Conceptual and reduced vehicle models performance enhancement through parameter estimation and neural-networks coupling

M. Gubitosa\textsuperscript{1}, T. Tsukano\textsuperscript{1}, S. Donders\textsuperscript{1}, W. Desmet\textsuperscript{2}
\textsuperscript{1}LMS International
Interleuvenlaan 68, B-3001, Leuven, Belgium
email: marco.gubitosa@lmsintl.com

\textsuperscript{2}K.U.Leuven, Department Mechanical Engineering
Celestijnenlaan 300 B, B-3001, Heverlee, Belgium

Abstract
A detailed 3D vehicle dynamics benchmark case has been defined, which has been modeled in LMS Virtual.Lab Motion. The simplified models are created in the 1D multiphysics environment of LMS Imagine.Lab Amesim, in which several mathematical vehicle representations have been adopted, tested and functionally correlated with the reference case. Primary importance has been given to the reliability of results together with numerical efficiency of the finalized models. For this purpose, the reduced models have been enhanced by implementing a Neural Network model in parallel, which can be trained to extend the model’s capability to reproduce the complex dynamics of the real system, while remaining as simple as possible for online computation. This new methodology is validated on numerical examples, including an industrial-level vehicle dynamics application case.

1 Introduction

Simplified vehicle models are often used by control engineers for control design and online implementation of on-board safety systems. Typically, these models are intensively used for efficient cycle computation within a limited validity range. In these simplified models, several physical phenomena and dynamics effects are not included, meaning that from the analysis of these models, the controls engineer does not learn about possible interactions of the neglected dynamics with the control law and furthermore, the effect of uncertainty in the input parameters on the controlled system performance is not assessed. For a proper insight in the physical performance of the end product (i.e. the active vehicle on the road), an improved engineering process is needed to guarantee the vehicle and the controller performance even in the presence of unmodelled physical effects and uncertainty in the input parameters. This paper contributes to resolving this need, by presenting an engineering methodology that starts from detailed information regarding the dynamics phenomena (either from test or detailed prediction), cascading into conceptual vehicle dynamics models with modular level of complexity. Such a simplified model can subsequently be used in the field of conceptual vehicle modeling, functional performance prediction and controller evaluation. The limited range of validity of such a modeling approach is then extended in the range of nonlinear vehicle behavior thanks to the adoption of black box structured systems: in the specific case Artificial Neural Networks (ANN, addressed to in the following also as simply Neural Networks).

Before setting the next step into the identification of the vehicle dynamics using Neural Networks, in section 2 a brief overview of the system identification techniques adopted in the field of vehicle modeling is presented, summarizing the approach already reported in [1]. In section 3 the vehicle models are presented, describing the high fidelity and simplified models. In section 4 the results of the scheme adopted for parameters estimation is briefly described, introducing the use of Neural Networks for models enhancement. Finally in section 5 the findings are summarized and conclusions are drawn.
2 Vehicle dynamics through model reduction

Functional 1D system simulation models are intensively used in an early stage of the vehicle dynamics development chain – in the so-called concept phase – while detailed 3D multi-body simulation models are typically used later on, when more detailed information becomes available. Between the 1D and 3D modeling approaches, there is a historic disconnect. At some point in the vehicle design process, the step must be set from the optimized functional 1D model to the initial geometric description of the vehicle chassis. From that point onwards, there exists a 1D-3D disconnect. Furthermore, there is a separation between controls engineering departments and vehicle functional performance engineering departments. Because of this gap, the functional system models, that are essential for controls engineering, often have to be “reinvented” by the control engineers. Frequently, these models are very coarse, not precise, and not connected to the much more accurate 3D system models developed by the functional performance engineers. A key challenge is to achieve parameter consistency between 1D and 3D models, enabling a seamless continuation of the engineering effort throughout the vehicle development cycle. Moreover the simplified models are intended to be a representation of single – or reduced set of – overall vehicle behavior, targeted indeed to the scope of the analysis. It is well understood that the existing vehicle models span a wide range in the complexity spectrum [2], which makes the selection of the right level of details a non-trivial task for the vehicle engineer. A systematic way of addressing this challenge is to start with a high fidelity model and cascade it down to a series of simplified representations of the separate dynamical ranges. The high fidelity vehicle model will contain a large number of parameters, but not all of them are important factors for the purpose of understanding the performance. To help decide which parts of the model can be reduced to a low model order such as look-up tables or synthesis parameters (e.g. the cornering stiffness for tire’s lateral force generation) and which must be translated directly into the reduced model representation, sensitivity analysis can be used. The question is finally to determine which high fidelity model parameters most affect a given metric or likewise whether the required performance are reachable by varying the reduced set of definition parameters in the simplified scheme.

