A modular and efficient approach to computational modeling and sensitivity analysis of robot and human motion dynamics

Martin Friedmann[1,*], **Janis Wojtusch**[1,**], and **Oskar von Stryk**[1,***]

[1] Technische Universität Darmstadt, Department of Computer Science, Hochschulstr. 10, 64289 Darmstadt, Germany

In this paper a new class library for the computation of the forward multi-body-system (MBS) dynamics of robots and biomechanical models of human motion is presented. By the developed modular modeling approach the library can be flexibly extended by specific modeling elements like joints with specific geometry or different muscle models and thus can be applied efficiently for a number of dynamic simulation and optimization problems. The library not only provides several methods for solving the forward dynamics problem (like articulated body or composite rigid body algorithms) which can transparently be exchanged. Moreover, the numerical solution of optimal control problems, like in the forward dynamics optimization of human motion, is significantly facilitated by the computation of the sensitivity matrix with respect to the control variables. Examples are given to demonstrate the efficiency of the approach.

1 Introduction

In this paper `MBSlib`, an object-oriented library for the efficient calculation of the forward dynamics of multi-body systems (MBS) written in C++ is presented. Following the discussion of the applied modeling methodology the provided methods for the calculation of MBS forward dynamics and sensitivities is presented. The paper closes with application examples.

2 Modeling of multi-body-systems and drives

A modular approach for the modeling has been chosen to allow for an easy extension of the library. MBS are modeled as a tree consistig of the basic modeling elements fixed or free base, fixed translation and rotation, variable translations and revolute joints, forks, and endpoints. Additionally, custom modeling elements like nonlinear springs, Hill-type muscles, or specific joints can be created and added to the library. Similar approaches to the modeling of MBS have been applied successfully to industrial robots [1], biomechanical systems [2] and autonomous robots [3].

To control the motion of the MBS, models of drives providing forces can be attached to the joints or the endpoints. Muscles and springs are modeled as polygons that are in contact with two or more endpoints of the MBS. Forces acting along these polygons are transformed into forces at these contact points. Note that even though the dependency of this transformation is highly non-linear w. r. t. the state of the model, the transformation itself is a linear one. Arbitrary functions can be integrated to calculate the forces along the polygons or in the joints. These functions may depend on a vector $\mathbf{u} \in \mathbb{R}^{\mathbf{m}}$ of control variables as well as state variables.

Fig. 1 shows the model structure of a pendulum with return spring. The pendulum is connected to a fixed base and comprises a revolute joint, a rigid link, and an endpoint. The anchor point of the return spring consists of an endpoint and is shifted by a fixed translation. The return spring is connected with both endpoints.

3 Calculation of forward dynamics and sensitivities

The motion dynamics of an MBS is described by the well known equation

$$\tau = M \cdot \ddot{\mathbf{q}} + C(\mathbf{q}, \dot{\mathbf{q}}) + G(\mathbf{q}) + \mathbf{F}_{ext} \tag{1}$$

with $\mathbf{q}$ being the vector of the joint positions, τ being the forces resp. torques acting in the joints and M being the mass matrix. $\mathbf{F}_{ext}$, $\mathbf{G}$ and $\mathbf{C}$ are additional torques resp. forces in the joints resulting from external forces, gravitation and Coriolis forces. Solving this equation for $\ddot{\mathbf{q}}$ is known as the calculation of the MBS forward dynamics. `MBSlib` provides two well known methods for solving the forward dynamics. Method one uses the Composit Rigid Body Algorithm (CRBA) [4, 5] to explicitly calculate M and subsequently solves the linear system of equations, leading to an $O(n^3)$ runtime for a system with n joints. Method two uses the Articulated Body Algorithm [6], which has $O(n)$ runtime. In [5] it is shown, that method

* email friedmann@sim.tu-darmstadt.de
** email wojtusch@sim.tu-darmstadt.de
*** email stryk@sim.tu-darmstadt.de

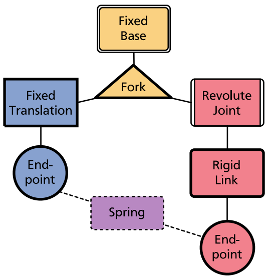

Fig. 1 Structure of a pendulum with return spring.

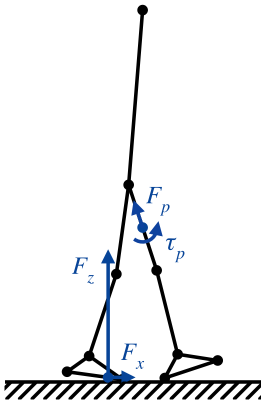

Fig. 2 Model of the human locomotor system.

