A Full Driving Simulator of Urban Traffic including Traffic Accidents

Jesus Félez
Joaquin Maroto
Gregorio Romero
Jose M. Cabanellas
Graphics Engineering and Simulation Group
ETS Ingenieros Industriales
Universidad Politécnica de Madrid
Jose Gutiérrez Abascal, 2
28006 Madrid, Spain
jfelez@etsii.upm.es

This paper describes a model for traffic simulation of an urban environment and its implementation in a driving simulator. The simulator is also able to reproduce realistic traffic accidents. In order to attain real-time simulation, the simulation environment has been partitioned considering the city as divided into segments of road, junctions, and sectors that minimize the interaction between the cars involved in the traffic simulation and the traffic simulation is considered only in a control zone centered on the driven vehicle. Simplified dynamic vehicle models have also been used when vehicles are not involved in the accident, allowing for a sufficiently realistic behavior. A traffic light regulation only in the area next to the driven vehicle is also included. A complex model for the vehicles involved in traffic accidents has been developed, including multibody components and different collision models. The developed model is then immediately applicable to large scale driving simulators.

Keywords: Simulation, driving simulator, traffic, traffic simulation, traffic model traffic accidents, crash

1. Introduction

Traditionally, traffic simulation models have been used to carry out traffic studies in order to design and modify the road layout, as well as to implement traffic light regulations on existing roads. However, during the last few years, there has been growing interest in driving simulators because they have proved themselves to be a highly useful tool for training and instructing drivers, particularly professionals (bus, truck, emergency vehicle drivers, etc.).

Lately, physicists have been paying special attention to the theoretical and empirical foundations of traffic flow physics [1, 2]. Vehicle movement taken individually has many peculiarities as it is basically a result of driver behavior together with some physical conditions.

In order to describe individual vehicle dynamics, a wide range of microscopic models has been put forward. Models based on cellular automata consider simple microscopic models with a logic that usually consists of a few operations having high performance [3]. On the other hand, car-following models provide more realistic modeling of driver and vehicular behavior.

Car-following models use realistic driver behavior and detailed vehicle characteristics that require higher computational resources. Models like INTRAS [4], CORSIM [5], CARSIM [6], and INTELSIM [7] are also models that use detailed acceleration models.

The development of these models has represented a very important advance in macroscopic model simulation, since it has been possible to perform the simulation from a microscopic dynamics basis. Nevertheless, when the results obtained are compared to reality, there are still important discrepancies, both in macroscopic behavior as well as in the description of individual car dynamics [5].

SIMULATION, Vol. 83, Issue 5, May 2007 415–432

DOI: 10.1177/0037549707083109
Figures 4, 5, 15–17, 19–22 appear in color online: http://sim.sagepub.com

Several traffic simulation tools have been developed recently, such as MITSIM [8], PARAMICS [9], AIMSUN [10, 11].

More sophisticated tools such as ARTEMiS [12] have also been developed, used for example to provide a general evaluation tool for advanced traffic management applications such as congestion and incident management, public transport priority and dynamic route guidance.

Due to the fact that the developed traffic model is for application in a driving simulator, the behavior and movements of vehicles surrounding the user-driven vehicle must be represented as realistically as possible. Therefore, a purely microscopic type traffic model will be implemented.

One of the characteristics of this simulator is that it is able to be used for simulating traffic accidents. Simulation models for the analysis of traffic accidents can be basically grouped in two types, based on the basic laws of physics used in the formulation of the model: dynamic models and analytic models.

The analytic models [13, 14] use the linear momentum conservation law, which is also applied to the point of impact, creating a group of simultaneous algebraic equations without the appearance of forces and which are immediately solved.

Having developed numerous analytic collision models, among the best known are CRASH (in its different versions EDCRASH, CRASH, CRASH3) [15] and OLDMISS [16].

The dynamic collision models use the numeric integration of Newton–Euler equations forming a system of differential equations. The dynamic models apply numeric time integration, like the computer simulation methods, advancing from the cause to the effect. The difficulty of these models is that in a traffic accident, the effect rather than the cause is known. The reconstruction of accidents by these models calls for applying iterative methods testing the initial starting conditions until arriving at the desired effect. Some examples of dynamic models can be found in [17] and [18].

Dynamic models have two important advantages compared to analytic models: They permit the consideration of the implied vehicle body's elasto-plastic behavior by means of models that are very close to reality, and they can be complemented with behavior simulation models of the vehicles. With this, the vehicle movement can be simulated before (pre-collision), during (collision), and after the collision (post-collision). This is an important facility for the analysis of the reasons that caused the accident or collision.

Another way to study traffic accidents is to apply the finite element method and structural analysis models to study the behavior of the vehicle's structure on collision. The best known programs of this type applied to crashworthiness are PAMCRASH and DYNA3D [19]. Some virtual reality applications have been developed based on this method [20], but they are simple postprocessors of simulation results.

The very complex parameters necessary to calculate these simulations and the high-performance computer needed, however, make these tools suitable for the structural design of the car body but unsuitable for a computer simulation program that works with a database of hundreds of vehicles.

For these reasons, and due to the very characteristics of virtual reality applications, a dynamic model to obtain vehicle movement is the best way to implement these theories in a virtual environment.

Research related to the development and application of a driving simulator is not new. Driving simulators began to appear in primitive forms in the 1970s. With the advent of computer technologies, Daimler-Benz launched a high-fidelity driving simulator in the 1980s [21], which created wide interest throughout the world. Since then, many automotive makers and research institutions worldwide have developed their own simulators. There are various levels of complexity of driving simulators ranging from a simple static base simulator to the most advanced simulator that are capable of simulating the dynamic motion and scenes of an actual vehicle.

There are several driving simulators developed by Companies or Universities such as TRUST [22] for heavy vehicles, Adams Transportation [23] for military and civil vehicles, NADS – National Advanced Driving Simulator – [24], Umtri [25], UCF Driving Simulator [26] or SIM2 [27].

A typical advanced driving simulator, such as NADS in Iowa, normally consists of a visual system, angular and translational motion systems with six degrees of freedom, vehicle cab system, control feel system, auditory system, and vehicle dynamics. An advanced simulator is capable of producing a high-fidelity simulation output. However, the construction cost of this system is very high. On the other hand, a static base simulator is a low cost and may be sufficient for applications that do not require very high fidelity simulation. The virtual reality-based driving simulator developed in this research will lay the groundwork for future research in transport safety, vehicle design and ergonomic evaluation, intelligent vehicle highway systems and driver education and testing, its main contribution being that the simulator is able to reproduce traffic accidents in real time.

The advanced computational capabilities in modern personal computers have made possible for consumers to experience simulations with a high degree of verisimilitude through simulation games. In recent years, the technology exchange between game and simulation technology, along with other factors, has contributed to the confusion as to what makes a simulation game and what makes a simulator. Several studies have been developed in order to clarify the differences [28].

In recent years, as a result of this technology diffusion, simulation games, serious games, and training simulators have become popular. As simulation games improve in realism, the use of the technological elements of simulation games in simulators has increased, and has contributed to the technology exchange between simulation games and simulators. The word "simulator" has been used in the titles of many simulation games (e.g., Microsoft Train Simulator or Microsoft Flight Simulator) and, in recent years, many simulation games have been used as simulators. These are some of the reasons that have contributed towards the confusion in distinguishing among games, simulation games, and simulators, but the main difference between games and simulators is the real physical behavior that is present in the simulators, rather than the verisimilitude present in games.

The main contribution of this paper is the development of a model for real-time traffic simulation for use in a driving simulator with the possibility to simulate traffic accidents. The developed simulator is based on a microscopic model, but it is not a classical microscopic traffic simulator because it only reproduces the traffic in the surroundings of the user-driven vehicle.

In the driving simulator, there are two types of vehicle: the user-driven vehicle and other vehicles, called automatic vehicles. The user-driven vehicle is fully simulated with tridimensional models and is used to reproduce the vehicle movement. On the other hand, the user-driven vehicle is also considered as an input of the general traffic model, in which the behavior and movements of vehicles surrounding the user-driven vehicle must be represented as realistically as possible. The complete traffic model that the user-driven vehicle must interact with reproduces the behavior of the other vehicles in the simulation. Each automatic vehicle has two models associated with it: the vehicle dynamic model and the driver model. The vehicle model reproduces its own movements responding to the driver's actions. The driver model establishes which actions are exerted on the vehicle, and when, and in what circumstances they are exerted.

Section 2 describes how the simulation environment is modeled and partitioned so that it can be simulated in real time. In order for the traffic to be represented in real time in a high-performance visual system, a minimum image refresh frequency of 30 Hz is needed. This frequency has a considerable bearing on all the solutions adopted for processing the simulation environment.

