

ANALYSIS OF STORED DATA HELP TO PROPOSE AND GENERATE NEW TRACKS

VÁCLAV JIRKOVSKÝ

Czech Technical University in Prague, Faculty of Transportation sciences, Department of vehicle technology.
Kovarikská 20, 110 00 Prague

correspondence: vaclav.jirkovsky@volny.cz

ABSTRACT. During experiments on vehicle simulators a large amount of data is stored. On these data, it is possible to trace some similarities in the behavior of drivers in certain areas or when performing the same task. We can assume that if the driver performs a certain type of experiment, his behavior exhibits certain traits. These elements of common behavior can be used to create virtual track for experiments. Which elements and how they can be used is described in this article. The algorithm for automatic creation of virtual track based on type of experiment is provided. It will help us to define the purpose of measurement and the track could be generated automatically.

KEYWORDS: Vehicle simulator, measurement, virtual track.

1. INTRODUCTION

Conducting of experiments is one of the most important things related to the operation of driving simulators at the Faculty of Transportation Sciences. There are many types of them. They are focused on investigating the driver behavior in fatigue[1–3], testing systems for predicting microsleeps[4], human-machine interaction[5], interaction with surrounding traffic[6, 7] and many more. During their performing the data are collected, which are mostly of a technical nature, and on the other hand, they give us information about the state of the monitored driver. All information about a vehicle, terrain and environment belong the first category. The data collected on a driver's body - such as heart beat, EEG, eye view direction, blink rate etc. belong to the second category[8]. After the data evaluation we are able to describe a driver's behavior in particular states of experiment and determine a crisis situations or areas. Hierarchical structure of measured data is shown on Fig.1

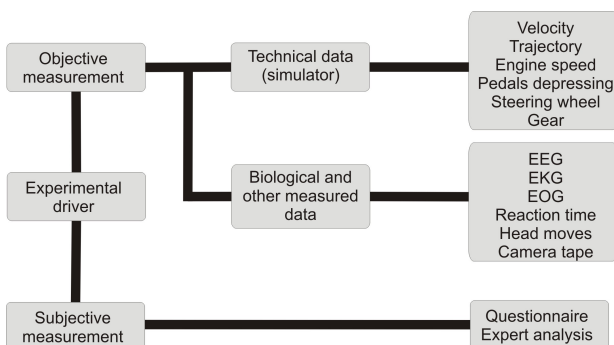


FIGURE 1. Hierarchical structure of measured data

2. MEASURED DATA ANALYSIS

If the amount of the measured data is sufficiently large, we are able to detect with help of mathematical tools how the drivers react to a given event, incentive or transport situation. We can find out if their reactions are different or identical. It is possible, based on the analysis and using of appropriate tools, to predict the behavior of drivers on virtual tracks if these tracks are only modeled or created without performing of experiments with data collection[9]. On the basis of structure and character of the track, it is possible to determine the place and the moment when the driver will have to solve the crisis situation. But the opposite procedure will be applied. It means to design and construct a virtual track according to the requirements of the type of crisis.

On thus obtained results we can create new scenarios with a specific focus. For example the results of the analysis shows that while operating the radio in rugged terrain, a large percentage of drivers do not hold the vehicle on an ideal path and run off the road. It's evident that the equally segmented track is suitable for testing similar devices. Based on these derived addition it is possible to design and construct virtual tracks with greater efficiency. It will also allow it to specialize in one specific purpose.

The figure No.2 shows the experimental track. The yellow marks show the places where the value of deviation from the ideal track overlaps 1.5 m. This value indicates that the car doesn't stay in the correct road lane and crosses the middle line to the opposite road lane. This can cause a serious traffic situation with possibility of frontal collision.

The chart shows that the deviation from the ideal track is caused by splitting the focus of the driver between the driving and the device manipulation. For more detailed evaluation we can divide the tasks into easy and difficult. An easy task can be done by one

