CZECH TECHNICAL UNIVERSITY IN PRAGUE
FACULTY OF ELECTRICAL ENGINEERING

DIPLOMA THESIS

Cooperative Agents for Cars Drive Simulation

Simulace jízdy automobilů pomocí kooperujících agentů

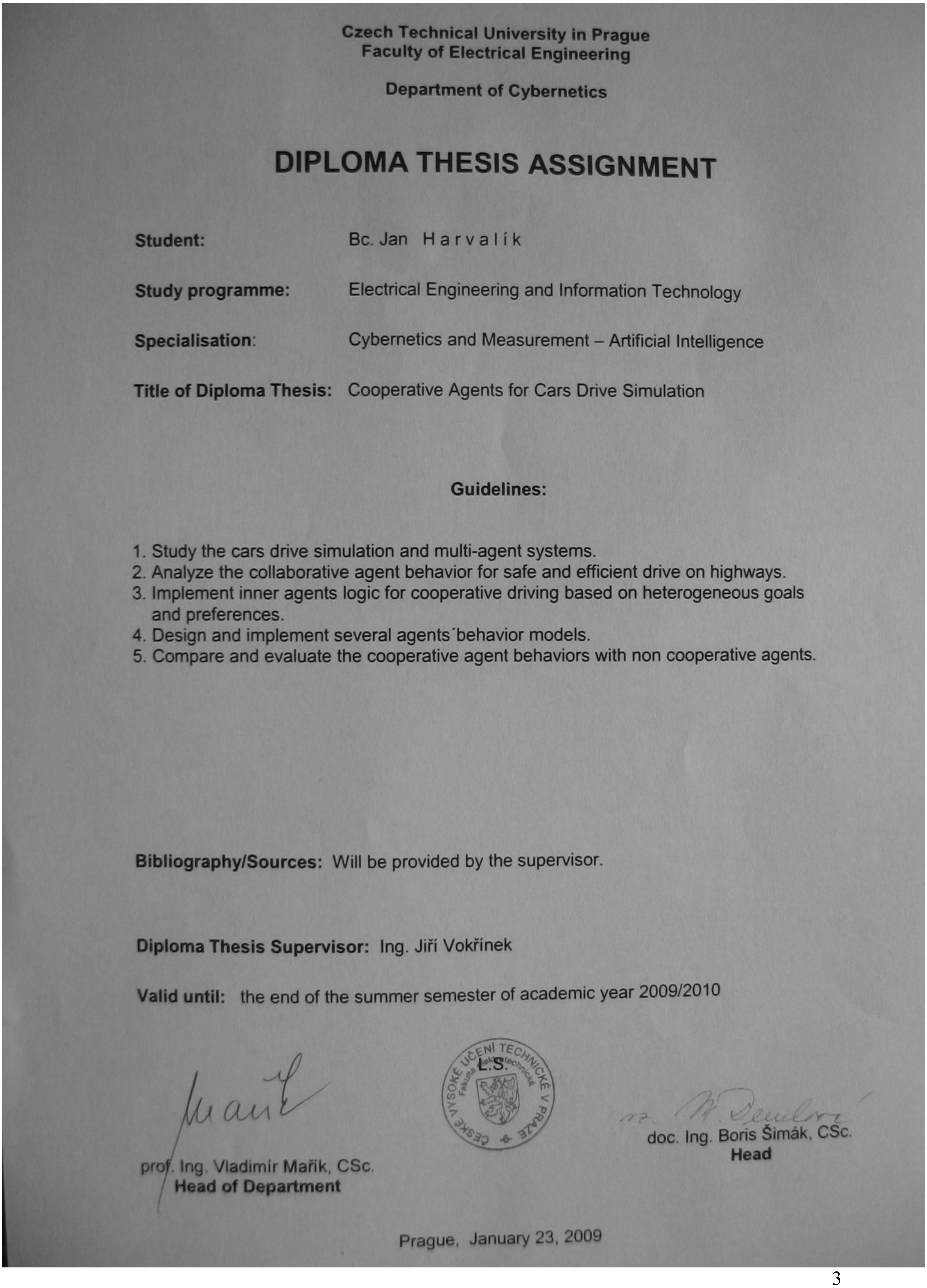
PRAGUE, 2009 Bc. JAN HARVALÍK

Prohlášení

Prohlašuji, že jsem svou diplomovou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW, atd.) uvedené v přiloženém seznamu.

V Praze dne 22. 5. 2009

Bc. Jan Harvalík



Czech Technical University in Prague
Faculty of Electrical Engineering

Department of Cybernetics

DIPLOMA THESIS ASSIGNMENT

Student:	Bc. Jan Harvalík
Study programme:	Electrical Engineering and Information Technology
Specialisation:	Cybernetics and Measurement – Artificial Intelligence
Title of Diploma Thesis:	Cooperative Agents for Cars Drive Simulation

Guidelines:

1. Study the cars drive simulation and multi-agent systems.
2. Analyze the collaborative agent behavior for safe and efficient drive on highways.
3. Implement inner agents logic for cooperative driving based on heterogeneous goals and preferences.
4. Design and implement several agents´behavior models.
5. Compare and evaluate the cooperative agent behaviors with non cooperative agents.

Bibliography/Sources: Will be provided by the supervisor.

Diploma Thesis Supervisor: Ing. Jiří Vokřínek

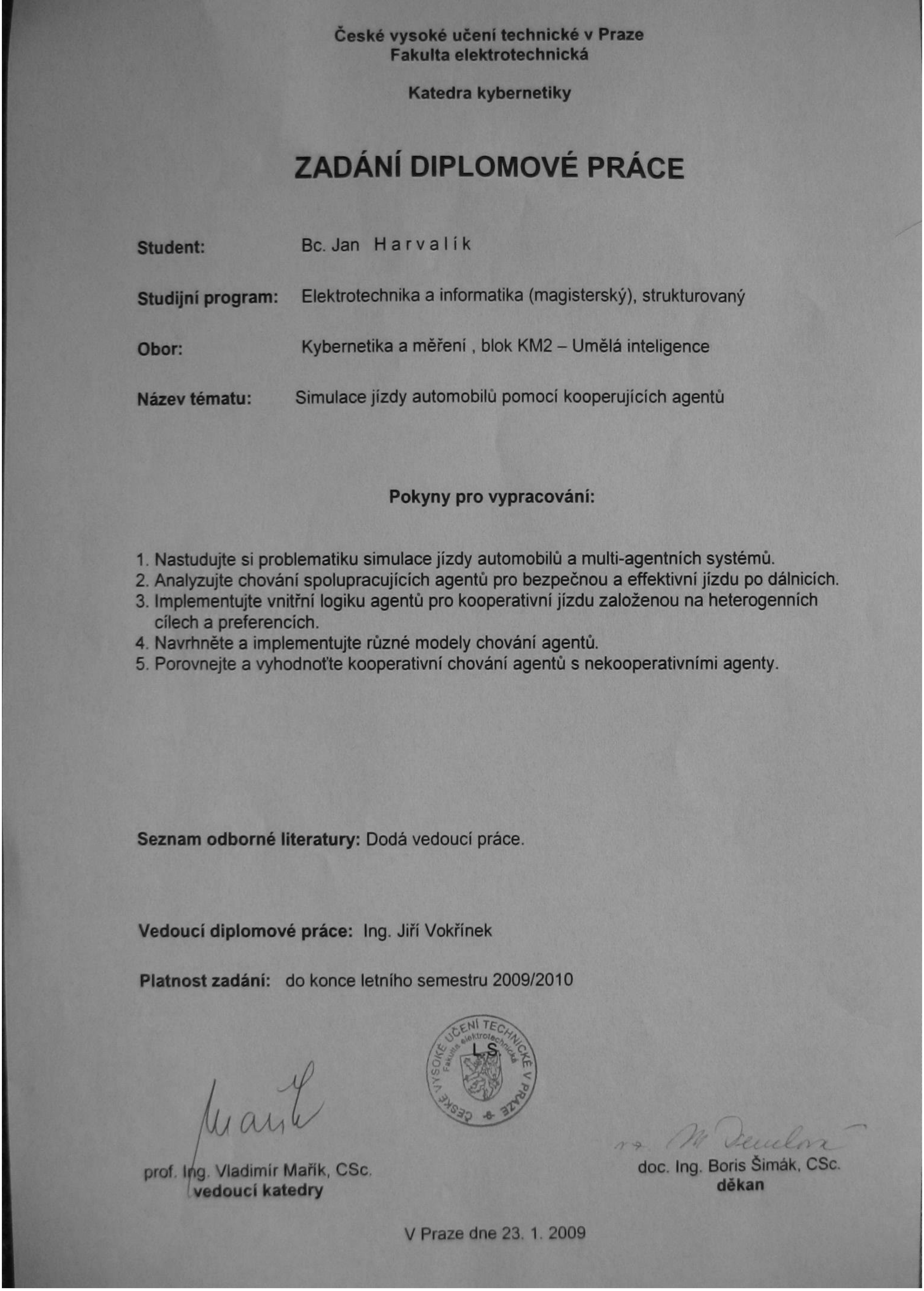
Valid until: the end of the summer semester of academic year 2009/2010

L.S.

prof. Ing. Vladimír Mařík, CSc.
Head of Department

doc. Ing. Boris Šimák, CSc.
Head

Prague, January 23, 2009



České vysoké učení technické v Praze
Fakulta elektrotechnická

Katedra kybernetiky

ZADÁNÍ DIPLOMOVÉ PRÁCE

Student: Bc. Jan Harvalík

Studijní program: Elektrotechnika a informatika (magisterský), strukturovaný

Obor: Kybernetika a měření , blok KM2 – Umělá inteligence

Název tématu: Simulace jízdy automobilů pomocí kooperujících agentů

Pokyny pro vypracování:

1. Nastudujte si problematiku simulace jízdy automobilů a multi-agentních systémů.
2. Analyzujte chování spolupracujících agentů pro bezpečnou a effektivní jízdu po dálnicích.
3. Implementujte vnitřní logiku agentů pro kooperativní jízdu založenou na heterogenních cílech a preferencích.
4. Navrhněte a implementujte různé modely chování agentů.
5. Porovnejte a vyhodnoťte kooperativní chování agentů s nekooperativními agenty.

Seznam odborné literatury: Dodá vedoucí práce.

Vedoucí diplomové práce: Ing. Jiří Vokřínek

Platnost zadání: do konce letního semestru 2009/2010

L.S.

prof. Ing. Vladimír Mařík, CSc.
vedoucí katedry

doc. Ing. Boris Šimák, CSc.
děkan

V Praze dne 23. 1. 2009

Abstract

This diploma thesis concerns the agent simulation. Its main purpose is to achieve a safe and comfortable driving on a highway. The basic rules for movement on a highway are described so that the agents can assert their individual preferences and achieve many different goals. This project is not an accurate description of neither the physicals parameters of the movement on a highway by kinematics and dynamics, nor the exact building of a highway. The main goal is to compare cooperative and non-cooperative agent behaviour and explain their influence to density and flow of the highway traffic and to find out which properties can improve the vehicle-to-vehicle negotiation, interaction and communication with their vicinity.

Abstrakt

Tato diplomová práce se týká agentní simulace dálnice, jejichž hlavním účelem je bezpečná a pohodlná jízda na dálnici. Jsou zde popsána základní pravidla pohybu na dálnici, tak, aby agenti mohli uplatnit své individuální preference a dosáhnout mnoha odlišných cílů. Tento projekt není pečlivým popisem kinematických a dynamických fyzikálních jevů, ke kterým může docházet při pohybu po dálnici, není ani popisem stavby silnic a dálnic. Hlavním cílem je porovnat kooperativní a nekooperativní chování agentů a ozřejmit jejich vliv na hustotu a propustnost provozu, zjistit vlastnosti, které může silniční dopravě přinést mezi vozidlová komunikace a výměna informací s okolím.

Table of contents

Introduction

The increasing amount of vehicles and the demand for mobility in our society poses challenges to both computer science and artificial intelligence. It is not always possible to build an additional capacity of highways, so a more efficient use of the available transportation infrastructure is necessary. This project engages in the domain of road traffic. Our main topic is: how can information be helpful for creating a fluent road traffic and an efficient use of highways?

The idea is that the information received from the vehicle environment and the information about the manoeuvres is transmitted from one car to the other cars in the vicinity. This allows early reactions like braking – e. g. if a vehicle has to use emergency brakes in front of another – and so helps to prevent rear-end collisions. This is the basis for our project of highway traffic simulation.

The agent-based simulation is very suitable for this domain because of its distributive properties. The agents are entities that are capable of perceiving their environment and carrying out goal-directed actions. Each agent can describe one vehicle or an intelligent traffic sign. The agents have their own computing devices, they are highly adaptable and they react to changes in the environment on an individual level.

We used Aglobe [1] for implementation of this project, which is suitable for real-world simulations including mobile units (e.g. vehicles) and static units (traffic signs). Aglobe [2] is an agent platform designed for testing the experimental scenarios featuring agent position and communication inaccessibility (range of vision of a vehicle, range of the transmitter and the receiver).

State of the art

Due to increasing car traffic, the effort to improve it and the financial background of such research is also increasing. At the same time there is a pressure on the development of such technologies which will improve the road safety and so eliminate property damage and loss of life.

The following projects are examples of where the research can lead, both theoretically and practically.

AHSRA

The Advanced Cruise-Assist Highway System Research Association (AHSRA) [3] develops its road infrastructure (AHS) in the following phases: “danger warning” (AHS-i), “assistance for driving” (AHS-c) and “automated highway system” (AHS-a). The developed system detects and recognizes the road traffic environment and gives a feedback to the driver.

