

# The DLR FlexibleBodies library to model large motions of beams and of flexible bodies exported from finite element programs

Andreas Heckmann\*, Martin Otter\*, Stefan Dietz<sup>◇</sup> and José Díaz López<sup>‡</sup>  
\*German Aerospace Center (DLR), Institute of Robotics and Mechatronics  
Oberpfaffenhofen, 82234 Wessling, Germany  
<sup>◇</sup>INTEC GmbH, Argelsrieder Feld 13, 82234 Wessling, Germany  
<sup>‡</sup>Dynasim AB, Ideon Research Park, SE-223 70 Lund, Sweden

## Abstract

The new DLR FlexibleBodies library enables and supports the object-oriented and mathematically efficient modelling of flexible bodies as components of multi-body and of arbitrary physical systems. It provides Modelica model classes to model (a) beams and (b) general flexible bodies exported from finite element programs. The motion of a flexible structure is defined by superposition of a in general large, non-linear motion of a reference frame with small elastic deformations. This paper gives an overview on the background, concepts and ideas on which the library is based and how the Modelica user may take advantage of it.

*Keywords:* Flexible body, modal representation, standard input data (SID), floating frame of reference

## 1 Introduction

The DLR FlexibleBodies library is based on the "Standard Input Data of flexible bodies" (SID) which is an object-oriented data structure that was developed at the DLR-Institute of Robotics and Mechatronics to generally describe the properties of elastic bodies, see [13]. It facilitates the use of data which may originate from a finite element description or from continuum models and has been used by various multibody codes, especially by SIMPACK [10], in industrial applications since the early 1990's. The implementation of this general, stable and well-established approach offers new possibilities for multiphysical modelling tasks in Modelica and is nevertheless open for further development and improvements, e.g. concerning multifield problems [6].

## 2 Modelling Capabilities

The DLR FlexibleBodies library is a commercial Modelica package. It provides two basic Modelica model classes: the Beam model and a general ModalBody model, see Figure 1.

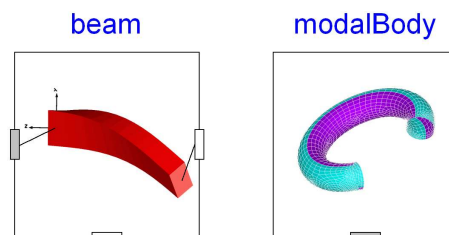


Figure 1: Icons of Beam and ModalBody models.

If a Beam object is instantiated, a dialogue menu, see Fig. 2, supports the definition of the geometrical and physical properties of a straight, homogenous and isotropic beam. For the specification of the cross section an additional menu offers predefined cross section profiles such as tubes, U-pipes or T-beams, see Fig. 3. The parameters defined there are also used for animation purposes. The choice *general* in Fig. 3 enables the direct input of the mechanical essentials, i.e. the geometrical moments of inertia of the cross section.

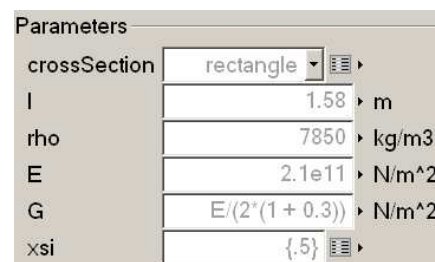


Figure 2: Cutout of the user interface to specify parameters of the Beam model.

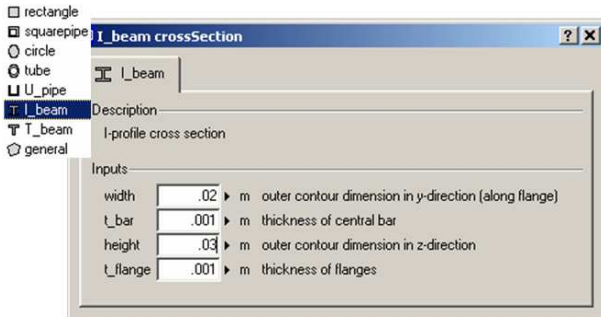


Figure 3: Input for predefined cross section profiles of the Beam model.

The beam model takes bending in two planes, lengthening and torsional deflections into account. The deformations are discretised by eigenforms, i.e. the analytical solutions of the eigenvalue problem of the Euler-Bernoulli beam, see Sec. 3.

Only those eigenforms and eigenfrequencies respectively that are specified by their ordinal numbers are considered in the model. In Fig. 4 the first, third and fifth eigenmode are selected, whereas a damping coefficient of 0.02 is assigned to the first, 0.03 to the second and 0.01 to the third eigenfrequency. This feature gives the experienced modeller the possibility to reduce the number of degrees of freedoms, since some eigenmodes might not contribute to the modelling problem.

Additionally, appropriate boundary conditions that constrain the deformation field of the beam with respect to its floating frame of reference have to be defined. As a general rule, the boundary condition should correspond to the joint where the beam is attached to.

