006 Multibody Dynamics Software for Controlled Vehicle Simulation

WERNER O. SCHIEHLEN

Institute B of Mechanics, University of Stuttgart,
W-7000 Stuttgart 80, Germany

The method of multibody systems achieved major importance in vehicle dynamics for traditional and advanced applications. In the context of concurrent engineering and advanced vehicle design, multibody formalisms represent the core of every software package for modeling and simulation. The paper presents a multibody system datamodel and the database RSYST linking software models. In detail the moduls NEWEUL for the symbolical generation of equations of motion and NEWSIM for the simulation are considered. As an example the multibody dynamics software is applied to a van with actively controlled suspensions.

Keywords: Computer Application, Modeling, Simulation, Vehicle Dynamics, Attitude Control

1. INTRODUCTION

Rail and road vehicles can be modeled properly as multibody systems for the design and the analysis of components like suspensions, attitude controllers, shock absorbers, springs, mounts and steering assemblies as well as brakes and antiskid devices. The complexity of the dynamical equations called for the development of computer-aided formalisms a quarter of a center ago. The theoretical background is today available from a number of textbooks authored by Wittenburg [1], Schiehlen [2], Roberson and Schwertassek [3], Nikravesh [4], Haug [5] and Shabana [6].

In addition, a number of commercially distributed computer codes was developed, a summary of which is given in the Multibody System Handbook [7]. The computer codes available show different capabilities: some of them generate only the equations of motion in numerical or symbolical form, respectively, some of them provide numerical integration and simulation codes, too. Moreover, there are also extensive software systems on the market which offer additionally graphical data input, animation of body motions and automated signal data analysis. There is no doubt that the professional user, particularly in the automotive industry, prefer the most complete software system for dynamical multibody system analysis.

On the other hand for the design of advanced vehicle control systems symbolical equations of motion are required by the control engineer. This means that for one and the same vehicle system different computer codes have to be applied resulting in repeated data input and a lack uniqueness. Therefore, a standard for multibody system data seems to be necessary, an approach which is successfully used in concurrent engineering, too, see e.g. Haug [8]. In Germany, during the last four years, a multibody system data model was defined and agreed by 14 academic and industrial research groups, see Ref. [9].

In this paper, the datamodel and the corresponding database RSYST are presented. Then, two moduls for the generation of equations of motion using this datamodel, and for the simulation of the vehicle motion will be discussed. Finally, an actively controlled vehicle is considered. A multicriteria optimization of the attitude control parameters shows the possibilities and the limits of performance improvement of a vehicle designed by multibody dynamics simulations.

2. THE MULTIBODY SYSTEM DATAMODEL

The method of multibody systems is based on a finite set of elements like bodies and joints shown in Ref. [2] or by Roberson and Schwertassek [3]. These elements can be used to define a unique datamodel applying classes and objects of software engineering databases.

The datamodel has been defined as a standardized basis for all kinds of computer codes by Otter, Hocke, Daberkow and Leister [10]. The following assumptions were agreed upon:

1. A multibody system consists of rigid bodies and ideal joints. A body may degenerate to a particle or to a body without inertia. The ideal joints include the rigid joint, the joint with completely given motion (rheonomic constraint) and the vanishing joint (free motion).
2. The topology of the multibody system is arbitrary. Chains, trees and closed loops are admitted.
3. Joints and actuators are summarized in open libraries.
4. Subsystems may be added to existing components of the multibody system.

A datamodel for elastic bodies is under development and will be completely compatible with the rigid body datamodel.

A multibody system as defined is characterized by the class *mbs* and consists of an arbitrary number of the objects of the classes *part* and *interact*, see Fig. 1.