One technique to reduce the vehicle model is the generation of look-up tables for the suspensions [3]. Often a kinematic map between the spindle motions relative to the chassis can be used instead of modeling all the separate suspension bodies. However, a kinematic model alone is not sufficient in some cases. As a substitute, an elasto-kinematic map should be used. This approach requires an a priori deep knowledge of the target vehicle and the possibility to run experiments of a (in case virtual) test rig event to retrieve the necessary information. The work here presented allows avoiding such a situation by allowing to generate (a set of) models able to predict the vehicle behavior in defined dynamics ranges by making use of gradually increasing information regarding the model definition, to finally allow building a communication bridge between the 1D and 3D simulation domains.

2.1 The parameter identification challenge

Systems identification is a general term to describe mathematical tools and algorithms that can be used for the creation of dynamic (time-domain) system models from measured data. Nowadays, a range of different approaches and a long list of algorithms are available, which can be classified based on the knowledge of the system and hence the modeling process adopted: black, gray and white-box identification. In the approach presented in [1] and here continued for the study case of vehicle lateral dynamics analysis, a piecewise distinction of the range of interest is obtained with a time rectangular windowing of the output signal from the high fidelity model on which a selection of optimization algorithms is tuning different set of parameters defined based on a priori sensitivity studies.

In this paper, the process of parameters estimation has been focused on the conceptual distinction of parameters and their domain of existence. As clarified by the following table, the distinction has been made between Assured, Calculated and Estimated parameters.
Table 1: Phenomenological classification of system’s parameters

A systematic approach has been proposed, which comprises two steps:

1) in the first step, the Assured and Calculated parameters are provided as input to the simplified models and considered as fixed
2) in the second step, optimization loops run to determine the Estimated parameters in an optimization loop that aims to minimize multiple objectives.

Figure 1: Scheme for parameters estimation

3 Vehicle models

3.1 High fidelity modeling approach

A high fidelity multibody 3D vehicle dynamics benchmark case has been defined, which has been modeled in LMS Virtual.Lab Motion.

Virtual.Lab Motion is the heir of LMS DADS, notorious for its solver accuracy and stability. It is a general purpose MB (Multi Body) package based on a Cartesian coordinates approach for the assembly of the equations of motion [4]. The solver uses Euler parameters to represent the rotational degrees of freedom (avoiding therefore the intrinsic singularity of the angular notation) and Lagrangian formulation for the assembly and generation of equations of motion.
The joints between bodies are expressed in a set of algebraic equations, subsequently assembled in a second derivative structure, resulting finally in a set of Differential Algebraic Equations (DAEs) that can be written in the following form (see also [5]):

\[
\begin{bmatrix}
M(q) & \Phi(q)' \\
\Phi(q) & 0
\end{bmatrix}
\begin{bmatrix}
\dot{q} \\
\lambda
\end{bmatrix}
= \begin{bmatrix}
Q(q,\dot{q},\gamma)
\end{bmatrix}
\]

Here,

- \( M \) is the mass matrix
- \( q \) is the vector of the generalized coordinates
- \( Q \) is the vector of the generalized forces applied to the rigid bodies
- \( \lambda \) is the vector of the Lagrange multipliers
- \( \Phi \) is the Jacobian of the constraint forces
- \( \gamma \) the right-hand-side of the second derivative the constraint equations

The analysis model of interest in this paper, represented in Figure 2, includes 52 bodies; therefore a total of 52 x 7 = 364 configuration parameters are used by the pre-processor to build the set of equations of motion. For the settling configuration, in which the vehicle is positioned on the ground, exposed to gravitational force & left to settle into an equilibrium steady-state position, joints and drivers are for a total of 234, therefore leaving the system with 130 degrees of freedom. While setting up a maneuver, instead, additional constraints are added to the system in terms of position driver on the steering wheel, commanded in open loop, and forces are acting on the wheel’s revolute joints to represent the driving torque. Moreover, non-zero initial conditions at velocity and position level are added to each of the bodies.