Section 3 describes the traffic model. To develop the traffic model a moving control area approach is used. The traffic is modeled and simulated only within this area, which moves along with the user-driven vehicle.

Section 4 presents the model and characteristics used for the user-driven vehicle. The model reproduces the full behavior of a vehicle depending on the driver inputs (steering wheel, gearbox lever and pedals). The obtained movements will be introduced in a six degree of freedom platform.

Section 5 describes the vehicle model used for automatic vehicles. This is the model that is used for all the vehicles in the environment, except for the user-driven vehicle. In order to endow the vehicles surrounding the user-driven vehicle with greater realism, a dynamic model based on differential equations has been used, which enables a more realistic behavior than with the traditional acceleration-based model. Using this type of model allows typical driver error situations such as skids or spins to be simulated.

Section 6 describes the driver model for the automatic vehicles. The driver model, basically describes the actions exerted by the drivers on the vehicles included in the traffic model, taking account of the traffic conditions surrounding them. Every vehicle has a path assigned to it that it must attempt to follow. With the aim of making the vehicle follow this path in accordance with the traffic conditions established, some PD (proportional-derivative) type controllers have been defined that act on the steering, the accelerator or the vehicle's brake. Together with these controllers, the decision-making criteria establish the action that the driver model must take in order to accelerate, brake or change lane, depending on the traffic and signposting conditions at any given moment. This section also introduces the traffic light regulation model, based on a simplified model with the only object to produce sensations of realism in the user-driven vehicle.

Section 7 introduces the collision model used in order to simulate traffic accidents and its implementation and Section 8 presents the characteristics of the virtual environment developed for the simulator.

Finally, Section 9 sets out the conclusions of this paper.

2. Geometric Description of the Traffic Environment

In order to carry out the traffic simulation, the traffic model needs to have information on the geometry and elements involved in the traffic flow environment. The starting point is the definition of the real geometry of a city on a two-dimensional CAD map, as shown in Figure 1.

When undertaking a geometrical description of a city, two different techniques have been traditionally adhered to.

In one technique, the lanes are divided into cells which can be simultaneously occupied only by one vehicle. Vehicle movement is controlled by taking account of the rules governing the occupation or liberation of these cells [29, 30].

The other technique is to consider the middle line of the lanes as a continuous path over which the drivers try to move. These paths are made of a series of entities (straight lines and arcs) that are divided into a series of points [31].

A schema of the paths definition can be seen in Figure 2, which shows the set of paths on an intersection with several directions, represented by discontinuous lines. Each segment represents a path. Each path has a driving direction assigned represented in Figure 2 by a

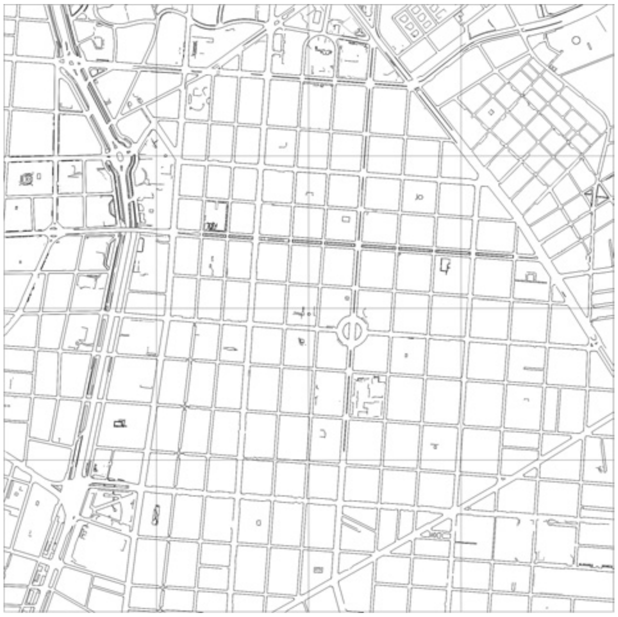


Figure 1. CAD map of a portion of the city

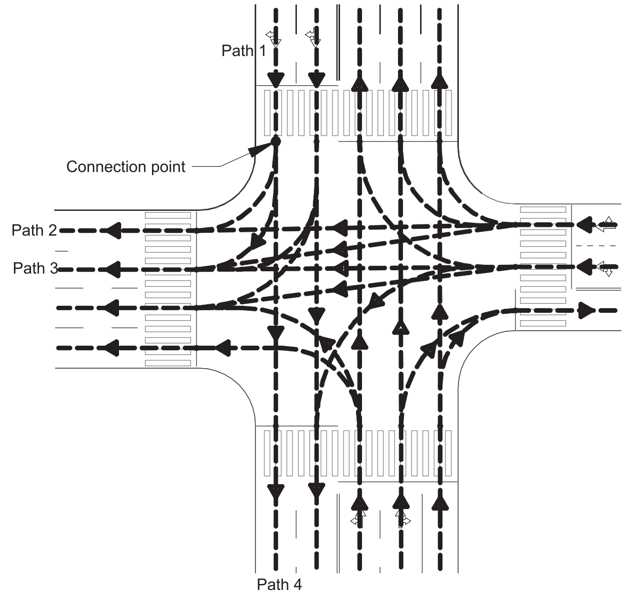


Figure 2. Path definition

triangle, and a circulation speed, determined by the road circumstances (maximum running speed, radius of curvature, etc.) At the end of each path there is a connection point that allows a path to be connected with adjacent paths. At each connection point the cars are distributed randomly toward each of the exit paths following the percentages established in the traffic environment design. In the example in Figure 2, when path 1 terminates at its connection point, it allows the traffic to be distributed toward paths 2, 3 and 4.

The geometric information of the city needed by the traffic model is the one relative to lanes, junction connectivities, signposting, etc. This information is grouped in different categories. Streets describe a section of street between two junctions, including a series of points with associated information, such as point co-ordinates, average vehicle speed, etc. Lanes comprise each section of street, including their direction of flow, their width and some special features, such as their being a bus lane, for example. Nodes include information concerning junctions and roundabouts in the city, including a list of all possible connectivities between access and exit lanes, with their corresponding probability. Signals and signs include a list of signals and signs existing in the city, such as traffic lights, stop signs, give way signs and pedestrian crossings. For each sign, data is specified to identify which lanes are affected by it and its relative position on each lane.

3. Traffic Model

Depending on whether the simulator is to be used for traffic studies or traffic regulation, or as a driving simulator, the simulation is either carried out over the whole of the city [32, 33], or only over a limited area [29, 34, 35]. In the first case, macroscopic models are used, while in the second case microscopic models are used. The present study has developed an implementation for a driving simulator, therefore the latter approach has been chosen.

The traffic is calculated and represented only in a zone surrounding the user-driven vehicle, which is called the control zone.

The zone used is rectangular in shape and is centered in front of the vehicle. Its configuration and size can be seen in Figure 3.

The size of the control zone and the location of the driven vehicle depend on the maximum number of vehicles involved simultaneously, the traffic density desired in the simulation, as well as the limit of visibility for the driver. The vehicle is not in the center of the control zone, it is situated rearwards instead. This is because the visibility limit must be greater in the forward direction so that the vehicles do not appear or disappear too close to the driver. In the rear direction, as the visibility is only due to the rear-view mirrors, the limit can be much closer to the driver.

The objects involved in the simulation are introduced at the points of intersection between this zone and the access lanes, and are destroyed at the points of intersection between the control zone and the exit lanes. In order to determine these points, the intersection of the control zone

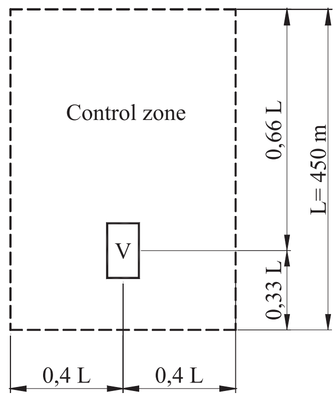


Figure 3. Control zone geometry

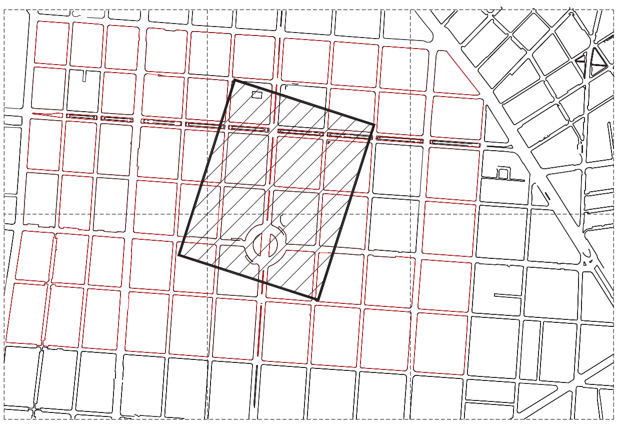


Figure 4. Partitioning the environment into sectors

with each and every one of the city's lanes, must be calculated. However, given that the total number of lanes in the city is very great, it is not possible to perform this process in every cycle because calculation speed decreases considerably. To solve this, the city is partitioned into square-shaped sectors of sufficient size that the control zone cannot be occupying more than four at a time, as can be seen in Figure 4. Therefore, it is only necessary to calculate intersections for the lanes in the sectors that are occupied by the control zone.

