IAVSD MB-Computer Codes, 1993, Page 57-60

MULTIBODY COMPUTER CODES IN VEHICLE SYSTEM DYNAMICS

Edited by

W. Kortüm and R.S. Sharp

Supplement to
Vehicle System Dynamics, Volume 22

SWETS & ZEITLINGER
AMSTERDAM ▮ LISSE ▮ BERWYN, PA
ACADEMIC PUBLISHING DIVISION

BOND GRAPH BASED MODELLING USING MACROS, AN INTRODUCTION TO THE PROGRAM BAMMS.

C.H. Verheul
TNO Road-Vehicles Research Inst.
Department of Vehicle Dynamics
Schoemakerstraat 97
P.O. Box 6033
2600 JA Delft, the Netherlands

H. B. Pacejka
Delft University of Technology
Department of Mech. Engineering
Vehicle Research Laboratory
Mekelweg 2
2628 CD Delft, the Netherlands

This contribution presents a versatile and flexible software package called BAMMS (Bond graph based Algorithm for Modelling Multi body Systems) written in FORTRAN 77. The BAMMS program can be used in various ways to analyze linear as well as non-linear systems. It can perform a time domain simulation or a frequency response analysis and, if desired, it can be used to produce the complete set of equations of a system in any conceivable domain (mechanical, electric etc.) [1]. The method of modelling is described, as well as a number of predefined sub models (macros available in a library for creating a model). The resulting -ready to run- models are simulated with the simulation package included in BAMMS.

1. INTRODUCTION

The bond graph method is a convenient tool for modelling a wide class of systems. The method is based on a graphical representation of the power flow in a system [2]. For simulation purposes, i.e. with the package TUTSIM or the simulation language ACSL, this graphical representation can be translated into a structure table or explicit equations. For relatively small systems this method is often one of the easiest and fastest ways of modelling. However, with larger systems, a increasingly large number of bonds and blocks must be introduced. A few typing errors may result in an unsolvable puzzle very easily. BAMMS has been developed specifically for simplifying the task of modelling, thereby reducing the risk of making errors. The program is based on a number of standard commands and 'user definable' macros. This way it allows for an optimal flexibility as users can accommodate the program to their specific desires. Some highly sophisticated macros as the 'Magic Formulae' have already been introduced and can be used and manipulated.
Furthermore, some extra features have been implemented in BAMMS (i.e. the modelling of kinematic constraints) that are considered a problem when working with bond graphs.

