

A physical multi-body car model using 3-D
(screw) bond graphs

G.R. de Boer

M.Sc. Thesis

Supervisors: Prof. dr. ir. Job van Amerongen
Dr. ir. Stefano Stramigioli
Dr. ir. Edward Holweg
Ir. Vincent Duindam

June 2002
Report Number Report
Number

Control Laboratory
Dpt. of Electrical
Engineering

University of Twente
P.O. Box 217
NL 7500 AE Enschede
The Netherlands

Acknowledgements

This project and report owe their existence to the help and guidance of several persons which do not deserve to remain unmentioned. First of all my appreciation goes out to Stefano Stramigioli for being an (almost) indefatigable supervisor and inspirator. Vincent Duindam deserves my thanks for all the time he spent demystifying mathematical and modeling difficulties. Also, my gratitude goes out to the 20-SIM team: Paul Weustink, Frank Groen and Christian Kleijn, for unravelling numerous obscure bug descriptions and returning useful tips and effective simulator adjustments. Last but certainly not least, I thank Martijn Visser for allowing me to adapt his model of the bouncing ball for my personal use of tire contact. Moreover I am grateful that he was willing to be my 'councilor-by-mail' in numerous occasions.

Summary

The goal of the project was to realize an infrastructure of a car in 20-SIM using (screw) bond graphs. Bond graph description is an energy based means of modeling physical systems. This model can be used for investigation of the X-by-wire concept and independent wheel steering. Enquiry into the basic features of a car provided for the necessary insight in the course of development of the car. Rigid bodies have been used to describe the most significant masses of the car. These are interconnected as so-called kinematic pairs. Within these kinematic pair connections, constraints are used to restrict the separate elements of the car to specific motions only.

Setting the steering angle has been realized with the use of the Ackermann algorithm, which distinguishes between the track (and thus also the radius of curving) of the two front wheels. For more realistic steering behavior, the steering axes of both front wheels have been canted slightly, according to the Kingpin and Caster angles in genuine cars. The resulting model is a realistic reflection of a car is obtained.

With the use of Gauss-maps on the surfaces of the tire and the ground, a contact determining algorithm has been developed. The velocity of the contact point of the wheel relative to the contact point of the surface is fed to a buffer returning an oppositely oriented force. Thus a forward propulsion is accomplished. Monitoring the force in the contact point and switching to coulomb friction behavior in case the force exceeds a set threshold value, enables the modeling of slip behavior. The slip in the horizontal plain has been coupled. Slip in one direction results in a loss of grip in the perpendicular direction as well.

It can be concluded that with the use of 20-SIM a physically correct, energy based model of a car has been realized. All basic functionality of a car is comprised. The model has been parameterized for easy customizing to specific desires. It is recommended that the (rear or front wheel) torque-based, PID controlled drive is replaced by a velocity driven system. A rear wheel version of the Ackermann algorithm has then to be implemented as well. Thus removing the correctional influence of the PID controller might seriously diminish simulation time. Moreover an investigation to minimizing the number of rigid bodies or kinematic connections in the model can prove to be worthwhile. This could also significantly reduce the simulation time of the model. Furthermore the center of mass of the car needs some looking into. It is currently situated in the middle of the car body. In real cars the engine and car framework are most weighty and they are predominantly in the lower part of the car body. Hence the center of mass should be shifted downwards and to the front of the car.

Contents

1	Introduction	9
2	Bond Graphs	11
2.1	Causality	11
2.2	Bond Graph Elements	12
2.2.1	Sources	12
2.2.2	Dissipators	12
2.2.3	Buffers	12
2.2.4	Transformers	13
2.2.5	Gyrators	13
2.2.6	Junctions	13
2.3	Screw Bond Graphs	14
3	The Car	17
3.1	The Suspension	18
3.2	The Wheel	19
3.3	The Steering	20
3.3.1	Ackermann Steering	20
3.3.2	Kingpin and Caster	21
4	The Model	23
4.1	Wheel and Axle	23
4.1.1	Rigid Bodies	24
4.1.2	Constraints	26
4.2	The Car Body	28
4.2.1	The Suspension	28
4.2.2	Four Wheels and a Car Body	29
4.3	The Steering	29
4.4	The Driving	31
4.5	The Tires	32
4.5.1	Contact Behavior	32
4.5.2	Wheel Slip	33
5	Parameters and Verification	35
5.1	Parameters	35
5.2	Verification	36

6	Conclusions and Recommendations	41
6.1	Conclusions	41
6.2	Recommendations	42
A	Tips and Details	45
A.1	Practical Tips	45
A.2	The Model Close Up	46
	A.2.1 the Global submodel	46
	A.2.2 The Suspensions	46
	A.2.3 The Steering and Driving Controllers	46
	A.2.4 The Contact Parameters	46

Chapter 1

Introduction

The design of a car nowadays involves a multitude of disciplines. Although in the course of time, the emphasis in the design has been extended from mere functional and technical cunning to e.g. aesthetics and originality, the focus of the research in this thesis will be on the former aspect. The report will enquire into the physical behavior of a car, and how to describe it appropriately.

Some other disciplines besides the functional behavior of enabling the driver to move from point A to point B, are safety and comfort. Much research has been performed in the field of airbags, seatbelts and energy dissipating frameworks in order to increase safety. Active driving control systems influencing the power and brake distribution to the tires, such as ABS (Anti-lock Braking System) and ESP (Electronic Stability Program) provide the user with more safety as well.

More comfort nowadays is achieved with systems like active suspensions, adaptive seats, noise reduction in the cockpit and a variety of electronically remote-controlled features for e.g. windshields, mirrors and headlights.

A field of research covering both safety and comfort is the X-by-wire approach. Similar to the fly-by-wire systems in aeroplanes, the goal is to replace all mechanically or hydraulically controlled functions of a car by electronic actuation. This means no more hydraulic brakes, no mechanical gas lever, no steel steering shaft. Electrical control of these functions enable even more fundamental driving control systems.

Hence, steer-by-wire implies that the steering wheel will not be mechanically coupled to the front wheels. the steering wheel will be attached to an electrical actuator to give a force feedback and a sensor to measure the position of the angle set by the driver. On the other hand, the tire angles will be actuated by an electrical motor. In between, there will be a computer which will evaluate the drivers actions and actuate the tires accordingly.

A further step is the possibility to even uncouple the two front wheels and actuate them separately with two electrical drives. This gives the possibility to have different relative turning angels depending on the speed and on the driving situations. A control strategy needs to be found for this independent steering.

The assignment is to develop a physically correct infrastructure to investigate the dynamic consequences of such an independent steering approach on the driving stability of the car. This investigation requires a multi-body dynamic model, comprising the main physical features of a car. The total 3D multi-body

structure of a car should be considered; especially the contact between the tires and the ground. The required model will be developed using (screw) bond graph techniques.

Of course there are quite a few software tools available aiming specifically at car design. Most of these can be categorized as either industrial (CAD) design tools for constructional design or mathematical models for physical descriptions. To the authors knowledge, no car model has yet been developed using screw bond graphs.

Screw bond graphs have been chosen for the modeling because they represent an energy based approach. This approach facilitates energy consistent implementation and it reduces the error proneness of the model. Moreover, screw bond graphs provide for a means of modeling, independent of the physical domain of the model. They can be applied whether a specific part of the model belongs to the electrical, mechanical or chemical domain. This means that all model parts (e.g. engine, suspension, steering controller) can be modelled in the same structure.

For the bond graph design, 20-SIM has been chosen. This is a very apt modeling tool, combining powerful model calculation features with useful visualization features. Most admirable is its DirectX 3D environment, allowing an instant 3D display of the model.

This report is organized as follows. In Chapter 2, a brief introduction to bond graphs and an explanation of the used bond graph elements is given. Chapter 3 gives an analysis of the basic functions of a car and will anticipate on some design choices that have to be made. The actual modeling trajectory and implementation in 20-SIM and a more detailed discussion of the decisions made in the design are described in Chapter 4. Chapter 5 contains the parameter choices and a simulation, displaying some typical experiment outcomes and car-like behavior. The final chapter discusses the resulting model, and gives conclusions and recommendations for further research. A detailed discussion and explanation of the model is found in Appendix 6.2.

Chapter 2

Bond Graphs

In the history of design and analysis of complex systems, the need for accurate models of these systems has increased. Modeling based on energy flows has proven to be very rewarding. When connecting two systems, energy can be exchanged between them. Bond graphs are a means of describing an interconnection of systems on such an energy flow-basis, independent of the physical domain of the model. They were first introduced by Paynter (Paynter, 1961).

Bond graphs are built up as follows. The positive orientation of the power is indicated by the direction of a half arrow, called a bond (see Figure 2.1). This

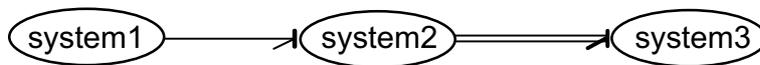


Figure 2.1: a bond and a multi bond

figure also shows a so-called multi bond. This is a bond which transports energy flows with more than one dimension (e.g. for a movement along two axes).

A bond actually consists of two factors, the *generalized flow* and the *generalized effort*. For any of the physical domains in the model, a bond is available which consists of an appropriate version of these two variables. So the power P , represented by the bond graph, can be represented by the product of its effort e and flow f .

$$P = e \cdot f$$

In the electrical domain the effort and flow are respectively the *voltage* U and *current* I . In the mechanical domain they can be recognized as the *force* F and *velocity* v in case of translation, and as the *torque* T and *angular velocity* ω in case of rotation.

2.1 Causality

As can be seen in Figure 2.1, the tip of the bond has a small stroke attached to it. This is called the *causal stroke* and it indicates the direction of the effort in the bond. So in the figure, the effort is an input of system 2. The flow is always

directed in the opposite direction of the effort, so in the figure, the flow is an output of system 2.

Sometimes the causality can be freely chosen, but quite often an element has a preferred or even a fixed causality. When modeling, these restrictions have to be well considered. Neglecting them could result in undesired simulation results or even an inability of 20-SIM to evaluate the model.

2.2 Bond Graph Elements

20-sim provides for many basic elements to create an actual model. This section gives a short description of the elements used in the model of the car, and how they can and should be used. We can distinguish between *sources*, *dissipators*, *buffers*, *transformers*, *gyrators* and *junctions*.

2.2.1 Sources

Source elements (S-elements) are power generating elements. One can choose to use a Source of Effort S_e or a Source of Flow S_f . Other possibilities are the Modulated Source of Effort MS_e and the Modulated Source of Flow MS_f . These sources have the possibility of being controlled by an external signal. The causality of these elements is fixed, so S_e and MS_e elements are effort-out and S_f and MS_f are flow-out (or effort-in).

2.2.2 Dissipators

Dissipators are elements that can dissipate energy. In 20-sim its symbol is an R . The R-element can be used to model resistance or friction. The causality of the R-element is indifferent. The dissipator can be both effort-in as flow-in.