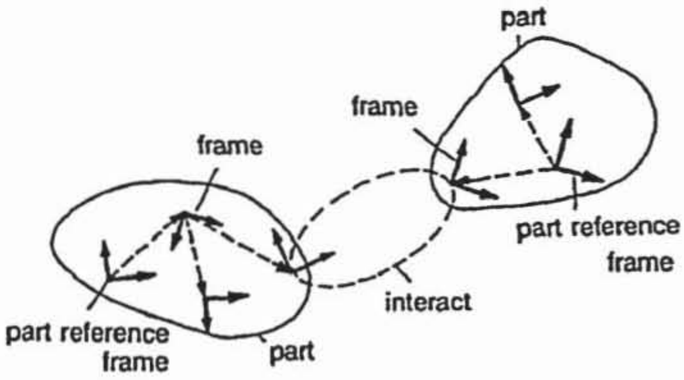


Fig. 1. Definition of a multibody system

The class *part* describes rigid bodies. Each *part* is characterized by at least one body–fixed *frame*, it may have a mass, a center of mass and a tensor of inertia.

The class *interact* describes the interaction between a frame on *part a* and a frame on *part b*. The interaction may be realized by a joint, by a force actuator or a sensor resulting in the classes *joint*, *force* or *sensor*, respectively. Thus, the class *interact* is characterized by two types of information: the frames to be connected and the connecting element itself, see Fig. 2.

class:	interact	
name	*class*	*description of component*
connect	connect	frames to be connected
member	member	connecting element

Fig. 2. The class *interact*

According to the assumptions, the datamodel represents a holonomic, rheonomic multibody system. If the formalism selected uses a minimal number of coordinates, like NEWEUL, then, the resulting equations of motion read as

$$M(y,t)\,\ddot{y} + k(y,\dot{y},t) = q(y,\dot{y},t)\,. \qquad (1)$$

Here, y is the position vector and t the time, M means the generalized inertia matrix, k and q represent the generalized gyroscopic and applied forces. However, Eq.(1) which represents a set of pure differential equations is not a consequence of the datamodel. By an appropriate formalism from the same datamodel a set of differential–algebraical equations may be obtained, too.

In addition to the mechanical elements of a multibody system, there exists also the possibility to use actively controlled elements resulting in the more general representation of dynamical systems, e.g.

$$\dot{x} = f(x,u,t,p) \qquad (2)$$

$$v = g(x,u,t,p)$$

where x is the state vector, v the output vector, u the input vector of controls , t the time and p the vector of mechanical and control parameters. Furt
her, f and g mean nonlinear vector functions. For more details see Ref. [10].

3. THE DATABASE RSYST

The scientific–engineering database system RSYST is considered to be a software tool. It supports the

- development of user programmes,
- development of program packages,
- execution of programmes,
- handling of large data sets,
- analysis of data.

One of the main applications of RSYST is the compilation of data and user programmes. RSYST is written in FORTRAN 77 and, therefore, it has an excellent portability to all kinds of computers. The fundamental elements of RSYST are the following:

- execution control,
- information system,
- dialogue system,
- output handling,
- dynamic storage handling,
- method and model base,
- database.

The RSYST system has been developed by Rühle and his staff at the Computing Center of the University of Stuttgart. A detailed description is given by Lang [11], Loebich [12] and Rühle [13].

Most important for the multibody system datamodel are the RSYST database and the handling of data objects. All the data in RSYST are considered as objects of a database. Such data objects are e.g. vectors, matrices, sets of parameters, texts or formally defined objects. The data objects are stored in the RSYST database subject to a very efficient handling, they are identified by special names.

Each data object in the RSYST database is characterized by a data description, identifying to the data type. The data description permits a correlation between data objects and possible operations.

RSYST offers the following operations on data objects which are completely internal executed:

- object generating,
- object changing,
- object deleting,
- object listing,
- objects relating to each other,
- objects storing and reading,
- handling of components of objects.