If a ModalBody is instantiated, a file name has to be specified by the user in which the SID structure is present. This SID-file has to be generated in a prepro-

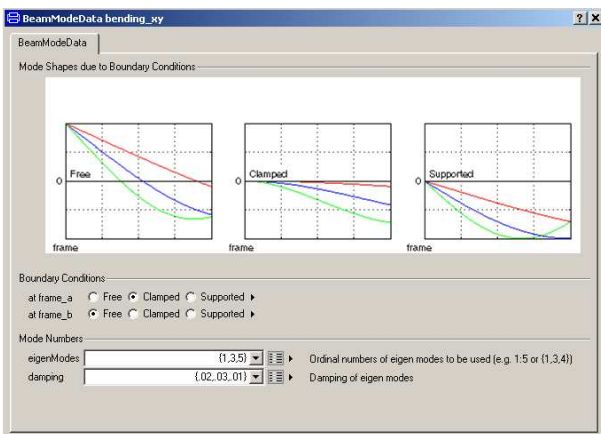


Figure 4: Menu to specify modes of the beam model.

cessing step from finite element data of the body. We recommend the preprocessor FEMBS, see Sec. 4 and [5], from the DLR spin-off company INTEC GmbH that is capable to prepare data from the FE-programs **ABAQUS, ANSYS, MSC.Nastran, NX Nastran, I-DEAS, PERMAS**.

For both the Beam and the ModalBody model the frame connectors of the Modelica.Mechanics.MultiBody library are used to define the connection of the flexible body instance to other system components such as joints, force or sensor elements. Besides the two frames at both ends an arbitrary number of intermediate frames may be defined as parameters of the Beam model. E.g. the parameter  $xsi=\{0.5\}$  in Figure 2 specifies an additional frame at the center of the beam. The ModalBody model has one vector of frame connectors that are associated with nodes of the finite element model.

### 3 Mechanical Background

The mechanical description is based on the floating frame of reference approach, i.e. the absolute position  $\mathbf{r} = \mathbf{r}(\mathbf{c}, t)$  of a specific body particle is subdivided into three parts: the position vector  $\mathbf{r}_R = \mathbf{r}_R(t)$  to the body's reference frame, the initial position of the body particle within the body's reference frame, i.e. the Lagrange coordinate  $\mathbf{c} \neq \mathbf{c}(t)$ , and the elastic displacement  $\mathbf{u}(\mathbf{c}, t)$ :

$$\mathbf{r} = \mathbf{r}_R + \mathbf{c} + \mathbf{u}, \quad (1)$$

where all terms are resolved w.r.t. the body's floating frame of reference ( $R$ ). That's why the angular velocity of the reference frame  $\boldsymbol{\omega}_R$  have to be taken in account when the kinematic quantities velocity  $\mathbf{v}$  and acceleration  $\mathbf{a}$  of a particle are derived:

$$\mathbf{v} = \dot{\boldsymbol{\omega}}_R \mathbf{r} + \dot{\mathbf{r}} = \mathbf{v}_R + \dot{\boldsymbol{\omega}}_R (\mathbf{c} + \mathbf{u}) + \dot{\mathbf{u}}, \quad (2)$$

$$\mathbf{a} = \mathbf{a}_R + (\ddot{\boldsymbol{\omega}}_R + \dot{\boldsymbol{\omega}}_R \boldsymbol{\omega}_R) (\mathbf{c} + \mathbf{u}) + 2\dot{\boldsymbol{\omega}}_R \dot{\mathbf{u}} + \ddot{\mathbf{u}}, \quad (3)$$

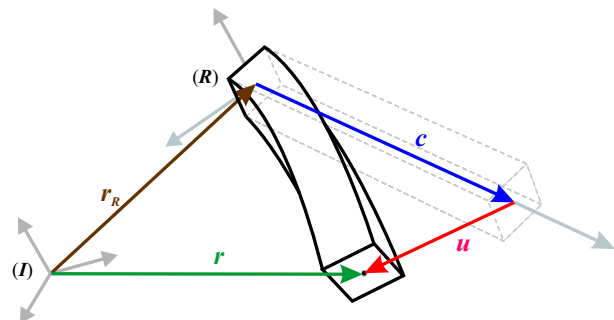


Figure 5: Vector chain of the floating frame of reference.



constants associated to specific boundary conditions, see [12, Ch. V].

To summarise, the Beam model class is based on an analytical continuum description of beams via (8) to (10). Contrary, the ModalBody class is supposed to represent bodies with an arbitrary geometry derived from a finite element description. The FE-model of the body is transformed and utilised in such a way that (1) to (6) also hold for the ModalBody class.

We recommend the use of the pre-processor FEMBS from INTEC GmbH that enables this transformation of FE-data as a reliable and sophisticated process, controlled via its own graphical user interface.

## 4 FEMBS an Interface between MBS and FEA-tools

FEMBS the time-tested interface between the multi-body system code SIMPACK and the most important finite element codes can also be used as a pre-processor for Modelica.

The number of degrees of freedom of multi-body systems is small, when compared with finite element models. Therefore, the main task of flexible body integration is the reduction of the number of degrees of freedom. Currently, this is to be done in two steps.

First, standard reduction schemes like Guyan reduction or Craigh-Bampton reduction are used to export the model from the finite element code into FEMBS's flexible body input file.

Reduction in finite element codes is performed based upon a user defined set of degrees of freedom at nodes which are to be retained. When using the Craigh-Bampton method proper dynamic behaviour of the reduced finite element model can be guaranteed within the frequency range of the application. This additionally requires the specification of a number of so-called dynamic degrees of freedom. They are the eigenmodes of the flexible body, whose retained degrees of freedom were fixed by constraints.