It offers three levels of automatization, from a low-level mode, which gives the driver additional information about the environment, to a high-level mode, which means a completely automated driving.

Currently, the AHSRA system is in process of application to practical use.

Automated Highway System

This experimental system [4] is based on vehicle platooning. A platoon is a line of vehicles which drive successively together. Vehicles in the platoon exchange information about their states with each other and negotiate manoeuvres to guarantee safety and smooth lane change.

The platooning increases also the permeability of the highway, allowing the flow to be twice or three times greater, because the gap between vehicles can be much smaller than under normal circumstances. It also increases fuel efficiency and mainly the passenger safety, guaranteed by automatic coordination of the vehicles and their velocities.

This system was successfully tested and demonstrated in 1997, but it was terminated later due to a lack of funds. Its main disadvantage concerning the real-world implementation was the need for a construction of special joining lanes for new cars wanting to join an existing platoon.

Chauffeur

This project [5] focused mainly on the automation of heavy trucks using “electronic tow-bar” operations where a leading truck with human driver is followed by a second one, fully automatic. It can increase fuel economy and reduce traffic blockage caused by trucks.

Chaffeur II is currently implementing the capability to follow ay type of vehicle and to platoon multiple trucks.

Cellular automaton

Cellular automaton is a discrete model which consist of a grid of cells. Each cell influences its vicinity and this influence obeys the rules of the model. Due to their simple design, cellular automaton models are very efficient in large-scale network simulations. We will demonstrate it on following examples.

Base of application

Cellular automata [6] are models that are discrete in space, time and state variables. Because of the need for a discrete space the highway system has to be separated to cells usually with a constant dimension. Each cell is either empty or engaged by one vehicle or a block. The width of the cell is usually equal to the a width of a lane. The length of the cell is statically constant and equal to the average of free space occupied by the vehicle, its immediate vicinity necessary for the safe drive of the vehicle and its distance from another object on the highway.

If the highway is modified according to this approach, we obtain a table of cells, which size is the number of lanes to the length of the highway, divided by the length of a cell.

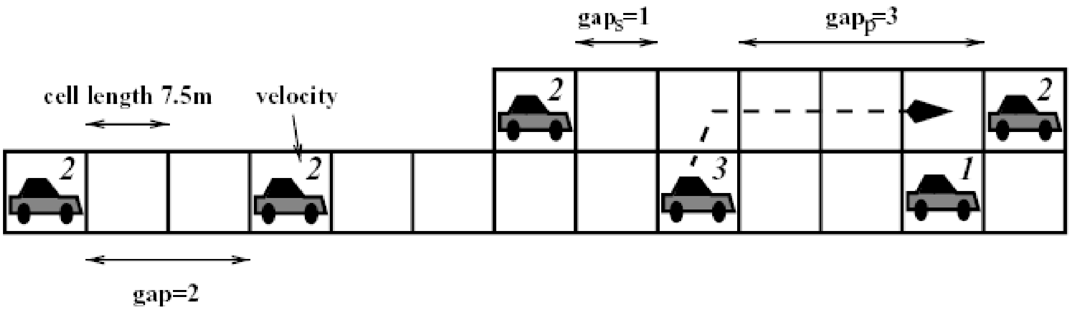
The vehicle movement is represented by a positive integer, due to the discretisation. This number is equal to the number of cells, which the vehicle passes during one simulation period, and has a bound, which denotes the maximum speed limit and in the simplest case is the same for all cars.

Rules of a single-lane traffic

Now one needs to specify the rules that define the temporal evolution of a given state. The simplest rule set, which leads to a realistic behaviour, consists of four following rules that apply at the same time to every vehicle (parallel or synchronous dynamics).

- **R1 Acceleration:** v_i ‹ $\min(v_i + 1, v_{max})$, for all cars that have not already reached the maximal velocity v_{max} acceleration by one unit
- **R2 Deceleration:** to avoid accidents: v_i ‹ $\min(v_i, gap)$, the gap denotes the number of empty cells in front of the vehicle; if a car has a lower count of the empty cells in front of it and its velocity v is larger then the gap, then it reduces its velocity
- **R3 Randomization:** with a certain probability p does v_i ‹ $\max(v_i - 1, 0)$
- **R4 Movement:** x_i ‹ $x_i + v_i$. After determining the new velocity v_n (velocity for one simulation period) for each car, a new position is determined, v_n cells ahead of the last position.

Figure 1: Description of movement in a cellular automaton



The first rule describes the desire of the drivers to drive as fast as it is possible (or allowed). The second rule encodes the interaction between the cars. In this simple model, the interactions only occur to avoid accidents.

The first two rules (R1, R2) describe a somehow optimal driving strategy. However, drivers do not always react in this optimal way: they overreact while decelerating or accelerate slower than they are able to.

This and the fluctuations of the velocity of the vehicles in the free flow is reflected by the braking noise p (R3). It corresponds – in a very simple way – to many complex effects that play an important role in the real traffic. E.g. a single car will not usually move with a constant speed, but there are always small fluctuations of the velocity. An important point are the overreactions when braking. A car that has to brake will, with a probability *p*, brake even more than necessary to avoid a collision. This kind of imperfect driving can lead to a chain reaction, if the density of vehicles is high enough. In the end it might lead to a stopping car and so to the creation of a traffic jam. It is also responsible for the spontaneous formation of jams, the so-called phantom jams, which occur without any obvious external reasons. It is also necessary to take into account the differences between cars and trucks. We introduced vehicles with varying v_{max}, i.e. we assigned lower maximum velocities to the trucks than to the cars. It shows the extreme importance of the step 3, which reflects the imperfect behaviour of the drivers. Finally, in the step 4, all cars move according to their new velocity.

Network structure

The highway is not composed of one lane only. The net of highways is a composite of junctions, intersections or triangular intersections. A few basic elements can be used to model a network of highways.

- **Nodes:** Connections between two links, or a sink/source at the boundary, or an entrance/exit ramp
- **Links:** Links are directional elements that connect the nodes.
- **Multi-lane links:** Main parts of the network. Every multi-lane link has its emission and absorption regions which are connected to the transfer links. These regions can be found at the beginning and the end of the multi-lane links.

Lane change

Free lane change

First, a vehicle checks if it is hindered by the predecessor in its own lane. This is fulfilled if gap < v. Then it has to pay regard to the security criterion, i.e., the gap to the successor gap and to the predecessor gap on the destination lane must allow a safe change. Then the vehicle moves to the other lane.

On- and off-ramps

On the highways the vehicles have to leave or enter the main road at the exit, respectively the entrance ramps. Thus, in this simulation we use merging areas to transfer the cars between the different links. Therefore, upstream of the exit ramp a preparation area is declared. This area is followed by an absorption zone. At the end of a transfer link an insertion area is defined. The vehicles drive on, up to this area and change line on the condition that there is sufficient distance to the successor on the destination lane.

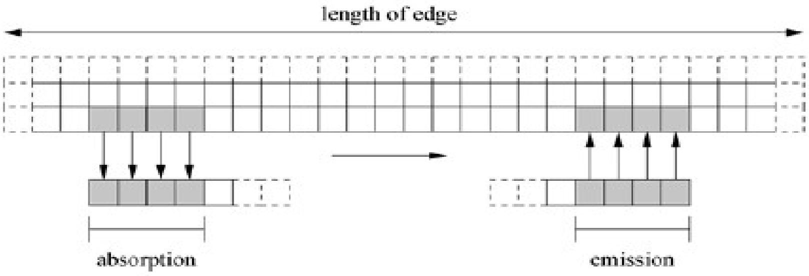


Figure 2: Exit and entrance ramps in a cellular automaton

Experimental application

The framework described above [7] is applied to the highway network of North Rhine-Westphalia, an area of about 34,000 km^2. The roads of the network have a length of 6,000 km. There are 67 highway intersections and 830 entrance and exit ramps. Every minute the aggregated amount of cars and trucks as well as their velocities are sent via permanent lines from the control centers to the simulation.

The simulation is supplemented with on-line data. Therefore, the algorithms have to be found to incorporate the real world measurements into the simulation. This is done at the so-called checkpoints, which are located at those places in the network where a complete cross-section is available, i.e. all lanes are covered by an inductive loop. In principle, the simulation results of the last minute have to be compared with the measured data and adjustments have to be made.

The driving force of the tuning strategy is the difference between simulated and real world data. In general there are two possible situations: If there are too many vehicles, some are removed. If the number of vehicles simulated is lower than the measured number, vehicles are added in an area up- and down-stream of the check-point. In this area, the mean gap of the vehicles is calculated. From the real world data a speed v in cells/time step is determined.

- Information about traffics density based on time
- Validation of the results of the simulator
- Dynamic route guidance systems

- Information about traffics density base on time
- Validation of the results of the simulator
- Dynamic route guidance systems

Conclusion

The cellular automaton is an interesting tool for highway simulation, especially in a domain, where we can study the impact of traffic density and flow to the highway net as a system. It is not suitable for a simulation from the vehicle point of view. The automaton is not meant and is not able to simulate the vehicle behaviour.

The cellular automaton achieves good results for the case of a highway point of view simulation. It means a successful approach to a realistic behaviour, a correct reproduction of especially occurring of the spontaneous jamming and the so-called fundamental diagram, i.e. the relation between flow and density. For small densities the flow is proportional to the density because there is almost no interaction between the cars. So they drive with their desired velocity $v_{\max}$ (up to fluctuations). At higher densities the interaction becomes more important and some deviations from the linear behaviour can be observed. Finally the interactions start to dominate and the flow decreases with increasing density.

Agent-based simulation

Agent system as a cellular automaton

This is one possible point of view for simulating a traffic system. The project [8] compared the agent and cellular automaton approaches to replicate the behaviour of a multi-lane traffic system (such as a highway) by integrating qualitative modelling concepts in a combined multiagent and cellular automaton approach. It proposed a spatial qualitative approach with several levels of abstraction and analyzed the neighbourhood of individual vehicles acting in a circuit of reference.

Traffic modelling

In the project described above, the rules for traffic modelling were identified according to two main principles, the so called “forward motion” and “lane changing rules”. The first one defines the way cars advance by accelerating and decelerating. The second one defines the rules for changing lanes, which again rely on two main criteria: the first indicates the need of the vehicle to change lanes in order to reach its maximum or optimal speed; and the second determines the possibility to change lanes if there is enough space in the target lane.

These approaches are generally calibrated and validated using the typical density inversion phenomena that mainly represents the fact that traffic flow increases nearly linearly with density until it reaches a maximum, from which the traffic flow decreases with increasing density.

Safe Highway System

This multiagent simulation, developed on the Aglobe platform is the direct predecessor of our project. The Safe Highway System [9] ses a highway divided distributively on individual segments, which are controlled by the simulation agent (the map agent). All entities situated on the highway are located in the same container; when passing from one highway to another, the agents also move between the containers. The highway representation resembles the cellular automaton, because the movement between the lanes is step-like. The highway was simplified to a straight segment, being the main disadvantage of the system, which complicated further expanding.

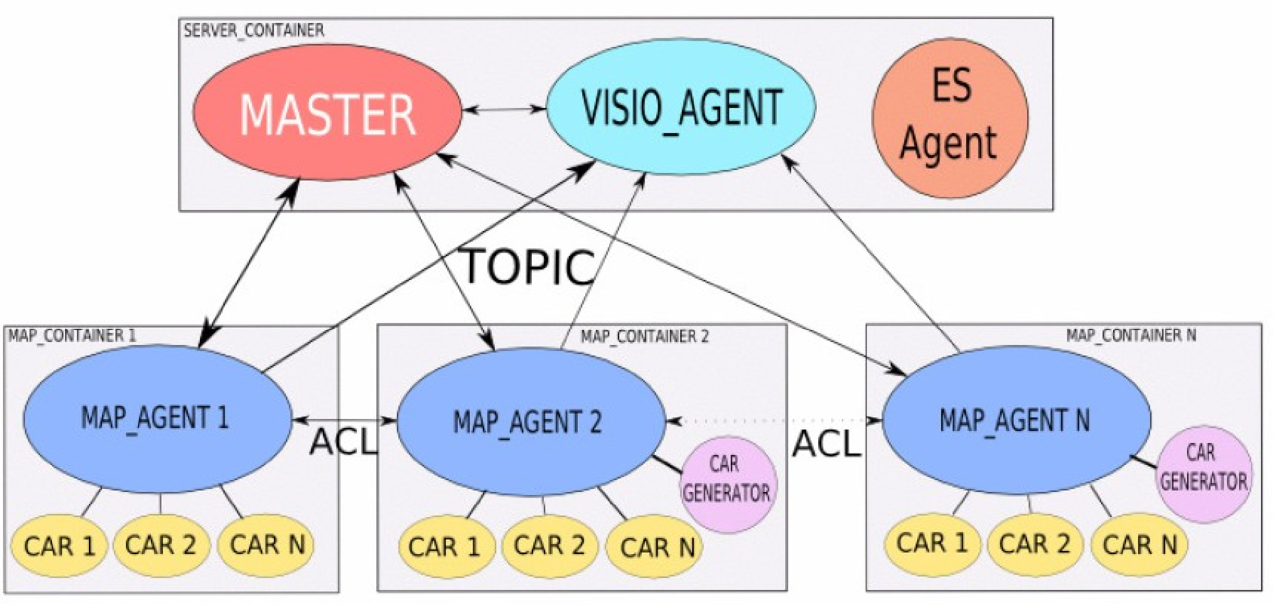


Figure 3: Architecture of the Safe Highway System

GA-INTACT

A genetic algorithm [10] search was adopted for the selection of the optimal tactical manoeuvres for the subject vehicle as it can provide a powerful way of obtaining optimal solutions for complex problems and eliminate the need for obtaining any training data.