4. The User-driven Vehicle Model

All the dynamic models have been developed using the bond graph technique. The main advantage of this technique is its multidisciplinary character, which enables the possibility of applying it to any field of physics and its capacity for integration in hierarchical structures such as those in the developed models.

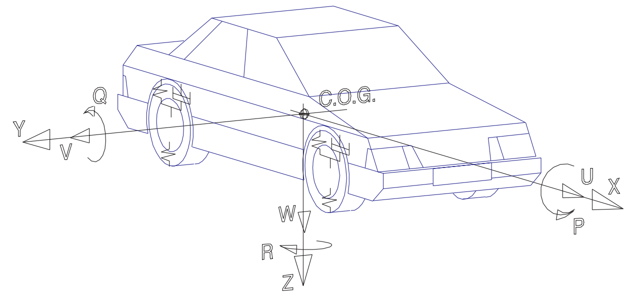


Figure 5. The vehicle

The vehicle model for the driven vehicle includes a three-dimensional model developed for the simulation of the complete vehicle's movement. The vehicle is simulated as a group of rigid bodies connected to one another, with the possibility to move in space. The vehicle is considered with two axes and four wheels with its steering and drive systems. The drive can be front, rear or four-wheel.

The vehicle model (Figure 5) contains fourteen degrees of freedom. Of these, six degrees of freedom correspond to the chassis, considered as a three-dimensional rigid body with movement capability in space. The remaining eight degrees of freedom are found in the wheels. Each wheel has two degrees of freedom, one of vertical displacement with regard to the chassis, and the other over its spin axis.

The wheels are connected to the chassis by means of the suspension system modeled with a spring and shock absorber in parallel, placed vertically with regard to the chassis. Furthermore, it allows the spinning of the wheels to be able to correctly simulate the force of traction and braking.

The front wheels can turn with regard to the vertical axis, controlled from the vehicle's steering maneuver. Therefore, these wheels will possess one more movement, even though it does not mean a degree of freedom because it is an input to the system.

In order to simulate the behavior of the tires, the 'magic formula' [36] is used, which reliably represents the experimental behavior of each kind of tire taking account of the effect of the variation of the vertical load acting upon the tire.

In order to develop the full multibody model for the vehicle, a multibond graph model was used. Figure 6 shows the bond graph of the vehicle. A more detailed explanation of this model can be seen in Félez et al. [37].

The drive set is modeled including the elements shown in Figure 7.

Vehicle interaction with the geometry of the road is introduced directly establishing a classical terrain-following algorithm between the wheels of the vehicle and the geometry of the ground.

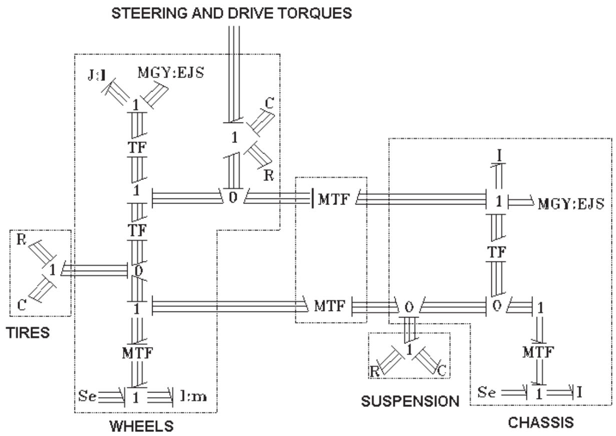


Figure 6. Multi-bond graph of the vehicle

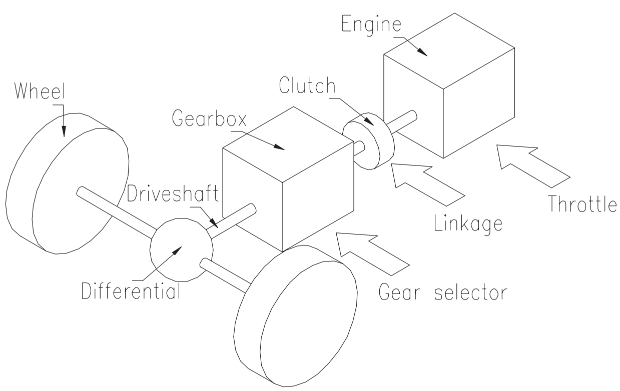


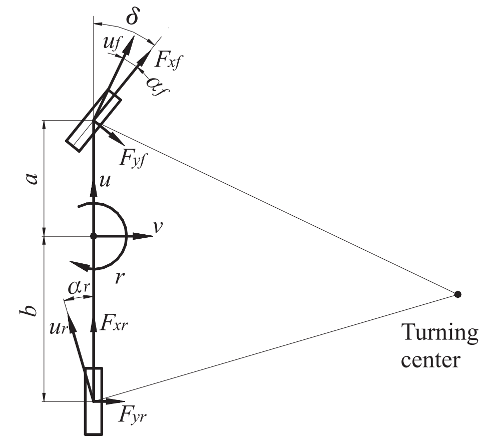
Figure 7. Drive set model. Sensor inputs

5. Automatic Vehicle Model

Kinematic vehicle models, like those used in MITSIM, PARAMICS or AIMSUN, have usually been used in traffic simulation in order to simplify simulator development and minimize the computation time needed. However, these models mean less realism in the simulation, since the behavior of the vehicles is pre-determined, so all of them move precisely along the same paths.

For this reason, and in order to endow the movement and behavior of the vehicles surrounding the user-driven vehicle with greater realism, a dynamic model was developed based on differential equations that enable more realistic behavior. With this model the vehicle does not keep strictly to its path, but tries to follow it by modifying and acting on the control parameters of the vehicle (steering, accelerator and brake), according to the specific parameters of the vehicle and driver.

Figure 8 shows the vehicle model and its geometry as well as its variables and parameters. The model is used



(b)

Figure 8. Vehicle model

with the assumption of small steering angles. The model has five degrees of freedom: longitudinal, transversal displacement and yaw angle.

Taking the steering angle into account and considering small angles, by approximating the sine of the angle to the angle and the cosine to the unit, equations (1), (2) and (3) define the movement of the vehicle body expressed in a body-fixed frame.

$$m \cdot (\dot{u} - v \cdot r) = F_{xf} + F_{xt} - \delta \cdot F_{yf} \quad (1)$$

$$m \cdot (\dot{v} + u \cdot r) = F_{yf} + F_{yr} + \delta \cdot F_{xf} \quad (2)$$

$$I_z \cdot \dot{r} = F_{yf} \cdot a - F_{yr} \cdot b. \quad (3)$$

In these equations, the transversal forces on the tires, F_{yf}, F_{yr} are obtained according to the slip angle α_i

$$\alpha_f = \delta - \frac{r \cdot a + v}{u} \quad (4)$$

$$\alpha_r = \frac{r \cdot b - v}{u}. \quad (5)$$

The meanings of the variables used are:

- u, v, r — longitudinal (x direction), transversal (y direction) and yaw velocities, expressed in a fixed body reference frame
- a, b — distance of the front and rear wheels to the vehicle center of mass
- δ — steering angle
- α_f, α_r — slip angle of the front and rear wheels
- u_f, u_r — velocities of the front and rear wheels that determine the vehicle turning center

F_{xf}, F_{yf} longitudinal and transversal tire forces in the front wheels
F_{xr}, F_{yr} longitudinal and transversal tire forces in the rear wheels
m total vehicle mass
$\dot{u}, \dot{v}, \dot{r}$ longitudinal, transversal and yaw acceleration corresponding to u, v, r velocities
I_z: vehicle momentum of inertia.

The longitudinal forces or rolling forces, F_{xf}, F_{xr}, are calculated by using the longitudinal slip coefficient

The inputs to the vehicle model are the three basic maneuvers that can be performed by the driver: steering angle, accelerator pedal percentage and brake pedal percentage.