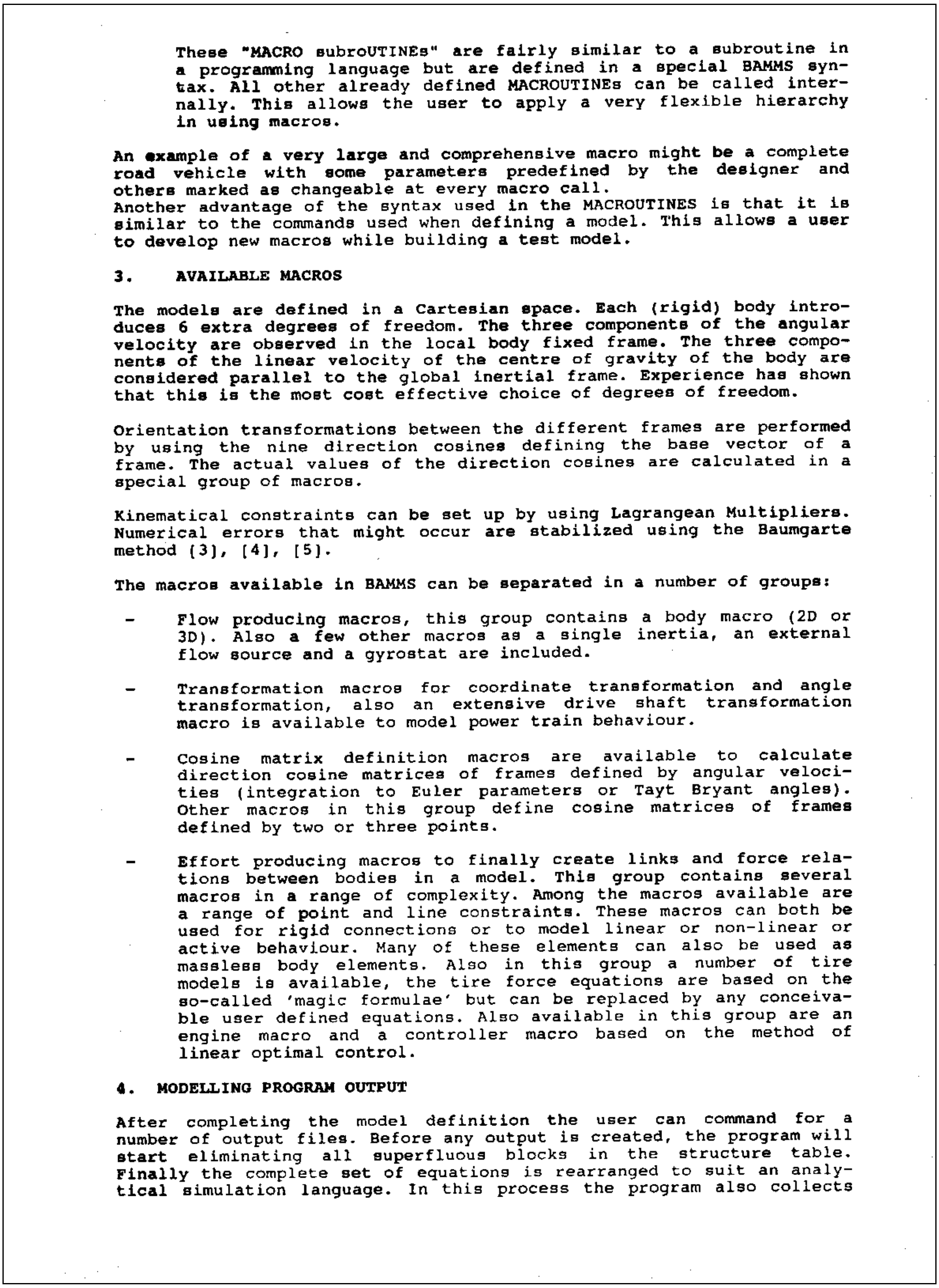
In principle, the BAMMS model is built up by connecting a group of predefined macros. A macro represents a subsystem utilizing bond graph elements and block diagram blocks to a maximum of 500 scalar or vectorial block elements. The complete set of equations for a model is composed by using the macro equations in the input file to create a system structure table. A uniform syntax is developed for interaction with the program and to utilize the macros.

2. GENERAL INFORMATION ON MACROS

The models are generated by connecting several independent macros. These macros have the following characteristics:

1) The subsystem equations are represented by a bond graph and block diagram table. The relations within the subsystem are already transformed into a causal bond graph format. Integral causality is established as much as possible throughout the system.

2) These equations must be copied from the input file to the system structure table and the appropriate connections must be defined. For this purpose the input library also contains MACROUTINES.



These "MACRO subroUTINEs" are fairly similar to a subroutine in a programming language but are defined in a special BAMMS syntax. All other already defined MACROUTINEs can be called internally. This allows the user to apply a very flexible hierarchy in using macros.

An example of a very large and comprehensive macro might be a complete road vehicle with some parameters predefined by the designer and others marked as changeable at every macro call.
Another advantage of the syntax used in the MACROUTINES is that it is similar to the commands used when defining a model. This allows a user to develop new macros while building a test model.

3. AVAILABLE MACROS

The models are defined in a Cartesian space. Each (rigid) body introduces 6 extra degrees of freedom. The three components of the angular velocity are observed in the local body fixed frame. The three components of the linear velocity of the centre of gravity of the body are considered parallel to the global inertial frame. Experience has shown that this is the most cost effective choice of degrees of freedom.

Orientation transformations between the different frames are performed by using the nine direction cosines defining the base vector of a frame. The actual values of the direction cosines are calculated in a special group of macros.

Kinematical constraints can be set up by using Lagrangean Multipliers. Numerical errors that might occur are stabilized using the Baumgarte method [3], [4], [5].

The macros available in BAMMS can be separated in a number of groups:

- Flow producing macros, this group contains a body macro (2D or 3D). Also a few other macros as a single inertia, an external flow source and a gyrostat are included.

