TRAFFIC SIMULATOR WITH NEURAL NETWORK DRIVER

Václav Sáblík
Doctoral Degree Programme (2), FEEC BUT
E-mail: sablik.vaclav@phd.feec.vutbr.cz

Petr Polách
Doctoral Degree Programme (2), FEEC BUT
E-mail: polachpetr@phd.feec.vutbr.cz

Jan Pohl
Doctoral Degree Programme (1), FEEC BUT
E-mail: xpohlj@stud.feec.vutbr.cz

Supervised by: Václav Jirsík
E-mail: jirsik@feec.vutbr.cz

ABSTRACT

The article deals with discrete-time simulation of multi-agent car systems. The entire concept of modular simulator is presented and its features are shown. Furthermore, the simplified model of vehicle dynamics is described, and the possibility of the usage of neural network as a driver simulation is drafted and analyzed. Models are also evaluated.

1. INTRODUCTION

In the past decade, the road traffic intensity has been increasing rapidly in almost every city on Earth. Road traffic monitoring has become the focal point of interest, and evaluation of gained results can considerably help to improve road safety. In spite of the results shows the way, in which the traffic flow can be optimized, certain analysis such as car crashes still have to be simulated.

The complex simulation represents a very complicated problem, and implementation always depends on what kind of results is required. In our case the study deals with organizational character of traffic situation, therefore only two-dimensional environments are used. Concerning that, the vehicle representation was chosen as a simple polygonal model, and for every element in simulation are important shapes of objects and key factors rather then accurate graphic representation.

2. SIMULATOR

Figure 1 shows layout of the application with section of road in the view port field. Design concept does not focus on multi MDI windows for adjusting parameters of elements, hierarchical structure of objects with using of properties features enables both of reading or writing values with property editor, which makes control simple and intuitive. The whole application is developed on .NET platform in C# language with the use of modern standards for object-oriented programming. For example, polygonal models of objects and entities in simulation are stored in XML form using .NET serialization system. It is also remarkable that program runs on multi-thread engine, which enables utilization of parallel computing if needed. [1] As was mentioned, the purpose of application is realistic simulation of vehicle models and driver behavior. It is obvious that driver and its vehicle characteristics are specific for every single case, and the simulation concept have to allow adjusting driver or vehicle qualities.

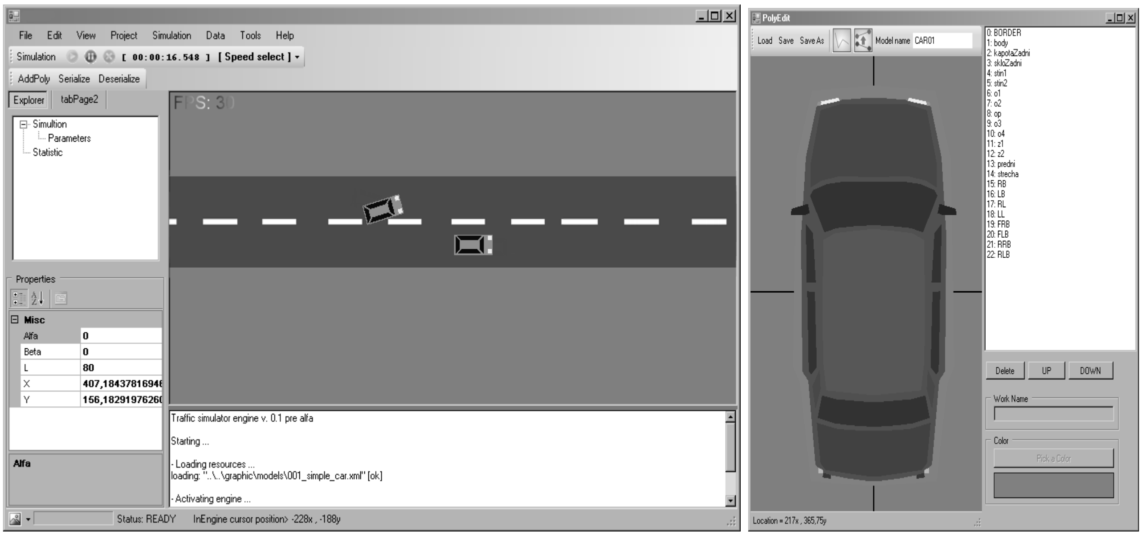

Fig. 1: Application screen

2.1. ENGINE

Specifically designed engine provides real-time (with respect to hardware capability) or full-speed simulation. Distribution of time is optional, starts at 1ms step and should follow users needs. As can be seen on simplified class-diagram of application (Fig 2.), there are two accumulators for static and dynamic elements. Every accumulator contains references to the base classes of stored elements. Function of the static objects, like a road or trees, is just non-dynamic. That means, the object has its location and shape defined but no state values are present.