Assumptions

Only the vehicles have the ability to perform intelligent tactical manoeuvres. The driving scenarios of other vehicles are predetermined and it is also possible to obtain prior information with regard to each vehicle's position and speed, at any stage.

A flowing traffic without interruptions, roundabouts or junctions, is assumed. A five-second interval is the updating period for the tactical driving steps as it allows an adequate time to perform realistic speed changes and lane transitions.

The two-second minimum gap: A safe distance is the distance of two seconds at the current speed to the car ahead.

$D(2\ \text{second}) = V\ 2$,

where D(2 second) means the two-second distance in meters, and V is the vehicle speed in (m/s).

Driver Behaviour Model

The Driver Behaviour Model (DBM) input takes the parameters of the driver’s age, gender, experience and personality, and returns a 0 to 1 output, corresponding to a range of very risky to very cautious driving behaviours, as follows:

$DBM = w1\ DBMa + w2\ DBMe + w3\ DBMp/g$,

where DBMa, DBMe and DBMp/g are the different DBM contributions resulting from the driver’s age, experience and personality/gender factors, respectively.

Data representation

The genetic algorithm in the solution GA-INTACT is a tactical decision in the form of change in speed, acceleration rate, lane and the type of lane transition, which is then optimized with respect to the tactical driving criteria. The author uses a 14-bit binary string that encodes the speed, acceleration, lane parameters, as follows:

- VSubject is the target speed of the subject vehicle at the end of the five-second interval. The 7 bits allow for a 1-kph resolution of a 0 to 128 kph speed range.
- ACS (Acceleration Scale): this scaling factor corresponds to the change in acceleration/deceleration over the five-second interval. A 3-bit field allows for scenarios such as (000) for a “very sharp” acceleration rate and (111) for a “very smooth” deceleration rate.

- LN is the subject vehicle's lane number at the end of the five-second interval where LN = 0 is for the slow lane (the rightmost lane in the European system, the leftmost in the UK system) and LN = 1 is for the fast lane (vice versa).

- LTS (Lane Transition Scale): this reflects the way in which the lane change is performed, in terms of smoothness/sharpness. The LTS ranges from "very smooth" fast-to-slow lane transition (LTS = 7/7) to "very sharp" slow-to-fast transition (LTS = 0/7).

Fitness function

There are four main optimization criteria, namely safety, speed, driving behaviour, and "keeping in the slow lane". The overall GA fitness function can be described as the weighted sum of the four evaluation measures, as follows:

GA fitness = w1 f(safety)+w2 f(speed)+w3 f(behavior)+w4 f(slow)

Conditions

Three conditions were imposed to reject unrealistic or unlawful driving solutions at every five-second interval manoeuvre, based on the minimal gap, maximal speed and acceleration limits:

- $G \geq G$ min: The gap to the vehicle ahead which is below the minimum safety gap

- $V \geq V$ Opt: The subject vehicle drives faster than its optimum speed VOpt

- $a \geq a$ Max: The vehicle exceeds its maximal acceleration/deceleration rate amax

Fitness

Safety fitness (f Safety)

The objective of the safety function is to drive the subject vehicle in such way that the gap to the vehicle ahead is reduced, while maintaining the minimum safety gap, as follows:

fSafety= SSafety/(SSafety +(G(t+-T)-Gmin(t+T))) where: T is the five-second interval. G(t+T) is the actual gap ahead of the subject vehicle at the end of the manoeuvre. Gmin(t+T) is the minimum gap ahead of the subject vehicle at time (t+T). The gap G is the distance from the subject vehicle to the vehicle ahead. Gmin is based on the two-second rule, with the addition of the DBM related term. SSafety is a scaling parameter based on the initial Gap G(t) and the minimum gap Gmin(t).

Speed fitness (f Speed)

For speeds near or equal to the optimum speed:

fSpeed= SSpeed/(SSpeed +(Vopt(t+-T)-Vsubject(t+T)))

VSubject(t+T) is the subject vehicle speed at the end of the five-second interval VOpt(t+T) is the optimum subject vehicle speed at the end of the five-second interval SSpeed is the fSpeed function scaling parameter based on VSubject(t) which is the initial subject vehicle speed and VOpt(t) is the initial optimal subject vehicle speed.

Driving behaviour fitness (f Behaviour)

It reflects the compatibility of the subject vehicle manoeuvre with the driving behaviour (DBM).

f Behaviour = f(LTS,ACS,SCS,DBM)

DBM is the subject driver behaviour model output value, LTS is the lane transition scale, ACS is the acceleration rate scale and SCS is the speed change scale.

A heuristic model was formulated with the data points that define the correspondence between the DBM, LTS, SCS and ACS input terms and the output value of f Behaviour.

“Keeping in slow lane” fitness (f Slow lane)

This fitness term takes a noticeable effect, when there are “almost” equal gaps ahead of the subject vehicle to the immediate vehicles in both slow and fast lanes, with these vehicles driving at similar speeds. The f Slow lane function causes that when similar traffic conditions are present in both lanes, the subject vehicle would end up in the slow lane.

f Slow lane = (LNmax - LN)SSlow lane/(SSlow lane+|VSlow -VFast|+|GSlow -GFast|) where VSlow and VFast are the speeds of the vehicles ahead, in the slow and fast lanes, respectively. GSlow and GFast are the gaps of the vehicles ahead in the slow and fast lanes, respectively.If there are no vehicles ahead, then GSlow/Fast = 200 m, and VSlow/Fast = 128 kph. LN is the lane number of the subject vehicle at the end of the five-second interval. LNmax is the maximum number of lanes per traffic direction number. The SSlow lane scaling parameter adjusts the sharpness of fSlow lane based on DBM.

Simulation

In the following text there are some examples of tactical driving results produced by GA-INTACT.

The GA-INTACT traffic scenario has the following conditions: Design speed=120 kph, Subject vehicle starting lane LN=0, Starting speed =80 kph, speed of the first vehicle ahead on the slow lane=90 kph and speed of the first vehicle ahead on the fast lane=100 kph.

In order to observe the effect of the driving behaviour on the tactical manoeuvres, the same traffic conditions were simulated with three DBM values: DBM=0.9 “overcautious” driver, DBM=0.5 “standard” and DBM=0.1 “risky” driver. UK road layout standards were used (left slow lane).

Overcautious driver

1. The subject vehicle accelerates from 80 kph to 102 kph to match its optimum speed (VOpt), thus reducing the gap ahead from 60m to 58m, in a “smooth” manner (ACS=0.4), in the same lane.

2. The gap ahead of the subject vehicle is reduced to the optimum gap from 58m to 50m in the same lane by decelerating smoothly (ACS = 0.7) to a speed of 92 kph, a speed level near that of the vehicle ahead.

3. The optimum gap of 48m is matched, by readjusting the subject vehicle’s speed to that of the vehicle ahead in the left lane. The speed is reduced smoothly (ACS=0.7) from 92

kph to 90 kph.

Standard driver

1. The subject vehicle accelerates from 80 kph to match its optimum speed of 104 kph, thus reducing the gap ahead from 60m to 57m, close to the optimum gap. The subject vehicle increases speed in a "smooth" manner (ACS=0.4).

2. The gap ahead of the subject vehicle is decreased to 52m and is therefore kept near the optimum gap of 50m. This is achieved by decelerating smoothly (ACS = 0.6) to a speed of 91 kph, and making no lane change.

3. The subject vehicle carries out a smooth lane transition (LTS=0.4) from the slow to the fast lane (LN=1). The speed is increased from 91 kph to the optimum of 104 kph on the fast lane, in a smooth manner (ACS=0.4). The gap to the "new" fast-lane vehicle ahead, which had been travelling at 100 kph, is now very large.

Risky driver

1. The subject vehicle accelerates from 80 kph to 107 kph, a speed close to its optimum 111 kph. This speed allows a 56m gap to the vehicle ahead which is a near-optimum gap length, as the minimum gap is 55m. The vehicle also stays in the slow lane and carries out the speed increase in a "very sharp" manner (ACS=0.1).

2. The subject vehicle carries out a very sharp lane change (LTS=0.1) from the slow to the fast lane (LN=1). This allows it to increase its speed from 107 kph to the optimum 111 kph. A sharp acceleration type was carried out (ACS=0.3).

3. No speed or lane change, reducing the gap from 90 to 80m.

Conclusion

The "overcautious" drivers have the tendency of travelling in the slow lane and optimizing the safety distance. The "standard" drivers produce a driving action that follows the general need to optimize the speed and the safety distance, changing lanes when necessary. The "risky" drivers increase their speed and travel in the fast lane.

The current 14-bit GA solution size means that the problem under consideration has a relatively small search space.

Results used for our simulation

In the concept of cellular automata, a virtual world is divided into a uniform grid of "cells" in which cellular automaton move at each "step", i.e., at each discrete unit of time. The essential features of a given cellular automaton are: its state, which is an array of parameters, its neighbourhood, defined by the presence or absence of other cellular automaton in the nearest cells and its behaviour, that is the set of rules that define its evolution at each time step. These rules are generally derived from its current state and neighbourhood. We used a similar approach to divide the highway into visibility cells.

Agent-based simulations have provided a new approach for the simulation of complex physical systems which include many independent variables acting and reacting together. All agents have an ability to sense the environment they are "living in" and make actions based on what they perceive. In terms of degree of autonomy, such systems can be classified in different categories from reactive and non-autonomous agents to intelligent and highly autonomous agents. Reactive agents are only subject to local interaction and communication with their neighbouring agents and environment.

Our multiagent approach offers a way of bringing the simulation closer to reality. It is a theoretical approach which approximates the reality even more than other systems mentioned above because it takes account of the interaction and behaviour of the agents.

We also used the genetic algorithm approach to divide the drivers according to their behavioural characteristics.

Highway simulation

The simulation is a simplified reflection, a model. We are not able to exactly evaluate all real-world mechanisms, so we have to manage with a system on some level of abstraction and ignore a lot of variables, which are describing the real world.

Our simulated world contains several things, which are important for the highway simulation. Most important are the vehicles and the highway.

Architecture

Aglobe is an agent platform designed for testing experimental scenarios featuring agent position and communication inaccessibility, but it can also be used without these extended functions. The platform provides functions for residing agents, such as communication infrastructure, store, directory services, migration function, deploy service, etc.

The highway simulation uses the Aglobe framework. We made use of all of its advantages. The architecture is based on the principles of the server-client application. The simulation has distributive properties. The agents can communicate with each other using their container communication level, which can run on different machines.

The simulation needs for all of that a central part, a main agent of a simulation. This agent is responsible for simulating an environment of surrounded subjects in a simulation. It is located on the master container.

The communication can be held by using topics or messages. Topics are used for the communication between the containers and mainly between the server and the client; messages are used for negotiation between the agents.

Usually, there is one container for each agent, because of the amount of topics incoming to the container. Topics are received by all agents located on one container. Our settings break this rule. Several agents can be located on one agent container. The upper bound of the number of agents on a single container was experimentally determined as 50 agents. Messages were chosen as the main transfer medium for a server-client communication in the direction from server to client instead of topics, because of the quick increase in the number of topics, which is caused by making copies of the topic for each agent on that container. The topics are a transfer medium only in the direction from the client to the server and only for the kind of information, which is intended for the whole content of the container.

There are several types of agents in the simulation.

- Simulation agent – keeps all relevant information about the simulation
- Visio agent – is responsible for the video representation of the state of the simulation world
- Vehicle agent – keeps information about position, state and actual situation of the vehicles in its visibility range, controls the vehicle with regard to this information
- Highway agent – it can be more than a checkpoint for statistic in our experiment. We can use dynamic traffic signs as parameters of the simulation.

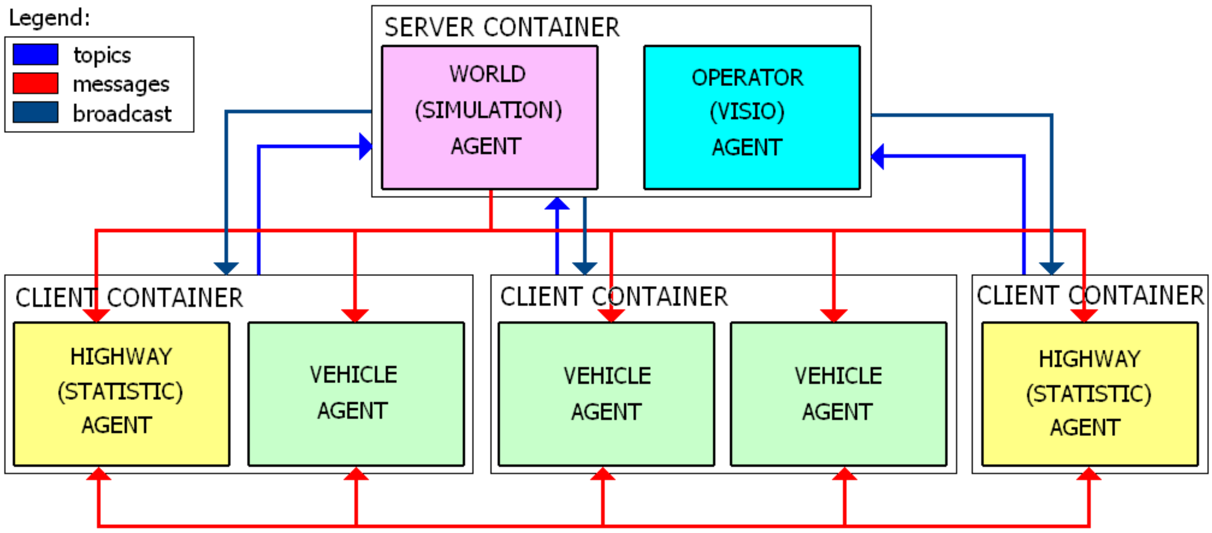


Figure 4: Architecture of the multiagent highway

The agents in the slave container gain the information about their environmental vicinity. The information about objects within the visibility range of the vehicles is periodically sent to the simulation agent.