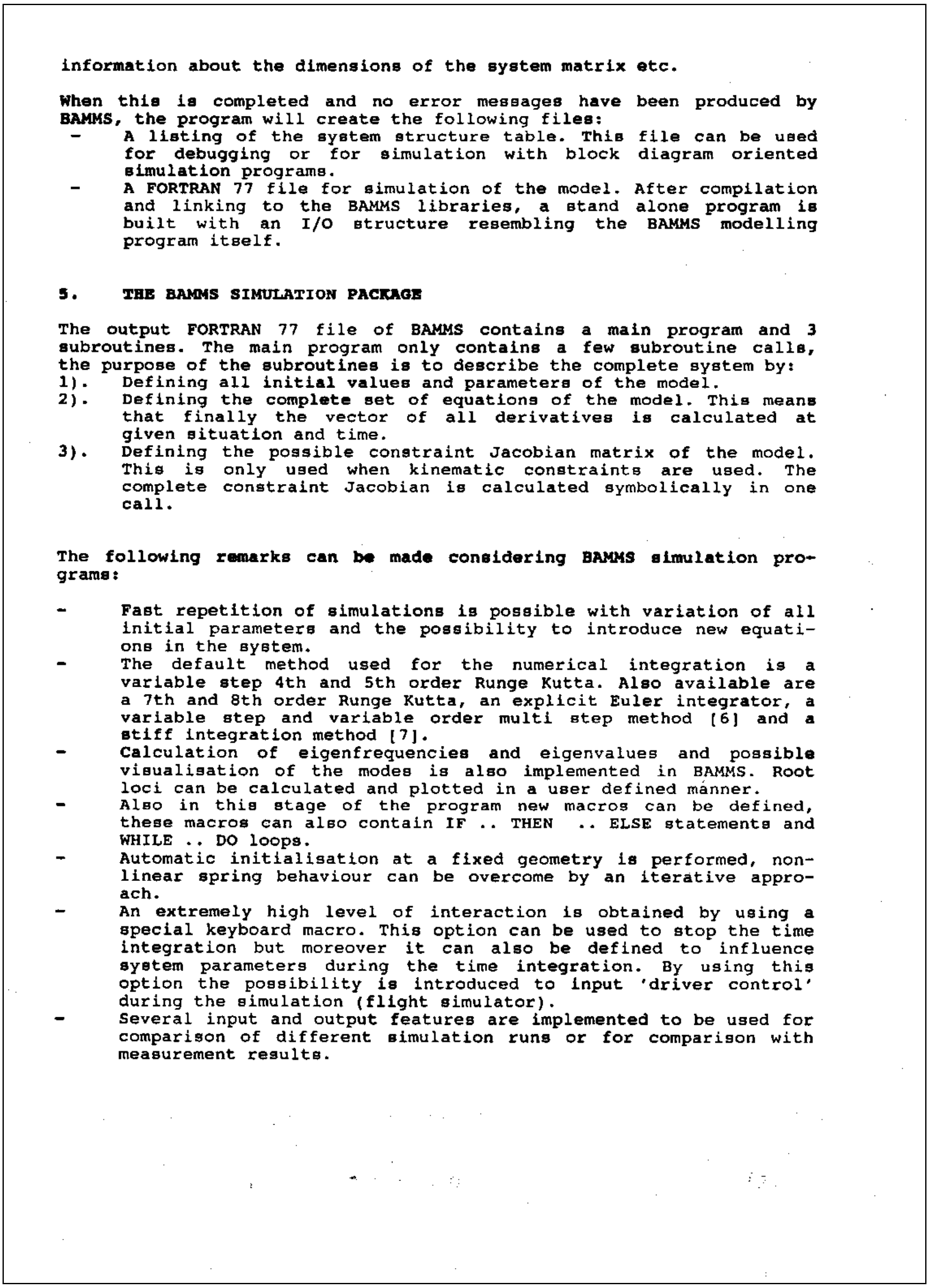
- Transformation macros for coordinate transformation and angle transformation, also an extensive drive shaft transformation macro is available to model power train behaviour.