This model includes a front double A-arm suspension and rear multi-link attached to the sub-frame. Front and rear antiroll bars are built-in but not modeled as flexible 3D elements: their effect is taken into account through lumped torsional springs. A steering rack-and-pinion system is included, which is directly supported by the rigid vehicle body. The suspension force elements and non linear elastic connections (i.e. bushings) are included, granting the model a total of 116 degrees of freedom. An accurate modeling of the tire force elements is achieved by including the so called TNO-MF tire (version 6.0), which is based on the renowned ‘Magic Formula’ tire model of Pacejka [6]. The model takes as input a series of parameters (i.e. a vector with more than 100 elements) for each calculation to be performed, which are empirically determined coefficients that address the complexity of the model.

Figure 2: Representation of the front and rear axles and screenshot of the vehicle performing a slalom maneuver
3.2 Simplified modeling approach

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\psi, \dot{\psi}$</td>
<td>Yaw angle, yaw velocity and yaw acceleration</td>
</tr>
<tr>
<td>$\varphi, \dot{\varphi}, \ddot{\varphi}$</td>
<td>Roll angle, roll velocity and roll acceleration</td>
</tr>
<tr>
<td>$\beta, \dot{\beta}$</td>
<td>Car-body sideslip angle, velocity at center of gravity</td>
</tr>
<tr>
<td>$v$</td>
<td>Absolute car-body velocity at center of gravity</td>
</tr>
<tr>
<td>$\delta_1, \delta_2$</td>
<td>Steering angle of the front wheels and rear</td>
</tr>
<tr>
<td>$\lambda_1, \lambda_2$</td>
<td>Coefficient for camber angle induced by roll (front and rear)</td>
</tr>
<tr>
<td>$a_1, a_2$</td>
<td>Front and rear wheelbases</td>
</tr>
<tr>
<td>$b_1, b_2$</td>
<td>Front and rear half tracks</td>
</tr>
<tr>
<td>$h$</td>
<td>Relative position of roll center with respect to car-body CG</td>
</tr>
<tr>
<td>$M$</td>
<td>Total mass of the vehicle</td>
</tr>
<tr>
<td>$I_{xx}$</td>
<td>Roll inertia of the vehicle</td>
</tr>
<tr>
<td>$I_{zz}$</td>
<td>Yaw inertia of the vehicle</td>
</tr>
<tr>
<td>$K_{\varphi}$</td>
<td>Total anti-roll stiffness $(K_{r1} + K_{r2})$</td>
</tr>
<tr>
<td>$b_{\varphi}$</td>
<td>Total roll damper rate</td>
</tr>
<tr>
<td>$C_1, C_2$</td>
<td>Camber stiffness, resp. for front and rear axle</td>
</tr>
<tr>
<td>$g$</td>
<td>Constant of gravity (defaulted to 9.80665 m/s)</td>
</tr>
<tr>
<td>$\delta_{\text{axle,elas}} + \delta_{\text{tire}}$</td>
<td>Total front and rear sideslip (axle + tire)</td>
</tr>
<tr>
<td>$\delta_{\text{axle,kin}}$</td>
<td>Steering angle of the wheel due to axle kinematics - Front axle</td>
</tr>
<tr>
<td>$\delta_{\text{axle,elas}}$</td>
<td>Toe angle of the wheel due to axle elasto-kinematics - Front axle</td>
</tr>
<tr>
<td>$\delta_{\text{axle,kin}}$</td>
<td>Steering angle of the wheel due to axle kinematics - Rear axle</td>
</tr>
<tr>
<td>$\delta_{\text{axle,elas}}$</td>
<td>Toe angle of the wheel due to axle elasto-kinematics - Rear axle</td>
</tr>
<tr>
<td>$M_{\text{axle,-k,el,,b,axle}}$</td>
<td>Mass, stiffness and damping of the axle in lateral direction</td>
</tr>
<tr>
<td>$v_{y1}$</td>
<td>Lateral deformation velocity of the axle (front and rear)</td>
</tr>
</tbody>
</table>

