, real one.

The experiments' results are read from telemetry panel provided by VPP (Figure 2) and from controller application. All data are stored during the experiments in text files and analyzed off-line.

B. Test and reference vehicles

The reference vehicle is built using VPP components only. It has an active suspension, 4-wheel steering, automatic gearbox, 4-wheel drive with active differential, and following active safety systems: ABS, Traction Control System (), ESC, ASR. Most of its implementation is hidden and unknown, but can be calibrated using built-in configuration panels, visible on the right part of Figure 2.

The test vehicle has the same physical parameters (weight 1200kg and equal weight distribution per wheel, wheels localizations, engine power/torque curves, etc.) and abilities (4-wheel drive, 4-wheel steering, controllable transmission, differential and suspension). The difference is that, in the test vehicle, the input from a driver is not sent to the vehicle itself, but transferred to an external application implementing the proposed algorithm. All in-vehicle and environment-related sensors data are handled in the same way. The application sends back control variable for each actuator separately, in separated threads. In this vehicle, there is no other driver support system implemented.

C. Performance results

All experiments are conducted using i7-7700k 4.2Ghz processor, 16GB RAM, SSD hard drive and Windows 10 64bit operating system. Both simulation (Unity3D) and control (external application) are performed on the same machine, because its performance is sufficient for current test scenarios. RAM usage never exceeds 10GB and CPU load is always below 20%, when simulation framerate 40fps is preserved.

The situation changes when all external sensors (21 radars and 8 cameras) are running. Then the simulation occupies about 4GB more RAM (which is still irrelevant), but exhausts all CPU abilities, reducing simulation framerate to 10-15fps (depending on a scenario).

D. Dynamic experiments

Three kinds of experiments are proposed. All test rides have been conducted 4 times, each with the same driver,

TABLE II. MOOSE TEST RESULTS

enter speed	reference solution			proposed solution		
	A	*B*	*C*	*A*	*B*	*C*
80	1.6	-4	170	0.1	2	91
80	1.8	-7	172	0.1	-1	84
80	1.7	-8	168	0.1	0	83
80	1.6	-5	192	0.2	1	86
100	1.6	-6	99	0.1	2	100
100	1.6	-6	97	0.2	11	87
100	1.7	-10	78	0.1	-2	81
100	1.7	-6	89	0.1	0	82
120	1.8	-9	145	0.2	1	92
120	1.5	-9	150	0.2	4	91
120	1.7	-2	154	0.2	4	97
120	2	-4	140	0.2	-1	95
140	failed			0.1	2	110
140	1.6	-40	180	0.4	2	101
140	2.1	-36	165	0.2	4	105
140	1.9	-38	79	0.3	1	87
160	1.9	-48	145	0.2	-2	89
160	1.8	-60	138	0.3	1	90
160	2.2	-40	139	0.3	2	98
160	failed			0.3	2	115
180	failed			0.4	2	97
180	2.1	-68	165	0.3	2	95
180	failed			0.3	1	96
180	failed			0.2	1	97
200	2	-20	66	0.2	2	94
200	Failed			0.3	1	119
200	1.9	-30	81	0.4	-6	126
200	failed			0.3	-1	83

 ISBN: 978-1-61208-720-7

same conditions. In Test 1 and Test 2, vehicle roll angle A (degree), speed change B (km/h), and maximum steering wheel angle C (degree) during the test are evaluated. Lower roll angle means better comfort. Lower loss of speed means higher safety (shorter maneuver time) and also better efficiency (energy loss). Lower steering wheel rotation angle is considered as sportier and also safer behavior, allowing driver to turn faster with holding the steering wheel with both hands all the time. In the last test, the most important evaluation parameter is minimum D and maximum E tires slip (m/s). Lower slip is equal to better handling and safer ride. The test is passed when the car fits the 4.3m lanes during the entire test.

1) Moose test

The test scenario is to rapidly change a lane and go back to the original one on a straight road with velocity in range 80-200km/h (changing by 20km/h).

Experiments results are shown in Table II.