- Cosine matrix definition macros are available to calculate direction cosine matrices of frames defined by angular velocities (integration to Euler parameters or Tayt Bryant angles). Other macros in this group define cosine matrices of frames defined by two or three points.

- Effort producing macros to finally create links and force relations between bodies in a model. This group contains several macros in a range of complexity. Among the macros available are a range of point and line constraints. These macros can both be used for rigid connections or to model linear or non-linear or active behaviour. Many of these elements can also be used as massless body elements. Also in this group a number of tire models is available, the tire force equations are based on the so-called 'magic formulae' but can be replaced by any conceivable user defined equations. Also available in this group are an engine macro and a controller macro based on the method of linear optimal control.

4. MODELLING PROGRAM OUTPUT

After completing the model definition the user can command for a number of output files. Before any output is created, the program will start eliminating all superfluous blocks in the structure table. Finally the complete set of equations is rearranged to suit an analytical simulation language. In this process the program also collects



information about the dimensions of the system matrix etc.

When this is completed and no error messages have been produced by BAMMS, the program will create the following files:

- A listing of the system structure table. This file can be used for debugging or for simulation with block diagram oriented simulation programs.
- A FORTRAN 77 file for simulation of the model. After compilation and linking to the BAMMS libraries, a stand alone program is built with an I/O structure resembling the BAMMS modelling program itself.