Table 2: List of symbols for the simplified model

First, the domain of lateral vehicle dynamics is investigated. As mentioned before a range of possible approaches has been reported to model the dynamics of a vehicle. Depending on the field of study and the accuracy required, the details to be included vary considerably. The solution for this dilemma, and a trustworthy help to the vehicle dynamics engineer comes from the adoption of a modular simplified modeling approach. A key element herein is the usage of numerical simulations, with a particular focus on conceptual (1D) modeling and simulation. Tools have been developed to meet those needs, merging industrial competences and advanced research activities (e.g. Imagine.Lab Amesim vehicle dynamics library) [7][8]. The simplified model for lateral dynamics studies proposed in the following is a four wheels chassis model with medium wheel approach for front and rear axles. The structure of this model, well known in the literature, is meant for handling modeling and lateral dynamics studies and has 3 DOF: yaw velocity ($\dot{\psi}$), car body sideslip angle at center of gravity ($\beta$) and roll angle ($\varphi$). The equation of motion are obtained by cascading the overall dynamics to the following set of differential equations, expressed in what are normally called quasi-coordinates and generated by forces and moments balance of the Newton-Euler approach. Moreover, linearization in the McLaurin series (assuming to be in steady state conditions and close enough to the equilibrium position) brings to the condensed formulation:

$$\begin{cases} 
M [v (\dot{\psi} + \dot{\beta}) - h \dot{\varphi}] = \sum F_{y \,\text{tires}} \\
I_{zz} \ddot{\psi} = a_1 F_{y \,\text{tire}1} - a_2 F_{y \,\text{tire}2} \\
(I_{xx} + Mh^2) \ddot{\varphi} - Mh [v (\dot{\psi} + \dot{\beta}) - h \dot{\varphi}] + b_\varphi \dot{\varphi} + k_\varphi \varphi = h_0 \sum F_{y \,\text{tires}} 
\end{cases}$$

(2)
Hence a global motion is allowed with respect to the ground, including also the car-body roll effect on the generalized sideslip and yaw velocity (due to roll center heights and axle kinematics). In the state equations, the relative position of roll center with respect to the car-body center of gravity can be computed with relative height of roll center above front and rear axle:

\[ h = h_G - h_0 = h_G - \left( \frac{a_2}{a_1 + a_2} h_{1G} + \frac{a_1}{a_1 + a_2} h_{2G} \right) \]  

(3)

In addition the load transfer between left and right is included (effect while negotiating a turn) which brings in the variation of lateral force available at the tires, computed via:

\[ F_{\text{tire}} = \left( a_3 \sin \left( 2 \cdot \arctan \left( \frac{F_a}{a_4} \right) \right) \right) \delta_{\text{tire}} \]  

(4)

The total axle sideslip angle gives an extra contribution to the lateral force acting on the axle due to the synthesis parameters of camber stiffness \( C_1 \lambda_1, C_2 \lambda_2 \) and lateral stiffness and damping. Therefore the combination of axle kinematic steering angle \( \text{axle kin} \) and axle compliance contribution \( \text{axle elas} \) is considered; the tire slip angle can be hence written as:

\[ (\delta_{\text{tire}})_{\text{front}} = \beta + \frac{a_4 \psi - v_{y1}}{v} - \delta_{\text{axle kin}} - \delta_{\text{axle elas}} \]  

(5)

In addition the relaxation length is considered as a simple first order filter with fixed time constant.

![Figure 3: Representation of the vehicle model in Imagine.Lab AMESim for studying lateral dynamics](image)

For more details and a more complete overview of such a modeling approach, refer to [7][8][9]. The set of equations obtained for this model has been presented in the form of an ODE system, but with an implicit loop for the computation of the lateral force on the tire, which depends on the lateral slip (which in turn is computed from the lateral force).
4 Case study of Lateral Dynamics

In general the motion equations governing a mechanical system are in the form of second order differential equations. Three types of solutions can be computed from this mathematical formulation, corresponding to three types of driving circumstances: the steady-state, the stability solution and the frequency response. To be able to explore those three domains, different maneuvers have been selected as comparison between the high fidelity model and the simplified model: slalom, step steer and a form of open loop double lane change.