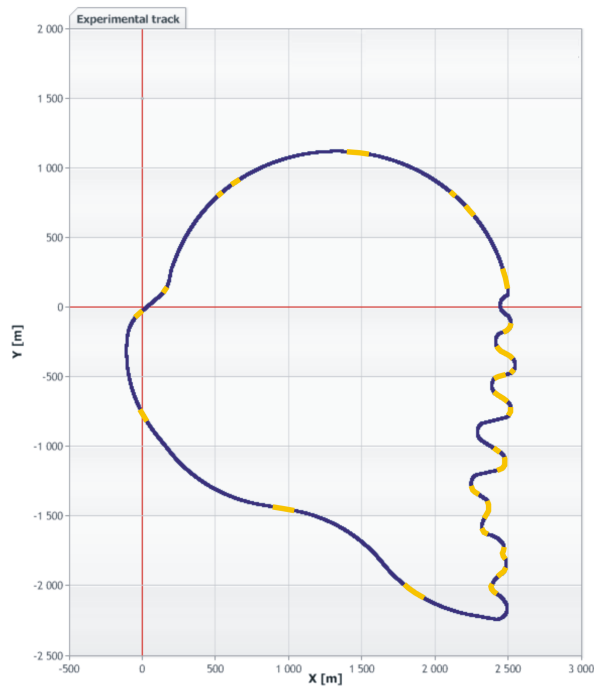


FIGURE 2. Experimental track

or two clicks (touch). Difficult task needs more time to finish.

3. IDENTIFYING COMMON ELEMENTS OF BEHAVIOR

During data analysis I've found out that the common behavior patterns could be observed for most types of experiments. With a very high probability we can assume that if the driver performs the measurements under the same conditions as the previous driver, his behavior will show the same elements. By examining the experiments I received the following connection:

- Assistive devices - ragged track
- Research on driver fatigue - monotonous, minimally curved track with minimal distractions
- Research driver's responsiveness to sound stimuli - monotonous track
- Research driver's responsiveness to obstacles in riding - complex traffic situations, poor visibility
- Driving precision - special testing polygon (slalom, braking precision, ride along a line etc.)
- Overtaking research - track with poor clarity (bend, curl, heavy traffic)

Individual connections determine the characteristics. With their help, I set up a knowledge base. That gives us an idea of which features driver's behavior occur in different types of experiments. That will tell us what behavior we expect on a particular experiment, but in the opposite case it will help us, when examining on the basis of activity to determine the shape of

TABLE 1. Experimental characteristics

the track. Characteristics of each experiment can be found in the following table.

Experiment	Properties							
	Complex track	Recommended velocity	Surrounding traffic	Light traffic	Heavy traffic	Good visibility	Corrugated track	Objects
Reaction speed on sound effects	NO	-	NO	-	-	YES	NO	NO
Reaction speed on road obstacles	YES	-	YES	NO	YES	NO	-	YES
The accuracy of the ride	YES	-	NO	-	-	-	NO	NO
Driver's fatigue	NO	90-130 km/h	NO	-	-	YES	NO	NO
Assistance devices handling	YES	50 km/h	NO	-	-	-	-	-
Overtaking	YES	-	YES	NO	YES	NO	YES	-

4. PROPOSAL TRACK BY TYPE OF EXPERIMENT

A knowledge base is essential for the design of the track, which I described in the last chapter. To create the track I proposed algorithm, which is described below. The entire process can be divided into several steps:

- The user determines variables, which aims to examine
- Distinguishing features are loaded from knowledge base
- The track is automatically proposed based on those distinguishing features
- The result is a polyline characterizing the road axis in space

First the user enters input data. It is primarily a type of experiment. The next steps of propose are fully automatic and the user cannot affect them. Based on input values the distinguishing features are obtained from the knowledge base. It means maximal velocity, curvature, longitudinal profile etc. These go to input of an algorithm as a specific values (arcs with radius from 100m to 350m, narrow lines of length from 200m to 1000m etc.). Based on the input data the algorithm proposes the whole track. Geometric properties of individual building elements comply with standard CSN 73 6101. The result is a polyline that determines the

points of the track. It can be subsequently imported into other software where it's finalized[10].

5. THE ALGORITHM FOR AUTOMATIC GENERATION OF EXPERIMENTAL TRACKS

The automatic generation of a track is as old as the game industry itself. The limitation of existing platforms did not allow the distribution of a large pre-defined content. Ad-hoc algorithmic procedures were widely used to generate game content on-the-fly. It was called procedural content generation. Now, when the distribution of games is not limited by memory space, the automatic generation of the content is still used mainly to reduce design costs.

The principle of automatic generation of the track is based on evolutionary computation to evolve track for a simple two-dimensional car racing game[11]. The basic idea is to connect individual segments (lines, arcs) so that they don't cross the other parts. And they gradually point to beginning. The connection of the parts is strictly defined. For example between each line and arc must be placed a spiral. The segment properties are based on the rules mentioned above. Additional parameters - slope - can be added to extend into 3D space to make virtual track more realistic[12].