2.2.3 Buffers

Buffers are elements that can store energy. We can distinguish between two types: I-elements and C-elements. The linear I-element relates the flow and the integrated effort p . The energy ($E(q)$)function of the I-buffer is

$$E_I(p) = \frac{1}{2I}p^2 \quad \text{with} \quad p = \int e \, dt + p(0)$$

$$f = \frac{\partial E}{\partial p} = \frac{1}{I} p$$

The linear C-element on the other hand relates the effort and the integrated flow q .

$$E_C(q) = \frac{1}{2C}q^2 \quad \text{with} \quad q = \int f \, dt + q(0)$$

$$e = \frac{\partial E}{\partial q} = \frac{1}{C} q$$

Their causality depends on whether the constitutive equations use an integration or a differentiation to calculate the physical state of the storage function. The preferred causality is always such that an integration can be used.

2.2.4 Transformers

Ideal transformers are depicted by a TF-element. It is a power-continuous multi port element, relating both the efforts and the flows of the input and output. The effort relation is described as follows.

$$e_{in} = n e_{out}$$

The demand of power continuity appoints the constitutive relation of the flow.

$$f_{out} = n f_{in}$$

Similar to the modulated sources, these TF-elements also come in a modulated version. These elements are depicted as *MTF*. In an MTF, the transformation factor n is set externally and may vary in time.



Figure 2.2: transformer element

2.2.5 Gytrators

A *GY*-element is also a power-continuous multi port element, and it relates the input effort to the output flow and vice versa. So these are its constitutive relations.

$$e_{in} = g f_{out}$$

Power continuity requires the second relation to be:

$$e_{out} = g f_{out}$$

Similar to the transformer element, also a modulated version of the gyrator, *MGY*, can be applied. In these elements, the gyration factor g is applied externally and can have a varying value.



Figure 2.3: gyrator element

2.2.6 Junctions

To be able to manufacture any structure whatsoever with these elements, connection elements are needed. These elements need more than two ports. Two junctions can be recognized, the *0-junction* and the *1-junction*. They have to answer the following two requirements.

Power continuity which means that the junction can not dissipate or generate energy, so that the algebraic sum of the power of all the ports is zero.

Port symmetry implying that the constitutive relations of the ports are exchangeable. One can develop these relations to the effort of flow of any of the connected elements.

The 0-junction behaves analog to *Kirchhoff's junction rule* in the electrical domain, so the sum of all flows entering and leaving the junction is equal to zero and all the efforts have the same value. For example, a *parallel connection* in the electrical domain is modeled in this fashion. In the 1-junction on the other hand, the sum of all the efforts in the junction is zero and all the flows are equal, just like in *Kirchhoff's loop rule* for the electrical domain. An example of such a junction in the electrical domain is a *series connection*.

2.3 Screw Bond Graphs

Before proceeding to the model, the concept of screw bond graphs has to be reviewed. Consider a 3 dimensional Euclidean space, $\mathcal{E}(3)$. We can describe the position and orientation of every body by an element of the Special Euclidean group $SE(3)$. Now consider two right-handed coordinate frames within this space, Ψ_i and Ψ_j . A general change of Cartesian coordinates in $\mathcal{E}(3)$ from Ψ_i to Ψ_j can be expressed with a homogeneous matrix of the form:

$$H_i^j = \begin{bmatrix} R_i^j & p_i^j \\ O_3^T & 1 \end{bmatrix} \quad (2.1)$$

where R_i^j is a rotation matrix (element of the special orthonormal group $SO(3)$) and p_i^j is a vector in \mathbb{R}^3 . H_i^j denotes the change of coordinates from a right-handed coordinate frame Ψ_j to another right-handed coordinate frame Ψ_i . Thus it can be used to for example describe the position and orientation of a body (with attached coordinate frame Ψ_j) relative to a reference (inertial) coordinate frame (Ψ_i).

The instantaneous velocity of a body i with frame Ψ_i relative to a body j with frame Ψ_j can be represented by a *twist* $T_i^{k,j}$, with

$$T_i^{k,j} = \begin{bmatrix} \omega_i^{k,j} \\ v_i^{k,j} \end{bmatrix} \quad (2.2)$$

where $\omega_i^{k,j}$ denotes the angular velocity of body i relative to body j expressed in coordinate frame Ψ_k . On the other hand, $v_i^{k,j}$ denotes the instantaneous lateral velocity (relative to frame Ψ_j) of the point fixed in frame Ψ_i that passes through the origin of frame Ψ_k . A twist can be regarded as the derivative of a homogeneous matrix in the following way:

$$\tilde{T}_i^{j,j} := \begin{bmatrix} \tilde{\omega}_i^{j,j} & v_i^{j,j} \\ 0 & 0 \end{bmatrix} = \dot{H}_i^j H_j^i$$

where $\tilde{\omega}$ is the skew-symmetric matrix equivalent to $(\omega \times \cdot)$. Similar to the homogeneous matrix, describing the change of coordinates for two coordinate frames, an Adjoint matrix can be used to change the coordinate frame in which the twist is expressed. We define the adjoint operator $Ad_{H_j^k}$ as

$$T_i^{k,j} = Ad_{H_j^k} T_i^{j,j} = \begin{bmatrix} R_j^k & 0 \\ \tilde{p}_j^k R_i^k & R_j^k \end{bmatrix} T_i^{j,j}.$$

(Duindam and Stramigioli, 2002b)

Chasles' decomposition theorem states that any element of $SE(3)$ can be described as a pure instantaneous rotation around an axis plus a pure instantaneous translation along the same axis:

$$\underbrace{\begin{bmatrix} \omega \\ v \end{bmatrix}}_{\text{twist}} = \omega \parallel \underbrace{\begin{bmatrix} \hat{\omega} \\ r \wedge \hat{\omega} \end{bmatrix}}_{\text{rotation}} + \lambda \underbrace{\begin{bmatrix} 0 \\ \hat{\omega} \end{bmatrix}}_{\text{translation}} \quad (2.3)$$

where $\omega = \|\omega\| \hat{\omega}$ and λ is called the *pitch*. Examination of this formula with reference to figure 2.4 shows that v is the velocity of an imaginary point passing through the origin of the coordinate system in which the twist is expressed and moving together with the object. The twist can be associated with a geometrical line, namely the line passing through r and spanned by ω , which is left invariant by the rotation.

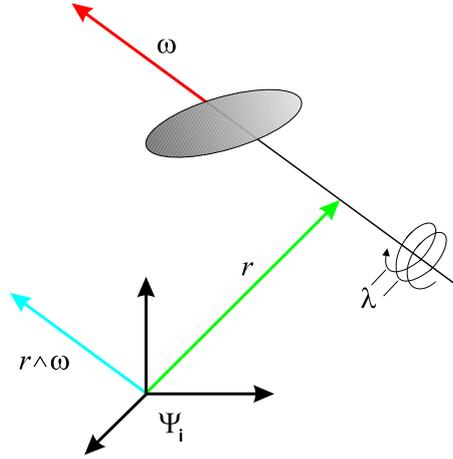


Figure 2.4: intuition of a twist

More information on twists can be found e.g. in (Murray, Li and Sastry, 1994; Stramigioli, 2001).

We can also define a wrench W_i^k (the dual of a twist), which describes the generalized forces acting on body i and expressed in frame Ψ_k , as

$$W_i^k = \begin{bmatrix} \tau_i^k \\ F_i^k \end{bmatrix} \quad (2.4)$$

where F_i^k denotes the linear force and τ_i^k the momentum, acting on the point in the origin of frame Ψ_k . The dual product of a twist and a wrench (when expressed in the same coordinate frame) is equal to a power flow.

The decomposition of twists using Chasles' theorem, can be given for wrenches by its dual analogous theorem, called *Poinsot's theorem*. This theorem states that any element of the dual SE^*3 (which is the vector space of *linear operators*

on $SE(3)$, i.e. the space of wrenches) can be split as a sum of two terms:

$$\underbrace{\begin{bmatrix} m \\ f \end{bmatrix}}_{\text{wrench}} = \underbrace{\begin{bmatrix} r \wedge f \\ f \end{bmatrix}}_{\text{force}} + \lambda \underbrace{\begin{bmatrix} f \\ 0 \end{bmatrix}}_{\text{momentum}} \quad (2.5)$$

Analogue to the twist, Figure 2.5 shows how a wrench can be visualized. In

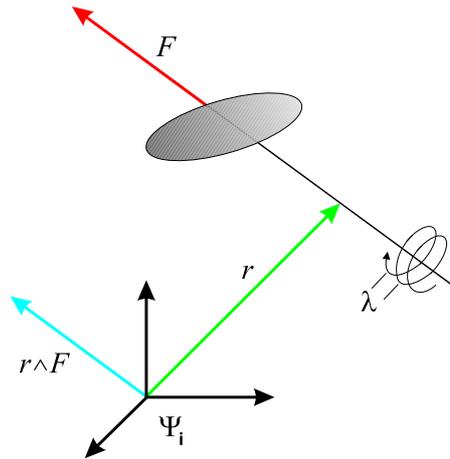


Figure 2.5: intuition of a wrench

Figure 2.5, the first element of equation 2.5 is representing a pure force applied along the line passing through r . The second is instead a pure momentum which does not need to be associated to a finite line, but to an infinite line which is again the polar of the line representing the pure force. (Stramigioli and Bruyninckx, 2001)

The concepts of twists and wrenches can be easily visualized in terms of so-called screw bond graphs. These are generalizations of ordinary bonds to six-dimensional bonds. A twist with zero pitch applied to a screw corresponds to a pure rotation. When a twist with infinite pitch is applied to a wrench, a pure translation results. This also holds for wrenches. A wrench applied to a screw with zero pitch corresponds to a pure momentum and a wrench with infinite pitch applied on a screw corresponds with a pure force.

Chapter 3

The Car

Before diving headfirst into the modeling steps and development trajectory, some key features of a car need to be mentioned. This chapter will specify basic attributes of a modern car and touch some choices that have been made in the design of the model. Using the characterization of these features and subsystems, the model can be iteratively enhanced, adding more subsystems and features in every step.

In order to study a car, it is necessary to specify one or more coordinate systems to measure the position of the vehicle. First of all there is an Earth-fixed coordinate system Ψ_0 . Within this analysis this frame will be considered inertial. Next, a body-fixed frame has to be assigned to the car body, Ψ_{car} . The z-axis is chosen in the forward driving direction. The y-axis is vertically upwards and the x-axis is chosen such that a right hand coordinate system is realized. Figure 3.1 shows this configuration. Each element (or group of elements) in the

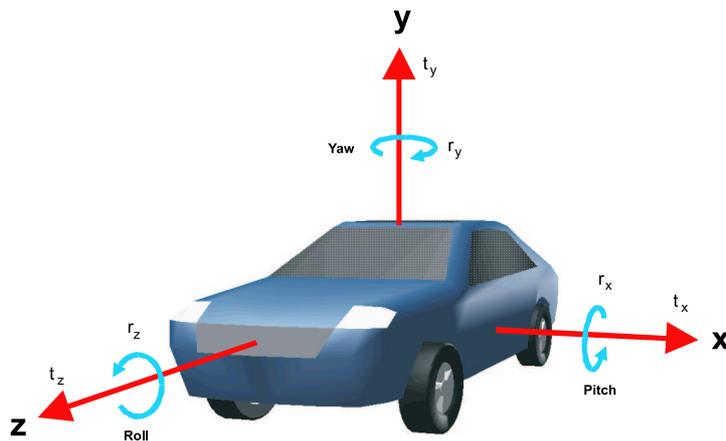


Figure 3.1: the different motions in the vehicle-fixed frame

model which has to be able to move with respect to the car body, will be placed

in a separate coordinate frame (e.g. the wheels, axles and bearings).