Figure 4: Comparison of the behavior of simplified and reference model for lateral acceleration and yaw rate in 1) step steer, 2) double lane change, 3) slalom, 4) polynomial steering angle, 5) Step steer (large angle)
It is commonly accepted that the yaw rate relates mainly to what a driver sees, whereas the lateral acceleration relates to the human feelings. Because of providing such complementary information regarding the driver perception, both yaw rate and lateral acceleration will be considered as output parameters. The identification method illustrated in detail in [1] is then based on the error minimization calculated in three different time windows considering basically the equilibrium starting condition, transient behavior, and steady state after distortion (steering input). Before the optimization, an accurate sensitivity analysis is run. Since the number of parameters is high, the importance of selecting the appropriate excitation for the target parameters is a crucial step, since one must avoid ending up with an ill-posed inverse problem. The objective functions of each stage are hence defined by results of the sensitivity analysis.

The subsequent optimization has been divided into three stages, cascading from the highest (most contributing) to lower sensitive parameters with respect to the selected cost functions. From the optimization process, an optimal parameters list has been obtained. The final comparison between the high fidelity model and simplified lateral dynamic model are shown in Figure 4, together with events for which the model was not previously tested. In particular Figure 4 proposes in the 4th row a cross validation of the model by applying to it a random steering action in time, and in the 5th row an event involving higher lateral acceleration (i.e.: step steer maneuver with large steering input).

As well known and already observed in [1], the validity of the simplified model must be assessed. Typically, these models are intensively used for efficient cycle computation within a limited validity range: several physical phenomena and dynamics effects are not included. Furthermore, the effect of uncertainty in the input parameters on the controlled system performance is not assessed. An example is indeed in the notable gap between the prediction of the simplified model and the high fidelity multibody simulation when large lateral acceleration is required. For a proper insight in the physical performance of the end product (i.e. the active vehicle on the road), an improved engineering process is needed to guarantee the prediction performance even in the presence of unmodelled physical effects and uncertainty in the input parameters.

A possible solution for this dilemma can be found in the degrees of freedom granted by black-box identification tools. Neural Networks can support and compensate the simplified model output, in case that the simplified model lacks in accuracy. This compensation can be done in a systematic process, which has been shown in Figure 5.

![Figure 5: Schematic representation of the integrated approach of Neural Networks to enhance simplified models performance](image-url)
Some previous papers have investigated the possibility to adopt Neural Networks in the field of vehicle dynamics, either for state estimation \cite{10}\cite{11}\cite{12} or for control applications generating an adaptive control architecture. Neural Networks can be used for system identification purposes \cite{13} as they offer a means of parameter identification without the necessity of detailed model knowledge and with the possibility of providing real-time updates in online simulations. In the specific case here presented we are applying Neural Network compensation at full system level. For an overall conceptual understanding of the applicability of Neural Networks as solution of the problem, a positive answer to the following questions is required:

- Good amount of data is available
- A non-linear multidimensional input-output system is under investigation
- Time is not a constraint for the initial training of the Neural Network

In the field of function evaluation (i.e. dynamic systems emulation), different architectures and internal characteristics of the Neural Network can be used, which provides a lot of degrees of freedom to the problem statement. It is therefore necessary to have an a priori selection criteria based on some general considerations, so that the most appropriate choice can be made. In particular, it is recommended to use the simplest possible configuration that is able to estimate the worse case scenario. The following decisions must be made regarding the Neural Network setting:

- Neural Network family
- Configuration of the layer(s) and neurons transfer functions
- Training algorithm and learning rule