Highway representation

A road is an identifiable route, way of path between places. This road is not the shortest path between start and goal destination, a straight line between these two places, but it is adapted to a ragged environment.

Many approach, how to describe roads, was investigated in early version of highway simulation. In follow text we compare classic engineering and used representation.

Engineering representation

The highways are a combination of these elements:

- Line: a straight segment
- Circular Arch: a curve with constant curvature
- Clothoid (transition curve, cornu spiral): A clothoid is a curve with infinite length and linear curvature change. Its curvature is the inverse value of its radius. A clothoid curvature is at any point proportional to the distance along the curve,

> measured from the origin. This property makes it useful as a transition curve in highway and railway engineering, because a vehicle following the curve at constant speed will have a constant rate of angular acceleration.

In transport engineering the problem with 2D description of 3D curves is solved with the help of following parameters:

- **Axis** – projection to a horizontal plane
- **Level** – projection to a vertical plane
- **Crosscut** – perpendicular to the axis of a road

For description and calculation a tangential (directional) polygon is used. Polygon is a broken line described by the length of its sides and the central vertex angles.

Simulation representation

This engineering description is not suitable in the highway simulation and instead of that we used a description by the beziér curve. The main reason for this choice is that the beziér curve is easier to compute and it is more suitable for graphic representation.

We can simply represent a 3D world via spline. We will use the beziér curve for the representation of the road; in our simulation this 3D curve describes the line of the central crash barrier.

For a full description of the highway additional data is needed, above all the information about the number of lanes on the highway. We have used a description with a uniform lane width.

Another important path of description is mapping. This utility keeps information about mapping one highway to another highway. We use this tool to make an intersection in our simulation.

Highway rules

One can also make a difference between symmetric (rules used in the USA) and asymmetric systems (rules mostly used in the Europe). In the symmetric system a vehicle in the central lane can overtake by the right or by the left; all lanes are equivalent in terms of speed average. Whereas in the asymmetric system, a vehicle has to keep right, using the center and the left lanes only when it needs to overtake (the contrary in UK-based systems). In our simulation, the Continental Europe system is used.

Properties

The highway description ensures that we can describe a random highway by nets of beziér curves, which are connected by mapping. This description is not exact (the imperfection is caused for example in the turning movement – when two vehicles drive in parallel lanes with the same velocities, the vehicle on the inner arch travels a shorter distance then the vehicle on the outer arch of the curve), but for our purpose suffices.

Server agents

The world and the operator are both implemented for running on a central server container. They are initialized from the store after starting the server container by the platform.

World (simulation) agent

The world agent is the main agent of a simulation. These are the base activities of the whole system, which are performed by the world (simulation) agent:

- Simulation control – we are able to start, pause and change speed of simulation using a gui for the world agent
- Measurement control – if an automatic measurement is not selected via xml, we can use the gui for measurement control
- Environmental configuration – information about the environment – such as the plans of the highway net and the location of the blocks or signs in the world are uploaded during initialization.

Every simulation period (0.1 second) the world agent has to compute the following:

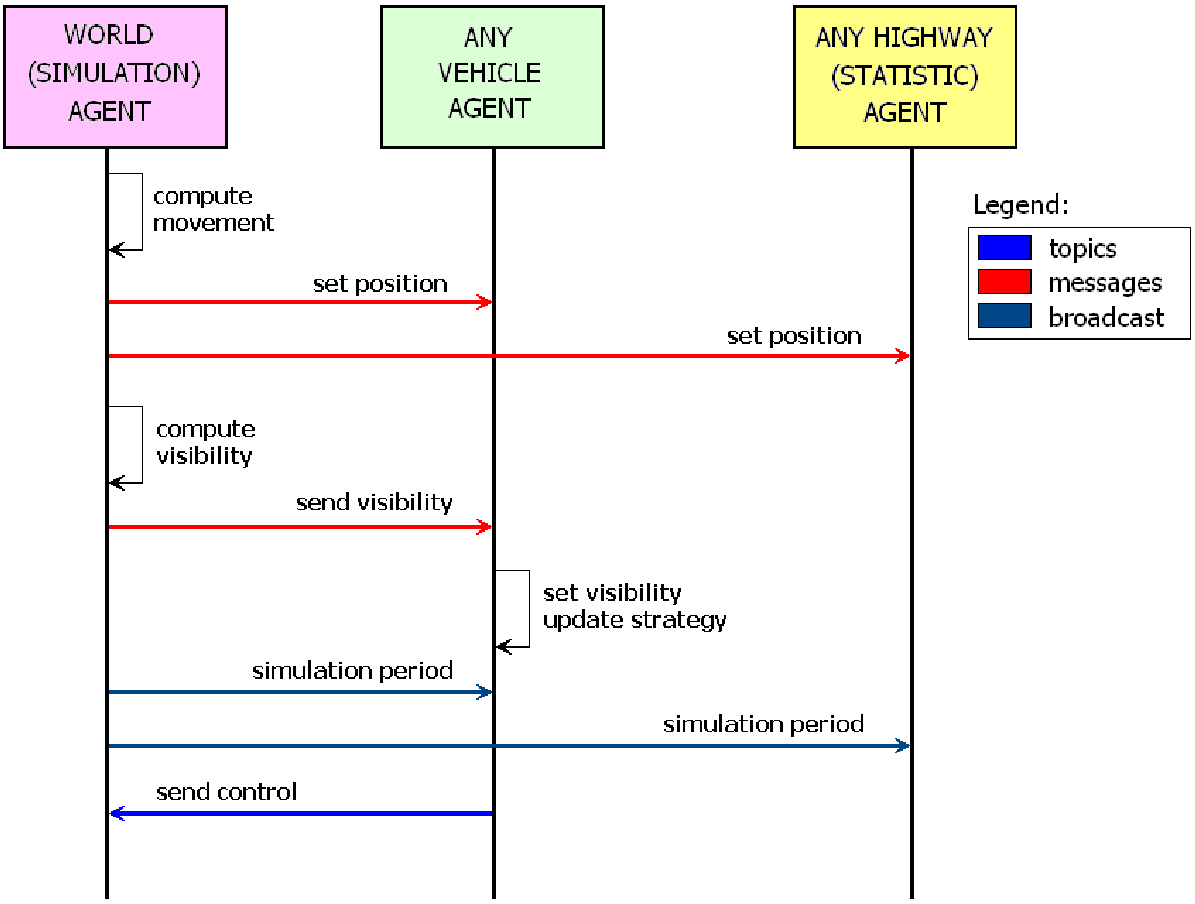
- Movement change computation – the agent is responsible for the change of the position for each vehicle per every simulation period.
- Send movement change message – the world agent informs the vehicle agents about the change of their position.

- Send a visibility message – the agent has to send information about visibility to every vehicle.

- Visibility data distribution

- End of simulation period broadcast

Figure 5: Diagram of the agent communication during a simulation period



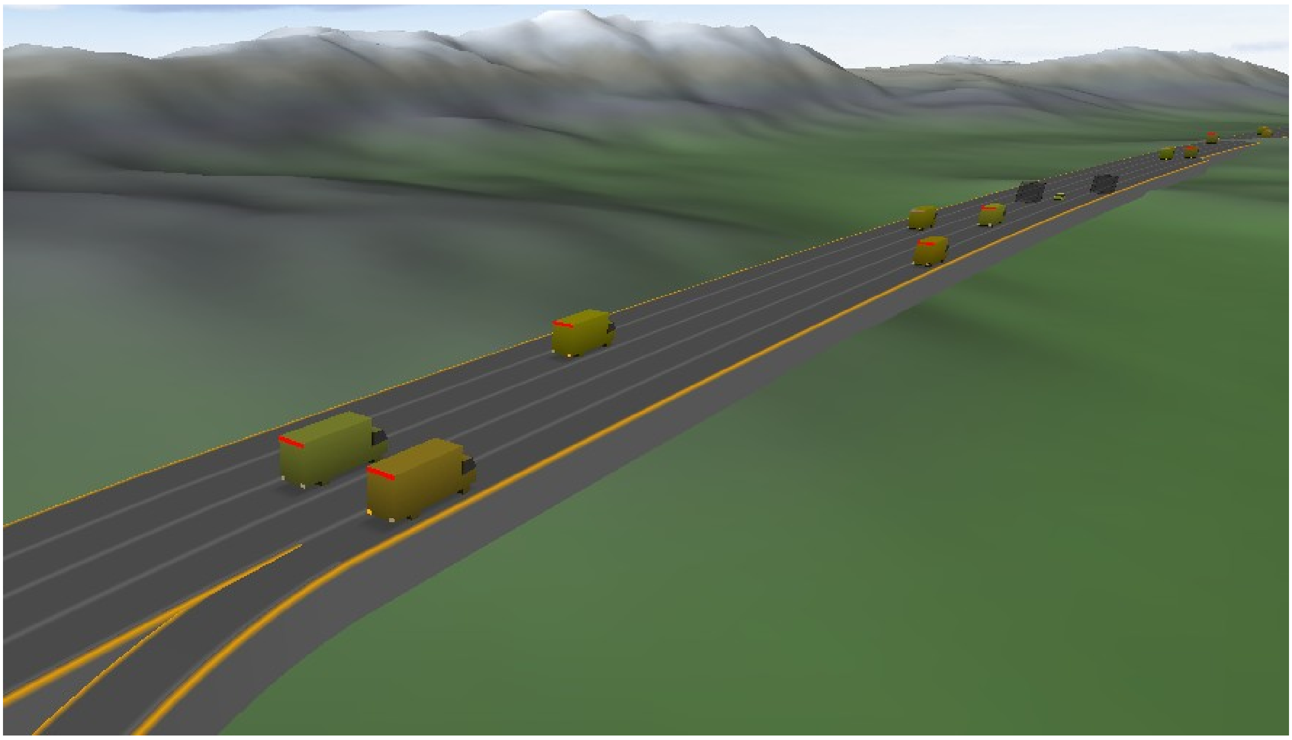
The main purpose of the operator (VisioAgent) is to collect and visualize information collected from the simulation. It receives a static model of the highway from the world agent shortly after initialization.

Figure 6: Snapshot of the visio

The agent receives the change of position and properties of all dynamic entities in the simulation (vehicles) per each simulation period. This information is sent by the world agent. The user has a possibility to change the point of view. The update frequency is equal to the simulation period.

Vehicle agent

The vehicles are represented by dynamic objects in our highway world. Each agent can be described as a unique control unit of a vehicle (driver). The main purpose of this agent is to plan a safe and comfortable drive control depending on its priorities and goals. The action control is sent to server per every period. The agent also receives information about its new position and objects in its visibility and communication range per every period. The agent can keep information about its vicinity and share it with other agents. This way it is possible to reach new sources of sensor information.



Driver

This part of the simulation models the human behaviour. We can simulate the human traffic behaviour by several significant parameters.

- **reactionTime** This is the parameter of driver's reaction time. It is a multiple of the simulation interval.
- **preferredSpeed** This is the driver's optimum speed of moving. The driver has the intention to equal the preferred speed to the average speed.
- **visibilityFront** This is the driver's front visibility.
- **visibilityRear** This is the driver's rear visibility.
- **preferredChangeLaneScale** This is the parameter of speed for changing lanes. This parameter assumes the values in interval 0 (no change of lane) ..1 (changing lane as quickly as possible).
- **blinkingTime** This is the time for blinking before the driver changes lane and direction. It is a multiple of the simulation interval.
- **slowDownScale** This describes the ability to drive behind a slow vehicle. When a driver closes a gap, he has to slow down and accommodate his speed. If the future accommodate speed is lower than the slow down percent scale of the preferred speed, the driver has the intention of passing the slower car, else he would have to slow down to current speed of the front vehicle. The parameter assumes values in the interval 0..100 (percent).
- **forbearanceTime** This is the parameter for the threshold of the forbearance of a driver, when he drives a car slowed down by the car driving in front of him. In this situation the current speed is lower then the preferred speed. This threshold depends on direct proportion of the difference between decrease of speed and time in the slow down mode. When this value oversteps the limit, the driver has the intention to pass the slower car.
- **passingMaxSpeed** This is the parameter for the threshold in the situation when it is necessary to speed up to pass a slower vehicle, but it is also necessary to have a higher speed than the preferred speed. For some of the drivers there are subjective highest limits of safety speed.
- **preferredFrontGap** This is the driver's personal minimum of front safety gap.
- **preferredRearGap** This is the driver's personal minimum of rear safety gap.
- **decelerationScale** This is the parameter of using the brake pedal, when it is not necessary to use it (it is not emergency braking). This parameter assumes values in interval 0 (no braking) ..1 (full braking of the vehicle).
- **accelerationScale** This is the driver's approach to using the acceleration pedal.

This parameter assumes values in interval 0 (no acceleration) ..1 (use full acceleration of vehicle).