Besides translation along the axes of its frame, the car body can rotate around these three axes. Figure 3.1 also shows these motions and the terminology that will be used (Rahnejat, 1998), (Dixon, 1991).

With these six Degrees Of Freedom (DOF) all elements which constitute a screw motion are known.

The car can be divided in several subsystems, each representing an elementary function of a car. These are the wheels (and axes), the suspension and the steering. First of all we will have a closer look at the suspension.

3.1 The Suspension

Suspensions come in various types, depending on the desired behavior. A first selection is made when the desire for independent springs is uttered. This rules out all variations of torsion suspension, which damp both wheels on either side of an axle. When the requirement for separate suspensions is added, only single wheel suspensions remain. Figure 3.2 shows some variations of single

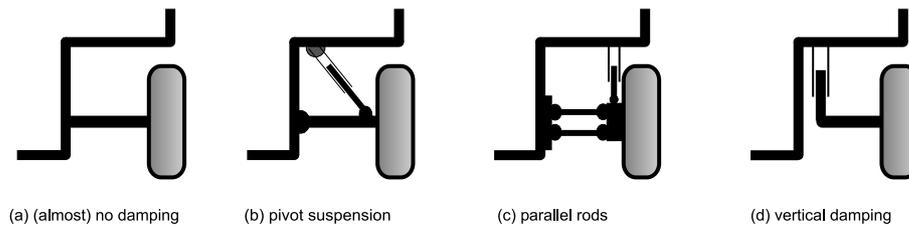


Figure 3.2: some (simple) suspension types

wheel suspensions. As can be seen, Figure 3.2 (a) only damps vertical motions slightly because of the finite stiffness of the axle. Figure 3.2 (b) damps the vertical motions using an axle connected to a hinge point. The wheel describes a circular rather than a vertical motion around this point. This configuration is not commonly used anymore. Item (c) of Figure 3.2 uses two parallel rods to enable an almost vertical motion for the wheel. Several wishbone suspension types are based on this concept. This is the most realistic variation in the figure. Finally, Figure 3.2 (d) shows a truly vertical motion, albeit not realistic in construction. However, the simulator does not care for realism very much (in this respect), and this representation does meet our need for vertical damping in an incomplex way. That is why the suspension in the final model resembles type (d).

Constraining a rigid body to certain motions only, can give rise to some difficulties, since some screw motions are coupled. For a rigid body, circular motions are mechanically coupled to translational motions. So when a rotation as well as a translation is desired, a third, undesired motion can occur. This problem can be solved by projecting the motion of the body onto the feasible directions and ignoring the forces in the constrained direction, using the *Lagrange-d'Alembert equations*. We can avert these relatively complex mathematics by assigning the motions to separate bodies, so that each body only

performs one particular movement. In the model, the latter solution has been applied by defining a bearing. This way, the axle can be constricted to sole rotations and the bearing to mere vertical translations.

It be noted that of course in real cars, the wheel is not only connected to the car body by an axle, but e.g. by an above mentioned wishbone-type suspension. A separate axle accounts for the rotation. But as we have chosen the simple suspension type (d) and we want to avoid the coupled motions, the bearing is needed. Figure 3.3 shows a view of the model in which this bearing is clearly visible. As can be seen in Figure 3.3, the axle of the car goes right through this

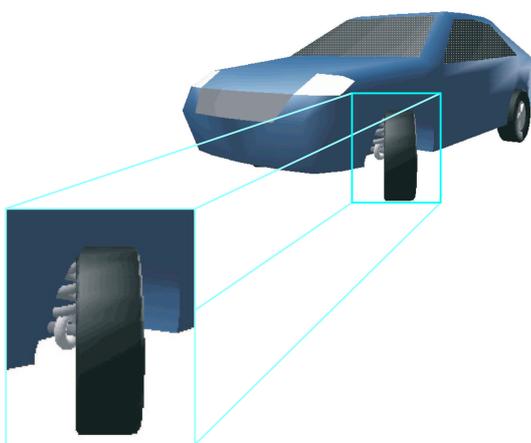


Figure 3.3: the vertical suspension and bearing on the car axle

bearing. The bearing is connected to the axle on a fixed position, and can only translate vertically. The axle can now rotate around the x-axis trouble-free.

3.2 The Wheel

The wheels play a significant role in the model, since they enable the contact to the surface. To provide the reader with an elementary insight in the processes involved in this contact, some fundamental behavioral aspects are reviewed.

In vehicle handling analysis, the tires are generally represented by empirical models; these are non-phenomenological, that is, the tire behavior is not derived from the material properties and structure of the tire. The material of a tire is a multi-layered, non-uniform, anisotropic, cord-rubber composite. So to understand tire behavior, there is a pressing need for simplification.

A number of tire models have been developed, e.g. the elastic foundation, the string and the beam model (Dixon, 1991). These models however, all assume a contact area (refer to Figure 3.4 (b)). These models provide for a surplus of simulation outputs (like pneumatic trail, wheel camber, wheel center line displacement, etc.), which are of little interest for an inquiry into the general behavior of a car. Hence a contact point rather than a contact area has been

implemented (Figure 3.4 (a)). At the contact point, forces are exerted from the

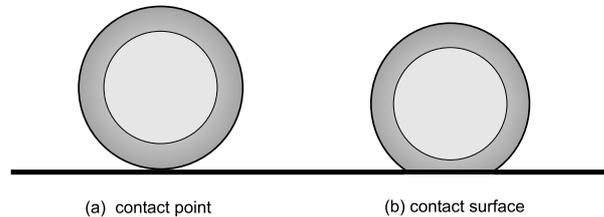


Figure 3.4: representations of wheel contact

wheel to the ground surface for vehicle support, guidance and maneuvers. These characteristics are all influenced by the operating parameters of a tire such as its inflation pressure, its rolling velocity and its size and shape (of Transportation, 1985). When implementing a contact model, these parameters have to be taken into account.

Surface parameters influence the driving behavior of the car to a great extent as well. Rough tarmac surfaces will feature more grip for the tires than e.g. wet asphalt or a dirt road. However, also in this respect the need for accuracy is not very urgent. We mainly want to distinguish between slippery and not-so-slippery surfaces. This can just as well be simulated by varying the friction parameters of the tire model. With these considerations a sufficient tire contact model can be constructed.

3.3 The Steering

The final basic feature of the car is the ability to steer. In almost all cars, this happens by rotating the front wheels around the vertical axis of the wheel. Setting both wheels to a certain equal angle, and thrusting the car forward, will make the car follow a curved course. This way the car can move freely on a plain, and one can maneuver the car to any desired location. However, true cars do not steer both wheels by an equal amount. Besides, steering does not actually happen by rotating the wheels around a vertical but rather around a slightly slanted steer axis. The following sections show how these attributes can be implemented for the car model to steer in a realistic fashion.

3.3.1 Ackermann Steering

When the car is steering in a low-speed situation, i.e. the side slip of the front wheels can be neglected, the trail the inner tires cover is not identical to the track of the outer tires. The inner tires follow a circle with a smaller radius than the outer tires. This means the front wheels have to be set to a different angle too, corresponding to the radii of their trajectories. Disregarding this property will eventuate in excessive tire scrub and poor steering behavior. Figure 3.5 displays how these dissimilar trails are to be interpreted.

The distinction between the inner and outer steering angle is known as the Ackermann steering concept, although it was actually invented by Langensperger (Dixon, 1991).

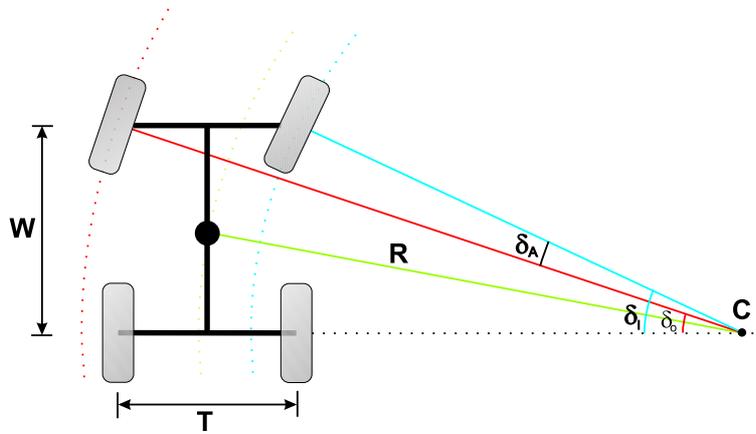


Figure 3.5: the Ackermann steering concept

In the figure, R depicts the radius of the curve measured from the car center, C is the center of rotation, T is the track width and W is the wheel base of the car. The following relations can be derived.

$$\delta_o = \tan^{-1} \left(\frac{W}{R + \frac{T}{2}} \right) \quad \delta_i = \tan^{-1} \left(\frac{W}{R - \frac{T}{2}} \right) \quad (3.1)$$

Where δ_o is the angle of the outer tire and δ_i the angle of the inner tire. The Ackermann angle δ_A is the difference between these angles.

$$\delta_A = \delta_i - \delta_o$$

Now, when the radius of the curve (R) is set to the desired value using the steer angle as in

$$R = \frac{WheelBase}{2 \cdot \sin^{-1}(SteerAngle)}$$

both the inner (δ_i) and outer (δ_o) angle can be determined. The radius of the curve has been determined from the center of the wheelbase ($WheelBase/2$).

3.3.2 Kingpin and Caster

These are not the names of some heroes in a morning daily-cartoon, but rather the two factors defining the inclination of the steering angle. In terms of the wheel coordinates, the Kingpin angle is the inclination around the z-axis of the wheel. The Caster angle defines the cant around the x-axis of the wheel. The initial orientation of the wheel frame is equal to the frame of the car. Figure 3.3.2 shows this more clearly. In a car, the Kingpin angle helps to give space to the brakes. Also, when the steering arms are forward (the steering rack in front of the wheel centers) it also gives room to angle the steering arms for Ackermann geometry. Moreover, it reduces the Kingpin offset. This offset works as a lever for the force exerted on the steering axle. The smaller the offset, the less force needed to steer the wheels.