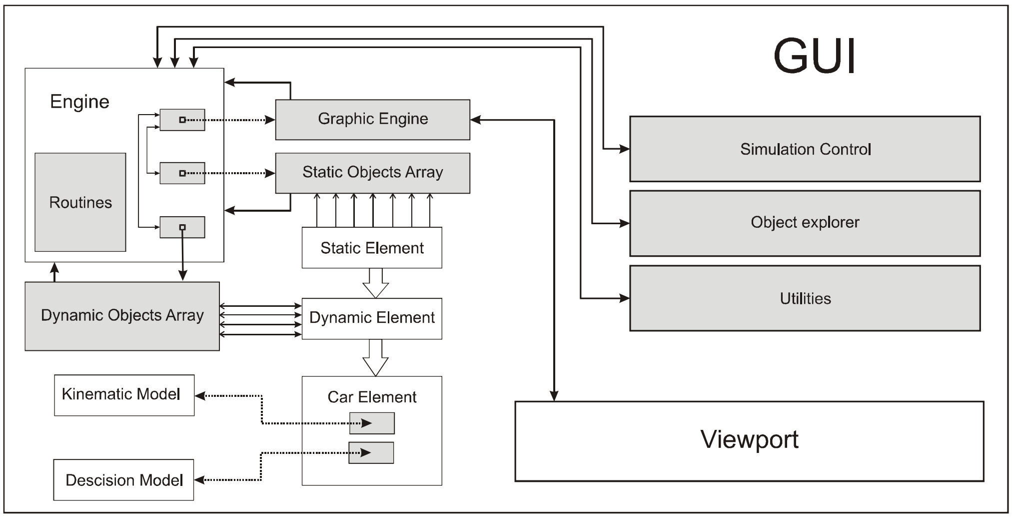

Fig. 2: Simulator application block diagram

There are two important dynamically adjustable properties at the level of *Car element* class. At first, *Kinematical model* class provides possibility of making own vehicle behavior based on different physical models, which are implemented. At second, *Decision model* class includes the control system of the vehicle, and represents driver model with his individual behavior. Both of these classes can be set individually for every vehicle element in simulation. The intention was to build universal open platform for traffic simulations with simple extensibility for vehicle and driver behavior in form of external models. [2]

3. DRIVER MODELING

It is evident that modeling of driver is now divided on two separated parts equally to the real systems where driver controls the vehicle. The simulation platform is universal so we have to restrict models for at least one case suitable for neural network as a part of driver control system.

3.1. VEHICLE DYNAMICS

At first, vehicle dynamic have to be described in form of equations, which converts the state of the vehicle in time t to the state in time $t + \Delta t$. The time is distributed trough simulation in discrete-packages, and it is an input value to the vehicle model, too.

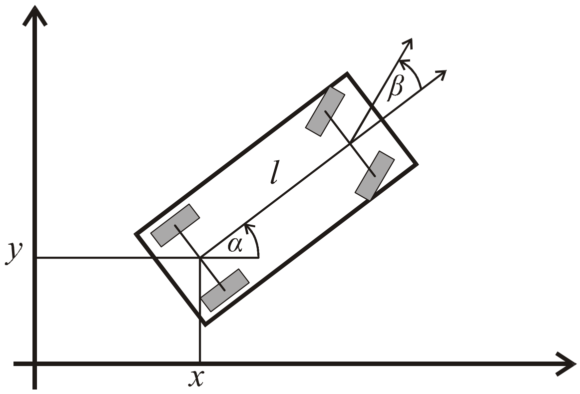

Fig. 3: Vehicle state description

Figure 3 shows the other state values of the vehicle. The location in two-dimensional space is defined by coordinates x and y at the center of the rear axle. In addition, non-state value l represents distance between rear and front axle. The angle α is relative rotation relate to global coordinate system, and the angle β is relative rotation of front wheels relate to the vehicle coordinate system. With this description should be vehicle dynamic described by following equations (1). [3]

$$\begin{bmatrix} x(t+\Delta t) \\ y(t+\Delta t) \\ \alpha(t+\Delta t) \\ \beta(t+\Delta t) \end{bmatrix} = \begin{bmatrix} x(t) \\ y(t) \\ \alpha(t) \\ \beta(t) \end{bmatrix} + \begin{bmatrix} \cos\alpha(t)\cos\beta(t) \\ \sin\alpha(t)\cos\beta(t) \\ (\sin\beta(t))/l \\ 0 \end{bmatrix} v_1(t)\Delta t + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} v_2(t)\Delta t \qquad (1)$$

Besides the time step value, there are two velocity values v_1 and v_2 as the input parameters where both of them can reach positive or negative values. The velocity v_1 indicates a speed of vehicle where positive value means moving forward and negative means moving backward. The other velocity v_2 indicates how fast are the front wheels turning left (negative values) or right (positive values). It is evident that in this case is vehicle model simplified to one-track vehicle with one wheel in center of each axle. Any more complex model should be implemented but for our reasons is this description sufficient.