Once the time consuming reduction has been finished the finite element model has less than 1500 degrees of freedom. This first reduction step is to be done in order to keep FEMBS's flexible body input file small. The contents of the *FEMBS* input file are:

- The mass and stiffness matrix of the reduced model.
- the retained nodes

- The eigenmodes which are to be obtained by a modal analysis of the reduced finite element model.
- Geometric stiffening matrices which are to be obtained by extra static analyses of the reduced finite element model. These analyses consider the pre-stresses which are relevant for the application.
- The mesh of finite element model. This data is taken from the FEA input file that contains the elements and nodes of the original, non-reduced model.

Based on this data, the second reduction step [9] in FEMBS is to be performed by selecting modes for multi-body simulation. The user is recommended to select all eigenmodes which correspond to the frequency range which is important for the current application.

Additionally the user should calculate so-called frequency response modes in FEMBS in order to improve the accuracy of the modal representation [3]. Frequency response modes represent local deformations which may occur near the attachment points of the flexible body, where it is connected by force elements, constraints and joints with the surrounding multi-body system.

To generate the frequency response modes (11) in FEMBS the user selects first the nodes and then the directions of the forces and moments which may be transmitted by the force elements or constraints. The frequency response modes  $\mathbf{u}_i$

$$(\mathbf{K} - \Omega^2 \mathbf{M}) \mathbf{u}_i = \mathbf{p}_i \quad (11)$$

are based upon the unit load cases  $\mathbf{p}_i$  for each coupled degree of freedom  $i$ , the mass matrix  $\mathbf{M}$  the stiffness matrix  $\mathbf{K}$  and an excitation frequency  $\Omega$  which is set to the half of the minimum eigenfrequency of interest.

Frequently, only a subset of the frequency response modes has significant influence on the flexible body deformation. The superfluous frequency response modes may be detected by their large frequencies, which follow from a modal analysis which is automatically performed in FEMBS. Thus, FEMBS can automatically select the important frequency response modes based on a cut-off frequency, whose default value is five times greater than the maximum frequency of interest.

As described, the finite element model with say millions of degrees of freedom is transformed to a

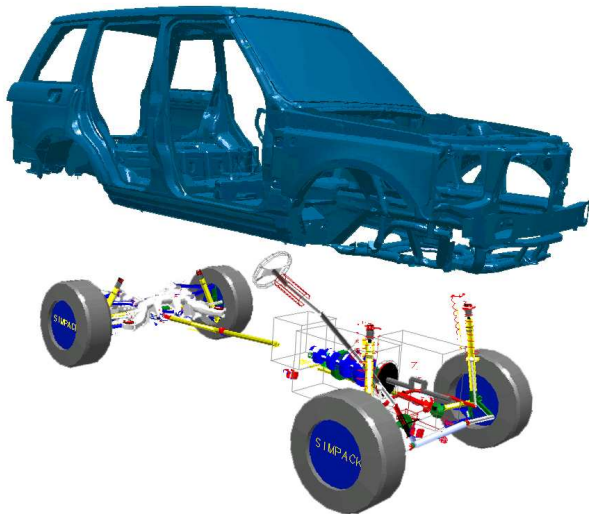


Figure 6: The multi-body system model with the flexible chassis

modal representation of typically about fifty to hundred degrees of freedom. This modal representation is stored in the standard input data file [9] that is input for SIMPACK and Modelica, respectively.

The static deflection of many different flexible bodies such as railway and automotive car bodies, chassis, engine parts, subframes etc. was calculated in the multibody system code SIMPACK. They were compared with the results of static analyses performed in the finite element code based upon the original, non-reduced model. For the static deflection a difference less than one percent can be expected between the results of the finite element model and the multi-body system. Also the eigenfrequencies of the flexible body calculated in SIMPACK are very close to the eigenfrequencies, which are obtained by corresponding finite element analyses.

Thus FEMBS provides efficient and accurate input for multi-body system analyses.

Detailed FEM models of a sport utility vehicle's chassis and also its front and rear subframes were integrated into the multi-body system model which was created using SIMPACK for comfort analyses, see Figure 6. The chassis consisted of about 2.4 millions degrees of freedom. All finite element models were dynamically reduced within NASTRAN. Different sets with 30 to 40 modes consisting of eigenmodes and frequency response modes were used for multi-body system analyses. CAD files of the finite element mesh were also generated by FEMBS and used for the graphical representation.

Fig. 7 shows the good correlation of the model to measured data in the frequency domain up to 25 Hz.

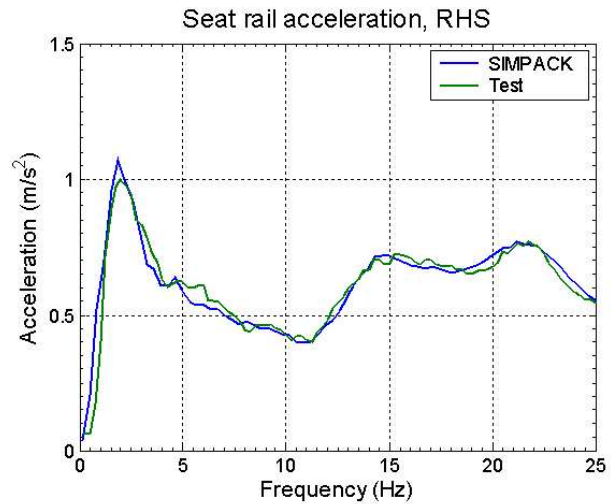


Figure 7: Correlation of the model to measured data: Seat rail acceleration at the right hand side

## 5 Animation of Modal Bodies

In this section, we describe briefly some visualisation aspects of the ModalBody. The visualiser object used in FlexibleBodies library has been designed to meet the specific requirements of flexible bodies animation in MBS context. It can be accessed from a Modelica model via a built-in function provided by the Modelica modelling and simulation environment Dymola [4].