The objects have to be interpreted for the identification of their information. A set of objects with same rules of interpretation are called a class specified by a name. For example, the four components of the class "frame", are shown in Fig. 3.

class:	frame					
name	*type*	*min*	*max*	*default*	*unit*	*description of component*
rframe	name			'␣'		name of reference frame on the same part
origin	param (3)			3 * 0.	[m]	origin of frame
axleseq	int (3)	−3	3	1, 2, 3		axlesequence of elementary rotations
angles	param (3)	-2π	2π	3 * 0.	[rad]	rotation angles

Fig. 3. Objects of the class "frame" in RSYST representation

The database RSYST offers a suitable software engineering concept for multibody system dynamics as shown in Fig. 4. This concept provides the opportunity to use a modular structure of the software, i.e. different multibody formalisms may be combined with different simulation programs via standardized interfaces.

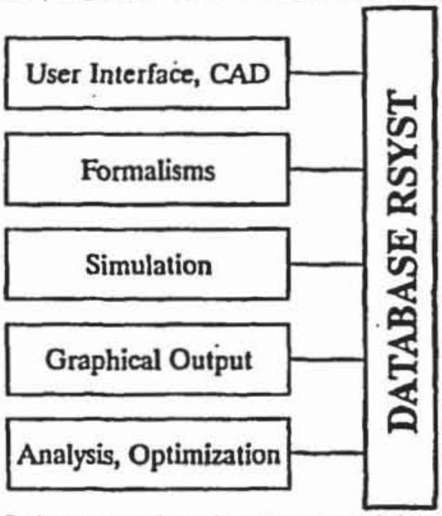


Fig. 4. Software engineering concept of the multibody system dynamics

4. THE MODULS NEWEUL AND NEWSIM

The software moduls NEWEUL and NEWSIM correspond completely with the multibody system datamodel and the database RSYST. However, they can be also used as a stand alone programme package.

NEWEUL is a software package for the dynamic analysis of mechanical systems with the multibody system method. It comprises the computation of the symbolic equations of motion by the modul NEWEUL and the simulation of the dynamic behavior by the modul NEWSIM.

Multibody systems are mechanical models consisting of

- rigid bodies,
- arbitrary constraining elements (joints, position control elements),
- passive coupling elements (springs, dampers), and
- active coupling elements (servo motors).

The topological structure of the models is arbitrary, thus possible configurations are

- systems with chain structure,
- systems with tree structure, and
- systems with closed kinematical loops.

The scleronomic or rheonomic constraints may be

- holonomic or
- nonholonomic.

The software package NEWEUL has been successfully applied in industrial and academic research institutions since 1979. The major fields of application are

- vehicle dynamics,
- dynamics of machinery,
- robot dynamics,
- biomechanics,
- satellite dynamics,
- dynamics of mechanisms.

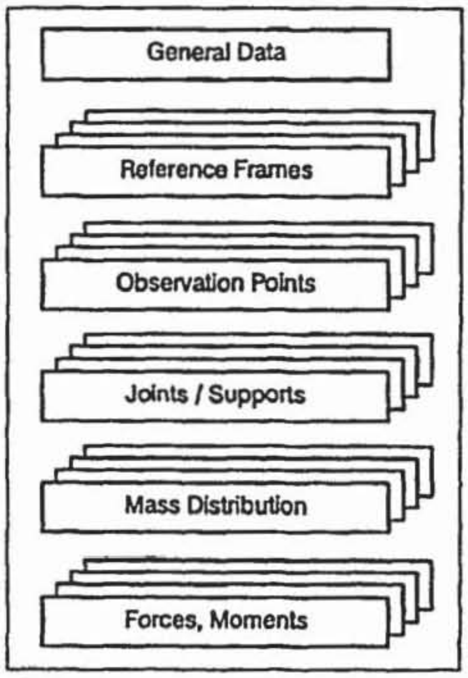


Fig. 5. Structure of a NEWEUL input file

The software package NEWEUL offers two approaches for multibody system modeling. These are

- the successive assembly approach using the kinematics of relative motions, and
- the modular assembly approach based on subsystems.

The input data for NEWEUL have to be entered in input files prepared with *prompts* and *comments*, see also Fig. 5.