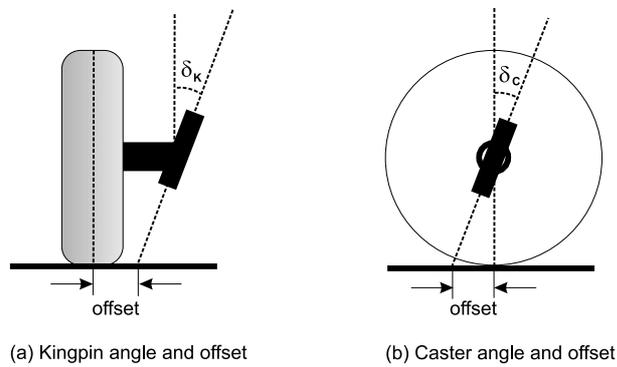


Figure 3.6: the distinction between the Kingpin and the Caster angle

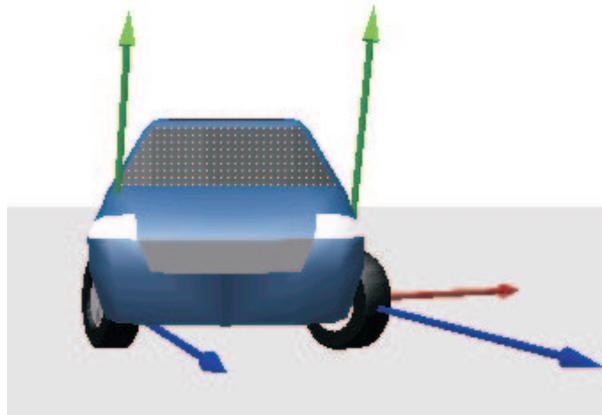


Figure 3.7: the effect of the Kingpin and Caster angle

The Caster offset, together with the Kingpin angle, affect the camber angle of the tire (the tilt of the tire around the z-axis) in a steering motion. Figure 3.7 shows a picture from the 3D model in which the effect of Kingpin and Caster angle are distinctly visible. To increase clarity, the coordinate frames of the front tires have been displayed as well. The vertical axes of the front wheels show different inclinations, due to the Ackermann steering concept (which is also visible by the different directions of the z-axes of the tires).

The inner tire which, due to the Ackermann geometry, has the greater steering angle, also exhibits more camber. Both these angles and offsets influence the handling of the car, especially in small-radius corners. In actual cars, such steering behavior is defined by a pair of ball joints, rather than an inclined steer axle. The resulting steering motion however remains similar.

When implementing these parameters in the model along with Ackermann steering, a realistic behavior of the steering of a car can be realized.

Chapter 4

The Model

Now we have arrived at the main chapter of this thesis. This chapter will elaborate on the modeling steps that were implemented to obtain a fully functional model of a car. The first endeavor is familiarizing with screws and constraints. This will be done using a wheel and axle configuration as described in Section 4.1. With this knowledge, iteration steps can be taken to gradually increase the complexity and veracity of the model. Each of the sections will cover partial enhancements until the desired functionality is reached.

4.1 Wheel and Axle

We commence the research with a simple wheel and axle construction. They are connected by a kinematic pair, and in the connection all motions but a rotation around the longitudinal axis of the axle are constrained. When an wrench is imposed on the first element of the screw bond in the kinematic connection (the rotation around the x-axis, as can be seen in equation 2.4), the wheel and the axle start rotating in opposite direction. This behavior is consistent with the physical law of conservation of momentum. Figure 4.1 shows the iconic representation of the implementation of this subsystem in 20-SIM.

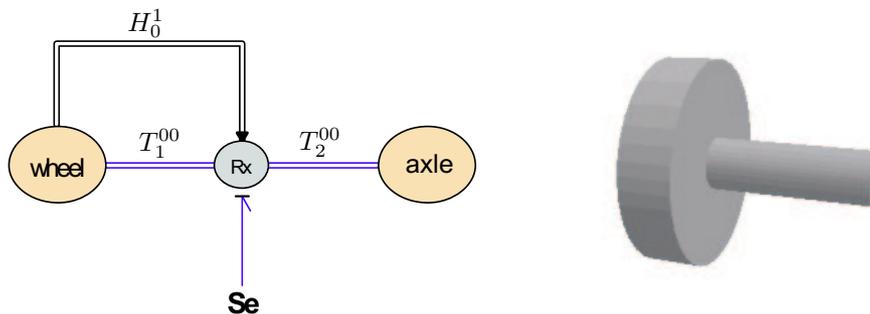


Figure 4.1: the iconic and 3D of a wheel and axle

The icon of the kinematic pair maintains the annotation definitions from Figure 3.1, so Rx indicates that this kinematic connection constrains all motions

but rotations around the x-axis. The right hand side of the figure shows a depiction of the 3-D representation of this bond graph model. The 3-D environment has been frequently employed for an intuitive verification of the model.

In this elemental subsystem, some essential submodels have been applied. Since they will turn up in several other regions of the model as well, they are briefly explicated in the following sections.

4.1.1 Rigid Bodies

The wheel and axis submodel have been based on the so-called rigid body model. This section will give a brief explanation of these submodels.

Since we are modeling in 3D, all objects in the model have to be able to move with 6 DOF. Hence, a model of a rigid body is needed that comprehends all the six elements that draw up the 3D screw motions. Moreover, dynamical aspects like acceleration of gravity, moments of inertia and coriolis forces have to be implemented.

We start by taking the momentum of a point mass m

$$p = mv$$

where p is the impulse or momentum and v is the velocity of the mass. This equation is known as *Eulers equation*. By taking the derivative of both sides, we obtain *Newtons second law of dynamics*, which is defined as

$$F = \dot{p} = m\dot{v} + \dot{m}v$$

where \dot{v} is the acceleration of the point mass. For most cases, where the mass has a constant value, the formula simplifies to

$$F = \dot{p} = m\dot{v}$$

Including the rotational domain as well, we take the momentum for the rotational inertia J defined by

$$p_r = J\omega$$

where p_r is the angular impulse or angular momentum and ω is the angular velocity. Again we take the derivative of both sides to get

$$\dot{p}_r = \dot{J}\omega + J\dot{\omega}$$

where $\dot{\omega}$ is the angular acceleration. \dot{J} is zero for most cases, so that we get the simplified form of

$$\dot{p}_r = J\dot{\omega}$$

Noting that \dot{p} is equal to the force F and \dot{p}_r is equal to the torque τ , we can generalize *Eulers equation for a rigid body* by

$$P = IT \tag{4.1}$$

where $P = [p_r^T p^T]^T$ is the generalized momentum and $T = [\omega^T v^T]^T$ is the twist.

I is a 6x6 inertia matrix. Finding or calculating this matrix can be simplified by using the *principal inertia frame*. For any rigid body, a special frame Ψ_k can be found, that is placed in the center of gravity, and oriented such, that

$$I^{k,i} = \begin{bmatrix} J_i & 0 \\ 0 & m_i I \end{bmatrix}$$

where

$$J_i = \begin{bmatrix} j_x & 0 & 0 \\ 0 & j_y & 0 \\ 0 & 0 & j_z \end{bmatrix}$$

I can then be found by

$$I^i = Ad_{H_i^k}^T I^{k,i} Ad_{H_i^k}$$

When we differentiate Eq. 4.1 on both sides we obtain the generalized formula of Newtons law which is defined in an inertial frame as

$$\dot{P} = W \quad (4.2)$$

where \dot{P} is the time derivative of the momentum of the rigid body, and $W^T = [\tau^T F^T]$ is the applied wrench. If we use the indices to indicate the frames, we get \dot{P} of body i relative to frame Ψ_0 defined in frame Ψ_0

$$\dot{P}_o^{i,0} = W_0^{0,i}$$

To be able to make use of the principle inertia, we transform the formula from the reference frame ψ_0 to the body frame ψ_i . After some manipulations, we arrive at the power balance for a rigid body:

$$\underbrace{\dot{P}^i T_i^{i,0}}_1 = P^i \underbrace{ad_{T_i^{i,0}} T_i^{i,0}}_2 + \underbrace{W^i T_i^{i,0}}_3 \quad (4.3)$$

In Eq. 4.3, the first term on the left (1) is the stored energy. The second term on the right (2) is the supplied energy. We see that an extra term has appeared (2), that always has a power value of zero, due to the structure of $ad_{T_i^{i,0}}$. Nevertheless, this term must not be omitted, and can be found directly in the used 20-SIM models. The extra term is a consequence of the position of our reference frame. In Eq. 4.2 the momentum is defined with respect to the reference frame of the observer. In Eq. 4.3 the reference frame is the body frame itself.

To model this in bonds graphs we get a model as depicted in Fig. 4.2. We can see that the model is a direct result of the above equation. The dimension of one multi-bond is 6. We would like to apply wrenches to the rigid body, that are defined in the reference frame of the rigid body. We need to apply a

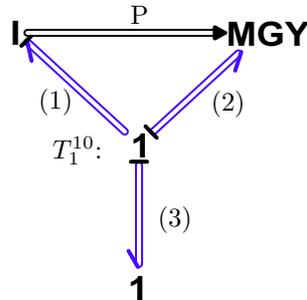


Figure 4.2: 3D Rigid Body

coordinate change in order to do this. We get

$$T_1^{1,0} = Ad_{H_0^1} T_1^{0,0}$$

where $Ad_{H_0^1}$ is the adjoint mapping from frame ψ_0 to frame ψ_1 . This transformation is realized by an MTF-element, with an extra modulation input signal H_0^1 that changes the transformation ratio. H_0^1 can be calculated according to

$$H_0^1(t) = \int_0^t \dot{H}_0^1(t) + H_0^1(0)$$

where

$$\dot{H}_0^1 = \tilde{T}_0^{1,1} H_0^1$$

and

$$H_0^1(0)$$

is the initial configuration.

To calculate the integral we must use an initial H-matrix (H_{init}), which can be chosen by looking at the starting configuration. The gravity force (G) is implemented in the same submodel causing an externally applied wrench, which is defined in the reference frame (Cs). We find τ , caused by the gravity force, by taking the cross-product of G and the displacement vector p , which is part of the homogeneous matrix we have calculated above. A complete model for a

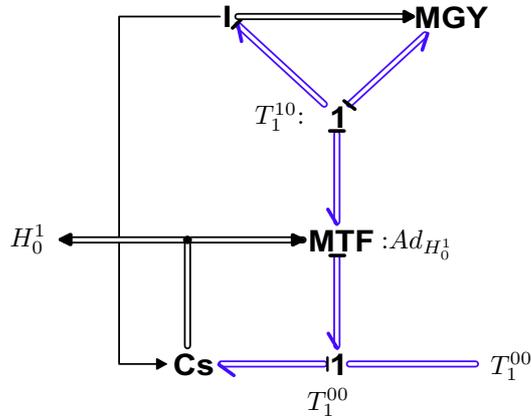


Figure 4.3: 3D Rigid Body

three dimensional rigid body that can be used for screw bond graph modelling is obtained. To define the screw motion, two coordinate frames are used. The reference frame Ψ_0 and the body frame Ψ_1 are located at the center of gravity. We write the twist as T_i^0 . This is the rigid body model as applied in the car model (Keukens, 2001).