### 5.1 Interpolation scenario

Consider a body  $\Omega$ . The typical animation scenario for  $\Omega$  is depicted in Fig. 8. As stated in (1), the numerical simulation of this modal body provides the position vector  $\mathbf{r}_R$  and the orientation of the body's reference frame as well as the elastic displacements  $\mathbf{u}(\mathbf{c}_i, t)$  of a finite set of nodes  $\mathbf{c}_i$  on the body. The set of space points that can be obtained from this information and setting  $\mathbf{u}(\mathbf{c}, t) \equiv 0$  is called *simulation points* set.

On the other hand, the preprocessor FEMBS provides a Wavefront file [14] that contains the mesh definition of the FE description in the undeformed state with respect to the reference frame of the body. This data is the basis for the flexible body animation. The set of node points of this description is called *animation points* in this section.

The basic interpolation problem is discussed at hand of the simple rectangular plate shown in Fig. 8. The animation set consists of the points in the grid. The simulation points are only the four corners of  $\Omega$ , that is,

$$\mathbf{c}_1 = (0, 0, 0), \mathbf{c}_2 = (1, 0, 0),$$

$$\mathbf{c}_3 = (0, 1.3, 0), \mathbf{c}_4 = (1, 1.3, 0)$$

and we define as deformation field  $\mathbf{u}(\mathbf{c}_i, t)$  the following

$$\mathbf{u}_1 = (0, 0, 0), \mathbf{u}_2 = (0, 0, 0),$$

$$\mathbf{u}_3 = (0, 0, 0), \mathbf{u}_4 = (0, 0, \sin 20\pi t)$$

This is the typical scenario for animation of flexible bodies. The discretisation of a flexible body, in the general case, ends up with a set of nodes such that the error of the FE discretisation is minimised. Then, a modal analysis is performed. For some special geometries, the resulting modes can be deduced in closed form on  $\mathbf{c}$ , but it is a difficult task in general, [9]. That is why in general, each mode is defined in discretised form, i.e. by the related displacements that are given at the node points only.

With these constraints in mind, the visualiser used in FlexibleBodies meets the following design criteria

- The solution of the equations of motion results in the displacements  $\mathbf{u}_i$  at simulation points  $\mathbf{c}_i$  for  $t \geq t_0$ . The visualiser has to determine the absolute position of the animation points whereas the instantaneous positions of the simulation points and the positions the animation points in the undeformed configuration are given.
- The topological information is just provided for the animation points. That is, the simulation point set is just a set of non-structured points in  $\mathcal{R}^3$ .
- The ModalBodyVisualizer uses a special interpolation technique that results in visually appealing images under the assumption that the simulation and animation point sets describe an elastic deformation field. No general interpolation technique (like polynomial interpolation or splines) is used. Instead, an approach inspired by potential theory applied to elasticity is used, [8].

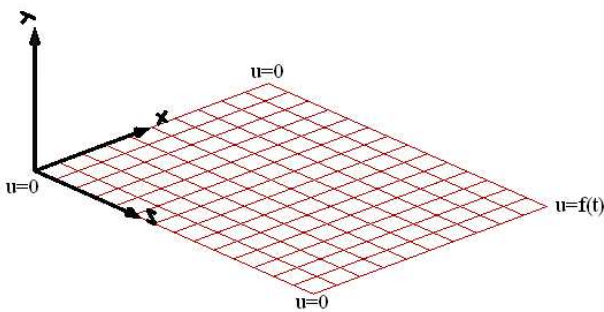


Figure 8: Interpolation problem setup with the rectangle plate

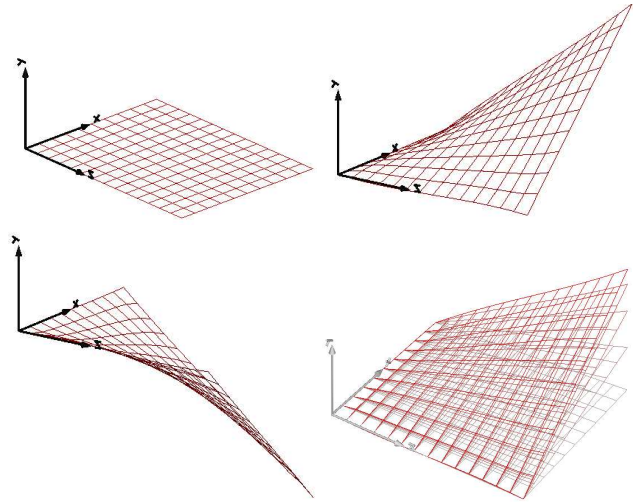


Figure 9: Interpolation result using the ModalBodyVisualizer

### 5.2 ModalBodyVisualizer in FlexibleBodies library

To show how ModalBodyVisualizer works, we applied it to  $\Omega$  in the setup described before. The results are depicted in Fig. 9. The body  $\Omega$  is originally defined in the plane  $xy$ , The outer most right point ( $\mathbf{c}_4$ ) is now moved up and down.

The initial position is depicted in the left upper figure. Then, when the elongation is maximally positive in the  $y$ -axis in the right upper figure. After some time, the deformation is maximally negative, depicted in the left lower figure. Finally, and to give some feeling about the animation, we present a frame with 4 past positions.