The steering angle is introduced into the model as can be seen in equations (1) and (2). This last equation determines the forward slip angle, and thus the transversal force on the driving wheel, which is what makes the vehicle turn. The value of the steering angle to be applied at each instant of simulation is determined by the steering controller defined in Section 6.1.

Vehicles are modeled as if they had an automatic gear box with the assumption of constant power, modeled with a hyperbolic function.

The accelerator and brake percentage value to be applied in the simulation is determined by the controllers defined in Section 6.2.

6. Driver Model for Automatic Vehicles

The aim of the driver model is to simulate real driver behavior. The driver model developed consists basically of an evolution of the vehicle following theory habitually used in traffic behavior studies [32, 34, 38].

The general behavior of the traffic model is based on the fact that each vehicle is associated with a lane whose middle line, called a path, it tries to follow. Each path has a list of the vehicles that can be found on it. In this way, the situation of the surrounding traffic is determined at each moment, and it is this information that conditions the driver's behavior. The driver model establishes several functions:

6.1 Steering Angle of the Driven Vehicle

In order to determine the maneuvers that the driver must carry out on the steering wheel, a reference point is established on the path followed by the vehicle at a certain distance ahead, towards which the driver must continually try to steer (Figure 9).

The reference distance value is given by equation (6):

$$d_s = t_s u + d_0. \qquad (6)$$

The more aggressive the driver, the smaller is the t_s coefficient.

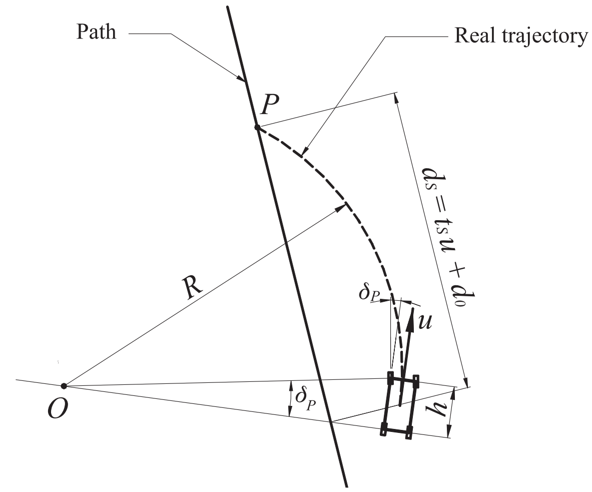


Figure 9. Determining the wheel steering angle

Figure 9 shows the calculation of the wheel steering angle. Once point P has been determined, it is necessary to calculate the radius R of the arc passing through this point, through the centre of the vehicle turning point O, and which is a tangent to the longitudinal velocity u direction. Therefore, by knowing the vehicle's wheelbase h, the theoretical wheel steering angle (equation (7)) will be:

$$\delta_P = \arctan\left(\frac{h}{R}\right). \qquad (7)$$

A proportional – differential (PD) controller is defined following the equation (8) in order to establish the steering angle value δ introduced into the vehicle model.

$$\delta = \delta_P + c_{d\delta} \cdot \frac{\mathrm{d}\delta_P}{\mathrm{d}t}. \qquad (8)$$

The proportional term coefficient of this controller has a value of 1. The differential controller coefficient $c_{d\delta}$ is defined as a result of a series of tests with vehicles moving along different types of path at different speeds, the more aggressive the driver, the greater the coefficient.

6.2 Acceleration and Breaking Maneuvers

The acceleration and braking controls establish the driver maneuver to be performed. The driver acceleration or braking maneuver depends on a set of references associated with its path, such as, for example, the position of the rear vehicle, the position of vehicles in adjoining lanes, the location of signposting, etc. A more detailed description of associated references can be seen in Maroto et al. [39].

A reference speed u_r is established, which the driver will try to keep to, and which is associated with a reference distance d_r. The set of both u_r and d_r determines the maneuver to be performed by the driver. The reference speed and distance used by the PD controllers and which determine the driver maneuvers, are the most restrictive among all the set of references that are considered.

Firstly, the vehicle acceleration a_r necessary to reach the required objective speed u_r is calculated, according to the equation (9), associated with a uniformly accelerated movement:

$$a_r = \frac{u_r^2 - u^2}{2d_r}. \quad (9)$$

The rate of acceleration is determined by means of a PD controller. The controller output expresses the accelerator or brake percentages as shown in the following expressions.

For acceleration (equation (10)):

$$acel = c_p \cdot a_r + c_d \cdot \frac{\mathrm{d}a_r}{\mathrm{d}t}. \quad (10)$$

The proportional controller coefficient is calculated in such a way that if the acceleration obtained a_r is less than the average acceleration corresponding to the type of driver a_m, the acceleration of the vehicle will be that calculated (equation (11)). On the contrary, if the acceleration obtained is greater, the acceleration of the vehicle will be equal to the average acceleration corresponding to the driver type (equation (12)). Therefore:

$$\text{If} \quad a_r < a_m$$

$$c_p = \frac{1}{\mu \cdot g} \quad (11)$$

$$\text{If} \quad a_r \geq a_m$$

$$c_p = \frac{a_m}{\mu \cdot g \cdot a_r}. \quad (12)$$

To determine the differential controller component value, successive tests have been made bearing in mind the relation between c_p and c_d.

The brake percentage is obtained using equation (13):

$$brake = \frac{a_r}{\mu \cdot g}. \quad (13)$$

The reference speed and distance used by the PD controllers and which determine the driver maneuvers, are the most restrictive among all the set of references that are considered [39].

6.3 Carrying Out Manoeuvres

The driver model decides which maneuvers will be carried out and executes them. The most usual maneuvers are: acceleration, braking, following a vehicle, lane changes and adapting to road signs.

In order to make a decision, the surrounding traffic conditions are taken into account. Each vehicle has information associated with the path where it is located, as well as the paths it will take if it is near a junction and has already chosen the corresponding connectivity. In turn, each path possesses information about adjacent paths. Since each and every one of the paths of the scenario contains a list of all the vehicles located on them at a given moment, one particular vehicle can obtain information about all the others around it, and whose behavior might affect it.

6.3.1 Behavior in Street Sections

When a driver is in a street section, their behavior is directly influenced by the vehicles preceding them in the same lane. Each driver has an objective speed established according to the street, their degree of aggressiveness, and the type of vehicle being driven. If the lane they are driving along is clear, they will carry on, tending to drive at their objective speed. However, if there are other vehicles ahead moving below their objective speed, the driver will try to change lanes in order to overtake them. To do this, they will check the state of the traffic in the adjacent lanes.

To decide if there is enough space free for the lane change, the minimum safety distance between the controlled vehicle i and vehicle j is determined.

6.3.2 Behavior at Junctions and Roundabouts

Lane changes are not allowed either at the junction access or inside it, which is why the associated references are not taken account of.

The vehicle driver acts according to two new additional references obtained from the intersections between the different connectivities making up the junction.

When the vehicle is close to the junction access or inside it, its behavior is determined by the position it keeps relative to other vehicles located on the junction connectivities. The priority at a junction without signals depends on the visibility between both vehicles.

The concept of degree of visibility can be seen in Figure 10. It consists of calculating the angle of the straight line joining the geometric centre of the vehicle with that of the other, and calculating the difference between this and the car's angle of orientation. The smaller this difference, the greater will be the driver's degree of visibility.

In general, the vehicle having the other in its angle of vision will concede the right of way. For example, in the situation of Figure 10, vehicle 2 will stop. If they cannot see each other, the one with the greatest degree of visibility will stop. The degree of visibility is determined when the vehicles come near the junction and switch to approaching mode.

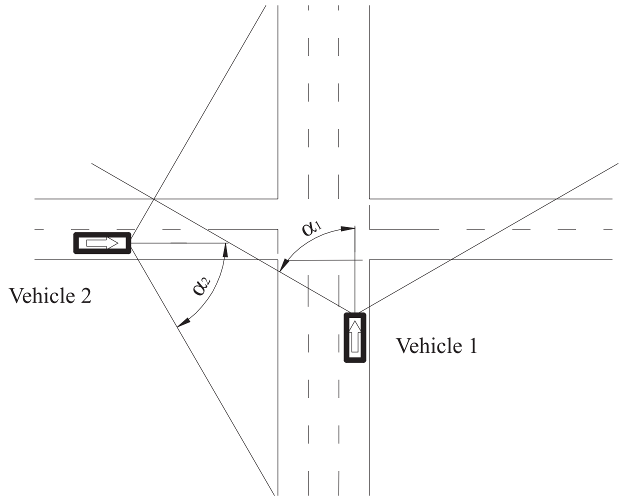


Figure 10. Angle of visibility of two drivers when approaching an intersection

6.3.3 Behavior with Regard to Signposting and Signals

The traffic model takes into account four types of signs and signals: traffic lights, stop, give way signs and pedestrian crossings. Below is a detailed description of how the driver model behaves with regard to each.