2) 90-degree turn

TABLE IV. 90-DEGREE TURN TEST RESULTS

enter speed	reference solution			proposed solution		
	A	***B***	***C***	***A***	***B***	***C***
10	0.2	2	253	0.2	-1	76
10	0.1	1	268	0.2	1	99
10	0.4	0	342	0.1	2	108
10	0.4	-2	268	0.2	0	104
20	0.8	1	372	0.2	-1	76
20	0.7	2	371	0.2	2	108
20	0.9	0	365	0.3	1	104
20	1	-1	312	0.2	0	98
30	1.2	2	290	0.2	2	96
30	1.2	1	246	0.2	1	94
30	1.1	-1	256	0.2	2	88
30	1.3	3	267	0.1	3	98
40	1.5	-10	160	0.2	-3	198
40	1.6	-8	381	0.2	-4	197
40	1.5	-12	271	0.3	-4	178
40	failed			0.4	3	174
50	1.9	-28	450	0.3	3	149
50	failed			0.2	-24	324
50	failed			0.3	-23	354
50	failed			failed		
60	failed			0.4	-38	450
60	failed			0.4	-39	450
60	failed			failed		
60	failed			0.4	-42	450

The test scenario is to turn right on a 90-degree intersection with velocity in range 10-60km/h (changing by 10km/h).

Experiments results are shown in Table III.

3) Long curve

In this test, the vehicle rides around a circle with a constant radius of 20m and speed in range 10-70km/h (changing by 10km/h). In this test, the capability of controlling all wheel speed separately is shown.

Experiments results are shown in Table IV.

V. CONCLUSIONS

When analyzing results, some considerations arise. The reference vehicle has failed in 29% of all 1st and 2nd tests' trials and the tested one failed in 4% of the same tests' trials. This is the main proof of improved safety in the presented solution. Secondly, the maximum roll of test vehicle is limited to less than 0.5 degree no matter of conditions, for all

TABLE III. LONG TURN TEST RESULTS

enter speed	reference solution		proposed solution	
	D	***E***	***D***	***E***
20	0.01	0.23	0.07	0.07
20	0.02	0.25	0.06	0.07
20	0.01	0.22	0.05	0.07
20	0.01	0.23	0.06	0.06
30	0.06	0.4	0.1	0.11
30	0.05	0.42	0.09	0.11
30	0.06	0.41	0.09	0.12
30	0.07	0.4	0.11	0.13
40	0.18	0.72	0.27	0.31
40	0.2	0.7	0.22	0.25
40	0.19	0.7	0.25	0.3
40	0.17	0.71	0.27	0.3
50	0.22	1.81	0.2	0.21
50	0.21	1.9	0.23	0.26
50	0.24	1.86	0.25	0.29
50	0.24	1.78	0.21	0.24
60	0.25	2.59	0.26	0.37
60	0.25	2.72	0.25	0.28
60	0.27	2.58	0.25	0.27
60	0.26	2.58	0.26	0,29
70	0.26	2.42	0.21	0.25
70	0.25	2.44	0.22	0.25
70	0.26	2.53	0.21	0.26
70	0.26	2.51	0.25	0.28

 ISBN: 978-1-61208-720-7

trials. The reason is the anti-roll bars work pro-actively, reacting to turn and speed, not to the roll itself. This behavior proves that the comfort of the ride improved.

The next thing to notice is that the maximum steering wheel rotation angle in test vehicle is significantly lower and fits into 90 degree for most cases. The cause is the steering wheel ratio being adjustable in a very wide range, due to the lack of physical connection (even simulated one). The function converting wheel angle and speed to the position of the center of a turn is adjusted to lower the minimum turn radius at high speed, when rapid turns are impossible anyway due to vehicle momentum.

The next observation is that the presented vehicle does not slow down during most of the tests, except the ones, when preserving speed is impossible, due to high vehicle inertia. The cause is that the driver does not press the brake, so all trajectories are calculated for the same speed. Stability is preserved with an active differential that transfers proper speed to all wheels to avoid a slip, with stiff connection between wheels and engine and without using brakes. This way, a maneuver can be finished faster, and the engine is never stalled by brakes, which also improves safety.