In this example, 165 nodes are interpolated from the information in the four corners of the rectangle. More general interpolation techniques are not so suitable for elastic deformation fields. The main reason is the lack of smoothness and loss of connectivity in the single body, making artificial cracks, peaks or weird artifacts to be shown as part of the simulation.

Taking just the simulation point set for animation can make hard to imagine the body behaviour, so a larger set of animation points is needed for good visualisation.

## 6 Example Models

The DLR FlexibleBodies library contains several examples that demonstrate the use of the provided capabilities.

## 6.1 Slider Crank Model

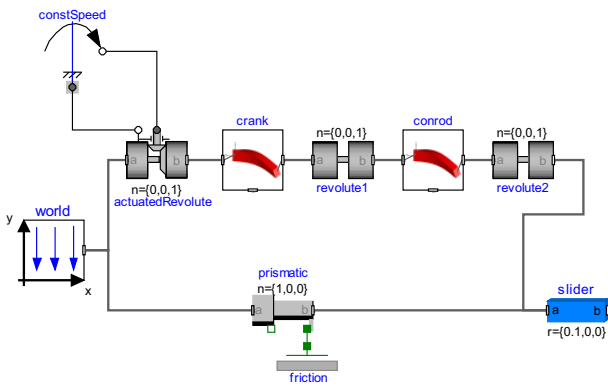


Figure 10: Diagram layer of the slider crank model.

One of these examples is a model of a slider crank mechanism with two beam instances that represent the crank and the conrod. The crank rotates with constant angular velocity.

Gravity forces are applied to all bodies of the system and an additional Coulomb friction force acts on the slider that has one translational degree of freedom.

The bending behaviour of the conrod is represented by one mode that is related to 1 Hz eigenfrequency, while the crank bending mode corresponds to 3.9 Hz eigenfrequency. The model aligns with an example from literature and may therefore also be used for verification purposes [9, Sec. 6.5.5].

Figure 10 depicts the graphical set-up of this model in Modelica. The model 3D animation is shown in Figure 11, where the menu option to show an additional exaggerated displacement field beside the exact in-scale deformations is activated.

Fig. 12 shows the simulation results of this plane, closed-loop mechanism with discontinuously applied friction force. For the integration of the 10-s-scenario 0.671 CPU-s were spent with  $10^{-6}$  integration tolerance on a 1.6 GHz Intel Pentium Laptop with WindowsXP.

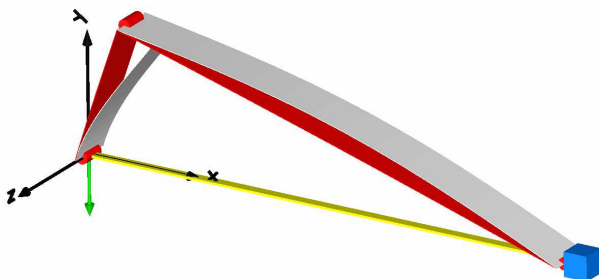


Figure 11: Animation of the slider crank model. The grey animations are scaled versions of the red animations and exaggerate the deformations of the beams.

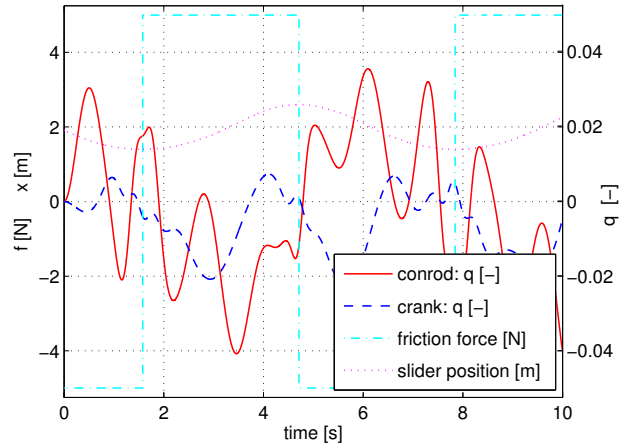


Figure 12: Slider Crank simulation results.

## 6.2 Buckling of a beam

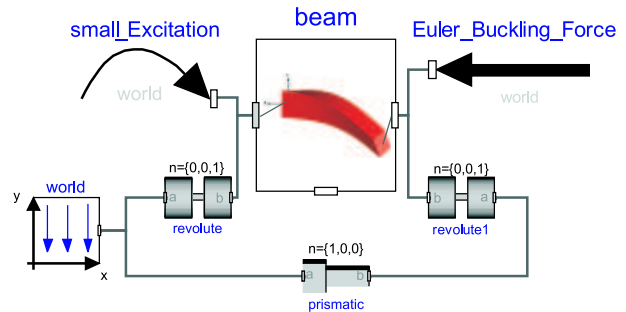


Figure 13: Set-up of the model Beam Buckling.

Considered is a bending beam with 18.8 Hz eigenfrequency that is supported but not clamped on both ends. At one end an external force is applied and is increased linear in time until the classical Euler buckling force (in this case: 16.6 kN ) is reached. This is the scenario of the model Beam Buckling of Fig. 13. It is supposed to document that the present approach is capable to cover the bending behaviour of the beam until buckling occurs, see Fig. 14. This requires to include stress stiffening terms in (6) that originate from the second order displacement field description, here.