The user has to provide only simple expressions for the description of kinematics and mass distribution with respect to arbitrary reference frames. Observation or sensor points, respectively, allow the determination of position, velocity and acceleration of arbitrary points of the multibody system.

NEWEUL generates the equation of motion of multibody systems in symbolic form. The computation is based on a Newton–Euler formalism with application of the principles of d'Alembert and Jourdain. The resulting equations of motion are

- linear,
- partially linearized, or
- nonlinear

symbolic differential equations. Constant parameters can be included in numerical form. Nonlinear coupling elements in kinematically linear models are also permitted.

For the output format of the equations of motion several options are possible. FORTRAN compatible output allows the equations to be included in commercial software packages for dynamic analysis and simulation such as, for instance, ACSL. Another output format allows the processing of the equations with the formula manipulation program MAPLE.

Control parameters for compression and factorization enable the user to change the structure of the output equations, Fig. 6 . For example, the user may want to obtain fully symbolic equations of motion in order to check the results for modeling and input errors. Later, computationally efficient compressed equations can be generated for the verified model.

The software module NEWSIM allows the simulation of the symbolic equations of motion provided by module NEWEUL. It automatically generates a problem specific simulation program. The user simply has to add the specification of

- force laws,
- system parameter values, and
- initial conditions.

The simulation results are stored in ASCII data files that can be visualized with arbitrary graphics packages.

The simulation results may contain

- the time history of the state variables,
- the kinematical data of observation points,
- data for animation,
- the time history of the reaction forces, and
- user–defined output data.

Apart from time simulations additional analyses can be performed with the module NEWSIM. These additional features include

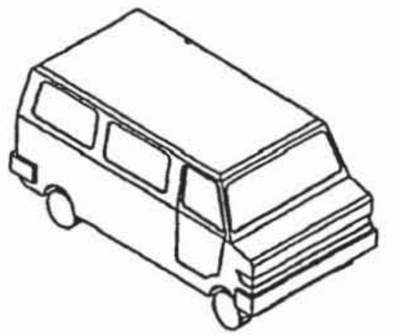
- the quasi static analysis,
- the computation of the state of equilibrium, and
- the treatment of the inverse dynamics problem.

The software package NEWEUL is written in FORTRAN 77 and can be implemented on any workstation or mainframe with a FORTRAN 77 compiler. NEWEUL uses its own formula manipulator. The theoretical background of NEWEUL is presented in more detail in Ref. [7].

5. ACTIVELY CONTROLLED VAN

The standardized object oriented datamodel for multibody systems as described in the previous section is very well qualified for vehicle dynamics applications. In particular, it offers the possibility for a very convenient benchmark analysis for all programmes adjusted to the standard of the multibody system data model.

As an example the van treated in detail by Otter, Hocke, Daberkow and Leister [10], is considered. The van consists of four bodies easily identified from the RSYST database, Fig. 7.



```
_PART
  _inertial
    _FRAME
      _iframe       frame iframe of part inertial
  _vanbody
    _BODY           van-body
    _FRAME
      _jlfwheel     frame of joint left front wheel of part
                    van-body
      _jrfwheel     frame of joint right front wheel of part
                    van-body
      _jraxle       frame of joint rearaxle of part van-body
        :             :
```

Fig. 7. Database structure of a van

Fully symbolic output:

```
C>   Inertia Matrix
      M(1,1)=M1*A**2+I1+
     +        M2*C**2+M3*C**2

      M(2,1)=-M2*B*C*SIN(AL1)*COS(AL2)+
     +        M2*B*C*SIN(AL2)*COS(AL1)
      M(2,2)=M2*B**2+I2

      M(3,1)=M3*B*C*SIN(AL1)*COS(AL3)-
     -        M3*B*C*SIN(AL3)*COS(AL1)
      M(3,2)=0.
      M(3,3)=M3*B**2+I3
```

Factorized output:

```
C>   Inertia Matrix
      M(1,1)=(C*C*(M2+M3)+
     +        (M1*A**2+I1))

      M(2,1)=C*B*M2*SIN(AL2-AL1)
      M(2,2)=(M2*B**2+I2)

      M(3,1)=C*B*M3*SIN(AL1-AL3)
      M(3,2)=0.
      M(3,3)=(M3*B**2+I3)
```

Fig. 6. Fully symbolic and factorized output of an inertia matrix

This van will be equiped with an actuator parallel to the spring–damper configuration in the suspension system, Fig. 8, as discussed by Bestle, Eberhard and Schiehlen, [14].