6. PSEUDO-CODE

As described in section four, the type of track is dependent on the type of experiment. It means each track consists of segments with a specific property. The track properties are set at the beginning of the algorithm. In the next step the random segment is randomly selected from the segment list. The list consists of narrow lines of different length and arcs of different length and radius. The end point of the previous segment is the start point of a next segment. The algorithm takes into account the crossing of the segments. If the actual segment crosses the track it is not placed and other segment is chosen. In some cases the algorithm doesn't offer the closed track, therefore, it must be done manually. To reduce the occurrence of these cases, the number of right-handed arcs is bigger than the number of left-handed arcs. Once the track has been built, it needs to be smoothen. In this step the spirals are placed between a line and an arc to provide a fluent transition.

- (1.) var property = SetTrackProperty()
- (2.) Track=null
- (3.) Object=choose random object from list
- (4.) Object.Properties=random(property)
- (5.) End_point=Object.EndPoint()
- (6.) Start_point=Object.StartPoint()
- (7.) Track.Insert(Object)
- (8.) Do while (Start_point != point)

- (9.) Object2=choose random object from list
- (10.) Object2.Properties=random(property)
- (11.) Object2.Start_point=End_point
- (12.) point=Object2.EndPoint()
- (13.) If(Collision(Track, Object2) == false AND SuitablePosition(Object2, Start_point) == true)
- (14.) Track.Insert(Object2)
- (15.) Calculate parts connection
- (16.) If(Distance(Start_point, point) < minimal_distance)
- (17.) break
- (18.) Else
- (19.) point=End_point
- (20.) Loop
- (21.) Connect Point and Start_Point
- (22.) Track.Smooth()

7. CONCLUSIONS

The measurements on a vehicle simulator and data analysis confirmed that the use of assistance and media devices can directly cause a dangerous behavior of the driver. Those devices should be designed to reduce driver's attention and thereby minimize dangerous situations on the road. Thorough tests could help us to determine which devices are suitable for using in a car and which are not.

The analysis also showed that there are connections between driver's behavior and track's properties and conditions of the measurement. The connections define rules that can be used during creating of the new tracks. In the future we will be able to define purpose of measurement and the track will be generated automatically.

The described algorithm for creating a track is one of many possible. It's easy to understand and fulfills the purpose. A number of track were generated during testing. Many of the tracks were useful and usable as a basis of a new created track. Their use is one of the steps to improve the quality of experiments on vehicle simulators.

REFERENCES

- [1] P. Bouchner, R. Pieknik, S. Novotny, et al. *Fatigue of car drivers – Detection and classification based on the experiments on car simulators*. 2006, WSEAS Transactions on Systems, 5 (12), pp. 2789-2794.
- [2] P. Bouchner. *A complex analysis of the driver behavior from simulated driving focused on fatigue detection classification*. 2006, WSEAS Transactions on Systems, 5 (1), pp. 84-91.
- [3] P. Bouchner, M. Hajny, S. Novotny, et al. *Car simulation and virtual environments for investigation of driver behavior*. 2005, Neural Network World, 15 (2), pp. 149-163.

- [4] P. Spurny, J. Andrs, P. Bouchner, et al. *Testing a system for predicting microsleep*. 2016 Lekar a Technika, 46 (2), pp. 51-54.
- [5] P. Bouchner, S. Novotny. *System with driving simulation device for HMI measurements*. 2005, WSEAS Transactions on Systems, 4 (7), pp. 1058-1063.
- [6] M. Matowicki, O. Pribyl, P. Bouchner. *Pragmatic overview of surrounding traffic implementation into driving simulator*. ELEKTRO 2016 – 11th International Conference, Proceedings, art. no. 7512111, pp. 423-428.
- [7] H. Klee. *Microscopic car modeling for intelligent traffic and scenario generation in the UCF driving simulator*.
- [8] P. Bouchner. *Driving Simulators for HMI Research*. PhD. Thesis, CTU.
- [9] R. Pieknik. *The Methodology of Creating Scenario Roads of Driving Simulator for Different Types of Experiments*. Conference of Driver-Car Interaction and Interface 2009.
- [10] A. Orlicky. *Automatic generation of road infrastructure in 3D for vehicle simulators*. Diploma thesis, Prague 2016.
- [11] J. Togelius, R. de Nardi, S. Lucas. *Making racing fun through player modeling and track evolution*.
- [12] D. Loicono, L. Caramone, P. Lanzi. *Automatic track generation for high-end racing games using evolutionary computation*.