Consider the physical force vector  $\mathbf{f} \in \mathbb{R}^3$  that is applied on the structure at the point  $\mathbf{c}_f$ . The equations of the model Beam Buckling then get the form of (7) whereas  $\mathbf{f}_q$  has to be defined using (4) and the Jacobian  $\mathbf{J}$ :

$$\mathbf{f}_q = \mathbf{J}^T \mathbf{f} \quad (12)$$

$$\text{with } \mathbf{J} := \frac{\partial \mathbf{u}(\mathbf{c}_f, t)}{\partial \mathbf{q}} = \Phi_{\mathbf{c}_f} + \begin{bmatrix} \mathbf{q}^T \Phi_x \\ \mathbf{q}^T \Phi_y \\ \mathbf{q}^T \Phi_z \end{bmatrix}_{\mathbf{c}_f} .$$

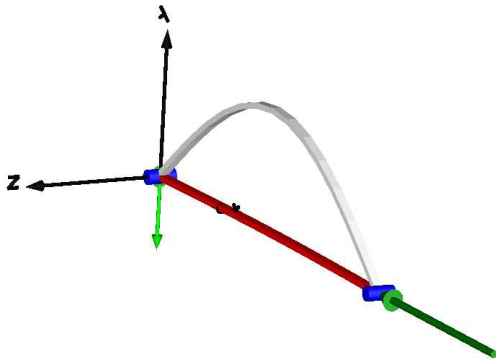


Figure 14: Animation of the model Beam Buckling in the instant the bifurcation occurs.

Consequently, an increasing force  $f$  may be additionally amplified by the increasing deflection  $q$ , so that the stability of the system may be affected. This is in particular the case for axial loads on slender beams as it is given in this model.

Since the deformation behaviour is not determined beyond the bifurcation point, an additional small harmonic excitation torque with 1 Nm and 1 Hz is applied at the other end at the beam. This is necessary to ensure that the buckling actually occurs and can be animated as it is done in Fig. 14.

The results of the simulation are shown in Fig. 15. At the beginning the harmonic excitation hardly influences the state of the beam. With increasing thrust force the bending behaviour is weakened until the buckling occurs after about 10 s. These results perfectly correspond to the theoretical prediction. The nonlinear characteristics of the beam are reproduced until buckling. However, it should be pointed out that beyond this point the model is no more valid.

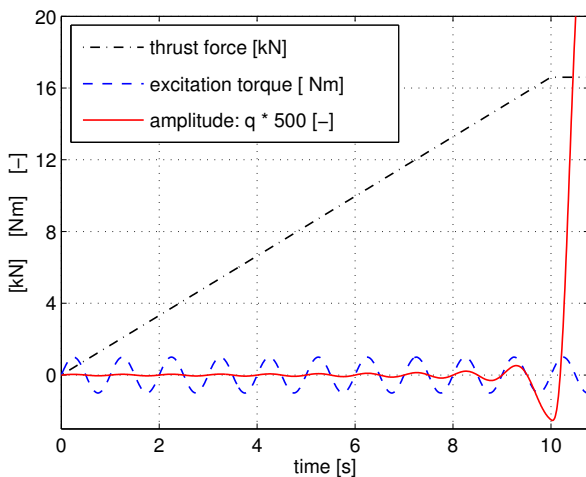


Figure 15: Simulation results: Buckling of a beam.

### 6.3 Helicopter Rotor

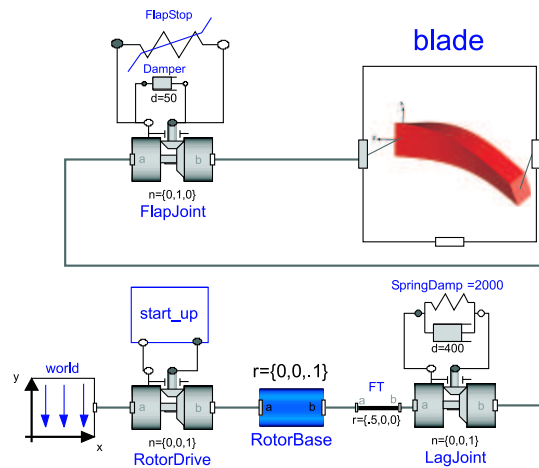


Figure 16: Set-up of the model Helicopter Rotor.

The model Helicopter Rotor in Fig. 16 mainly consists of a rheonom driven, cylindrical rotor base, two joints and one blade. The rotor base rotates around its cylinder axis that coincides with the global  $z$ -axis, while the lag joint allows for a rotation around the local  $z$ -axis at the outer radius of the rotor base. The flap joint defines a angular motion around the local  $y$ -axis at the circumference of the rotor base.

In its initial state the rotor base does not move and the flap stop, a bump stop modelled as a nonlinear spring, applies the torque to counterbalance the gravity of the blade. A linear spring-damper element actuates according to the state of the lag joint. The 6 m long blade is modelled as a flexible beam with 7 bending modes in its  $xz$ -plane and 2 bending modes in its  $xy$ -plane so that a frequency range up to 270 Hz is covered.