- **entranceRamp** The position on the entrance ramp in which the driver tries to enter the highway. This parameter assumes values in interval 0 (first opportunity) ..100 (last moment).
- **exitRamp** The position on the exit ramp in which the driver tries to exit the highway. This parameter assumes values in interval 0 (first opportunity) ..100 (last moment).

This system of parameters gives us tools to create random driver profiles. The following examples are the most common driver profiles.

Overcautious driver

An overcautious driver is described by the following behaviour. He has a relatively low preferred speed, low preferred change lane scale, long blinking time, high slow down scale, very long forbearance time, relatively low passing maximum speed, mainly big preferred front and rear gap, low deceleration scale, takes advantage of the first opportunity to enter exit ramps, but they exit the entrance ramp at the last moment.

If the agent receives a message with request for some manoeuvres, that agent agrees very often with that request, even if the manoeuvre is not profitable for this agent.

We can describe the drive of this driver as cautious or fearful. He passes the slower cars rarely, he rather slows down and adapts his speed to the front vehicle. When he changes his direction to another road, he tries to keep the safest and earliest way to reach his goal.

Standard driver

A standard driver is described by the following behaviour. He has a high preferred speed, appropriate preferred change lane scale, medium slow down scale, forbearance time and maximum passing speed, mainly long preferred front and rear gap, medium braking and acceleration scale.

If the agent receives a message with request for some manoeuvres, that agent agrees often with that request. The request is accepted mainly if the manoeuvre has some profitability for the agent or there is no reason for rejecting the request.

We can describe the drive of this driver as comfortable. He likes a smooth ride without quick speed changes.

Risky driver

A risky driver is described by the following behaviour. He has a high preferred speed, prefers a quick change lane scale, short blinking time (sometimes no blinking), low slow down scale, short forbearance time, relatively high maximum passing speed, mainly medium preferred front and rear gap, high deceleration and acceleration scale, he enters exit ramps at a last moment, but takes advantage of the first opportunity to exit entrance ramps. The driver slows down in a bad weather, but holds his speed at the edge of his skills.

If the agent receives a message with request for some manoeuvres, that agent agrees rarely with that request. The request is accepted only if the manoeuvre is profitable for the agent.

We can describe the drive of this driver like braking and acceleration, he makes last time decisions and trusts his driving ability. His behaviour is rude to other drivers.

	overcasual	standard	risky
accelerationScale	0.2	0.6	0.95
decelerationScale	0.2	0.6	0.95
blinkingTime	15	7	1
entranceRamp	0.1	0.5	0.8
exitRamp	0.8	0.5	0.1
forbearenceTime	1000	400	100
slowDownScale	70	85	95
passingMaxSpeed	1.1	1.15	1.2
preferredFrontGap	10	3	1
preferredRearGap	10	3	1
preferredChangeLaneScale	0.4	0.7	0.9
preferredSpeed	41.6	50	62.5

Figure 7: Table of driver parameters

Goal

The agents in the simulation are following some goals. These goals are related mostly to the driving style.

- **Smooth motion:** Drive with minimal active control of acceleration and deceleration

- **Keep preferred speed:** The agent tries to have minimal divergence from the preferred speed

- **Minimize time of drive:** The vehicle with this goal makes an effort to drive with maximum speed and to achieve the end of its path in the shortest time possible.

Highway (statistic) agent

The highway agent is primarily used as a data collector for the simulation evaluation in a statistical way. For every segment there is one highway agent assigned. More will be explained about the data collected by the highway agent in the chapter 5.1.

Data objects

In the following text we will introduce some of the structures which are used by the simulation system mainly as content of the messages and topics and have a crucial function in the field of memory saving and system processing. This class extends the abstract class SubjectInfo, which is set as its pattern. We will now introduce the main features.

Position properties

The simulation needs two representations of position. First, it is necessary to determine the position on the map and in the visio. The second property describes the position on the highway and it changes with movement; it is used for computing the new position of the subjects. The position change is recomputed every simulation period and the structure contains both position parameters.

Position on a map

These parameters are related to the map point (0,0,0). This data is very important for the simulation, because it determines the location of the subject in the simulation world.

The second important point is the position of the subject, which is related to its front center. This point is the reference point of the subject. The position and heading of the subject is determined by the following parameters:

- Position – represented by the class Point3d, including coordinates x, y, z
- Yaw – the yaw angle is the angle between the vehicle's heading and a reference heading on a horizontal plane.

- Pitch – the pitch is the angle between a vehicle's heading and a reference heading on a vertical plane.

Position on a road

These parameters determine the location on a road. This information is used for the orientation in the highway net and also on the highway. It is easier to process the information that a vehicle is in the distance of 10 meters from the center crash barrier then to work with several coordinates at once.

- disLane – a lane distance is the current distance from the center crash barrier
- distance – a distance from the beginning of the highway
- idHighway – identification number of the highway. It is necessary for path finding.

Inner and exact position on a road

We are using these parameters for computation of the movement on the bézier curves. These parameters are saved only on the server. They are used only for computing a movement.

- Identification number of beziér – one highway can consist of some amount of beziér curves, this is its identification.
- beziér – keeps all information for computation of the movement
- beziérT – parameter of the beziér curve

Subject information

The subjects are objects in a simulation which can be viewed by sensors. They have the following parameters:

- Dimension – this is the dimension of the subject, it includes its width, length and height

- Type – type of the subject

Information about vehicle

The vehicle information has three kinds of representations. Each has a different role in the system.

Vehicle state

The vehicle state is used as the content of the messages and topics, the server is sending the state as a result of a movement of the vehicle and the vehicle is sending the topic to the server about changing the control parameters; the serve then collects information including the vehicle state and sends it to the other vehicles as visibility information.

The vehicle state contains only necessary parameters that are of a public character. The state can keep the control parameters, acceleration and difference in lane, and also keeps information about the blinker and brake light.

Vehicle info

Vehicle info is located only in the memory client of the vehicle agents. This class contains information about the goal of the vehicle, driver behaviour and motion performance, that is the private information of a vehicle.

Vehicle behaviour

The vehicle behaviour is located only in the memory of the world agent server. This class contains the information about movement, which is used by the world agent to compute the new position in the simulation.

Movement

Computation of the movement is carried out by the method move, which is a part of the vehicle behaviour class owned by the world agent on the server container. Every simulation period the vehicle agent sends an action control from the client container to the server via topics. This control information contains two control parameters, acceleration and difference in lanes which is related to the highway coordinates system as a planned change of the vehicle coordinates within the highway. After the movement computation the world agent sends back a message with updated information.

Vehicle dynamic

This project is not an accurate description of the physicals parameters of the movement on a highway by kinematics and dynamics. The simulation uses a base of dynamic physics and many of the physical features are ignored as being irrelevant for the simulation.

This is because of the need for a quick computation of the movement results.

The changes of position in the point of view of the kinematic physic are described by the following equation:

$$\Delta s = \vec{v}\,t + \frac{1}{2}\vec{a}\,t^2$$

But in reality the acceleration and deceleration are not constant. There are several results of the pull of forces, which have a significant effect to the motion. This feature is approximated by the curve of acceleration and deceleration.

In the picture there are curves representing maximal acceleration and deceleration values [m/s^2] depending on speed [km/h]. The value of the action control acceleration is often lower; the values in the graph are limit values of a vehicle.

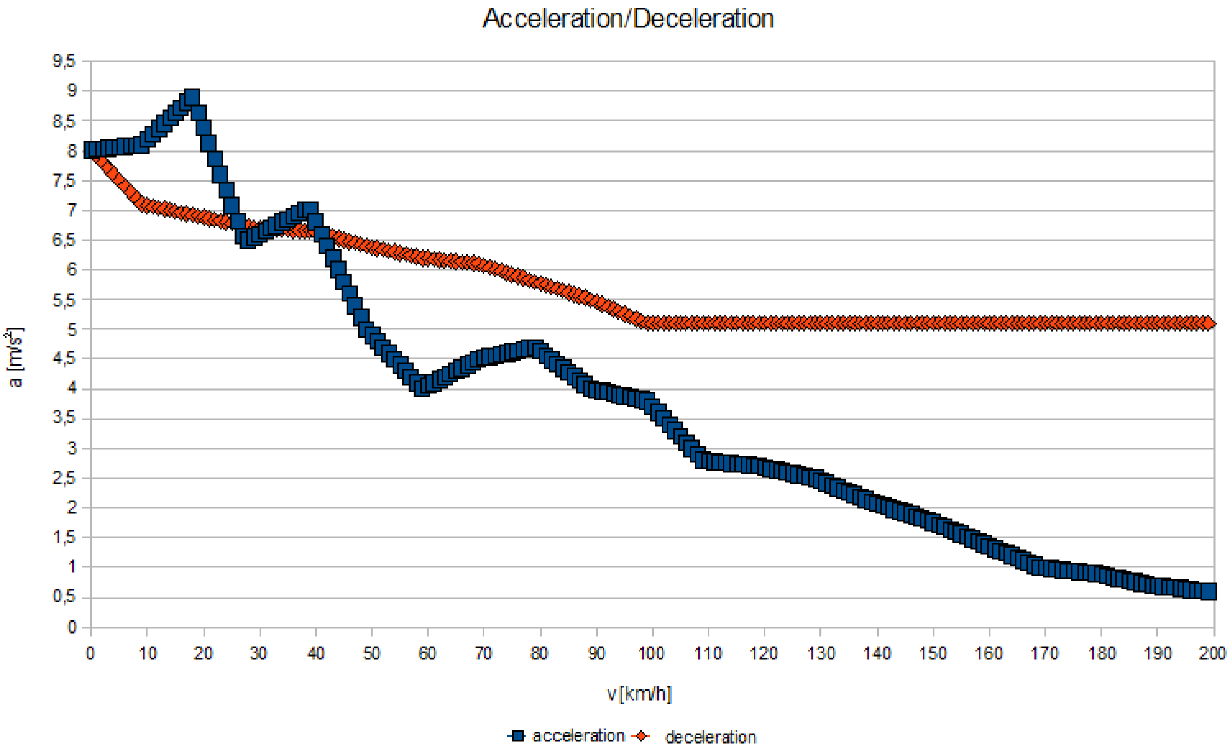


Figure 8: Graph of the acceleration and deceleration of the vehicles

The difference in lanes is a change of position in the coordinates of a highway. This variable changes with the distance from the central crash barrier. The position with regard to this coordinate is described by the following equation:

$$dis_{lane} = \frac{\cos(\frac{\pi \cdot t}{T_{man}}) - 1}{2} \cdot width_{lane}$$

t means time from the beginning of a manoeuvre, T_{man} means the total duration of a manoeuvre and width is the width of lane. The difference in lanes is equal to the difference between current and next distance value from the central crash barrier. The difference can be described by the following equation:

$$dif_{lane} = \left| \frac{\cos(\frac{\pi \cdot t}{T_{man}}) - \cos(\frac{\pi \cdot t + T}{T_{man}})}{2} \right| \cdot width_{lane}$$

where T is the simulation period.

In the simulation T_{man} is computed as a multiple of the constant time 2 seconds (this is equal to 20 simulation periods) and the variable called scale for changing lanes, which can be different for each driver. So this variable does not depend on current speed of the vehicle. The vehicle changes its distance from the central crash barrier just as quickly in 100 km/h as in 0 km/h.

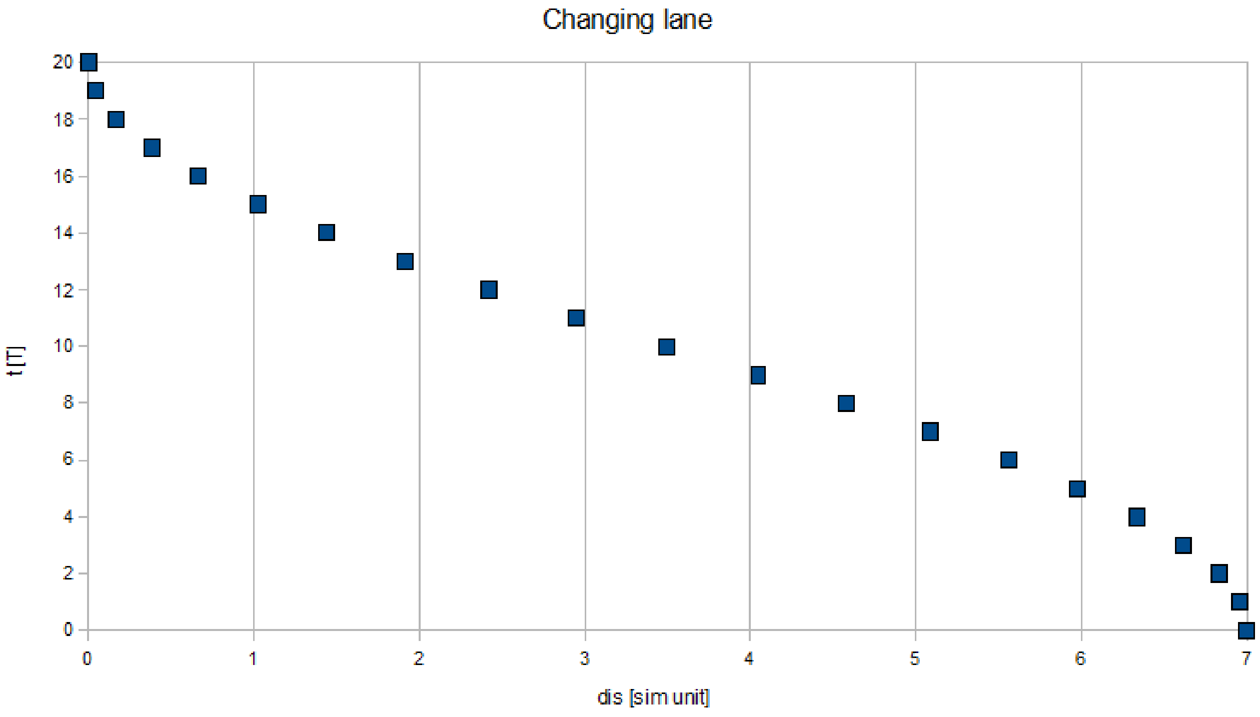


Figure 9: Graph of the lane change

In the previous figure there is the progress of distance from the central crash barrier during lane change depending on time.