- Traffic Lights.
  The behavior of a driver when faced with a traffic light depends on the phase of the light and the driver's degree of aggressiveness. If the light is green, the driver's behavior is not affected, no matter what their degree of aggressiveness. If the light is red, the driver's behavior depends on the distance from the lights. Normally, they will stop, except if they are very close and are an aggressive or moderate driver. If the light is yellow, the behavior of the vehicle also depends on the distance. In general, the more aggressive the driver, the more they will tend to accelerate to go through the light.

- Stop Sign.
  The behavior of a driver when approaching a stop sign depends basically on their degree of aggressiveness. If no other vehicles are present in the proximity of the junction, an aggressive driver will not usually slow down on approaching the sign. However, a moderate driver will reduce speed, while a passive driver will stop. If the possibility of an imminent collision exists, the driver will stop the vehicle whatever their degree of aggressiveness. If there are other vehicles in the proximity of the junction, and there is a possibility of collision, the behavior of the driver of the controlled vehicle is determined by the relation existing between the arrival times of each of the vehicles at the junction.

- Give Way Sign
  The behavior of a driver approaching a give way sign is determined in a similar way to that for a stop sign, depending basically on their degree of aggressiveness. In general, if there are no other vehicles in the immediate proximity of the junction, an aggressive driver will not slow down. However, a moderate driver will slow down and a passive driver will stop at the white line, as if it were a stop sign. If there is a risk of imminent collision, the driver will stop the vehicle whatever their degree of aggressiveness.

- Pedestrian Crossing
  If a pedestrian is on a crossing, the driver will stop the vehicle or reduce speed until the pedestrian has finished crossing the road. If there is no pedestrian, the behavior of the driver will depend on their degree of aggressiveness. If it is a passive driver, they will reduce the speed of the vehicle on approaching the crossing. However, if the driver is moderate or aggressive, the speed will be maintained.

6.4 Traffic Light Regulation Model

The algorithm used to carry out traffic light regulation only controls the traffic lights within the control zone. At each instant, the model selects a traffic light, called the master traffic light, that serves as reference for establishing the status of the other traffic lights. This traffic light will be the one at the entry to the intersection nearest to the user-driven vehicle on the same street and moving in the same direction. Once the status of the master traffic light has been set, the model will determine the status of the other traffic lights on the intersection.

The total cycle time is shared among all the streets having traffic lights at the entry to the intersection, in such a way that all the traffic lights at the entry to an intersection that are on the same street have the same status, and when those on an entry street are green or yellow, those at the entry on other streets will be red. In this way, an exit traffic light will remain green when those at the entry to the street are such, and will remain on yellow for all the other statuses associated with it.

Once the status of all the traffic lights at the same intersection has been established, the status of the other traffic lights in the control zone will be determined. For each intersection a new reference traffic light will be taken, from which the status of the other traffic lights will be calculated according to the algorithm described previously. The reference traffic light status will be calculated with a time lapse that is proportional to the distance existing between the master traffic light and the position of the new reference traffic light. The traffic light regulation associated with the crosswalks will be opposite to that for the vehicles, their remaining on green only when the status associated to the traffic is red, and red for the others.

7. Accident Simulation

The behavior model for the simulated vehicles included in the traffic model is based on the following theory, trying to follow pre-established paths avoiding vehicle collisions. Changing lane or overtaking maneuvers are only performed when there is enough free space between vehicles. For this reason, in the general behavior of the traffic model, it is not possible that traffic accidents happen.

Traffic accidents can happen only in exceptional circumstances, only in specific traffic conditions and when the simulator instructor decides that an accident must be produced. In these circumstances, the simulator instructor decides what vehicle is going to produce the accident. When the vehicle has been selected, references used by the driver model related to the near vehicles are deactivated. Then, the vehicle has a free behavior and can collide with other vehicles or with obstacles in the road. When collision with other vehicle happens, references of this other vehicle are also deactivated.

7.1 Collision Model

Collision detection is performed in two steps. The first checks the intersection between the bounding spheres of each object. When intersection between bounding spheres is detected, the bounding boxes intersection is checked for the previous intersection detected, establishing the center of the area of intersection in order to define the contact point. When the contact point is detected, the direction of the corresponding collision forces is calculated and collision forces are applied.

Once it is possible to simulate the movement of a vehicle being driven on a road, the next step is to take two vehicles and implement a collision model between them.

The first operation in the simulation of a collision is the detection of the contact point between both vehicles. Then, the vehicles are connected at the point of collision by sets of springs and dampers, placed in a longitudinal and transversal position to the body of the car, (Figure 11). These sets of springs and dampers are those which simulate the elastic–plastic behavior of the body of the car during the collision.

The parameters of the springs and dampers have been adjusted from testing, by classifying the vehicles by types and within these by category according to their weight and wheelbase.

The values of the stiffness of the springs are considered to be constant throughout the simulation process. For the dampers, where the plastic deformations are going to occur, the law of damping variation is utilized as Figure 12 indicates.

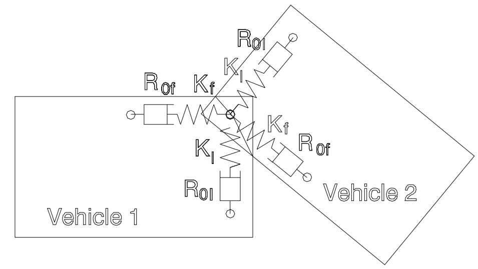


Figure 11. The collision model

Validation of the collision model was performed in Lozano et al. [18].

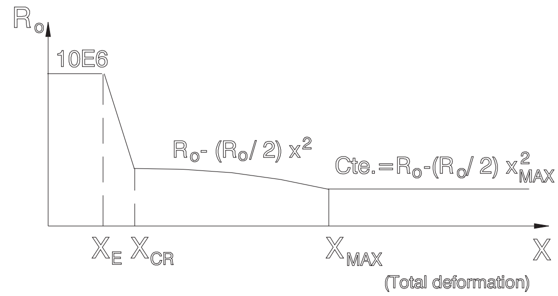
Two collision models have been developed. The first one corresponds to a car-to- car crash. In this case, no slip between cars has been considered. The second collision model corresponds to a crash between a car and a barrier. In this case, slip between car and barrier has been considered. Figures 13 and 14 show the bond graph of the collision models. More detailed information of these models can be seen in Lozano et al. [18].

7.2 Submodels Assembly

In order to optimize the time simulation during the crash simulation, a special algorithm has been developed optimizing the number of equations needed for each time step.

All the procedures developed up to now and used for traffic accident reconstruction build the entire set of dynamic equations of the vehicles and the collision models for all the possible combinations of vehicles or vehicles and obstacles. The developed procedure only considers the collision model between objects where there really has been a collision. Collision models have elements that permit the transmission of collision forces only when vehicles are in contact, but they keep calculating the possible collision forces.

The algorithm developed is based primarily on the detection of any geometric interference between two vehicles or a vehicle and an obstacle, as seen in Section 7.1. This process is firstly carried out by checking that the distance between the centers of gravity of two vehicles or between vehicle center and obstacle center is less than the sum of the radii of the bounding spheres of each object. Once this condition has been fulfilled, that is, when the bounding spheres of the two objects intersect, a second check is carried out to detect if the bounding boxes of the objects intersect. If this intersection occurs, the mid point coordinates of the intersection surface are calculated as local coordinates for each body. The forces generated by the collision model are then applied to these coordinates, which means that the collision model between two objects is only activated if interference is detected between the two bodies, thereby avoiding unnecessary calculations



Vehicle category		Front		Lateral		Rear	
		R_o (Ns/m) 10^6	K_f (N/m)	R_o (Ns/m) 10^5	K_l (N/m)	R_o (Ns/m) 10^3	K_t (N/m)
Car	small	1.132	81.7 R_o	1.230	81.7 R_o	2.222	81.7 R_o
	medium	3.580		3.052		3.756	
	large	4.236		3.694		4.868	
Van		5.946		3.525		4.178	
Truck	light	20		20		20	
	heavy	30		30		30	