Fig. 17 visualises the initial, static deformation state of the blade which is dominated by the first  $xz$ -

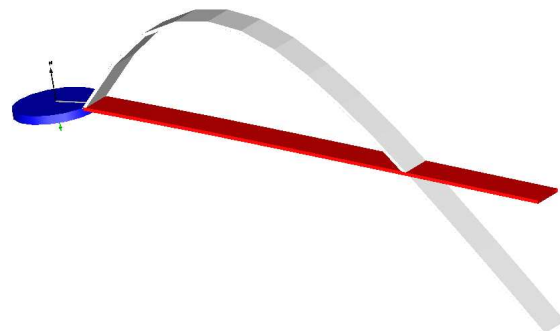


Figure 17: 3D-View: Helicopter Rotor in its initial state with red in-scale and grey exaggerated deflection.



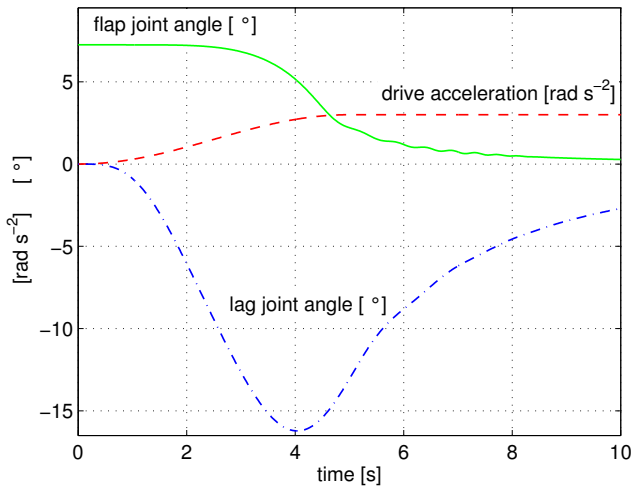


Figure 18: Time plot of joint variables of the model Helicopter Rotor.

bending mode at 7.9 Hz. The design, the geometrical and physical parameters of the model correspond to a typical helicopter configuration of the 1960's, see [7].

The result plot Fig. 18 shows the applied, smooth rotor drive acceleration of the simulated start-up scenario. The flap joint angle tends against zero with increasing rotor rotation so that the blade moves towards its vertical alignment. The instantaneous lag joint angle is associated to the torque applied by the linear spring-damper element which transmits the drive torque to the blade.

Fig. 19 presents the dominance of 7.9 Hz mode at the initial static configuration. The plot of  $xy$ -plane bending modes clearly correspond to the lag joint plot which is again due to the torque applied by the lin-

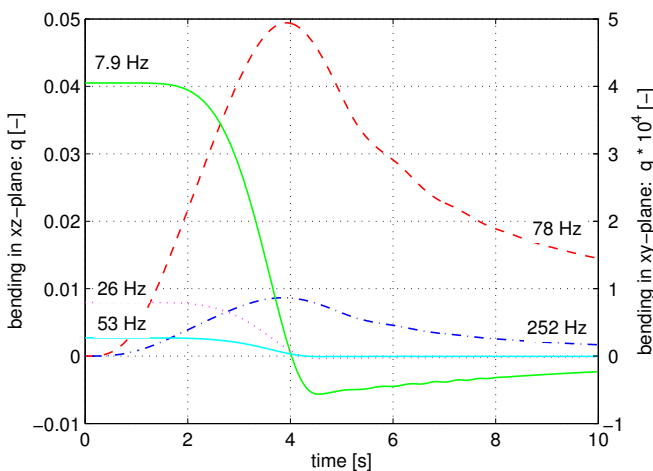


Figure 19: Plot of modal deformation amplitudes  $q(t)$  for 2 bending modes in  $xy$ -plane (78 Hz, 252 Hz) and 3 bending modes in  $xz$ -plane (7.9 Hz, 26 Hz, 53 Hz).

ear spring-damper element. Generally, all modal amplitudes decrease with the increasing angular velocity and clarify its stabilising influence. Note that the stabilising effect again relies on the second order description (4). Neglecting these terms would lead to completely wrong results for this scenario.

## 6.4 Piston Rod

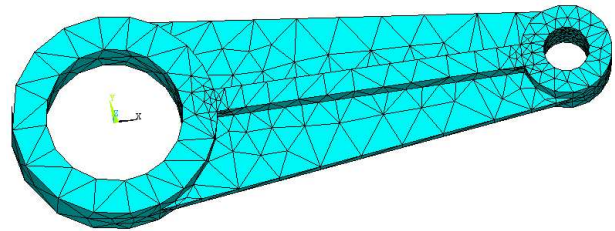


Figure 20: ANSYS finite element model of a piston rod.

In order to demonstrate the use of DLR FlexibleBodies library with body data that originate from finite element data, a simplified piston rod was modelled in Ansys with 2.788 tetrahedral elements of type Solid95 [1] and 15.492 degrees of freedom, see Fig. 20.

The model was preprocessed in FEMBS and three eigenmodes with 809 Hz, 1040 Hz and 1244 Hz eigenfrequency were selected to represent the deformation field of the piston rod on the multibody side. 97 of the

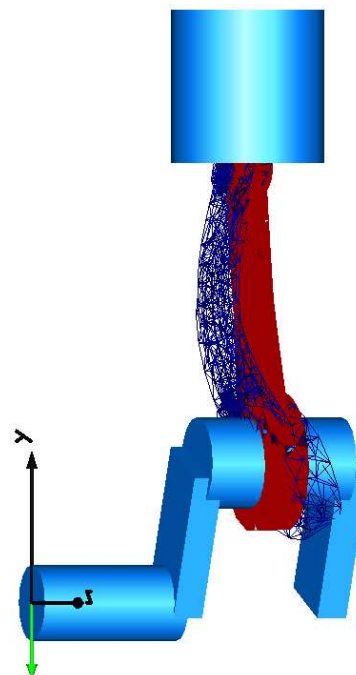


Figure 21: Modelica 3D-View of one cylinder combustion engine with flexible piston rod.