4.1.2 Constraints

As mentioned in section 4.1, the wheel-axle system is set up with two rigid bodies, connected as kinematic pairs. Within this kinematic pair submodel, the

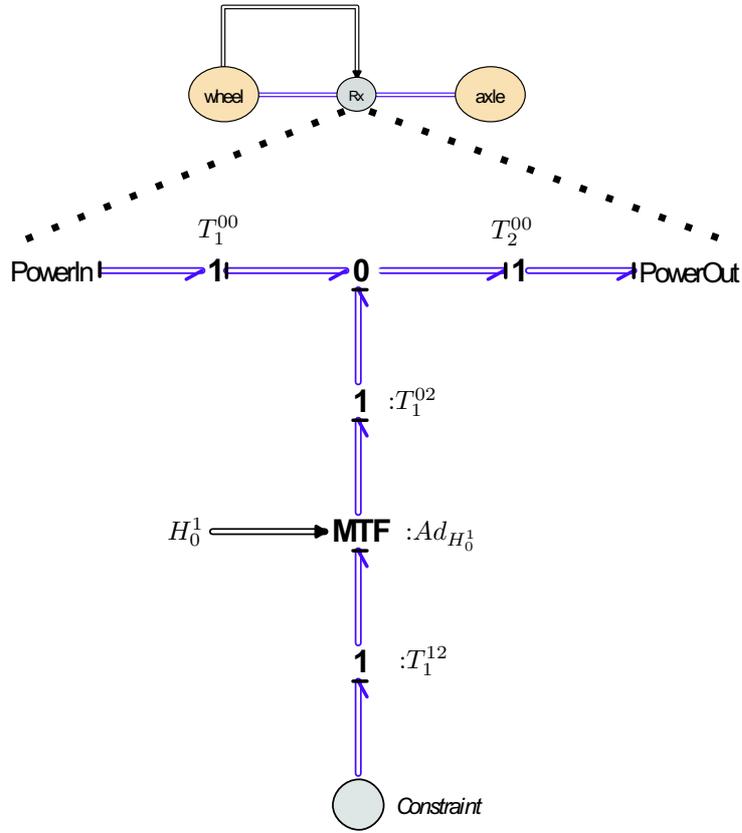


Figure 4.4: the insides of the kinematic pair connection

constraints on the motions are defined. This is done in the following manner. Figure 4.4 shows the transformations applied to the screw bond graphs. The bond graph model can be interpreted as follows. The top of the structure displays two 1-junctions, representing the twists of body one (the *wheel*) and body two (the *axle*), relative to and represented in the frame Ψ_0 (the Earth-fixed frame). The constraints concern motions in the frames of the bodies rather than in the Earth-fixed frame. So before we can effectively impose constraints on this system, we have to transform the screw bonds in such a way that the relative velocity of the bodies is represented in either one of their coordinate frames. As can be seen in the figure, the bonds are transformed so that body one is related to body two, and represented in the coordinate frame of body one (T_1^{12}). The twist (T_1^{02}) of body one, relative to body two is obtained simply by subtracting them using a 0-junction (the sum of all flows in a 0-junction is zero). Finally, using the H-matrix from body one to construct an Adjoint matrix ($Ad_{H_0^1}$) to change the frame of representation gives the desired result.

In this frame, the constraints can be enforced so that only rotations around the x-axis of the axle are allowed. This is done mathematically in the Constraint submodel by applying the *constraint-command* provided by 20-SIM. We refer to the Appendix for a detailed description.

With this simple analysis of a wheel and axle, an essential strategy has been drafted for the actual car model.

4.2 The Car Body

Finally we have arrived at the stadium where the real car model will be drawn up. The core of the car body is a rigid body like any other. It will only differ from rigid bodies for e.g. the wheels or bearings in mass and inertia, as can be seen in Chapter 5. The pith of the modeling are the connections and constraints with the other rigid bodies.

The car model will be built up stepwise, beginning with the wheels from the previous section. The car needs four of these subsystems. Each of these wheels-axle systems has to be connected to the car body with a suspension. Eventually, a torque has to be imposed on the wheels for the propulsion, and the steering has to be implemented.

4.2.1 The Suspension

The suspension requires a damped vertical motion as mentioned in section 3.1. This means that the kinematic pair of the wheel and axle have to be kinematically connected to the car body and constrained to pure vertical motions.

Figure 4.5 shows this construction for one wheel. The icon of a wheel and axle contains the subsystem of section 4.1. The restriction to vertical movements (T_y) is done in a similar fashion as the rotation of the wheel. It can be seen that the bearing discussed in section 3.1 comes into use here. The iconic depiction of

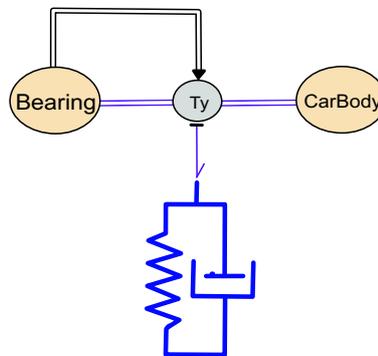


Figure 4.5: the bond graph model of the suspension

the suspension is an equation-submodel in 20-SIM, which integrates the velocity of the vertical movement to a relative position x . Subsequently, this position is used to return a force in the opposite direction of the velocity in the manner of equation 4.4. The concept of mechanical play has been applied to limit the suspension behavior to a region of 30 centimeters. This way the mechanical restriction to the amplitude of the damping motion is defined. Outside this zone, the stiffness and damping continuously increase to extreme values, to model the mechanical stop. Effectively, the damping motion will be restricted to approximately 30 cm.

$$\begin{aligned}
F = \frac{1}{2} \cdot k2 (x + \sqrt{(x - s/2)^2 + (ep \cdot s)^2} + x - \dots \\
\dots - \sqrt{(-x - s/2)^2 + (ep \cdot s)^2}) + k1 \cdot x + d \cdot v
\end{aligned} \tag{4.4}$$

Where

F = Force

$k1$ = stiffness in the nominal region

$k2$ = stiffness outside the nominal region

s = interval of the play i.e. the suspension region

d = damping, equal to $d1$ within the nominal region and $d2$ outside the region

ep = "sharpness" of the transition region

v = the vertical velocity

For a more detailed description of the suspension, refer to the Appendix.

4.2.2 Four Wheels and a Car Body

Little exertion is needed to extend the model to a complete car body with four wheels and suspensions. Figure 4.6 shows the schematic depiction of the car and the wheels, where the car body and wheels are expanded to display their contents. Although one is tempted to read the tiny submodels and signals, this figure only serves the purpose of showing the general structure of the model. The arrows show the direction of development from the center rigid body of the car to the wheels. The dotted outlines indicate where the car body part of the iconic diagram on top of the figure is to be found in the bond graph model.

As can be seen, the front wheels differ from the original wheel axle system of section 4.1. This is necessary to enable steering motions in the front wheels. The following section will discuss how the steering has been realized.

4.3 The Steering

Analog to separating rotational and translational movements to prevent interference, a new rigid body has to be defined which is confined to steering motions (rotations around the vertical axis). This body can be recognized in 4.6 as the **Rotator**. A kinematic connection (**Ry**) restricts its motions. As we know now, this rotation does not occur around a pure vertical axis, but rather a slanted steering axis. This means that within **Ry**, apart from the regular transformation in a kinematic connection, an additional transformation has to take place. This way, the constraints are imposed on a frame with a Kingpin and Caster inclination relative to the original frame.

An additional TF element has been implemented in the standard kinematic connection **Ry** between the **Axis** and **Rotator**, tilting the body fixed frame by a Kingpin and Caster angle (4.5).

$$T_{KC}^{KC,2} = Ad_{H_1^{KC}} \cdot T_1^{1,2} \quad \text{with} \quad Ad_{H_1^{KC}} = Adjoint(H_1^{KC})$$

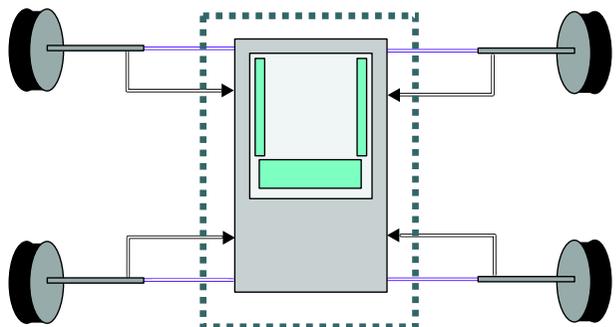
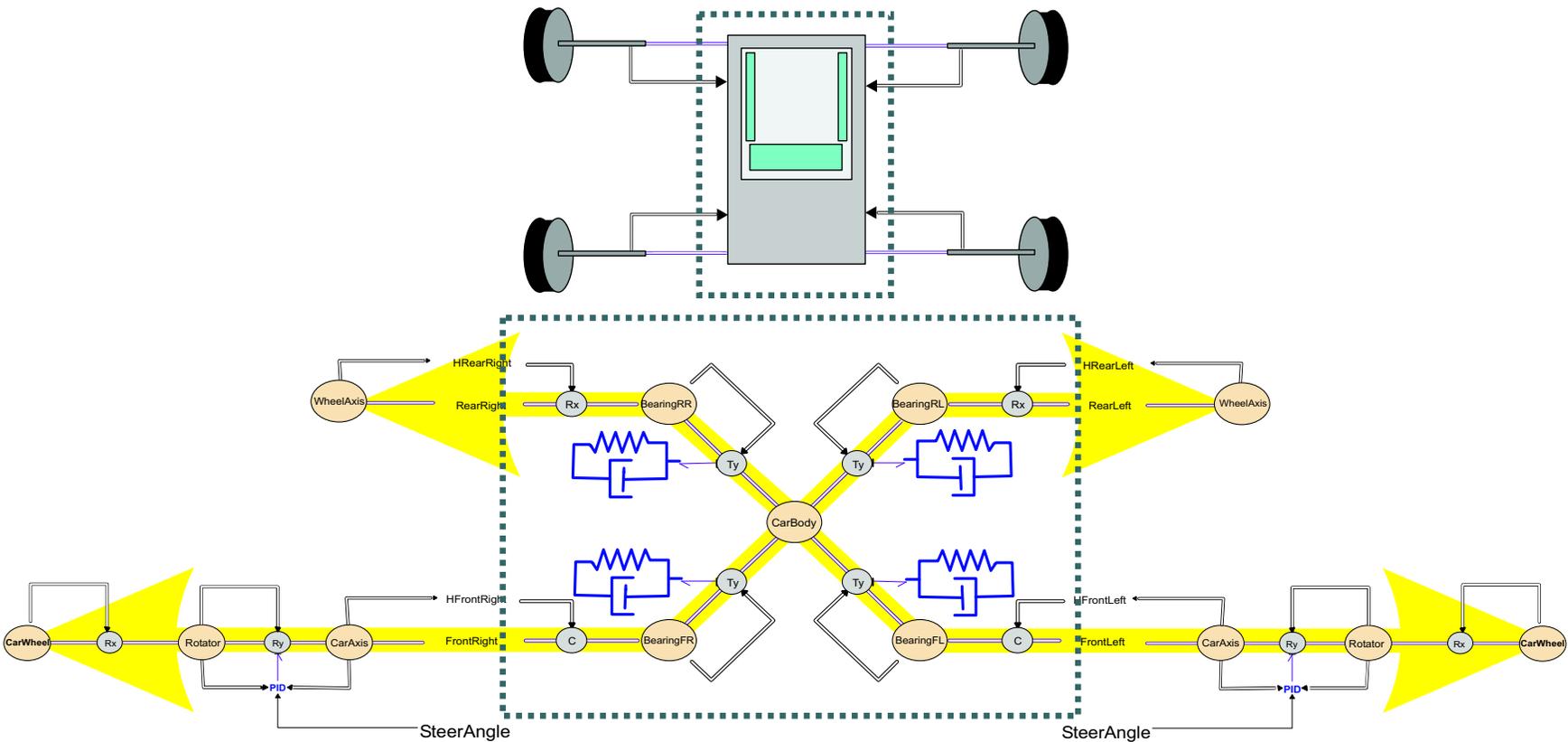


Figure 4.6: the complete car body bond graph

$$H_1^{KC} = H_K \cdot (H_C)^T = \begin{bmatrix} \cos K & -\sin K & 0 \\ \sin K & \cos K & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos C & -\sin C \\ 0 & \sin C & \cos C \end{bmatrix}^T \quad (4.5)$$

With K the Kingpin angle and C the Caster angle.