3.2. Neural network driver

When the vehicle dynamic is set, we should use neural network for driver simulation. In hierarchical structure of control is neural network placed between vehicle model and decision algorithm, which defines the path the vehicle is following. Usage of this implementation is in simulation trustful driver reactions obtained from measured characteristics. These characteristics are also used as training data for neural network that means the real driver is a “black-box” to us and we do not have to analyze behavior for trustful modeling. Another kind of driver or another mood of the same driver is only a new set of weights of neural network.

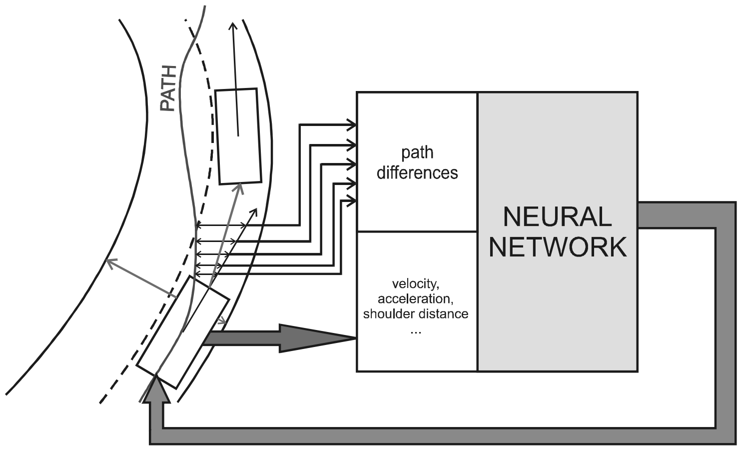

Fig. 4: Neural network driver - input values

There are many way how to learn neural network and many techniques of its usage. The presented solution (Fig. 4:) gives two kinds of inputs to the neural network. One of them is a vector of differences between actual direction of vehicle and shape of path which is the vehicle following. The other set of inputs represents distances between shoulders of road, vehicles border and distance of nearest car at the front, which is suitable when path collides with some objects. Output vector of the network contains velocities, used as input factors to the vehicle model at the same time. In addition, there is a good reason for implementing time factor as an additional input for synchronizing neural network reactions with actual time step, but is better not to change time step during the simulation if no adaptation algorithm is implemented.

Our model of neural network was multilayer-perceptron with many different topologies where number of neuron was in range not going over a hundred of neurons because it was sufficient for our purpose.

4. CONCLUSIONS

The suggested implementation has worked correctly. It is remarkable that there is many ways how to use standard regulation for shape following with known system of vehicle dynamics and in fact those methods would do better fit. However, what makes neural networks useful is the way of learning which comes out form conditions of real measured data. It is very hard to design a system that will be approximating measured characteristics on very abstract level of knowledge about the real driver. In many cases, no heuristic information is present over all. That is the part where neural networks are successful, the part where we have incomplete information about systems. Moreover, some disadvantages still remains such as number of computations needed for neural network operations. However, the simulation is not hard real-time at all.

Structure of simulation engine, which separates driver from vehicle, allows using neural network as a model of vehicle, too. This will be a great deal when all that we know about simulated vehicle is described in form of measured characteristics, and according to a theoretical assumptions, even a network with a small amount of neurons should greatly simulate a real driver or vehicle. With respect to fact, that all these features can be combined together, we receive a very promising platform, which capabilities go over the usage of stand alone neural network and soon will meet other technologies like fuzzy systems as well.

ACKNOWLEDGEMENTS

The paper has been prepared with the support of the research plan MSM0021630529 “Intelligent systems in automation”.

REFERENCES

[1] Nagel Ch., Evjen B., Glynn J., Watson K., Skinner M., Jones A.: C# 2005 Programujeme profesionálně. Computer Press, 2006, ISBN: 8025111814.

[2] Buckland M.: Programming Game AI by Example. Wordware Publishing, Inc., 2004, ISBN: 1556220782.

[3] Gu D., Hu H.: Neural Predictive Control for a Car-like Mobile Robot, International Journal of Robotics and Autonomous Systems, Vol. 39, No. 2-3, 2002.

[4] Novák M. a kol.: Umělé neuronové sítě – teorie a aplikace. C. H. Beck., Praha, 1998.

[5] Šíma J., Neruda R.: Theoretical issues of neural networks. MATFYZPRESS, Praha, 1996, ISBN 80-85863-18-9.

[6] Richtr T.: Řízení autonomního vozidla pomocí umělé neuronové sítě. Diploma thesis, FI MUNI, 2005.