A detailed description of the different possibilities of network design is beyond the scope of this paper, which focuses on providing an overall insight of the definition criteria in the remainder of this section. As reported in \cite{14} Neural Networks can be classified into two large families: static and dynamic networks. Static networks have their neurons organized in feed-forward architecture, using direct connections between input and internal layers, while dynamic behavior could be given by feedback loops and/or by delay elements\footnote{A more rigorous definition identifies feed-forward networks with delay as FIR (Finite Impulse Response) systems and feed-back, possibly with delay, as IIR (Infinite Impulse Response) systems.}, making the final output dependent on previous time step computations. Problems for which the decision boundary is not linearly separable, should avoid having linear threshold units as internal transfer functions. It is instead recommended to use differentiable functions, since Neural Networks can be trained more easily with differentiation based algorithms (like gradient descent or quasi-Newton iterations). Particular care has to be posed in the definition of the number of layers and the number of neurons per layer. Generally, it is not recommended to have more than 2 layers, since in most of cases two layers offer a proper level of complexity \cite{14}. General suggestion is hence the use of a network of two layers, in which the first contains sigmoid units and the second lists linear neurons. This network can be trained to approximate any function (with a finite number of discontinuities) with an arbitrary accuracy. Concerning the number of neurons one should observe that too few neurons could make your estimation too roughly approximating the target function, because of the limited number of degrees of freedom. On the other hand, too many neurons may make have the result that the training will make the network over-fitting the training data set, so that the network will lack generality for new situations. Moreover different possible learning procedures could be adopted. In the following section, the supervised learning algorithm based on different search criteria has been defined.

4.1 Case 1: cascade-forward Neural Network

The simplest possible network to capture the variable linear/non-linear behavior of the vehicle in lateral dynamics domain has been identified in a multilayer cascade-forward configuration with one internal layer of 20 neurons, which in addition to the standard feed-forward network includes a weight connection from
the input to each layer (and possibly from each layer to the successive layer). This has been programmed with the internal layer of sigmoid neurons and an output layer of 2 linear neurons. Such architecture allows the network to learn nonlinear and linear relationships between input and output vectors, having the linear layer granting the capability of producing values outside the range -1 to +1. As compared to the feed-forward architecture, this approach permits a faster learning, and allows to have a directly combined non-linear and linear influence of the same input on the output having a signal flow cascading in a double stepwise manner.

Figure 6: Schematic representation of the cascade-forward network architecture

The network receives as input the maneuver settings (i.e. steering angle and vehicle forward velocity) as well as the actual output in terms of lateral acceleration and yaw rate computed by the simplified model. The outputs produced are actually two signals to compensate the simplified model output again in lateral acceleration and yaw rate. The core point of implementing Neural Networks is the training phase. Particular care was therefore put in the selection of an appropriate set of input able to attribute a rich level of information to the network to permit the highest generalization possible. Consistent with the preliminary observation with respect to the training set to be adopted given in [10] it has been noticed that enough variability in the excitation should be provided. In particular, since the estimation focuses mainly on the non-linear range, maneuvers including high level of lateral acceleration have to be included in the training set, as well as small steering angle actions to keep the network able to behave correctly in the linear domain. For this seek of generality another event has been defined in addition to those for which the simplified model’s parameters have been estimated, which is involving in one single event both linear and nonlinear behavior.

The algorithm adopted for the training is based on the Levenberg-Marquardt, which uses an approximation of the Hessian matrix in the computation of a Newton-like update that aims at minimizing the mean squared error between the network output and the reference model output.

Figure 7: Comparison in lateral acceleration and yaw rate for the random large steering input
Figure 8: Comparison of the behavior of enhanced model with cascade-forward Neural Network and reference model for lateral acceleration and yaw rate in 1) step steer, 2) double lane change, 3) slalom, 4) polynomial steering angle, 5) Step steer (large angle)