A steering angle is set by implying a force on the kinematic connection between the axle rigid body and the rotator body. To be able to set the steer angle to a desired value, a PID controller has been implemented. This controller feeds a force to the kinematic connection in proportion to the difference between the desired angle and the actual angle. The rigid bodies only contain information of the position and orientation of the bodies relative to the Earth-fixed frame. Hence, a relative H-matrix has to be constructed, relating the position orientation of the Rotator to the Axis or vice versa. This can be realized with the following straightforward operation.

$$H_{relative} = (H_{Rotator})^T \cdot H_{Axle}$$

Figure 4.7 shows how the controller has been implemented. Chapter 5 reports on how the control parameters have been determined. The front wheels have

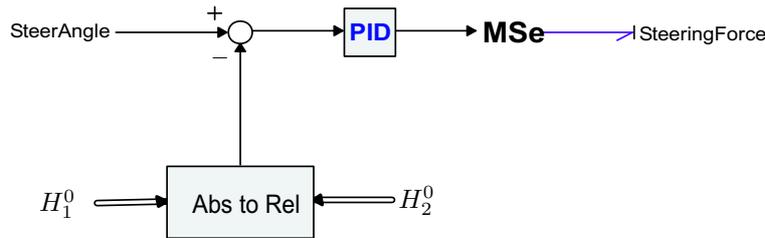


Figure 4.7: the PID controller implemented for steering

separate controllers, their setpoints following the Ackermann algorithm. A special submodel has been created which converts a steering angle to a separate Ackermann angle for each wheel. For a more detailed description on the steering as implemented in 20-SIM, refer to the Appendix.

4.4 The Driving

The model has been designed to allow both front, rear and four wheel driving. A signal generator provides for the speed the wheels have to follow. A PID controller ensures that this speed trajectory is realized, within a small error tolerance.

We are applying a torque on each wheel separately. In a real car, the torque is distributed to the wheels using a drive shaft and differential. The differential is needed because one engine has to drive two or even four wheels which do not necessarily spin at the same speed. Without a differential, the driven wheels would be locked together, forcing them to rotate at the same speeds. The discussion about the Ackermann algorithm (section 3.5) has clarified that in turning situations, each wheel travels its own course at its own speed, directly related to the steering angle.

However, in 20-SIM we are not limited to the use of one 'engine' (i.e. Effort Source). We can simply implement several sources. Moreover, the connection between the sources and the wheels is not locked like in actual cars. Applying an equal torque on e.g. both rear wheels does not confine the wheels to equal rotational speeds. The sources are ideal, which means that whatever speed is accomplished, the effort supplied to the wheels remains unaltered. This rules out the need for implementation of a differential. In the model it has been chosen to supply the rear wheels with two effort sources. When the need arises for front wheel driving, a similar configuration of controlled effort sources needs to be implemented using the Ackermann algorithm once more to reckon with the difference in track of the inner and outer wheel. Again, for a more detailed description of the implementation of the driving of the car, we refer to the Appendix.

4.5 The Tires

The most critical part of the model is the behavior of the tires. 20-SIM has to be programmed to distinguish between contact and non-contact situations and accompanying distinct behavior. A rigid body as defined in section 4.1.1 only needs mass and inertia (in 3 directions) for parameters. For its spatial behavior, the geometry of the object is insignificant (provided it moves in vacuum). However, since we want to implement a form of collision detection accompanied with switching behavior, the shape of the colliding objects are crucial. Section 4.5.1 examines the implementation of the contact behavior of the car. Wheel slip is another basic feature of a car. Section 4.5.2 gives an explanation of the realization of this behavior in the contact submodel.

4.5.1 Contact Behavior

Consider a rigid body with a smooth, oriented surface \mathcal{S} , embedded in the Euclidean space \mathcal{E} . To this body we rigidly attach a coordinate frame Ψ_O . In the frame Ψ_O , we can describe the surface (locally) as a mapping $f : \mathcal{D} \subset \mathbb{R}^2 \rightarrow \mathcal{S}$, which assigns to each pair of coordinates $(u, v) \in \mathcal{D} \subset \mathbb{R}^2$ a point of the surface. The mapping f is a (local) parameterization of the surface, and we assume this parameterization to be well-defined, i.e. the partial derivatives $f_u := \frac{\partial f}{\partial u}$ and $f_v := \frac{\partial f}{\partial v}$ are continuous and independent at all points, such that f_u and f_v span the tangent plane to the surface at all points.

Without loss of generality, we can assume that the surface is parameterized such that $f_u \times f_v$ points in the direction of the outward normal to the surface (if this is not the case, just switch the parameters u and v). Then, at each point p of the surface, the unit normal vector is given by

$$n(p) = \frac{f_u(p) \times f_v(p)}{|f_u(p) \times f_v(p)|}$$

where $|\cdot|$ denotes the Euclidean norm.

We define the *Gauss mapping* $g : \mathcal{S} \rightarrow \mathbb{S}^2$ as the mapping which takes a point p on the surface and returns a point $g(p)$ on the rigid body, corresponding to the unit normal of p (Duindam and Stramigioli, 2002a).

Now consider two objects with surfaces S_o and S_f which are touching at a contact point η . The motion that needs to be found is the motion of the points of contact across the surfaces of the objects in response to a relative motion of the objects. The motion of the contact coordinates, $\dot{\eta}$, as a function of the relative motion is given by

$$\begin{aligned}\dot{\alpha}_f &= M_f^{-1}(K_f + \tilde{K}_o)^{-1} \left(\begin{bmatrix} -\omega_y \\ \omega_x \end{bmatrix} - \tilde{K}_o \begin{bmatrix} v_x \\ v_y \end{bmatrix} \right) \\ \dot{\alpha}_o &= M_o^{-1}R_\psi(K_f + \tilde{K}_o)^{-1} \left(\begin{bmatrix} -\omega_y \\ \omega_x \end{bmatrix} + \tilde{K}_f \begin{bmatrix} v_x \\ v_y \end{bmatrix} \right) \\ \dot{\psi} &= \omega_z + T_f M_f \dot{\alpha}_f + T_o M_o \dot{\alpha}_o \\ 0 &= v_z\end{aligned}\quad (4.6)$$

where v_x, v_y, ω_x and ω_y are the local sliding and rolling velocities along the tangent plane at the point of contact. v_z is the linear velocity in the direction of the contact normal and ω_z is the rotational velocity about the contact normal. R_ψ is the orientation of the x- and y-axes of contact frame C_f of S_f relative to the x- and y-axes of contact frame C_o of frame S_o . K, M and T are the curvature, torsion and metric tensors respectively. For the surface, these tensors describe a plain. The wheels are described as spheres. This way, in normal driving conditions, natural behavior is achieved. However, for the sake of accuracy it must be recommended that a cylindrical description be implemented. For a more detailed description of contact kinematics, refer to (Murray et al., 1994). The contact parameters used in the model can be found in the Appendix.

These mathematics have been applied to find the contact point on the wheel and the surface. In non-contact situations, the points on the wheel and surface where the distance between both is smallest is determined. Both points are provided with a frame with the vertical axis directed toward each other. When the origins of these frames intersect, contact has been established.

Next, a 4-dimensional buffer is connected between these contact points: $C_{buffer} = [C_{T_y}, C_{t_x}, C_{t_z}, C_{t_y}]$. The first element refers to the stiffness of the wheel around the steering axis, the second to the stiffness of the wheel in the forward driving direction. The third element refers to the stiffness of the wheel perpendicular to the driving motion and the fourth element is the stiffness of the wheel in vertical translations (i.e. the air pressure the tires). Thus, a velocity of the wheel in one of these directions (see Eq. 4.7), returns a force reversely oriented. The other elements of screw motion are free. This force propels the car to the desired direction, and keeps it on track ((Visser, 2002)). However, extreme situations can be imagined where the wheels do not keep the car on track, or the car is not propelled in the desired direction anymore (breaking out of a curve or skidding in acceleration). The next section describes how wheel slip has been implemented.

4.5.2 Wheel Slip

Wheel slip is a hard to describe feature of car tires. When little or no torque is applied to the rear wheels, the rubber sticks to the surface and almost all supplied power will generate car movement. However, when much force is applied on the tires, the wheels can start skidding and the better part of the power will

be dissipated in frictional heat. When slipping, the tire can regain its grip only when the speed difference between the tire surface and the ground is (almost) equal to zero.

So the skidding behavior depends on the force on the contact point and the velocity of the contact point on the tire surface (relative to the speed on the plain). Moreover, when in slip in the forward direction, the grip in the perpendicular direction is lost as well. Steering the car in a forward slipping condition should result in a skidding motion in the perpendicular direction as well (consider show burnouts of racing cars where they slip in circles).

In the model, this has been accomplished by implementing a 'Coulomb'-parameter, which determines the force at which slip occurs. Also, when skidding, this parameter determines the coulomb friction force. Hence this parameter determines the forward propulsion in slipping conditions.

To implement this in 20-SIM, the horizontal movements are evaluated together for slip conditions in the following manner (using 'pseudo-code').

```

 $F_{x,z} = \sqrt{F_x^2 + F_z^2};$ 
if ( $F_{x,z} > \text{maximum}$ ) then
     $slip_{x,z} = \text{true};$ 
else
     $slip_{x,z} = \text{false};$ 
end;
```

In the situation of $slip_{x,z} = \text{true}$ the wheel slides in both the driving direction and perpendicular to the driving direction. Regaining grip after skidding is done in a similar manner.

```

 $v_{x,z} = \sqrt{v_x^2 + v_z^2};$ 
if ( $F_{x,z} < \text{threshold}$  and  $slip_{x,z} = \text{true}$ ) then
     $slip_{x,z} = \text{false};$ 
end;
```

Of course, these are not the only lines of code implemented in 20-SIM, however they represent the tenor of the instructions that 20-SIM has to obey. All other instructions serve the purpose of energy consistency and behavioral aspects of the tire movement (Visser, 2002).

Chapter 5

Parameters and Verification

This chapter will discuss how some essential parameters have been determined such as car mass and inertia, wheel mass, inertia and stiffness. Section 5.2 gives an account on an experiment displaying genuine car-like behavior. This will serve as an empirical verification of the model.