Figure 12. Parameters of the collision model

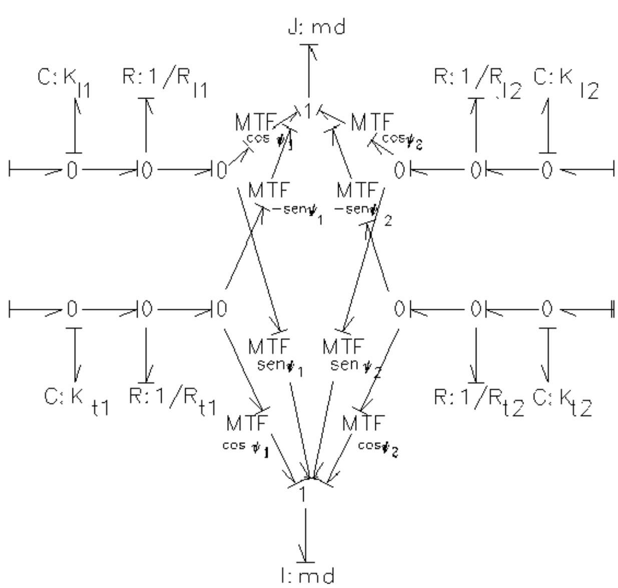


Figure 13. Car to car collision model

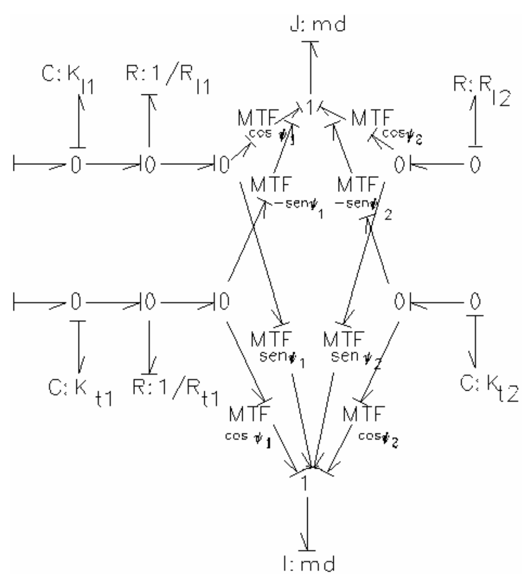


Figure 14. Car to wall collision model

in the collision models when such calculations are not needed in a collision situation.

The developed procedure only considers the collision model when a real collision exists (Figure 15). This figure shows how the dynamic models of five vehicles are activated. Only the collision models between vehicles 1 and 3, vehicles 1 and 2, and vehicle 1 with an obstacle and vehicle 5 with an obstacle, are activated. However, the collision model between vehicles 3 and 4 is not activated because no geometric interference between them was detected. With this procedure, a reduction of more than 50% of equations is obtained.

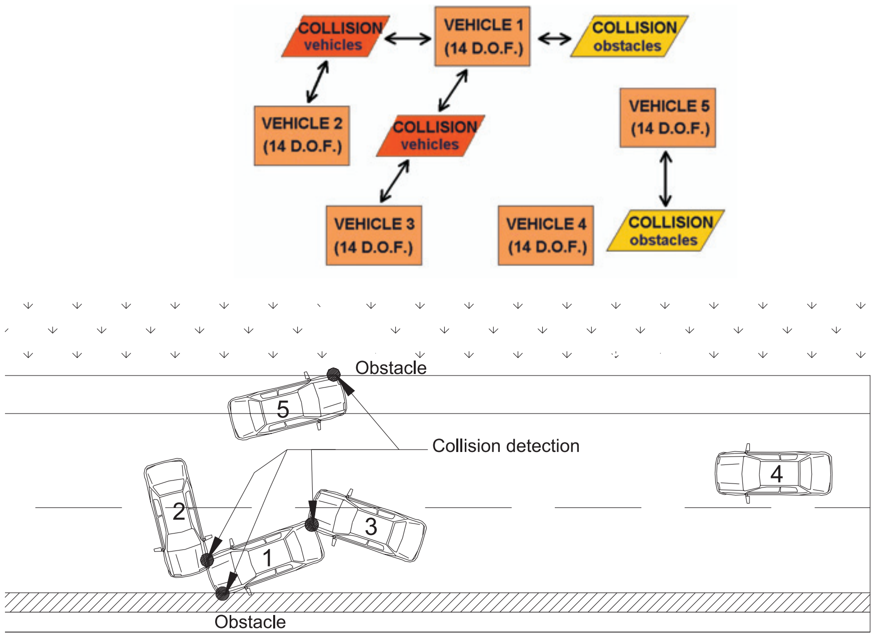


Figure 15. Assembly of submodels

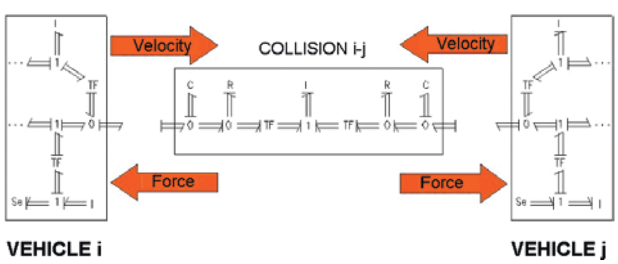


Figure 16. Assembly procedure

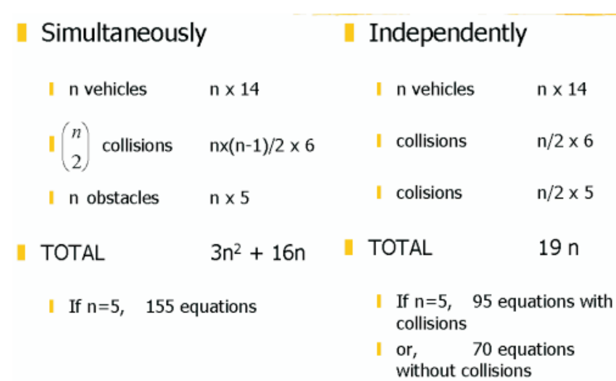


Figure 17. Simulation results

The assembly procedure is based on the exchange of flows and efforts in the connecting bonds between submodels. Each submodel is calculated independently, their having in common the flows and efforts present in the connecting bonds. In the vehicle model an effort corresponding to the collision force is introduced as a new variable. This variable is the reaction force obtained from the collision model. Then, in the collision model, a flow corresponding to the velocity of the collision point is introduced.

This implementation has been established so that it can be done with several processes running in parallel if necessary.

If all the vehicles and all the possible collision models are calculated simultaneously, a number of equations are obtained that correspond to the number of vehicles (n) multiplied by the number of equations of each vehicle, 14 $\times$ n equations, plus the combinations of possible collisions between two vehicles, i.e. $n \times (n - 1)/2 \times 6$ equations, plus the number of equations obtained from the collisions between car and obstacles, i.e. $n \times 5$ equations.

If the proposed method is considered, it is only necessary to solve the equations of vehicles, i.e. $n \times 14$, but during the main part of the simulation time, it is not necessary to solve the collision models. Also, it is not common for multiple collisions to occur simultaneously. The reduction of the number of equations is then very significant. Following the example of Figure 16, if there are five vehicles and five possible obstacles, 155 equations are needed if all the possible combinations are considered, but if we consider only the collisions that exist, 95 equations are needed. Figure 17 shows these results.

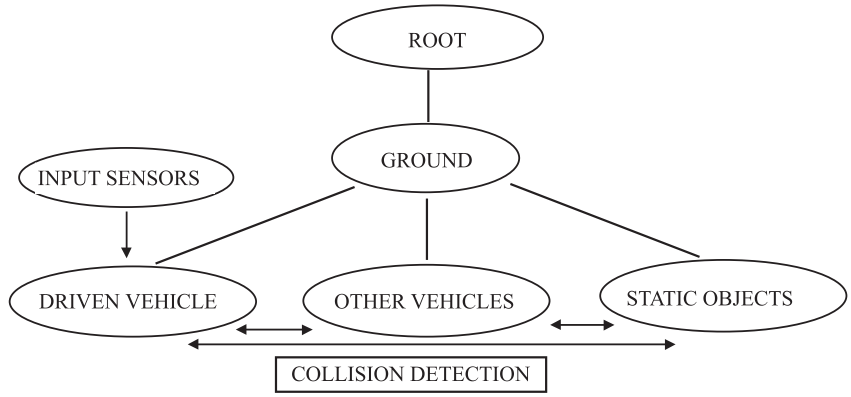


Figure 18. Scene graph of the virtual environment

8. The Virtual Environment

The virtual environment has been developed by basing it on a classical virtual reality application, where the following elements have been defined: geometry objects, lights, sound sources, tasks, behaviors, collision detection, and actions in the systems (driving actions in this case).