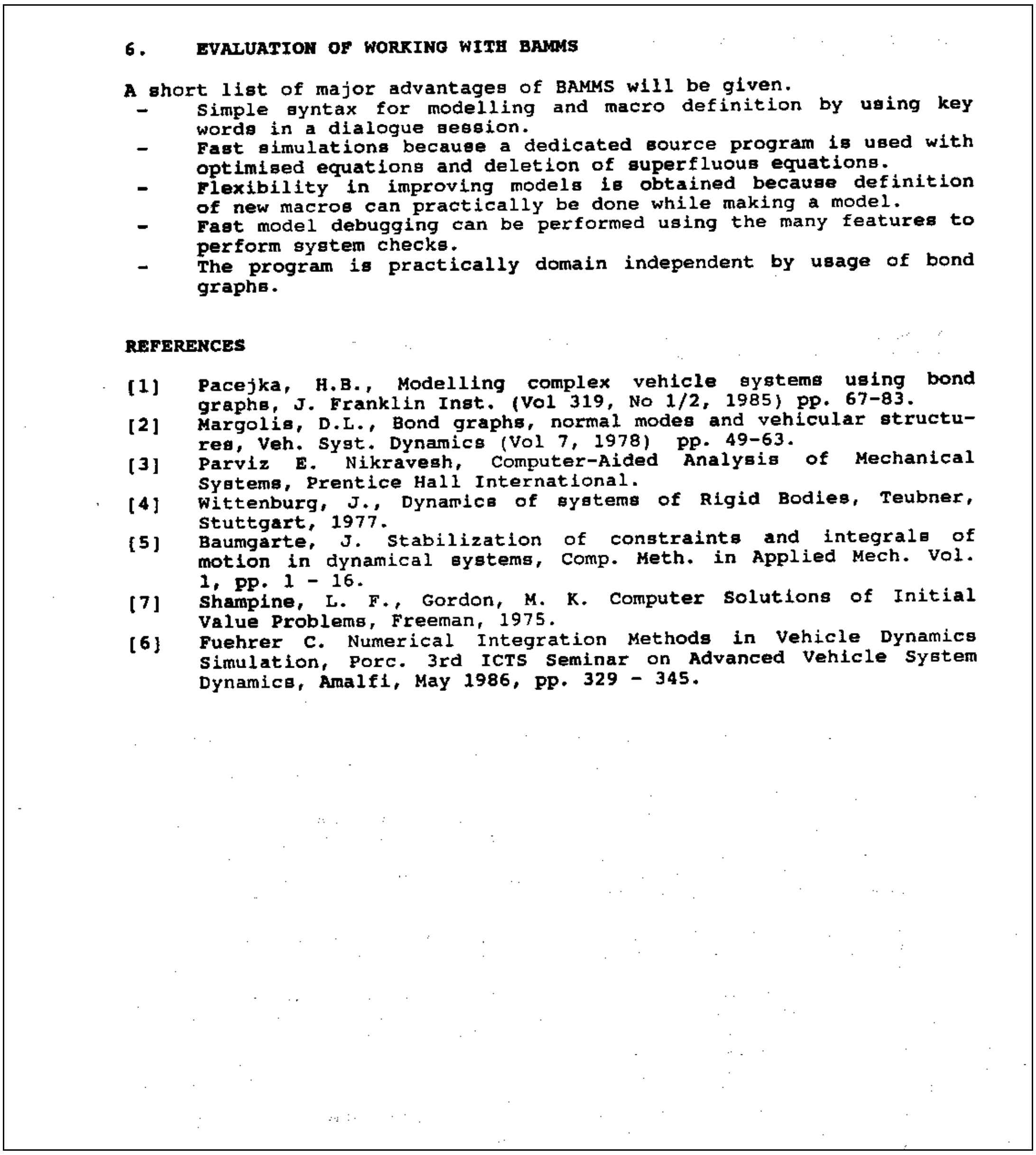
5. THE BAMMS SIMULATION PACKAGE

The output FORTRAN 77 file of BAMMS contains a main program and 3 subroutines. The main program only contains a few subroutine calls, the purpose of the subroutines is to describe the complete system by:

1). Defining all initial values and parameters of the model.
2). Defining the complete set of equations of the model. This means that finally the vector of all derivatives is calculated at given situation and time.
3). Defining the possible constraint Jacobian matrix of the model. This is only used when kinematic constraints are used. The complete constraint Jacobian is calculated symbolically in one call.

The following remarks can be made considering BAMMS simulation programs:

- Fast repetition of simulations is possible with variation of all initial parameters and the possibility to introduce new equations in the system.
- The default method used for the numerical integration is a variable step 4th and 5th order Runge Kutta. Also available are a 7th and 8th order Runge Kutta, an explicit Euler integrator, a variable step and variable order multi step method [6] and a stiff integration method [7].
- Calculation of eigenfrequencies and eigenvalues and possible visualisation of the modes is also implemented in BAMMS. Root loci can be calculated and plotted in a user defined manner.
- Also in this stage of the program new macros can be defined, these macros can also contain IF .. THEN .. ELSE statements and WHILE .. DO loops.
- Automatic initialisation at a fixed geometry is performed, non-linear spring behaviour can be overcome by an iterative approach.
- An extremely high level of interaction is obtained by using a special keyboard macro. This option can be used to stop the time integration but moreover it can also be defined to influence system parameters during the time integration. By using this option the possibility is introduced to input 'driver control' during the simulation (flight simulator).
- Several input and output features are implemented to be used for comparison of different simulation runs or for comparison with measurement results.



6. EVALUATION OF WORKING WITH BAMMS

A short list of major advantages of BAMMS will be given.

- Simple syntax for modelling and macro definition by using key words in a dialogue session.
- Fast simulations because a dedicated source program is used with optimised equations and deletion of superfluous equations.
- Flexibility in improving models is obtained because definition of new macros can practically be done while making a model.
- Fast model debugging can be performed using the many features to perform system checks.
- The program is practically domain independent by usage of bond graphs.

REFERENCES

[1] Pacejka, H.B., Modelling complex vehicle systems using bond graphs, J. Franklin Inst. (Vol 319, No 1/2, 1985) pp. 67-83.

[2] Margolis, D.L., Bond graphs, normal modes and vehicular structures, Veh. Syst. Dynamics (Vol 7, 1978) pp. 49-63.

[3] Parviz E. Nikravesh, Computer-Aided Analysis of Mechanical Systems, Prentice Hall International.

[4] Wittenburg, J., Dynamics of systems of Rigid Bodies, Teubner, Stuttgart, 1977.

[5] Baumgarte, J. Stabilization of constraints and integrals of motion in dynamical systems, Comp. Meth. in Applied Mech. Vol. 1, pp. 1 - 16.

[7] Shampine, L. F., Gordon, M. K. Computer Solutions of Initial Value Problems, Freeman, 1975.

[6] Fuehrer C. Numerical Integration Methods in Vehicle Dynamics Simulation, Porc. 3rd ICTS Seminar on Advanced Vehicle System Dynamics, Amalfi, May 1986, pp. 329 - 345.