5.1 Parameters

In order to make a realistic model, parameters have to be found which resemble those of real cars. First the mass and inertia of the car body and wheel are determined.

The car body mass can be set quite intuitively. A regular car weighs around one thousand kilos. Subtracting around 200 kg. for the wheels, axes, steering construction and other peripheral elements, results in a mass of the sole car body of 800 kilos. For a rough *estimate* of the inertia in 3 directions we consider the car as a solid block (the mass evenly distributed through the body). If we furthermore suppose the height equal to the width of the car, the inertia can be evaluated as follows. We are not striving for exactness in this respect, but we are merely interested in an approximation of the dimensions of the inertia.

$$J = m \frac{(h^2 + l^2)}{12} \quad [kg/m^2]$$

With m the mass of the car. The results of these considerations can be found in 5.1.

Similarly an estimate of the mass and inertia of the wheels has to be found. The mass of one wheel has been estimated to be 18 kg. When assuming the tire a solid disk, the inertia can be calculated using

$$J = \frac{1}{2} m r^2 \quad [kg/m^2]$$

with r the radius of the disk (for the rotation around the x-axis). The inertia in the other two directions can be estimated by applying the equation for a solid block once again, and reducing the outcome slightly. Similar calculations for the axles and bearings can be found in table 5.1.

$$\begin{aligned}
J_{axle} &= [0.0005, 0.33, 0.33] \text{ (kg/m}^2\text{)} \\
J_{bearing} &= [0.001, 0.0008, 0.0008] \text{ (kg/m}^2\text{)} \\
J_{wheel} &= [0.8, 0.5, 0.5] \text{ (kg/m}^2\text{)} \\
J_{carbody} &= [1500, 1500, 300] \text{ (kg/m}^2\text{)}
\end{aligned}$$

Table 5.1: the inertia parameters found for the model.

$$\begin{aligned}
k_{tire} &= 300\,000 \\
k_{suspension} &= 12\,000 \\
d_{suspension} &= 4\,000
\end{aligned}$$

Table 5.2: stiffness and damping applied in the model.

Next, the damping and stiffness of the suspension and the pneumatic tires have to be determined. The stiffness and damping for a regular passenger car are somewhere in the region of: $k_{tire} = 230000 \text{ (N/m)}$, $k_{suspension} = 32500 \text{ (N/m)}$, $d_{suspension} = 7500 \text{ (Ns/m)}$. Trying these parameters on the model and testing some variations until a quite natural behavior was achieved resulted in the values in Table 5.1. Finally, the conditions for slip have to be estimated. As can be seen in 4.5.2, the condition for slip is determined using the *Coulomb* variable. This value has to be set to a realistic value (for the lateral movements) to enable real life driving behavior. The Coulomb variable (as well as the stiffness) for steering motions (r_y) has been set to zero for convenience. In a real car they are non zero but still insignificantly small for general behavior studies.

5.2 Verification

For an intuitive verification of the model we will discuss an experiment in which several features of the car show life-like behavior. In the experiment the coulomb friction has been set to 1550 [N].

The car is thrust forward to such an extent that the rear wheels start spinning. First the car is steered left and after two seconds it is steered right. Next, the front wheels return to a non-steering position, and the car is decelerated. In the experiment some interesting behavioral aspects are visible. They will be referred to as *events*. The following events can be recognized. The following figures display the parameters where these events are clearly visible. Figure 5.1 shows the Ackermann steering of the front wheels. It can be seen that the inner wheel displays a slightly sharper angle than the outer wheel. The angle of the

- I The rear wheels start slipping during acceleration
- II The rear wheels regain grip
- III The car steers left
- IV The car steers right
- V The car stops steering
- VI The rear wheels start skidding during deceleration

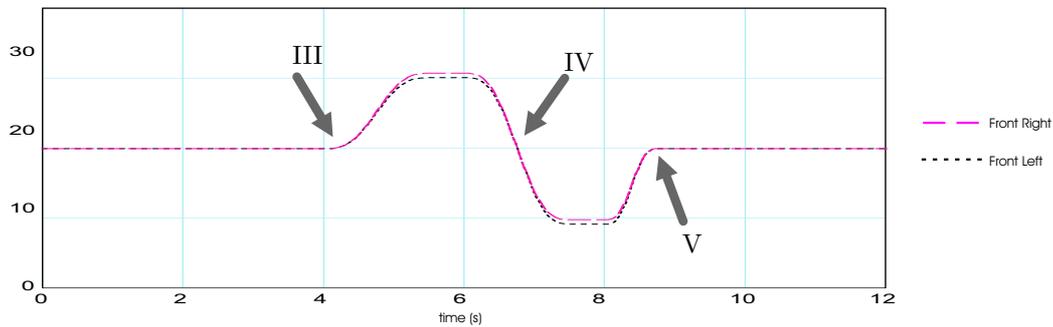


Figure 5.1: the Ackermann algorithm in effect

wheels (the vertical axis of the graph) can vary between 0 and 1. Zero angle means no steering (the radius of the curve is infinite). An angle of 1 means that the radius of the curve is zero. Both wheels are then turned toward the center of rotation (i.e. the inner wheel is rotated slightly more than 90 degrees and the outer slightly less). In practice the angle is never set to more than 0.5.

Figure 5.2 shows the speed of both rear and both front wheels. As can be seen in the figure, the plot of the rear wheel behavior does not feature all events. The graph of the front wheel behavior however exhibits all six events.

It can be seen that the Ackermann algorithm has been applied to the rear wheel speeds as well. The outer wheel is accelerated in steering situations whereas the inner wheel is decelerated slightly. It must be noted that only an approximation of Ackermann has been implemented for simplicity of the model. The curve of the front wheels displays all six events. The slipping of the rear wheels (event I) can be recognized by the front wheels not following the same speed trajectory of the rear wheels. The speed of the wheels is measured by regarding the rotational velocity of the wheels. Hence difference in speed trajectory implies that the rear wheels rotate faster than the front wheels. Since the front wheels are not driven, this means that the rear wheels must be spinning. The same goes for the skidding during deceleration (event VI).

Event II, the regaining of grip of the rear wheels, can be seen in the curve of the front wheels as well. At this point they reach the same speed as the rear wheels. The steering motions (event III and IV) are visible in a general slowdown. This is due to the fact that the Ackermann algorithm on the rear wheels is only an approximation. The moment the steering stops (event V) the front wheels rotate again at the same speed as before the steering.

Finally the suspensions of both rear wheels are regarded. In Figure 5.3 all six events can be recognized albeit slightly indistinct because of surrounding 'noise'. This noise is the result of the correctional activity of the driving PID. To keep the speed at the desired level, the controller corrects small speed errors very firmly, accelerating or decelerating the car up to the desired speed. The suspensions increase or decrease as a reaction on these corrections. In the recommendations (section 6.2) some remarks can be found on how to improve this behavior.

The graph shows that when the slip sets in (event I), the suspension length slightly reduces. This is because the forward thrust is less in slipping condition. The regaining of grip can be recognized as a slight stretch of the dampers. Since

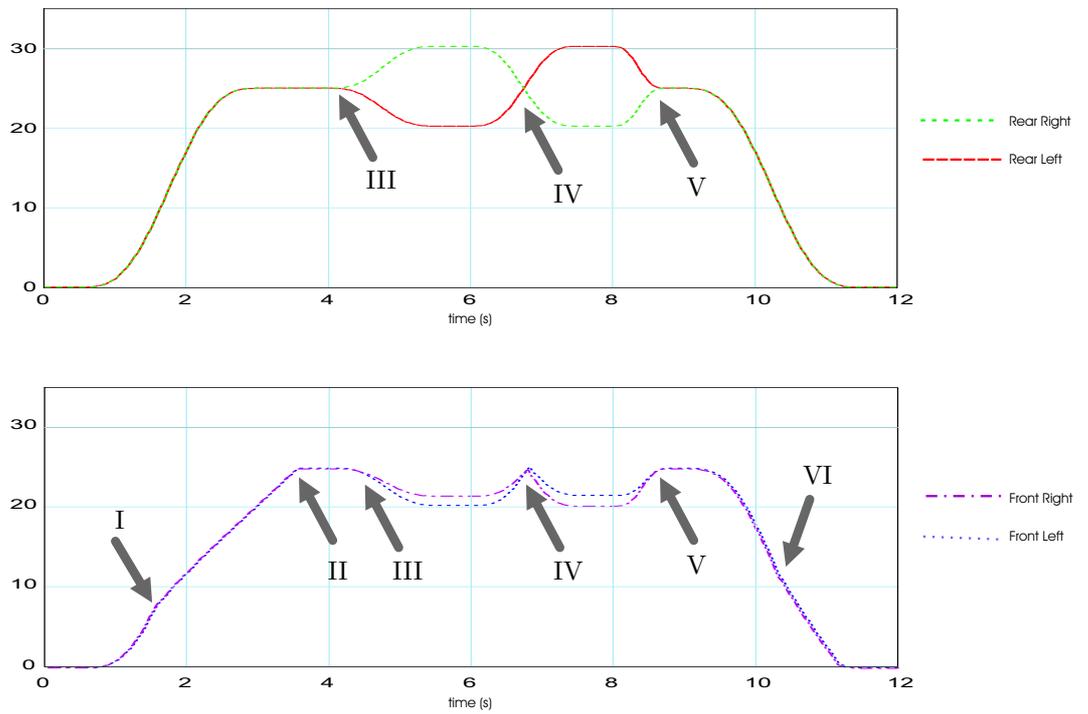


Figure 5.2: the velocity of the rear wheels and front wheels

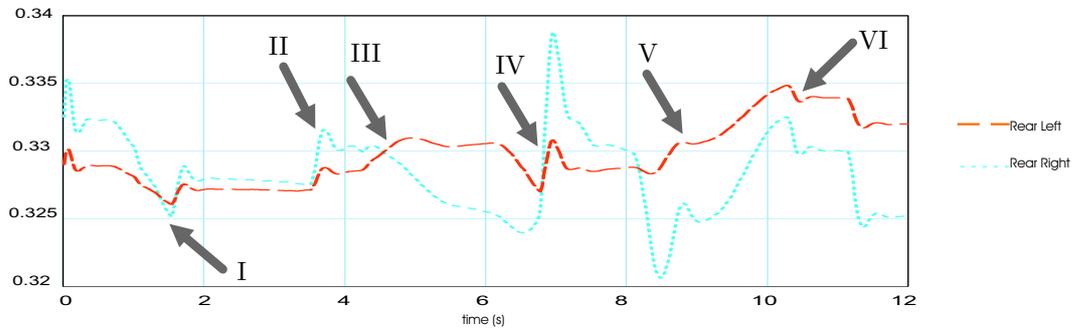


Figure 5.3: the length of the rear left and rear right suspensions

the desired speed is achieved, the forward acceleration is nullified resulting in a slight increase of damper length. The steering motions (event III, IV and V) can be recognized as respectively the stretch of the left suspension (the right suspension length decreases), the stretch of the right suspension (decrease of the length of the left suspension) and the regaining of approximately the initial length of the suspensions. The skidding during deceleration (event VI) can be recognized as a slight decrease in length of both suspensions.