Several types of objects have been defined. All the objects or nodes have been structured in a hierarchical way, in a tree structure. The base node is the so-called ground. The ground node includes a three-dimensional mesh representing the complete terrain geometry where the accident occurs. The second types of nodes are the vehicles. These are movable nodes, changing position in each simulation step.

Vehicle movement can be obtained in two ways. The vehicle driven from the console has several associated sensors controlling its movement. The input sensors are transmitted to the vehicle actions through a model simulating the behavior of the engine, clutch, gearbox, drive shaft, differential and wheel. Inputs are the steering wheel, the lever to the gearbox and the clutch, and the accelerator, and brake pedals. The output is the torque in the wheels that is applied to the dynamic model of the vehicle.

The other vehicles have their movement governed by a pre-established driver behavior and they try to follow a path. When a collision is detected, the collision model applies the corresponding collision forces to each object. At this moment, the established path is disabled and the vehicle movement is obtained from its dynamic equations.

There are other types of object in the virtual environment. These objects are generically called static objects. They are non-movable objects that have no behavior in the environment. Basically there are two types of static objects: the obstacles and the areas with different ground adherence. In the obstacles, the possibility of collision with vehicles is checked, and if collision occurs, the collision model is applied to the vehicle. In the zones of different adherence, interference with vehicle wheels is checked. If interference is found, the corresponding adherence coefficient is applied in the dynamic model to the vehicle wheel. The dynamic behavior of the vehicles is introduced by means of a task in the vehicle node. The collision model is also introduced as a task in both objects when collision is detected.

Geometry nodes can have two more characteristics: level of detail (LOD) properties, or switch characteristics. LOD nodes are used when the simulator is working or not with an external user or it is simulating a particular accident without external actions. The use of LOD nodes is useful to improve computation time. When simulation is performed with external actions, in order to obtain real-time simulation, time step size has to be controlled. Switch nodes are used to represent different geometries of a car with different deformations when collision exists. Figure 18 shows the virtual environment developed schematically.

One of the main deficiencies arising in many of the current virtual environments is the lack of realistic dynamic behavior. The objects in the scenario usually include only very simple and simplified physical routines, which are only valid in specific situations and which are practically never accurate for realistic simulations.

Although the model could be used for traffic flow and control studies, it has been designed for the simulation of near traffic. Its most suitable application is for training vehicle drivers. The type of vehicle, car, van, truck or motorbike, only determines the model of the user vehicle.

An implementation of the model is being developed in the University for a multi-purpose simulator in which different types of vehicles can be simulated, such as road vehicles, cars, trucks, motorbikes, railway trains, ships and planes. The simulator has a six degrees of freedom platform with three visual channels mounted on it. Sounds and reactive steering wheel and controls, in conjunction with the movement provide a sensation of very realistic driving. In Figures 19 to 22 some pictures of the simulator and their visual images can be seen.

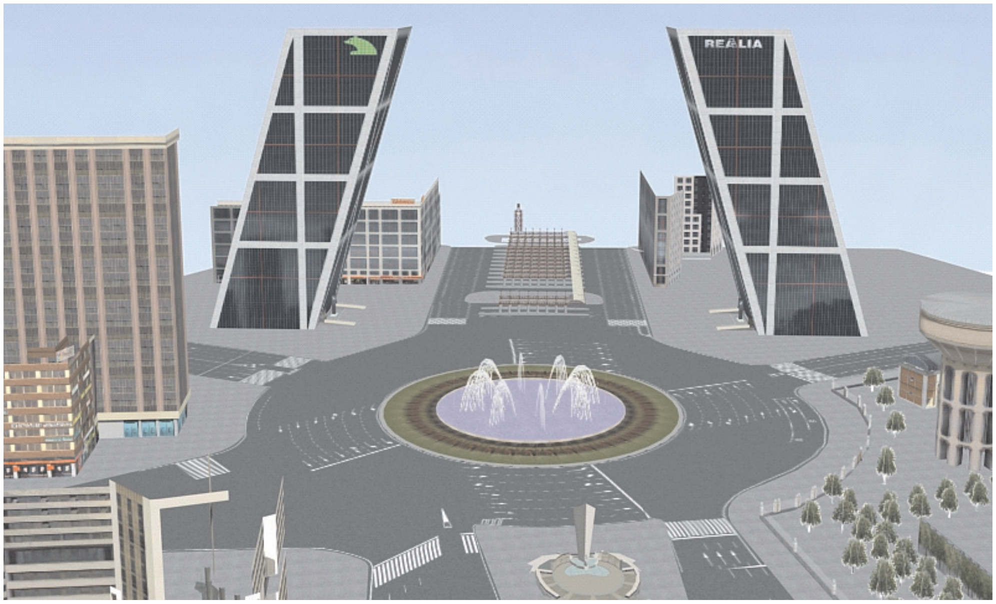


Figure 19. Model of Plaza de Castilla in Madrid, Spain

Figure 20. Driver view point, including rearview mirror

9. Conclusions

In this work, a traffic model, which is also able to reproduce traffic accidents, has been developed for implementation in a driving simulator in an urban environment. Therefore, the model needs to be suitable for real-time simulation and, in turn, provide a high degree of realism so that the user will become immersed in the environment.

The developed model is capable of functioning in real time using only a small percentage of the available computational resources imposed by the requirements of the number of vehicles simulated and control zone size. It has thus been shown that it is possible to simulate a great number of vehicles in a large control zone. This is because all the roads are considered as continuous paths along which the vehicles move, and lists are used to associate the latter with their paths.

The use of simplified dynamic models to simulate vehicle movement is highly realistic, without requiring a significant increase in the computational resources needed.

Figure 21. Top view of the environment

It has also been shown that the use of a purely microscopic model does not affect traffic realism, particularly when dealing with a generic city. Individual driver behavior and their decision making in the zone nearest to the driven vehicle can be faithfully reproduced without the need to solve complex traffic flow calculations throughout the entire city.

The traffic light simulation implemented can be executed with few resources being consumed and, at the same time, the state of all the traffic lights affecting the traffic simulated in the control zone is provided following a logical sequence.

Specific models for simulating crash and traffic accidents have also been included in the simulator.

10. Acknowledgments

This work was part of the project DPI2002-2198 and has been developed under the support of the Spanish Administration in the programs of the Ministry of Science and Technology related to Research in Transportation and Design and Industrial Production.

The authors would also like to extend their thanks to the recently deceased Professor Carlos Vera for his learned advice and contributions to the work included in this paper.

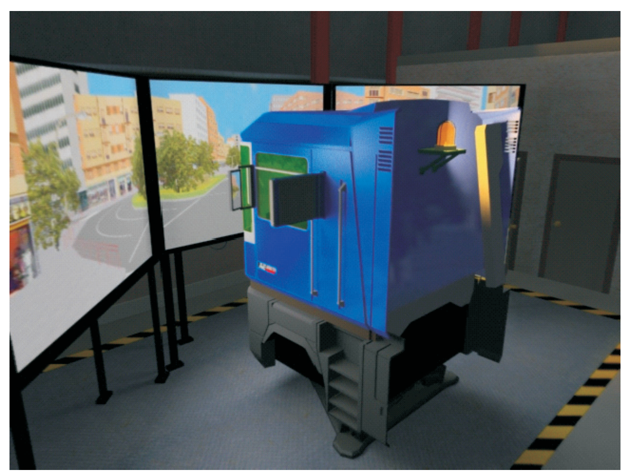


Figure 22. Driver cabin

11. References

[1] Chowdhury, D., L. Santen, and A., Schadschneider. 2000. Statistical physics of vehicular traffic and some related systems. *Physics Reports* 329: 199–329.

[2] Helbing, D. 2001. Traffic and related self-driven many-particle systems. *Reviews of Modern Physics* 73, 1067–1141.

[3] Bham G. H., and R. F. Benekohal. 2004. A high fidelity traffic simulation model based on cellular automata and car-following concepts. *Transport Research, Part C: Emerging. Technology* 12(1), 1–32.

[4] FHWA. 1980. *Development and Testing of INTRAS, A Microscopic Freeway Simulation Model, Vol. I: Program Design, Parameter Calibration and Freeway Dynamics Component Development*. Report No. FHWA/RD-80/106. US Department of Transportation.

[5] Halati, A., L. Henry and, S. Walker. 1998. CORSIM-corridor traffic simulation model. In: *Proceedings at the Traffic Congestion and Traffic Safety Conference*, Chicago, ASCE, pp. 570–576.