On the other hand, the reference vehicle uses brakes to preserve comparable stability which causes a significant loss of speed.

In the last test, the minimum slip, occurring for inner wheels, is comparable for both vehicles, but active differential implementation offered by VPP is not as effective as the tested one. Moreover, in different conditions, when outer wheels travel on grass instead of tarmac, the differences between the reference and test vehicle are even bigger (Figure 3). The reason is, again, that tested standard safety systems react by breaking wheels already slipping, and the test one controls the wheels behavior proactively, calculating its speed before any slip occurs using mainly engine and differential, not brakes. This result also proves the higher ride safety.

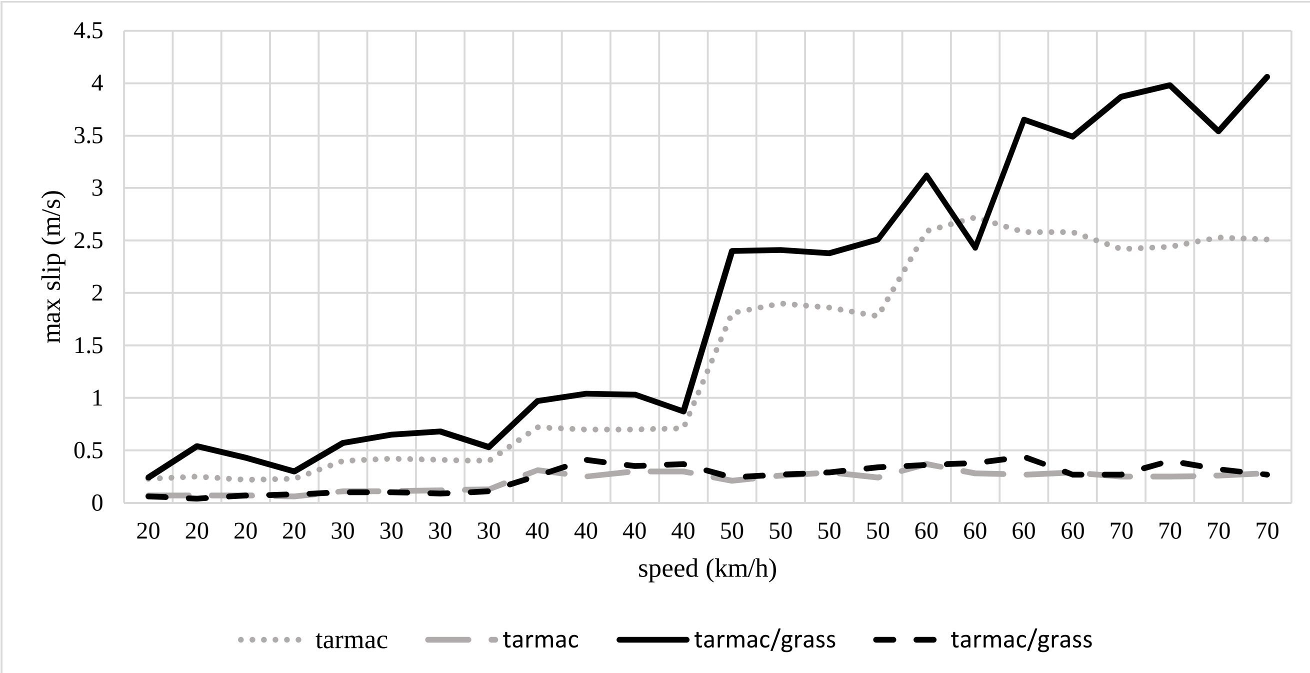

Figure 3. Slip test on different surfaces

VI. FUTURE WORK

Although the current results are promising, a lot of work is planned. For now, all processes are triggered by data or time. The event-base trigger (rapid condition change) is planned. Besides that, disruption analysis with fuzzy functions [21] usage will be introduced. When environment analysis process is established, the full evaluation will be conducted with more test cases and more drivers. And lastly, all processes implementations of the method use very simple algorithms so far, but they are designed to be replaceable, so the best combination is to be found.