This description of movement for the simulation is not exact but in contrast to the physical description, there is a possibility to quickly compute results and determine a new control action.

Position change

Moving of a vehicle causes a change in its position represented in the hash table, which importance for the visibility will be explained in the following text, in the memory of the world agent. The key changes correspond to the movement of the subject from an old destination to a new one. There are three possibilities of change:

- Little position changes often do not change the hash key.
- If a change in position causes a change of the key, then also the position representation in the hash table has to be changed. So the representation of vehicle information has to be removed from the old destination and moved to another.
- If the movement exceeds the end of the highway, then the key has to be removed from the hash table corresponding with the old highway segment and added to the hash table for the new highway segment.

Visibility

The main advantage of a multi-agent system is in the feature of the communication inaccessibility. A vehicle as an agent in the simulation has not all information about every other vehicle (or subject) in the simulation. That leads to limited visibility for the driver and limited radio range for communication. The driver can see only those subjects within the bounds of sensor visibility. The visibility structures will be introduces in this chapter.

Visibility representation

Let's assume that the position related to the highway coordinates is important for a vehicle's visibility (note that the opposite direction of a highway is considered a different highway segment). It means that a vehicle is not interested in any subject, which is not located on the same or an adjacent highway. "Adjacent highway" means a previous or next segment of the highway net. Ergo a vehicle ignores any subjects on the other parts of the highway net (meaning also the opposite direction lanes). This behavioural feature does not apply to the real world, because accidents are frequently caused by distractions from the other parts of the highway, which should be ignored (for example when the driver is watching an accident on the opposite direction of the highway).

The simulation needs an efficient approach to storing the visibility data. That is because every agent needs the data about its environmental vicinity for every simulation period. Every simulation period the reciprocal position of all subjects in the simulation is changed. The inaccessibility is an advantage under these conditions, because it decreases the extent of searching the domain of results.

All visibility information is located in the memory of the world agent. The preconditions for the structure which stores the visibility information are as follows:

- Separability – subject in the visibility range must be in a separated structure based on its reciprocal distance. Subjects that are situated on the same highway, have to be located near each other in the visibility structure too. Subjects which are located in other parts of the highway net have their visibility representations located far from each other.

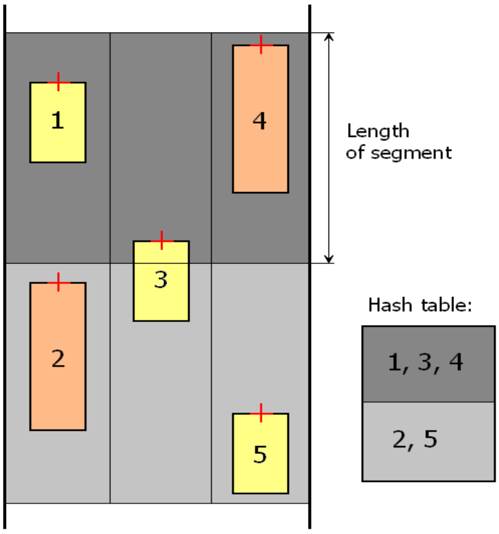
- Quick accessibility to the target data – this demand is connected with the amount of access to the data. The simulation supposes that the visibility of a vehicle is in several segments. The assumed amount of access to the structure is quite higher than the number of the input.

- Relatively quick change of the location of a representation – the location of a representation in the memory is related to the subject position. If the subject is a vehicle, the location of its visibility representation must be changed with its position.

For this domain a hash table is used. All of the highways are divided into segments with equal size. Each highway is represented as a hash table and each segment is represented by one cell of the hash table in this simulation. Each cell of the hash table contains a linked list with information for the subjects which are actually located in this part of the highway.

The hash table has generally a default load factor of .75 which offers a good trade off between time and space costs. Higher values decrease the space overhead but increase the time cost to look up an entry (which is reflected in most hash table operations, including get and put) [11].

There are several types of hash functions, which use a key as a pointer to the hash table.



We chose a key to the hash table as a hash function for an integer number. The following text show how to reach a key. The cells of a hash table have a length as the minimal size of the visibility range of a vehicle. The length is evaluated as 40 simulation dimension units for this simulation.

Figure 10: Representation of the highway splitting into hash table

When the system wants to get the visibility information for a vehicle from the hash tables, the hash table corresponding with the highway has to be chosen and taken, and the key of this hash table is deduced from the vehicle position on the highway.

The key is acquired as floor of the quotient of the distance from the start of the highway and the length of the highway segment to get an integer number and then the result is modulated by the number of the cells in the current hash table.

```
key=(int)Math.floor(distance/segment_lenght)%size_hashTable;
```

Visibility of a vehicle

A vehicle in the simulation has two ranges. The first range corresponds with the visibility range of a driver and the second corresponds with the range of a communication unit. These two ranges are for simplification united to one. This complex range information is then sent to a vehicle. So the system has to collect information only within a border of a range and the information processing and the filtration is the duty of the vehicle.

First of all a system of the world agent has to collect the visibility data for a vehicle. This means to collect data from all the hash tables which are within the borders of a complex range of the vehicle. If the visibility range is stretched to more than one highway, then the system has to collect visibility information from all of them. So the hash tables have to contain a pointer to the adjacent highway.

The range of visual visibility is a variable related to vehicle speed. The values of visual visibility are computed as follows:

frontVisibility=speed*(preferred time distance)+50

rearVisibility=50+max(50-speed,0)

Accidents

The computation of accident consequences on the highway is not trivial. Let us assume that the vehicle stops completely after an accident.

We use such a simplified point of view for our simulation.

Agent communication

The main purpose of the domain of communication is to reach new, commonly unknown data. Communication in the highway simulation can work on the same basis. A drive can be called cooperative if it is made more efficient by communication.

In this chapter we will describe the manner, purpose and goals of the communication which can be achieved.

Each vehicle with the ability to drive cooperatively has communication equipment with limited range. This feature is realized by a filter which is not able to send a message when the message receiver is out of range.

The protocol base on standard FIPA is implemented as the interaction protocol specification for vehicle-to-vehicle communication.

Communication query

The query protocol [12] is used in the simulation for discovering another communicating agent in the simulation.

If a vehicle is not cooperative, then it has no communication and cooperation agent. The only exception is a refuse reply to a query question.

A non-cooperative agent is able to send only one type of message – the information that it is not able to communicate (and so it is non-cooperative). This feature is a simplified form of a communication protocol; the agents in the vicinity do not have to wait for a message timeout to verify an agent’s non-cooperative status.

If an agent receives information from an agent out of its visibility range but in the range of communication, the agent is able to send a query message for more information about the unknown agent. This information extends the data from the sensors.

This approach was first required because of a problem with synchronization. The query message for communication is sent immediately after computing a received message/topic with the visibility information from the master container. This visibility information is subsequently sent from the master agent to the vehicle agents, so the time of receiving is not the same. The receiver system cannot process messages and save the sender like a communication agent, because the sender was not in its memory. These algorithms solved this problem.

Subscribe information

The subscribe protocol [13] is used to process the sensor data from another agent. The agent-initiator has to know that the other agent, the participant, is communicative.

The agent-initiator chooses its informant on the basis of these rules:

- The vehicle is within the bounds of its visibility range. This means that the sensor information will be valuable, because the number of sensor information different from the original data will rise.

- The informant has approximately the same speed. This rule ensures that the partnership lives long and prospers.

If any of the negotiating vehicles predicts that the second agent is going to leave the communication visibility range, it finishes the communication.

The communication is sensible also if the communication range of a unit is lower than the visibility range. The communication can always bring new data because the sensor data of the vehicle is constrained by its visibility bounds. An object can block the vehicle's view and the information about the blind spot can be contained in the received data.

Requesting manoeuvres

The request protocol [14] is used to manage the negotiation about manoeuvres. The agent-initiator has to know that the agent-participant is communicative. Request for manoeuvres is relevant in the case when both negotiating agents are in vicinity.

The initiator can request one of following manoeuvre:

- slow down/speed up

- turn left/turn right

The content of the request message has to be related to the vehicle, which is asked to perform the required manoeuvre.

Vehicle logic

In this chapter we will describe the vehicle logic as a set of rules for a safe and comfortable drive. This set of rules is based on the evaluation of the environment vicinity information. The vehicle is searching for a strategy, which will efficiently deal with the situations on the highway.

Non-cooperative logic

The non-cooperative logic control is used if the vehicle is unable to use negotiation with another vehicle. The vehicle logic reflects only the information which can be mined from the visibility information of the world agent. This information is then stored and periodically updated in the memory of the vehicle.

Sensor information

An up-to-date part of the information is searched to acquire important sensor result information. The space around the vehicle is parsed into two main domains. The first domain keeps information about the space behind the vehicle and the second domain keeps information about the space in front of the vehicle. The line which forms the border for the rear and front visibility passes through the reference point of the vehicle. So the sensor information about a subject situated on the side of the vehicle is located in the rear visibility, and when the reference point of the subject is situated in front of a vehicle or next to it, it is located in front visibility.

The whole width of the highway is divided into several sectors. This division corresponds with the sensor sensibility. An ordinary lane is divided into 21 sectors in our simulation. If we set the lane width in the simulation as 7 simulation dimension units, then the width of a sector will be 1/3 simulation dimension units. This width is a sufficient representation of vicinity information. For each of this sector we gain important data for the vehicle control system.

- Road distance – this is a distance between two subjects which is determined as the difference between the distance parameters of the subjects. The subjects keep information about their distance from the start of the highway. This distance is reduced by the length of the front vehicle.

- Direct distance – this is a distance between two vehicles which is determined as the difference between their referent points. This distance is reduced by the length of the front vehicle. This distance can be quite different from the road distance. It is important because of the communication range.

- Time distance – how much time is left to a collision if we take into account the velocity of the subject; in case the detected subject is a vehicle, we assume that it starts braking immediately.

Every period the sensor data answers the questions whether the vehicle is compelled to brake, able to accelerate or if the vehicle has a free manoeuvring space for turning.

This information is filtered from the visibility information.

Safety gap

The most important thing for the vehicle control is to guarantee preventing of collisions. This demands the need to keep a safe distance between vehicles during the ride. This safe distance depends on the speed of the vehicles and their ability to stop in time.
Every second we have to expect that the vehicle before our vehicle starts to brake. The time needed for a complete halt is described with this equation:

$$t=\frac{s-\left(v_{rear}TIME+\frac{1}{2}a_{rear}TIME^{2}\right)+\frac{\left(v_{front}+a_{front}TIME\right)^{2}}{2a_{max}}}{v_{rear}}$$

The time corresponds to the worst situation that could happen on the highway (for example that the vehicle in front starts to brake) and the result is compared with the time needed for the vehicle to stop completely. This compared value is:

$$t = \frac{v}{x\,a} + t_{const} + t_{reac}$$

We are using linear enhancing of x: it is 0.85 for the safe time distance and 0.65 for the preferred time distance. We are also adding the driver's safety gap constant and the reaction time of the driver and the vehicle.

Spheres of influence

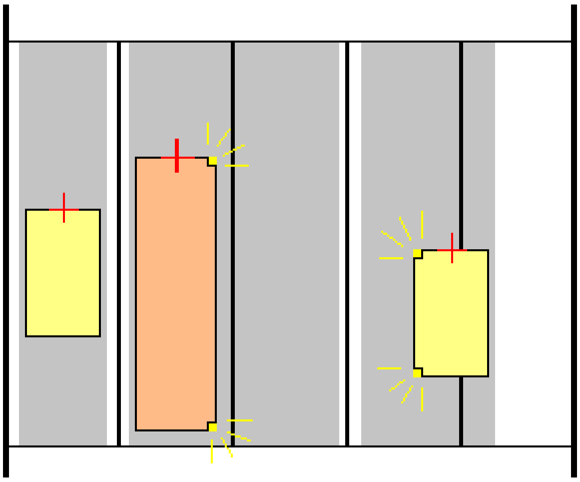
The subjects in the simulation occupy a given space. This fact is reflected in the sensor info and the vehicle memory representation. As we mentioned, the main data designated for the vehicle control is related to the position on the highway. The dimension of a subject and the space which is reserved for the future position of the subject are very important variables, crucial for both the vehicle itself and other vehicles in the vicinity.

Because of this demand the sensors have to know the values of the left and right sides of the subject and if possible also the values in their sphere of influence for the near future. These values are represented by the coordinates of the distance from the central crash barrier. The sphere of influence (grey areas in the figure 10) corresponds with the dimension of a vehicle in the vicinity and its impact to the vicinity.

The standard sphere of influence is defined for a subject or for a vehicle, which does not change its distance from the barrier. This standard means that the left side of the influence sphere is located on the left side of the subject and the right side of the influence sphere is located on the right side of the subject. The whole sphere occupies in this case the same space as the vehicle does.