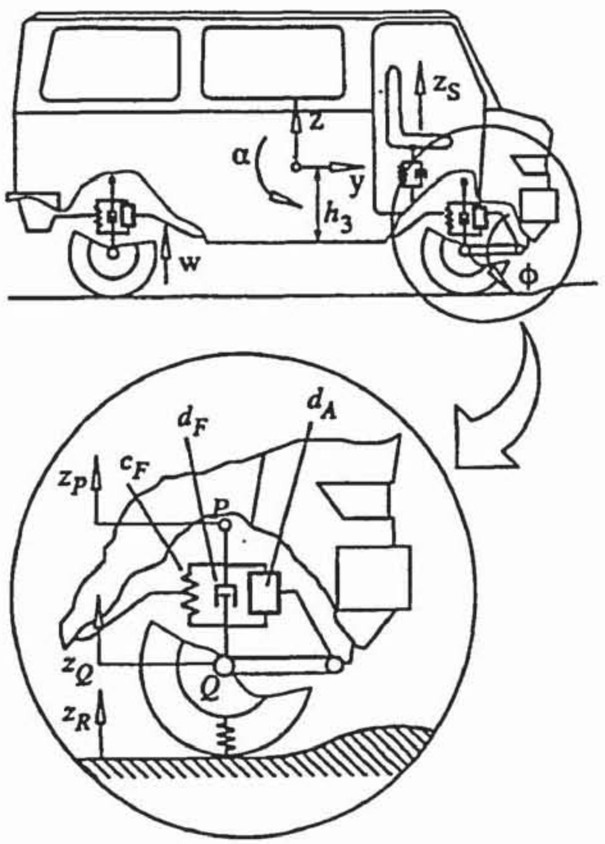


Fig. 8. Actively controlled van

To some extent, the designer of a suspension system has a free choice of the values of the stiffness coefficients c_F, damping coefficients d_F, and control parameters d_A. Any of these parameters may be chosen as a design variable. The resulting generalized forces $\mathbf{q}$ have a direct influence on the dynamical behavior of the vehicle. But geometrical data like the height of the center of gravity of the carbody or the masses and moments of inertia of some bodies may be changed within given ranges, too. This will result in a different behavior due to changes in the inertia matrix $\mathbf{M}$ and the gyroscopic forces $\mathbf{k}$, respectively.

After summarizing all design variables in a vector $\mathbf{p}$ and introducing generalized velocities $\mathbf{z} = \dot{\mathbf{y}}$, the equations of motion (1) has to be rewritten as

$$\dot{\mathbf{y}} = \mathbf{z},$$
$$\mathbf{M}(t, \mathbf{y}, \mathbf{p})\, \dot{\mathbf{z}} + \mathbf{k}(t, \mathbf{y}, \mathbf{z}, \mathbf{p}) = \mathbf{q}(t, \mathbf{y}, \mathbf{z}, \mathbf{p}) . \qquad (3)$$