Future experiments will be conducted using two computers with a direct network link. Physical experiments with real vehicles are not planned so far. Very sophisticated, highly equipped vehicle (with active differential, suspension, etc.) with an open access available to all in-vehicle actuators and sensors and also a large set of extra sensors are needed, which makes such experiments too expensive for the current stage of research. This kind of research is possible after full simulation evaluation.

 ISBN: 978-1-61208-720-7

REFERENCES

[1] J. Ernst, “Mercedes-Benz and the invention of the anti-lock braking system: ABS, ready for production in 1978”, in: Daimler Communications, July 2008.

[2] A. van Zanten, "Bosch ESP Systems: 5 Years of Experience" in: SAE Technical Paper 2000-01-1633, January 2000.

[3] M. Knorr, “Self-Calibration of Multi-Camera Systems for Vehicle Surround Sensing”, KIT Scientific Publishing, December 2018.

[4] B. Ranft, and C. Stiller, “The role of machine vision for intelligent vehicles”, in: IEEE Transactions on Intellifent Vehicles, Vol. 1, pp. 8-19, March 2016.

[5] M. Z. Liu, and M. Sun, “Application of Multidimensional Data Model in the Traffic Accident Data Warehouse”, in: Applied Mechanics and Materials, Volumes 548-549, pp. 1857-1861, April 2014.

[6] D. Badura, “Prediction of Urban Traffic Flow Based on Generative Neural Network Model”, in: Management Perspective for Transport Telematics. TST 2018. Communications in Computer and Information Science, volume 897, pp. 3-17, September 2018.

[7] Z. H. Ren, K. Zhang, L. X. Xue, and Y. L. Gao, “Mandatory Lane Change Model and Time Delay under Traffic Emergency Incidents”, in: Applied Mechanics and Materials, Vols. 644-650, pp. 2627-2631, September 2014.

[8] H. Abut, J. Hansen, G. Schmidt, K. Takeda, and H. Ko, “Vehicle Systems and Driver Modelling”, De Gruyter, September 2017.

[9] T. Nitsch, “Sensor Systems and Communication Technologies in Autonomous Driving”, GRIN Verlag, April 2017.

[10] S. Liu, L. Li, J. Tang, S. Wu, and J. Gaudiot, “Creating Autonomous Vehicle Systems”, Morgan & Claypool Publishers, October 2017.

[11] D. Vaishnavi, E. Sundari, T.V. Sangeetha, S. Shrinidhi, and P. Saravanan, “Design and Development of Computational Intelligence for Enhanced Adaptive Cruise Control Using Arduino”, in: Applied Mechanics and Materials, Vol. 852, pp. 782-787, September 2016.

[12] T. Denton, “Automobile Electrical and Electronic Systems”, Chapter 10, Routledge, July 2007.

[13] D. Hoffman, "The Corvette Acceleration Slip Regulation (ASR) Application with Preloaded Limited Slip Differential," SAE Technical Paper 920642, February 1992.

[14] E. D. Dickmanns, “Dynamic Vision for Perception and Control of Motion”, chapter 7, Springer, June 2007.

[15] P. G. Guest, “Numerical Methods of Curve Fitting”, Cambridge University Press, December 2012.

[16] C. Cheng, C. Liu, and C. Liu, “Unit Commitment by Lagrangian Relaxation and Genetic Algorithms”, in: IEEE Transactions on Power Systems, Vol. 15, pp. 707-714, May 2000.

[17] A. Yenter, “A Multi-kernel Convolutional Neural Network with LSTM for Sentimental Analysis”, ProQuest, 2017.

[18] https://unity3d.com/, last retrieved May 2019.

[19] https://vehiclephysics.com/, last retrieved May 2019.

[20] https://www.logitechg.com/pl-pl/products/driving/drivingforce-racing-wheel.html, last retrieved May 2019.

[21] J. J. Buckley, and E. Eslami, “Fuzzy Functions” in: An Introduction to Fuzzy Logic and Fuzzy Sets. Advances in Soft Computing, Physica, 2007.