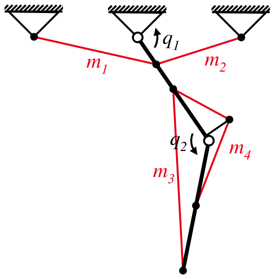

Fig. 3 Model of a robot with muscle-type actuation.

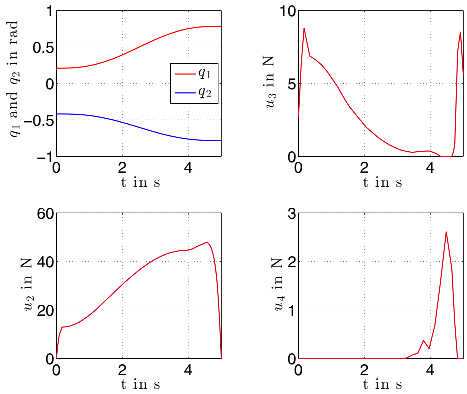

Fig. 4 Optimized control variables for desired joint trajectories.

two nevertheless has a worse runtime than method one for systems with less than 9 joints. In `MBSlib` the methods can be interchanged transparently to allow the use of the better performing method for any MBS.

Two methods are provided to calculate the derivatives of $\ddot{\mathbf{q}}$ w. r. t. the control variables $\mathbf{u}$. The first method calculates the derivatives of $\ddot{\mathbf{q}}$ w. r. t. each component u_i of $\mathbf{u}$ separately. Each of these calculations makes use of the ABA, leading to an $O(n \cdot m)$ runtime for the calculation of the whole sensitivity matrix. The second method uses automated differentiation based on the ADOL_C library[1]. This allows for the calculation of sensitivities w. r. t. to the control variables as well as w. r. t. any modeling parameter, thus allowing further applications like parameter estimation for a model. The drawback of this method is, that the calculation of the derivatives is much slower, as ADOL_C is based on a special floating-point type which allows to keep a record of calculations and afterwards interprets this record in order to calculate derivatives.

4 Examples

The capability of `MBSlib` is illustrated by two biomechanical application examples featuring the modular modeling approach in combination with efficient motion dynamics computation procedures.

The first example is an inverse dynamic simulation of the human gait with a transfemoral prosthesis in order to estimate the torque τ_p and normal force F_p at the user-prosthesis-interface in sagittal plane. Fig. 2 shows the applied two-dimensional model of the human locomotor system. The model consists of eight rigid links representing the body segments and seven revolute joints representing ankle, knee, hip, and sacroiliac joints. The model is actuated by given joint trajectories and ground reaction forces F_x and F_z [7].

The second example is a forward dynamic optimization to find energy-minimal muscle control variables u for a robot with muscle-type actuation. Fig. 3 shows the applied robot model comprising two rigid links, two revolute joints q_1 and q_2, as well as four muscles m_1 to m_4. The optimal control variables are computed for desired joint trajectories by applying the direct collocation method for the numerical solution of optimal control problems `DIRCOL` [8]. The desired trajectories and the computed optimal control variables are plotted in Fig. 4. Muscle m_1 does not participate in the motion and is not displayed.

`MBSlib` is a major step towards modular and efficient modeling and sensitivity analysis of MBS. The two given application examples demonstrate the computational abilities and the applicability for tree-structured, biomechanical systems and robots.

Acknowledgements The research presented in this paper was supported by the German Research Foundation DFG under grant STR533/8-1.

References

[1] R. Höpler, A unifying object-oriented methodology to consolidate multibody dynamics computations in robot control, PhD thesis, TU Darmstadt, 2004.
[2] M. Stelzer and O. von Stryk, ZAMM, Journal of Applied Mathematics and Mechanics **86**(10), 828–840 (2006).
[3] M. Friedmann, Simulation of Autonomous Robot Teams With Adaptable Levels of Abstraction, PhD thesis, TU Darmstadt, 2009.
[4] M. W. Walker and D. E. Orin, Journal of Dynamic Systems, Measurement, and Control **104**, 205–211 (1982).
[5] R. Featherstone, Rigid Body Dynamics Algorithms (Springer, 2008).
[6] R. Featherstone, Intl. J. of Robotics Res. **2**(1), 13–30 (1983).
[7] S. W. Lipfert, Kinematic and dynamic similarities between walking and running, PhD thesis, Friedrich-Schiller Universität Jena, 2010.
[8] O. von Stryk, Numerische Lösung optimaler Steuerungsprobleme: Diskretisierung, Parameteroptimierung und Berechnung der adjungierten Variablen, PhD thesis, TU München, 1994.

[1] http://www.coin-or.org/projects/ADOL-C.xml