The application of optimization methods requires the definition of performance criteria. An important function of suspension systems is to improve the riding comfort by isolating the carbody from forces generated by roadway unevenness at the wheels. The comfort of a vehicle can be evaluated by the acceleration $\ddot{z}_s$ acting on the driver. If the driving over a sinusoidal bump is considered as a test, accelerations may be penalized by the square of the time, too. An integral performance criterion can then be expressed by

$$\psi_C = \int_{t^0}^{t^1} t^2\, \ddot{z}_s^2\, dt . \qquad (4)$$

Another important task of a suspension system is to provide safety which is related to the dynamic load of the wheels. A potential performance function evaluating the safety of a vehicle is

$$\psi_S = \int_{t^0}^{t^1} (z_Q - z_R)^2\, dt . \qquad (5)$$

Carbody isolation results in soft suspensions, yielding to large relative motions between carbody and wheels. For limiting the relative displacements, a criterion like

$$\psi_D = \int_{t^0}^{t^1} \left(\frac{z_P - z_Q}{s_0}\right)^6 dt \qquad (6)$$

may be used where s_0 is a predefined amplitude following from the vehicle design, which should not be exceeded to much.

Minimizing each of the criteria (4)–(6) individually represents a nonlinear programming problem which may be solved by any general purpose optimization algorithm. In general, the optimal points in the design space P, i.e.

$$\mathbf{p}_C: \min_{\mathbf{p}\in P} \psi_C(\mathbf{p}), \quad \mathbf{p}_S: \min_{\mathbf{p}\in P} \psi_S(\mathbf{p}),$$
$$\mathbf{p}_D: \min_{\mathbf{p}\in P} \psi_D(\mathbf{p}), \qquad (7)$$

will be different from each other. A design where all three criteria have simultaneously minimal values,

$$\psi_C^* = \psi_C(\mathbf{p}_C), \quad \psi_S^* = \psi_S(\mathbf{p}_S),$$
$$\psi_D^* = \psi_D(\mathbf{p}_D), \qquad (8)$$

does not exist in general. E.g., high riding comfort requires a very soft suspension whereas low relative displacement between carbody and wheels can only be achieved by a stiff suspension. Multicriteria optimization offers optimal designs in such conflicting situations, too. An often used method is the weighted objectives method. Instead of the individual criteria a scalar weighted–sum criterion has to be minimized:

$$\psi = w_C \frac{\psi_C}{\psi_C^*} + w_S \frac{\psi_S}{\psi_S^*} + w_D \frac{\psi_D}{\psi_D^*},$$
$$w_C + w_S + w_D = 1. \qquad (9)$$

Then, the multicriteria problem also reduces to a nonlinear programming problem.

The planar model of the van, Fig. 8, consists of 4 rigid bodies. The vector $\mathbf{y}$ summarizing the generalized coordinates reads as

$$\mathbf{y} = (\, y,\ z,\ \alpha,\ \phi,\ w,\ z_S)^T . \qquad (10)$$

The dynamic behavior will be described by a set of twelve first order differential equations based upon the equations of motion. The van is assumed to drive with a constant velocity of 20 m/s over a sinusoidal bump of height 0.1m and length 3m, often found in residential areas as "sleeping policemen". If the control parameter d_A is zero, the suspension is called passive, otherwise it is active.

In the following, the stiffness and damping coefficients of the front and rear suspension, i.e. d_F, d_R, c_F, c_R, the height of the center of gravity h_3, and in case of active suspension the control parameter d_A, are chosen as design variables. They can be summarized in the vector of design variables:

$$p = (d_F , d_R , c_F , c_R , h_3 , d_A)^T . \qquad (11)$$

Optimization of the vehicle has been performed using criteria (9) with different sets of weighting factors w_C, w_S, w_D.

The analysis of conflicting optimization criteria shows that the improvement of one criterion worsens the other criteria. Thus, only a multicriteria approach will give an engineering trade-off, Fig 9.