If a vehicle is going to change its position, the sphere of influence is widened to the chosen side. If a vehicle is able to reserve free space in the next lane, it demonstrates this with blinking. The other cars identify this blinking sign as reserving a space and the sphere of influence of the vehicle is widened to the chosen side for the width of the lane and gets narrower on the other side. The other cars now know that this vehicle is going to change the line in some of the next simulation periods.

Figure 11: Representation of the spheres of influence



If the subject is the nearest object in at least one sector, where its sphere of influence extends, then it becomes visible for the sensors of the vehicle.

Rules for passing

Lane changes happen in the moments when a vehicle passes a slower car or a block on the highway.

If the vehicle wants to change a lane then its sensor data must allow this movement. The sensor collects information to pass a test for this action. The sensor not only has to watch the safety gap, but also has to control the space behind the vehicle in the lane of its direction. It also has to keep a rear time distance. The vehicle does not start the passing manoeuvre without having enough time to change lane safely.

Finding phase is the time used for finding the most suitable manoeuvre and next position. This phase lasts until a new position is found and there is no obstruction for the manoeuvre.

Arrangements phase starts with the blinking on the side of the lane change. This is a sign for vehicles in the vicinity that the blinking vehicle is reserving free space in another lane and its sphere of influence is widened in their memory representation. The length of this phase depends on driver characteristics.

Changing lane phase – the vehicle changes its lane and moves to the goal lane

States relations

The vehicle can be represented by some set of states. These states describe the vehicle behaviour in the nearest simulation step and reduce the number of the following states. The basis of this approach is to describe the vehicle control mechanism as a state machine. The degree of change of the control parameters that can be achieved depends on the type of driver.

The states of the vehicle are separated into two descriptions, the forward mode and the side mode. This representational approach corresponds to two possible control actions which the vehicle can perform.

Forward modes

The forward modes have a crucial impact to acceleration change.

- Normal – vehicle keeps the same speed
- Broken – if the vehicle has a negative distance to another subject, then it is marked as broken and unable to move
- Accelerate – vehicle speeds up
- Starting – the vehicle is stopped, not broken and needs to speed up
- Stopping – continuous braking
- Full breaking – emergency braking

Side modes

The side modes influence the changes in another dimension than forward.

- Normal – vehicle keeps the same distance from a central barrier
- Prepare right move – vehicle prepares for moving to the right. It starts blinking with the right blinker and reserving free space on its right side.
- Prepare left move – vehicle prepares for moving to the left. It starts blinking with

the left blinker and reserving free space on its left side.

- Left move – turning to the left lane
- Right move – turning to the right lane
- Left move within lane – it is a small change of direction to the left
- Right move within lane – it is a small change of direction to the right
- Search safe – if a vehicle has not enough space in its current position then it is searching for free space on its sides

Cooperative logic

Cooperative logic is an extension of non-cooperative logic. Cooperative agents differ from the non-cooperative in the way that they can process requests.

Creating request

An agent can start a negotiation only with another agent in visual vicinity. An initiation could be danger of an accident or a pressure to achieves the vehicles goal.

An agent can only request one manoeuvre at a time. If it does not succeed, it can make another request immediately or repeat the first one.

Investigation of a request

Each vehicle can only attend to one requested manoeuvre at a time. If it is dealing with a requested manoeuvre it is not able to handle any other.

A vehicle accepts the request if it does not contradict its current state. It considers its real feasibility in the other simulation cycle. In the following table there are permissible states for the individual driver profiles.

Accepting a request is also a part of the driver's behaviour as was referred in the chapter 3.1.3.1 about driver behaviour.

For example, as seen above, the overcautious driver is able to interrupt manoeuvre preparations in order to receive a request.

Mining information

The vehicle can compute statistical data about motion of other vehicles. A vehicle can also collect hidden information about these vehicles. This way it can try to compute statistical characteristics such as the preferred and maximum speed, intensity of using speed acceleration and deceleration for the lane change. This information can be used for prediction of the vehicle behaviour.

Properties of control

The sphere of influence is a space which the passing vehicle reserves for its future position. This demand is then transmitted to the vehicles in the vicinity by the blinker. The sensor can easily compute the results of a vehicle control's specific query.

Measurement and comparison

The traffic flow theories are intended to provide understanding of phenomena related to the movement of individual vehicles along a highway as they interact with neighbouring vehicles. The theory can be divided into macroscopic and microscopic treatment. The macroscopic treatment views traffic as a continuum akin to a fluid moving along a duct which is highway. The microscopic treatment considers the movement of individual vehicles as they interact with each other.

In the following chapter we will explain why the multiagent system is suitable for simulating a highway.

Definition of metrics

The simulation is able to register a wide scale of data. The measured data can be related to a single vehicle or to a highway segment. The data for a single vehicle are relevant for evaluation of the driving strategy and the evaluation can influence the strategy improvement in an iterational way. Whereas the data related to a highway segment can be useful for the highway regulation, for instance position of the intelligent traffic signs, which are able to regulate the flow of automobiles, e. g. with modification of the maximum speed, or evaluate the highway flow data. Based on that information they can re-route a number of vehicles and improve the permeability of the highway.

Macroscopic metrics

As described above, the highway net is divided into segments and on their borderlines there are checkpoints. In every simulation period it is determined whether the vehicle belongs to the given segment. An important parameter for the macroscopic description is the area of the highway, which is determined by the number of lanes and the length of the segment. A restrictive parameter is the number of blocks on the highway segment, which influences the vehicle flow. All these parameters are invariable in time (for our purpose).

These are the main parameters for the macroscopic description:

- Density – the number of vehicles per km per lane
- Average flow – the number of vehicles per hour in the checkpoint

The following variables are measured every simulation period and are calculated per minute:

- Average vehicle speed on the given segment
- Average vehicle speed related to its type (car, truck)
- Histogram of the individual vehicle velocities on the segment
- Number of stopped (not crashed) vehicles
- Number of crashed vehicles
- Average vehicle speed on the subsegments (each segment is divided into five same-sized subsegments)

Microscopic metrics

These are the parameters related to a single vehicle.

The following variables are measured during the whole simulation:

- if the vehicle crashed
- Number of simulation periods elapsed without crashing
- Distance passed
- Number of simulation periods when the vehicle accelerated, decelerated, stopped and blinked
- Time elapsed when overtaking

Experiment conditions

For determining the properties of the implemented system we selected a number of experiments. The experiments will take place under almost identical starting conditions, so that the results can be compared. There is certainly the initialization influence effect, but it can be eliminated with higher number of measurements with random initialization. We did not conduct this number of experiments due to lack of time. On the other hand the quantity of the collected data is able to compensate for this disadvantage.

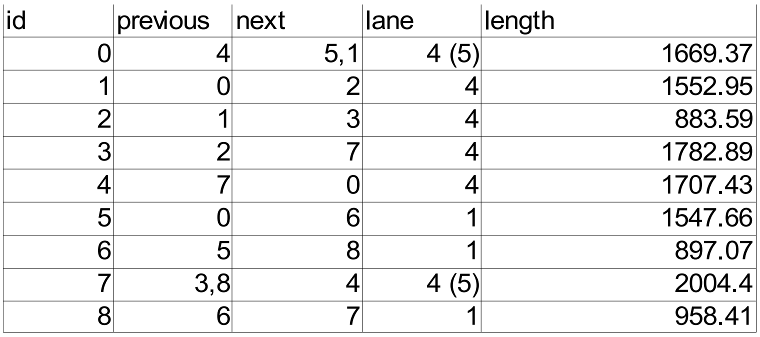
At the beginning the simulation agent is stopped. It is started only after initialization of all of the entities in the simulation. Every measurement begins automatically after one minute from the start of the simulation, in order to eliminate the initialization influence effect (no accidents occur during this period), and it takes 20 minutes.

Highway conditions

All measurements will take place on a simple highway net. It is a circle with one exit and entrance ramp, which consists of 9 highway segments. The visualization of the circle can be seen on the next picture. The red dots represents the checkpoints.

Figure 12: Division of the highway into segments

In the following table there are properties of the individual segments. The length is given in the simulation units. If the highway has an exit or entrance ramp, it is not regarded as a part of it. The number of lanes is increased by half of a lane to compensate it.



id	previous	next	lane	length
0	4	5,1	4 (5)	1669.37
1	0	2	4	1552.95
2	1	3	4	883.59
3	2	7	4	1782.89
4	7	0	4	1707.43
5	0	6	1	1547.66
6	5	8	1	897.07
7	3,8	4	4 (5)	2004.4
8	6	7	1	958.41

Figure 13: Table of highway parameters

Vehicle conditions

In the individual experiments we will observe the unique properties of the system and the other properties will be fixed in a constant ratio. Unless otherwise stated, 50 agents will be used, initiated in a constant ratio of the trucks and cars (2:3), the overcautious, normal and risky drivers (1:3:1) and the non-cooperative and cooperative agents (2:3).

- Traffic flow experiment: We will observe the influence of the increasing number of agents in the simulation on the traffic flow

- Communication and cooperation experiment: We will change the ratio of the cooperative and non-cooperative agents

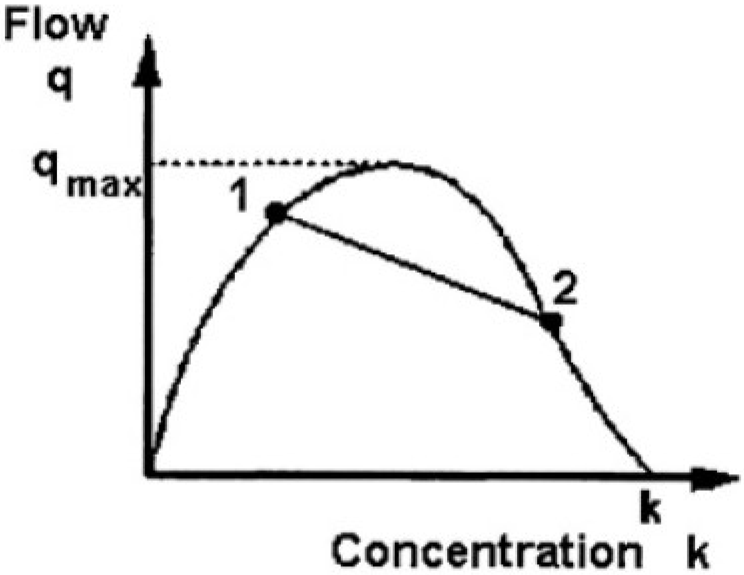
Hypothesis

Before the measurements we have to determine a few assumptions which should be valid for a highway simulation system.

Traffic flow

Further we assume that q and k are related in a fashion described by what came to be known as the “Fundamental Diagram”, shown in the figure 14, on the right. The rationale for the existence of a Fundamental Diagram is simple. Clearly, at zero density (k=0), there is no traffic and consequently q=0. The traffic flow is also zero at some “jam density”, when cars are stopped bumper-to-bumper.

Figure 14: The Fundamental Diagram



Assuming a continuous relationship between q and k, by Rolle’s Theorem there must be at least one maximum point between 0 and k_j. The assumption of one maximum point produces the Fundamental Diagram, which describes the following behaviour of traffic: As the density increases from zero, the flow also increases, starting with an almost linear rate of increase, and a lower rate as the density increases and the average speed of the traffic decreases. The flow reaches a maximum at some optimum speed and density, and then starts decreasing with increasing density, reaching zero at jam density, when the speed is also zero.

Positive impact of information propagation and cooperation

Information propagation by the vehicle-to-vehicle interaction has a possibility to make the highway traffic more fluent. The information like the sensor data propagation and the vehicle-to-vehicle negotiation is also able to prevent accidents.

The positive impact can be decreased by the visibility range related to the vehicle speed. The visibility is computed with regard to the preferred time distance and current speed; the visibility range is usually higher than the communication range and so the influence of the communication is significantly weakened.

Measurement

The measurement was influenced by a higher number of accidents, which eventually led to a complete standstill on the highway due to the blockage in the place of the accident. With regard to the accidents we were forced to change the original plans for the measurement methods. Originally there should be 100 agents moving in the simulation for the duration of 30 minutes. Due to several mass accidents after 20 minutes, which blocked the highway and led to a subsequent total inhibition of the highway flow, we changed the number of agents in the basic simulation to 50 and the measuring time to 20 minutes.

The classic SI units cannot be used in the simulation, because the simulation is scaled.

Flow

The flow rate can be computed as follows:

$$q = k \cdot v_{avg} = \frac{N \cdot v_{avg}}{T \cdot l \cdot n} ,$$

v_{avg} is the average speed in simulation units per second k is density. The density is computed as follows:

$$k = \frac{N \cdot 1000}{T \cdot l \cdot n} \text{ ,}$$

where N is the sum of vehicles traced on a highway segment during one simulation period, T is the sampling period, in this case equal to 600 simulation periods (1 minute), l is the length of the highway segment in the simulation units and n is the number of lanes of the highway segment.