4.2 Case 2: NARX Neural Network

At first sight, it may seem unnecessary to implement a dynamic network architecture, especially when considering the good results that have already been obtained with a cascade-forward configuration. However, when one considers the nature of the system under analysis, the potential added value of the dynamic architecture is clear. Therefore, also this approach has been investigated in this paper. The delays in the input and in the feedback of the output give an Infinite Impulse Response like behavior, which is actually typical of mechanical systems. In the case under investigation, the NARX (Non-linear AutoRegressive with eXogenous inputs) architecture has been chosen [14].
As represented in Figure 9 the NARX network has been adopted in a series-parallel feed-forward configuration to better perform the training with static back propagation algorithm, and converted in the feedback form once convergence is achieved. The network is configured with one inner layer of 10 sigmoid neurons and an output layer of two linear units. No delay elements have been introduced on the four inputs (steering angle, forward velocity, lateral acceleration and yaw rate of the simplified model) while 10 act instead on the feed-back loop. The type of training makes a substantial difference, granting the two networks a similar fitting of the trained time series but a different generalization capability. In both cases the training used is performed in batch, meaning that the step of variation for the weight and biases are updated after all the inputs and targets are presented. In the case considered here the Levenberg-Marquardt algorithm has been perfected with the usage of Bayesian Regularization [14] that minimizes a combination of squared errors and weights, and then determines the correct combination in order to produce a network that generalizes well. The training event considered here is the same as the previous case, producing the plots shown in Figure 10 and 11 as validation results, after the network inversion.

![Figure 9: Schematic representation of the non linear auto regressive with exogenous inputs network architecture for the training events (A) and after (B)](image)

Figure 10: Comparison in lateral acceleration and yaw rate for the random large steering input
Figure 11: Comparison of the behavior of enhanced model with NARX Neural Network and reference model for lateral acceleration and yaw rate in 1) step steer, 2) double lane change, 3) slalom, 4) polynomial steering angle, 5) Step steer (large angle)

5 Conclusions

In this paper a vehicle concept modeling methodology has been presented, which starts from detailed information regarding the dynamics phenomena, and cascades down into conceptual vehicle dynamics models. Simplified vehicle models are often used by control engineers for control design and online implementation of on-board safety systems. Typically, these models are intensively used for efficient cycle computation within a limited validity range. Typically, these models are intensively used for efficient cycle computation within a limited validity range. In these simplified models, several physical phenomena and dynamics effects are not included, generating visible discrepancies in the example case of lateral
dynamics here investigated. Mainly this is recognizable to be due to the non linear effects neglected in the
simplified model and given by:

- the suspension movement that involves angles variation during bump & rebound which generally
  affect the dynamic behavior; moreover the elasticity of the suspension is collapsed in its
  contribution in roll and no influence of the pitch movement is included
- big effect of discrepancy comes from the assumption of simplified tire model.
- time dependent parameters such as the variation of roll center position are not taken into account.

As a next step, the 1D vehicle model range of validity has then been extended in the range of nonlinear
vehicle behavior thanks to the adoption of black box structured systems: in the specific case artificial
Neural Networks. Two architectures have been trained and proved to give good accuracy, with preference
on the generalization capabilities observable in the adoption of a NARX dynamic network. Future work
includes the investigation of a more rigorous approach, in which a number of simple Neural Networks can
be positioned in parallel with single sub-components aiming at enhancing the synthesis parameters, which,
by definition, are often the nests of crucial simplifications.

Acknowledgements

The authors kindly acknowledge the EC Marie Curie ITN project VECOM (see www.vecom.org), funded
by the European Commission, from which the first author, Mr. Marco Gubitosa, holds a grant. Furthermore, we kindly acknowledge IWT Vlaanderen for their support of the ongoing research project
“From System Modeling to S/W running on the vehicle – MODELISAR”, which is part of the ITEA
project 07006 (in turn, supported by the European Commission). We finally acknowledge the
VULCANUS program of the EU-Japan Centre, from which Tsukano Takatoshi received a grant.

References

and real-time simulations of vehicle system dynamics”, FISITA 2010, Budapest, Hungary, May 30,
June 4, 2010

importance analysis”, Proc. of ASME DSCC, Dynamic Systems and Control Conference, Oct 20-22,
2008, Ann Arbor, Michigan, USA

using a Reverse Engineering Approach”, Proc. ASME IMECE, Nov. 2-6, 2008, Boston, MA, USA

Motion”, Vehicle System Dynamics, Vol 45 (S1), 2007, pp 199-206

Symposium on International Automotive Technology, Purne, India, Jan 1999.

2005

Functional Simulation”, Vehicle Dynamics Expo North America, Oct 2008, Novi, MI, USA

software to standard product and modularity for chassis subsystems”, SIA Congress, Sept 2009.