It can be seen in the figure that the suspensions do not exactly regain their initial lengths at the end of the graph. This is due to the relatively large

integration stepsize used to obtain these graphs. Chapter 6 gives an account on the necessity for this stepsize and how to change the model to prevent this.

Chapter 6

Conclusions and Recommendations

We have sought to formulate a multi-body physical description of car behavior using screw bond graphs. The model has been constructed incrementally, adding features step by step. This has been done using 20-SIM, an energy based modeling tool. This chapter contains some concluding remarks as well as recommendations for further research.

6.1 Conclusions

- The goal of realizing an infrastructure for measurements in the field of X-by-wire and independent wheel steering has been accomplished. By applying some simplifications to elements as the wheel contact and slip behavior or the suspension of the car, a car model has been developed which behaves like a genuine car. The model can easily be configured to specific user needs because of the implementation of a 'Global' submodel, containing all significant parameters.
- It can be concluded that using bond graphs to model dynamical systems, results in a model that is physically correct, as long as we use the elements the way they are meant to be used. When a model is physically correct, it means that the energy flow between the elements is not altered by changing one of the two dual power variables (except for the dissipative elements and the sources). If no dissipative elements are used, and we give the car some initial energy, the total sum of the energy remains constant.
- The use of screw bond graphs has proven that we can combine torques, forces, velocities and angular velocities without problems. Screw theory and bond graphs combine very well.
- 20-SIM has proven to be a very powerful tool for modeling multi-body dynamical systems. The use of graphical submodels greatly improves the clarity of the model. Moreover the 3-D environment provides an excellent means for verifying simulation results.

6.2 Recommendations

The model provides for a fully functional car infrastructure, albeit it can use some improvement in certain areas. The following items discuss some aspect that are suitable for further research.

- First of all, the PID controllers realizing the steering angle and the driving torque need improvement. At present, the correctional actions of both controllers influence each other. This results in continuous corrections, alternating between the steering force and the driving torque, slowing the simulation down to a great extent. Moreover, these correctional actions sometimes disable proper simulation because of the sharp transitions they entail. An error message like

```
"The integration was halted after failing  
to achieve corrector convergence even  
after reducing the stepsize by a factor of  
1e10 from its initial value"
```

might occur. The steering PID might be maintained, but the propulsion of the car needs some more thought. At first, the model has been designed for torque drive. When a fixed speed became desirable, the PID has been implemented, to realize velocity control. It might pay off to implement rear wheel driving using two *flow sources* since the controller can then be refrained from. However, the need for a differential of some sort becomes essential, since now the rotational speed of the wheels is fixed.

- For now a rudimentary Ackermann algorithm has been implemented to realize differential behavior. This algorithm needs some looking into since the front wheels display slowdown in steering situations. Of course one can also choose to use front wheel driving since the Ackermann concept is already implemented there.
- For more thorough research of car behavior a means can be found for parameterizing the position of the center of gravity of the car body. Now it is located simply in the center of the car body. Since most of the weight of the car is located in the engine and the body framework the center of gravity should be positioned lower and more in front of the car body. At present, taking curves results often in the car tipping over at speeds where this should not yet occur. Shifting the car center of gravity to a more realistic place would result in better curving behavior. More pressure on the front wheel also enables more realistic slipping behavior. In a curve, the rear wheels should start slipping before the front wheels do, since less weight rests on them.
- As mentioned in the first item, the model requires quite some simulation time. Apart from reshaping the torque supply of the engine part of the model, one could research other ways of implementing the behavior of the steering and suspension. This might lead to reduction of the number of rigid bodies or kinematic connections in the model, hence reducing the model calculations significantly. This reduction could be established by grouping several constraints in one kinematic connection, using the *Lagrange-d'Alembert equations* to uncouple the motions.

- The simulation step of the Modified Backward Differentiation algorithm is a critical parameter. Especially in extreme situations such as the severe corrections of the driving PID or in transitions between slip and non-slip. The simulator sometimes halts and returns the message

```
"After some initial success, the
integration was halted either by repeated
error test failures or by a test on
(relative tolerance)/(absolute tolerance).
Too much accuracy has been requested."
```

However, the simulation step can not be increased too much since that will result in significant numerical errors (refer to the shift in suspension length in the experiment of section 5.2).

Appendix A

Tips and Details

This appendix is aimed at describing the essential model parts and parameters to enable further research. This will include explanation of some design choices which at first hand might seem awkward. Furthermore some basic knowledge and experience concerning modeling with screw bond graphs will be shared.

A.1 Practical Tips

Working with 20-SIM requires some getting used to. This section aims at averting errors and exceptions that have occurred during the modeling process in further research.

The submodels have mainly been connected using iconic diagram bonds (no arrows, just a bond indication energy exchange). However, this can give rise to some confusion, since the direction and orientation of the energy flow is not clear-cut. The function "View / Show Terminals" shows the direction of the energy flow. This was especially crucial in the connection of the suspensions and the wheel contact submodel. Erroneous connection of the bonds resulted in inverted damping behavior (a force exerted on the suspension moved it in the the opposite direction of the force) and the car falling through the plain (the force that was meant to keep the car on the road (pushing it upward), flung the car downward through the surface).

Moreover, quite some time was lost with esthetical alterations resulting in bizarre model jams. These are errors that might be remedied in newer versions of 20-SIM, but are reported anyway, to be on the safe side. Changing the name of rigid body submodels is impossible. The model does not simulate anymore. This is the reason that the axle in the model is still called axis. Also, rigid bodies sometimes refuse more than one power connection (which normally is possible). Deleting this submodel and inserting a fresh one is the quickest solution. Lastly, beware of changing rigid body parameters (mass, inertia) after using the "Copy States" function of the simulator. The simulation might not start anymore. Resetting all initial values to their original values after changing the parameters of the rigid body forestalls this error.

A.2 The Model Close Up

As mentioned in the report, some important elements of the car will be discussed in detail to ease the break in. The elements that will be discussed are respectively the Global submodel, the suspensions, the Ackermann algorithm, the steering and driving controllers and the contact parameters.

A.2.1 the Global submodel

Since we are dealing with a car with four wheels, a lot of parameters need to be defined and initialized four times. To prevent defining these parameters and values manually (very error-prone), global variables have been used. They are defined in the Global submodel along with a *functional description*. They can be declared in the concerning submodels. This way parameters only have to be set once in the simulator.

A.2.2 The Suspensions

As mentioned, the suspensions are based on the submodel for mechanical play. Only in this context, the low friction and damping parameters inside the play area are set to the normal suspension parameters. The parameters outside the play area are set extremely high, to restrict the damping motion to the play area only (which is 30 cm.). The transition between these regions can be smoothed by the RoundOff parameter. The smaller this parameter, the less severe the transition kicks in. This is important because the simulation method (Modified Backward Differentiation) adapts its stepsize to the smoothness of the signals. Best is not to choose this parameter too high to prevent high simulation times.

A.2.3 The Steering and Driving Controllers

Both the steering and the driving of the car has been implemented by using PID controllers. The parameters have been determined empirically, although they are still far from optimal. This is because corrections of the driving controller affect the steering angle slightly, which then needs correction and vice versa. This elongates the simulation time (because of the sharp transitions of the correctional actions). For the front wheels a controller is necessary since only forces or velocities can be fed to the system directly. However, as the conclusional remarks already state, it might be worthwhile to prescribe a velocity to the rear wheels instead of a force. It be noted that one has to implement an Ackermann algorithm (i.e. the differential) for the rear wheel velocities as well.

If the desire for front wheel driving emerges, one has to supply the front wheel MSe with a PID controller as well. This has been omitted since the controller eliminates the passive behavior of the front wheels. It wants to follow a set speed value and if this value differs from that of the rear wheels the controller will counteract all driving motions supplied by the rear wheels.

A.2.4 The Contact Parameters

In the submodel for the contact, the following parameters are of significance. The parameter WheelRadius determines the radius of the wheel, which sets the

radius of the sphere used to parameterize the wheel. In the lower part of the submodel, the parameter description of the two surfaces is given. Martijn Visser ((Visser, 2002)) developed the model in a coordinate frame where the x- and y-axes span the horizontal plane and the z-axis is the vertical. The parameter descriptions of the plain and the sphere has been adapted such that the contact model fits in the coordinate frame of the car. The curvature, metric and torsion (K, M and T) tensors from Eq. 4.7 can be found as

$$\begin{aligned}
 K_f &= \text{curvature2} \cdot \text{temp3} \\
 \tilde{K}_o &= \text{temp2} \\
 M_f &= \text{metric1} \\
 M_o &= \text{metric2} \\
 T_f &= \text{torsion1} \\
 T_o &= \text{torsion2}
 \end{aligned}$$

Implementing a cylindrical description of a tire only concerns this part of the model.

Another parameter that needs explanation is the **Deformation** parameter. This four-dimensional element describes the deformation of the wheel. The first three elements of this parameter represent the deformation in respectively the r_y , t_x and t_z direction and can initially be set to zero. However, the fourth value is the deformation in the t_y direction. So it contains the position of the contact point relative to the contact point on the plain (relative to the tire, so this value corresponds to *minus* the height of the wheel surface from the plain). This value has to be initialized to the initial height of the wheel. Not setting this value results in an infinite tumble of the car. In the **Global** submodel this parameter has been set to its correct value, but when changing parameters as the wheel radius or other car dimensions, check the validity of this parameter.

N.B. initially, the car axle submodel was called **axis**. Since this can give rise to confusion with respect to the axes of the coordinate frames it is now referred to as **axle**. However, 20-SIM did not allow a change of name afterward, so the Figure still depicts the axle with the name **axis**. Throughout the model, all the rigid bodies depicting an axis of some kind, bear reference to an axle of the car.

Bibliography

- Dixon, J. C. (1991). *Tyres, Suspension and Handling*, Cambridge University Press.
- Duindam, V. and Stramigioli, S. (2002a). *Geometric Description of Contact Kinematics*, University of Twente, Drebbel Institute for Mechatronics.
- Duindam, V. and Stramigioli, S. (2002b). *A Novel Lumped Spatial Model of Tire Contact*, University of Twente, Drebbel Institute for Mechatronics.
- Keukens, F. (2001). *Screw Bond Graph Modelling of the Jumbo Container Crane*, TUDelft.
- Murray, R. M., Li, Z. and Sastry, S. S. (1994). *A Mathematical Introduction to Robotic Manipulation*, C.R.C. Press.
- of Transportation, U. D. (1985). *Mechanics of Pneumatic Tires*, U.S. Government Printing Office, Washington D.C.
- Paynter, H. (1961). *Analysis and Design of Engineering Systems*, The M.I.T. Press.
- Rahnejat, H. (1998). *Multi-Body Dynamics, Vehicles, Machines and Mechanisms*, Professional Engineering Publishing Limited, London and Bury St Edmunds, UK.
- Stramigioli, S. (2001). *Modeling and IPC Control of Interactive Mechanical Systems - A Coordinate-Free Approach*, Springer-Verlag.
- Stramigioli, S. and Bruyninckx, H. (2001). *Geometry and Screw Theory for Robotics*.
- Visser, M. (2002). Contact dynamics with friction.