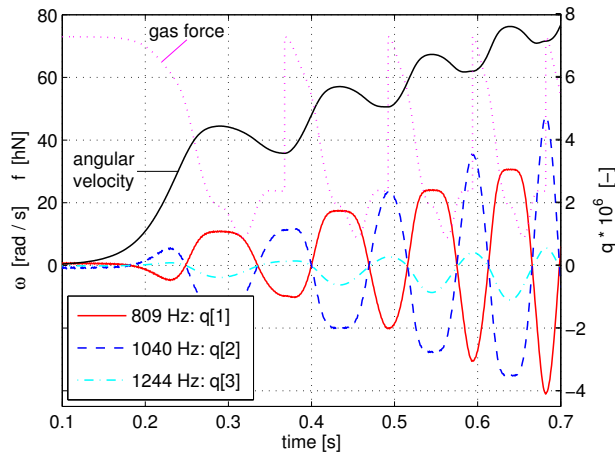


Figure 22: one-cylinder-combustion engine: simulation results.

5164 finite element nodes were selected in *FEMBS* so that the animation scheme has to map the motion of 97 simulation nodes on 5164 animation nodes.

Fig. 21 gives a 3D-View of a one cylinder combustion engine where the flexible piston rod is assembled. The dark-red, solid structure depicts the unscaled deformation state while the blue mesh exaggerate the deflections of the piston rod.

A start-up manoeuvre of this engine has been simulated applying a simplified gas force and no load. The plot of the gas force and the increase of the angular velocity of the crank shaft are given in Fig. 22. For each of the eigenmodes the related modal amplitude as function of time is plotted there as well.

## 7 Conclusion

This paper presents the new DLR FlexibleBodies library, that is based on the “Standard Input Data” (SID) file format for flexible bodies. The library provides the intrinsic capability to generate this data for beam-like bodies. For more general bodies, this data has to be supplied by an external tool on the basis of an available finite element model. This preprocessing is performed using the *FEMBS*-Software of the *INTEC* GmbH in Wessling, Germany, that supports all major FE programs.

The animation of general flexible bodies is based on new features provided in *Dymola*. Especially, wavefront files are supported in *Dymola* and a sophisticated algorithm is included to map the movement of simulation to animation nodes.

Furthermore, the mechanical background and the graphical user interface of the FlexibleBodies library

have been presented. In several application examples, the usage of the model classes *Beam* and *ModalBody* have been demonstrated.

As a final conclusion it may be stated that this library opens new chances for the set-up of multibody models in *Modelica* since flexible components may now be included and so that future application fields are numerous.

## 8 Acknowledgements

A first preliminary version of the *ModalBody* model was implemented by Gerhard Schillhuber with advice by Professor Dr. Oskar Wallrapp.

We would like to thank Hans Olsson from *Dynasim* for helping with some implementation details.

This work was partly funded by the Bayerisches Staatsministerium für Wirtschaft, Verkehr und Technologie which supported the Conceptual Design Laboratory (CDL) project. The authors are grateful for this support.

## References

- [1] ANSYS® Release 7.1 Theory Reference, ANSYS, Inc., Canonsburg, USA, 2003.
- [2] H. Bremer and F. Pfeiffer: *Elastische Mehrkörpersysteme*, Teubner-Verlag, Stuttgart, 1992.
- [3] S. Dietz: *Vibration and Fatigue Analysis of Vehicle Systems Using Component Modes*, Fortschritt-Berichte VDI Reihe 12, Nr. 401, VDI-Verlag Düsseldorf, 1999.
- [4] *Dynasim: Dymola Version 6*. Dynasim AB, Lund, Sweden, 2006, <http://www.dynasim.se/>
- [5] *FEMBS*: [http://www.simpack.com/downloads/pdf/datasheet\\_fembs.pdf](http://www.simpack.com/downloads/pdf/datasheet_fembs.pdf)
- [6] A. Heckmann. *The modal multifield approach in multibody dynamics*. Fortschritt-Berichte VDI Reihe 20, Nr. 398. VDI-Verlag, Düsseldorf, 2005.
- [7] W. Just: *Hubschrauber und Vertikalstartflugzeuge*, Verlag Flugtechnik Stuttgart, Stuttgart, 1963.
- [8] Perdomo, Y. *Reproducing kernels and Potential theory for the Bergman Spaces* Doctorate thesis in Mathematics, Lund University, 2004.

- [9] R. Schwertassek and O. Wallrapp: Dynamik flexibler Mehrkörpersysteme, Vieweg Verlag, Braunschweig, 1999.
- [10] SIMPACK: <http://www.simpack.com>.
- [11] A.A. Shabana: Dynamics of Multibody Systems, Cambridge University Press, Cambridge, 2nd ed., 1998.
- [12] S. Timoshenko: Vibration Problems in Engineering. D. Van Nostrand, Princeton, 1955.
- [13] O. Wallrapp: Standardization of Flexible Body Modeling in Multibody System Codes, Part 1: Definition of Standard Input Data, Mechanics of Structures and Machines 22(3):283-304, 1994.
- [14] Wavefront: <http://www.fileformat.info/format/wavefrontobj/egff.htm>.