[6] Benekohal R. F. 1989. *Procedure for Validation of Microscopic Traffic Flow Simulation Models*. Transportation Research Record 1320, TRB, National Research Council, Washington, DC, pp. 190–202.

[7] Aycin, M. F., and R. F. Benekohal. 1998. *Linear Acceleration Car-Following Model Development and Validation*. Transportation Research Record 1644, TRB, National Research Council, Washington, DC, pp. 10–19.

[8] Yang, Q., and H. N. Koutsopoulos. 1996. A microscopic traffic simulator for evaluation of dynamic traffic management systems. *Transportation Research Part C* 4(3): 113–129.

[9] Duncan, G. I. 1995. PARAMICS wide area microscopic simulation of ATT and traffic management. In: *Proceedings of the 28th ISATA Conference*, Stuttgart, Germany.

[10] Barceló J., and J. Casas. 2002. Dynamic network simulation with AIMSUN. In *Proceedings of the International Symposium on Transport Simulation*, Yokohama, Japan, pp. 1–25.

[11] Barceló J., J. F. Ferrer, D. Garcia, M. Florian, and E. Le Saux. 1996. The parallelization of AIMSUN2 microscopic simulator for ITS applications. *Proceedings of the 3rd World Congress on Intelligent Transportation Systems*. Orlando, 1996.

[12] Hidas P., and K. Behbahanizadeh. 1998. SITRAS: a simulation model for ITS applications. In: *Proceedings of the 5th World Congress on Intelligent Transport Systems*, Seoul, Korea, October, 1998.

[13] McHenry, R. R, and B. G. McHenry, 1997. Effects of Restitution in the application of crush coefficients. In: *Proceedings of the 1997 SAE Congress*, SAE Paper No. 97-0960. Also in SAE publication SP-1237.

[14] Woolley, R., C. Stronther, and M. James. 1991. Rear stiffness coefficients derived from barrier test data. SAE Paper 91-0120.

[15] Day, T. D., and Hargens, R. L. 1987. An overview of the EDCRASH computers Delta-V. SAE Paper 87-0045

[16] Prasad, A. K. 1991. Missing vehicle algorithm (OLDMISS) reformulation. Accident Reconstruction – Technology and Animation. Paper 910121, SAE SP-853, pp. 25–36

[17] Vera, C., J. A. Lozano, F. Aparicio, and J. Félez. 1995. SINRAT IV. A program for traffic accident reconstruction. In: *Proc. of the ICBGM'95*. SCS. Simulation Series, vol. 27, number 1, pp. 209–214

[18] Lozano J. A., C. Vera, and J. Félez. 1998. A computational dynamical model for traffic accident reconstruction. *International Journal of Vehicle Design*, 19(2): 213–227.

[19] Wirley, R. G. and B. E Engelmann. 1993. Automatic contact in DYNA3D for vehicle crashworthiness analysis. In: *Proceedings of the ASME Winter Annual Meeting. Symposium on Crashworthiness and Occupant Protection*.

[20] Kuschfeldt, S., M. Schulz, T. Ertl, T. Reuding, and M.Holzner. 1997. The use of a virtual environment for FE analysis of vehicle crash worthiness. In: *Proceedings of the IEEE 1997 Virtual Reality Annual International Symposium*, Albuquerque, NM, USA, IEEE Computer Society Press

[21] Drosdol, J. and F. Panik. 1985. The Daimler-Benz Driving Simulator, SAE Paper 85-0334.

[22] Thales. 2006. http://www.thalesgroup.com/training-simulation/civil/heavy_vehicles/1_0_725_10162.html

[23] Rheinmetall Detec. 2006. http://www.rheinmetall-detec.com

[24] University of Iowa. 2006. http://www.nads-sc.uiowa.edu

[25] University of Michigan. 2006. http://www.umich.edu/~driving/sim.html

[26] University of Central Florida. 2006. http://catss.engr.ucf.edu/default.htm

[27] INRETS. 2006. http://www.inrets.fr/ur/msis/simus/sim2.htm

[28] Narayanasamy V., K. W. Wong, C. C. Fung. and S. Rai. 2006. Simulation and games: Distinguishing games and simulation games from simulators. In: *Computers in Entertainment (CIE)*, 4(2): ACM Press

[29] Esser, J. and M. Schreckenberg. 1997. Microscopic simulation of urban traffic based on cellular automata. *International Journal of Modern Physics C*, 8(5): 1025–1036.

[30] Wahle, J., A. Bazzan, F. Klügl, and M. Schreckenberg. 2000. Decision dynamics in a traffic scenario. *Physica A: Statistical Mechanics and its Applications*, 287(3–4): 669–681.

[31] Willemsen, P. J. 2000. *Behavior and Scenario Modelling for Real-Time Virtual Environments*. Doctoral Thesis, University of Iowa.

[32] Lieu, H., N. Gartner, C. J Messer,.and A. K. Rathi. 2004. Traffic flow theory. U.S. Department of Transportation. Federal Highway Administration. http://www.tfhrc.gov/pubrds/janfeb99/traffic.htm

[33] Van den Berg, M., A. Hegyi, B. De Schutter, and J. Hellendoorn. 2003. A macroscopic traffic flow model for integrated control of freeway and urban traffic networks. In: *Proceedings of the 42nd IEEE Conference on Decision and Control*, Maui, Hawaii, pp. 2774–2779.

[34] Montero, L., E. Codina, J. Barceló, and P. Barceló. 2001. A combined methodology for transportation planning assessment. Application to a case study. *Transportation Research Part C: Emerging Technologies*, 9(3): 213–230.

[35] Bonakdarian, E., J. Cremer, J. Kearney, and P. Willemsen. 1998. Generation of ambient traffic for real-time traffic simulation. Computer Science Department, The University of Iowa. IMAGE Conference, Scottsdale, Arizona.

[36] Bakker, E., L. Nyborg, and H. Pacejka. 1987. Tire modeling for use in vehicle dynamics studies. SAE Paper 870421.

[37] Félez, J., C. Vera, I. San José, and R. Cacho. 1990. BONDYN: A bond graph based simulation program for multibody systems. Transactions of ASME: *Journal of Dynamic Systems, Measurement and Control*, 112: 717–727.

[38] Zhang H. M., and T. Kim. 2005. A car-following theory for multiphase vehicular traffic flow. *Transportation Research Part B: Methodological*, 39(5): 385–399.

[39] Maroto J., E. Delso, J. Felez, and J. M. Cabanellas. 2006. Real-time traffic simulation with a microscopic model. *IEEE Transactions on Intelligent Transportation Systems,* (in press).

***Jose Mª Cabanellas** received his Mechanical Engineer and Doctoral degrees from the Universidad Politécnica de Madrid in 1989 and 1998. He started as Associate Professor at the Technical University of Madrid in Spain (UPM) in 1991 and became Professor in 2001. His activities began in the robotic field, new technologies in industrial vehicles and actually are mainly focused in the simulator development field. His research includes simulation techniques, computer graphics, artificial sight, distributed communications architectures, virtual reality, real motion and reactive instrumentation. His research also includes training with simulators and innovative techniques for teaching. He has published over 30 technical papers and has been actively involved in over 15 research and development projects. He has served as thesis advisor for six master's theses and two doctoral dissertations.*

***Jesus Felez** received his Mechanical Engineer and Doctoral degrees from the University of Zaragoza in 1985 and 1989. He*

started as Associate Professor at the Technical University of Madrid in Spain (UPM) in 1990 and became a Full Professor in 1997. His main activities and research interests are mainly focused in the field of simulation, computer graphics and virtual reality. His research includes simulation techniques based integrating computer graphics and virtual reality techniques, mainly addressed to the development of simulators. He has published over 50 technical papers and has been actively involved in over 25 research and development projects. He has served as thesis advisor for 30 master's theses and six doctoral dissertations. Professor Felez is a member of ACM, SCS and IEEE and takes a very active part in these organizations.

***Joaquin Maroto** has received his PhD in 2005 and the Control Engineer Grade in year 2000 both at the Technical University of Madrid. At present, he is Assistant Professor at the Technical University of Madrid in Spain. His main research activity is based on traffic simulation models, computer graphics and computer vision. His PhD thesis was focused on the generation of a three-dimensional view urban traffic simulator, behaviour models generator and development of a graphic engine, including IT Management. He has published over 50 technical papers.*

***Gregorio Romero** has received the PhD in 2005 and the Control Engineer Grade in year 1999 both at the Technical University of Madrid. He is Assistant Professor at the Technical University of Madrid in Spain. His main research activity is based on computer simulation models, virtual reality and computer vision.*