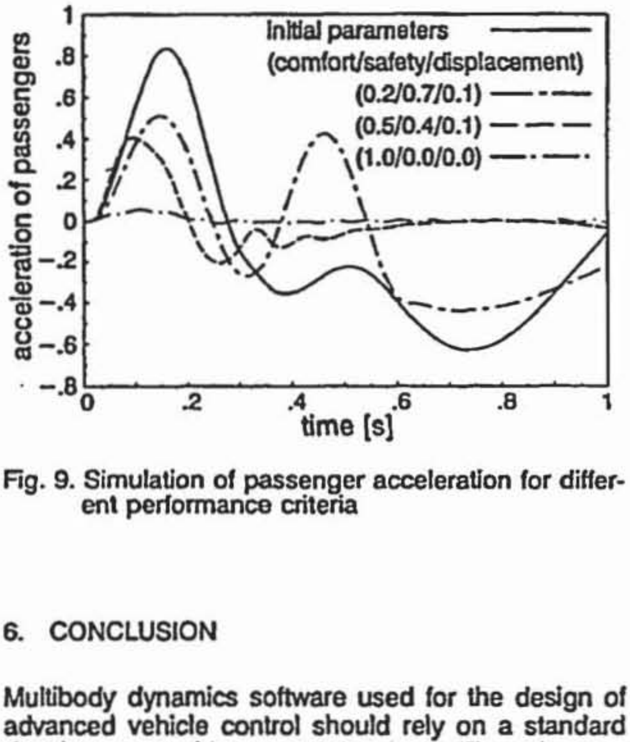


Fig. 9. Simulation of passenger acceleration for different performance criteria

6. CONCLUSION

Multibody dynamics software used for the design of advanced vehicle control should rely on a standard data base to avoid repeated data input. Then, the control design of suspension components as well as the analysis of the performance of the whole vehicle can be concurrently performed.

The programme package NEWEUL uses the multibody system datamodel for the input, the output offers symbolic equations of motion as well as time histories of state variables. It is shown that symbolic equations of motion represent a sound basis for the optimization of control parameters of the active suspension of a van.

REFERENCES

[1] J. Wittenburg: **Dynamics of Systems of Rigid Bodies.** Stuttgart: Teubner 1977.

[2] W. Schiehlen: **Technische Dynamik.** Stuttgart: Teubner 1986.

[3] R.E. Roberson and R. Schwertassek: **Dynamics of Multibody Systems.** Berlin/..., Springer Verlag 1988.

[4] P.E. Nikravesh: **Computer-Aided Analysis of Mechanical Systems.** New Jersey: Prentice-Hall 1988.

[5] E.J. Haug: **Computer Aided Kinematics and Dynamics of Mechanical Systems.** Boston: Allyn and Bacon 1989.

[6] A. Shabana: **Dynamics of Multibody Systems.** New York: Wiley 1989.

[7] W. Schiehlen (ed.): **Multibody Systems Handbook.** Berlin: Springer-Verlag 1990.

[8] E.J. Haug (ed.), **Concurrent Engineering Tools and Technologies for Mechanical System Design.** Berlin/... Springer-Verlag, to appear

[9] W. Schiehlen: **Prospects of the German Multibody System Research Project on Vehicle Dynamics Simulation.** In: Dynamics of Vehicles on road and tracks. Proc. 12th IAVSD Symposium. Amsterdam: Swet and Zeitlinger 1992.

[10] M. Otter, M. Hocke, A. Daberkow, G. Leister: **Ein objektorientiertes Datenmodell zur Beschreibung von Mehrkörpersystemen unter Verwendung von RSYST.** Institutsbericht IB-16. Stuttgart, Institut B für Mechanik, 1990.

[11] U. Lang: **Erstellen von Anwendungsmoduln in RSYST.** Stuttgart, Rechenzentrum der Universität Stuttgart, 1988.

[12] I. Loebich: **Einführung in RSYST.** Stuttgart, Rechenzentrum der Universität Stuttgart, 1988.

[13] R. Rühle et al.: **RSYST Unterprogramm- und Moduldokumentation, Version 3.5.0.** Stuttgart, Rechenzentrum der Universität Stuttgart, 1988.

[14] D. Bestle, P. Eberhard, W. Schiehlen: **Optimization of an Actively Controlled Vehicle System.** Cybernetics Engineering, to appear.