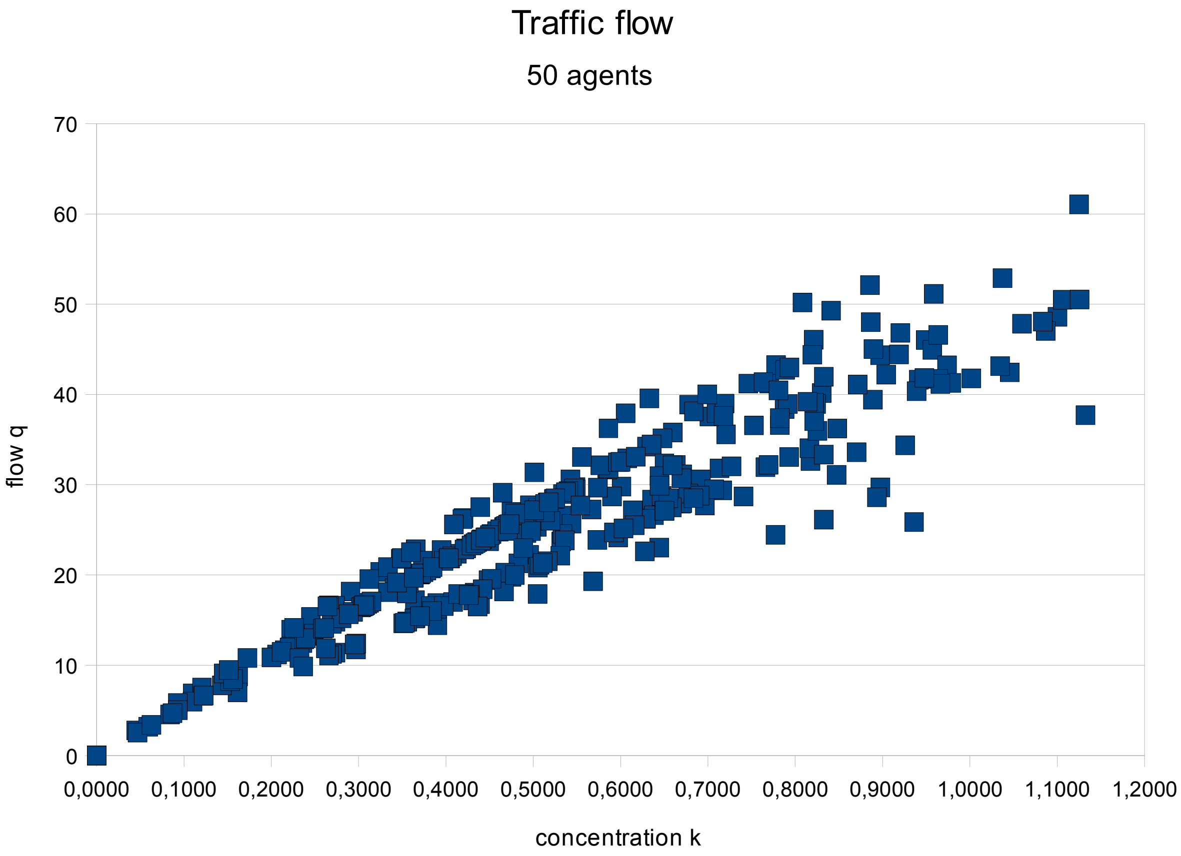


Figure 15: Graph of the traffic flow for 50 agents

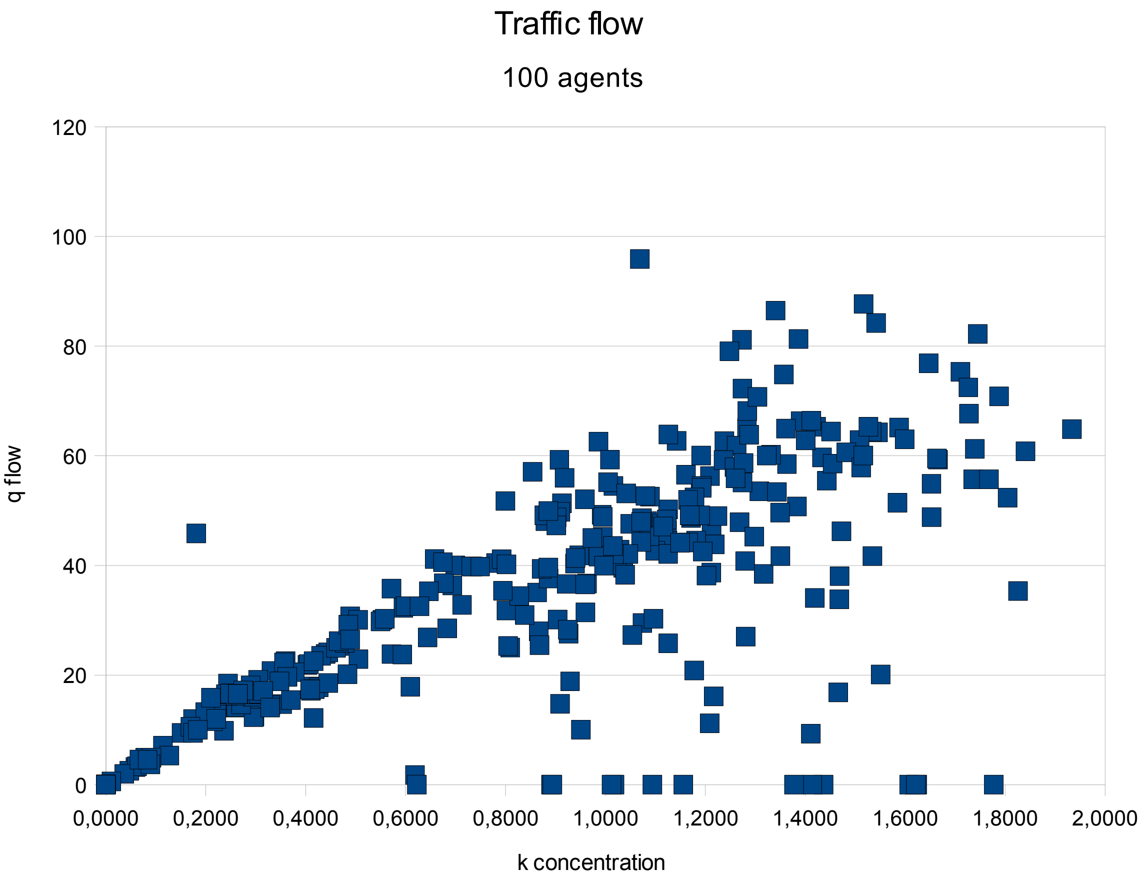


Figure 16: Graph of the traffic flow for 100 agents

As we can see on the measured values, the graph resembles the rising edge of the Fundamental Diagram. The concentration represents the density of the traffic (in vehicles per length of 1000 simulation units) and the flow represents the number of vehicles in one lane in one segment per minute (i.e. the sampling period). The values are increasing in a linear fashion – with increasing concentration the flow also increases. The values in the second figure are significantly more scattered around the linear line because with 100 agents the number of vehicle interaction increased substantially, which caused accidents and standstill of the vehicles. In extreme cases the flow can stop completely.

The measurement method was not ideal. A better method would be to measure only one highway segment during different pointed vehicle load – the data would be more uniform.

Communication

The following figure summarizes the data from two measurements – the white cells represent the non-cooperative agents and the grey cells the cooperative agents. From the measured data is evident that the communication and especially the request fulfilling has a negative effect on the traffic flow – under conditions of these measurements. Under identical circumstances the vehicles fulfilling requests are forced to manoeuvre more and this influences their average speed, which is then lower.

Figure 17: Table of differences between cooperative and non-cooperative agents

highway	acceleration	deceleration	speed
0	444,38	253,3	50,17
0	478,36	307,32	47,8
1	351,34	184,12	50,88
1	332,02	208,18	49,48
2	188,44	122,64	50,51
2	205,72	112,12	49,07
3	421,22	202,38	51,03
3	415,7	212,38	49,99
4	470,22	284,14	49,7
4	483,42	362,94	45,46
7	654,62	363,82	48,66
7	610,8	393,02	46,09

In the first column there is the ID number of the highway segment (5, 6 and 8 are missing, because these are the single-lane segments and therefore not as significant for our measurements as the others), in the second and the third column there are the average numbers of accelerations and decelerations during the measurement and in the fourth there is the average speed of the vehicles.

Results

The problems with the measurement show clearly that in the current state the algorithm is not a solution to a highway traffic movement. On the other hand the algorithm is doing well in standard situations, although it is not always acting naturally, as a human driver would do.

The algorithm strives for a best solution possible – under given circumstances – which is not always the best solution for the drive. It changes lanes very often only because of a slightly disturbed time distance. This could be solved by making the drive control more robust – to widen the sphere of influence and to increase the front and rear safety gaps.

However, we have reached an important finding. The cooperativeness of the agents is not an advantage (under the conditions of our measurement), because it causes a higher number of lane changes and therefore a higher number of accidents, and it also decreases the vehicles average speed and has a negative effect on the flow in general.

It is highly likely that this finding does not apply universally. We conclude that the experiment conditions were unsuitable and degraded the function of the communication. A solution to this problem would be to increase the communication range beyond the borders of the visibility range making the communication more effective.

Discussion

The algorithm which controls the vehicles in the simulation is not perfect, a number of accidents happen, especially at a greater traffic density. The accidents are usually very close, meaning that the overlap of the vehicles is minimal, and with regard to the coordination transformation and inaccuracies in the operations of the data type double it may not even be the case of actual accidents.

However, this is not an excuse. The vehicle should have a wider safety range around it, which would prevent this kind of accidents.

It is highly likely that it is caused by the algorithm’s strive to approach the preferred speed, which is too high with regard to the traffic density. Other possible causes are: forced fulfilling of a request and turning on the entrance and exit ramps, which are frequent causes of accident even in the real life. The highway traffic is a highly dynamic environment and the highway movement algorithm is a complex problem, which is extremely difficult to solve. A solution, which seems to be the best for a specific traffic density, does not have to be universally applicable. The safe time distance approach appeared as an adequate solution, but it fails at a greater traffic density.

Working with the simulation was also made more difficult by an inconveniently used visualization tool. Its disadvantages show for instance in the representation of the length units of the world, which was derived for the use of the visualization, in particular by pervasive scaling. If the problem of the visualization was solved, the results would be more definite and have a better interpretability.

We dare to say that our algorithm has not chosen a completely wrong way; yet regrettably we failed to take account of all possible cases of the algorithm usage.

Nevertheless, the multiagent highway system as it was developed during this project and this thesis is an ideal tool for testing the algorithm functioning.

Conclusion

The evaluation of the simulation using a wide scale of collected data offers a valuable basis for subsequent reseach. The data for individual vehicles are relevant for evaluation of the driving strategy and it can influence the strategy improvement in an iterational way. Genetic algorithms or other kinds of softcomputing approaches, which could use the simulation results as their fitness functions, can be used to adapt the drive. Theoretically, it would be possible to achieve an algorithm which could rival the abilities of the professional race drivers.

The simulations can be also used as a criterion for automobile sensor suitability. Other course of study which can be used for subsequent projects is the behaviour prediction of the vehicles in the vicinity. The negotiating algorithms can offer a valuable contribution for the improvement of safety and driving comfort. They broaden the existing possibilities with the element of negotiating.

It is possible to enrich the vehicle behaviour in the simulation with help of the implementation of a more exact physical model. In the current state, the higway is represented by a beziér curve. Its importance lies mainly in the area of mapping the automobile position on the world coordinates system. From the point of view of the drive the curve is degraded to a line segment, because no influence by the centripetal force occurs.

The physical model could also simulate the reality better under different weather conditions and road conditions. I could also determine accident causes and consequences. After examining the automobile dynamics it could be possible to compute an ideal drive of the vehicle adjusted to the fuel consumption, braking strategy and optimal velocity; it would lead to individualization of the vehicle drive.

Acknowledgements

This project was supported by the ČVUT Media Laboratory Foundation for Students and Corporations.

The author wishes to thank to Ing. Jiří Vokřínek, supervisor of this project, for his valuable insights and professional guidance, and Jan Jiránek for the cooperation on this project.

ČVUT MEDIA LAB

Literature

1: , A-globe: A-globe Agent Platform, 2009, http://agents.felk.cvut.cz/aglobe
2: Šišlák, D., Rehák, M., Pěchouček, M. Rollo, M., and Pavliček D., A-globe: Agent Development Platform with Inaccessibility and Mobility Support, 2005
3: , Advanced Cruise-Assist Highway System Research Association, 2007, http://www.ahsra.or.jp/index_e.html
4: California PATH, Partnersfor Advanced Transit and Highways, Vehicle Platooning and Automated Highways, 1997,
5: Benz, T., Braun, A., Ulken, U., Schulze, M., Schwarz, J., and Sonntag, J., CHAUFFEUR - TR 1009 - Report on Safety Analysis of System Components andHazard Analysis of Tow-Bar, 1999, ftp://ftp.cordis.europa.eu/pub/telematics/docs/tap_transport/chauffeur_d7.1.1.pdf
6: Schadschneider, A., Traffic flow modeling, 2000, http://www.thp.unikoeln.de/~as/Mypage/traffic.html
7: Wahle, J., Chrobok, R., Pottmeier, A. and Schreckenberg M., A Microscopic Simulator for Freeway Traffic,
8: Claramunt, Ch., Jiang, B., A qualitative model for the simulation of traffic behaviours in a multi-lane enviroment, 2001
9: Šeda, J., Inteligentní automobily řízené multiagentním systémem, 2008
10: Tawfik H., and Liatsis P., Modelling Tactical Driver Manoeuvres GA-INTACT, 2006
11: Sun microsystems inc., Java API,
12: The Foundation for Intelligent Physical Agents, FIPA Query Interaction Protocol Specification, , http://www.fipa.org/specs/fipa00026/SC00027H.pdf
13: The Foundation for Intelligent Physical Agents, FIPA Subscribe Interaction Protocol Specification, , http://www.fipa.org/specs/fipa00035/SC00035H.pdf
14: The Foundation for Intelligent Physical Agents, FIPA Request Interaction Protocol Specification, , http://www.fipa.org/specs/fipa00026/SC00026H.pdf

Appendix

Documentation for the simulation, source codes and the measurement data are located on an appendant CD.