ECCOMAS Multibody Dynamics 2013
1-4 July, 2013, University of Zagreb, Croatia

A Library of Computational Benchmark Problems for the Multibody Dynamics Community

Ramin Masoudi[1], Thomas Uchida[2], David Vilela[3], Alberto Luaces[3], Javier Cuadrado[3], John McPhee[1]

[1] Department of Systems Design Engineering, University of Waterloo, 200 University Avenue West, Waterloo, Ontario, N2L 3G1, Canada: rmasoudi@uwaterloo.ca,mcphee@real.uwaterloo.ca

[2] Department of Bioengineering, Stanford University, 318 Campus Drive, James H. Clark Center, Stanford, CA 94305-5448, U.S.A.: tkuchida@stanford.edu

[3] Laboratorio de Ingeniería Mecánica, University of La Coruña, Escuela Politécnica Superior, Mendizábal s/n, 15403 Ferrol, Spain: {david.vilela,aluaces}@udc.es,javicuad@cdf.udc.es

Abstract

The objective of this work is to establish an online Library of Computational Benchmark Problems designed and solved by multibody dynamics researchers. This library will serve as a comprehensive reference for current and future generations of researchers and students. By sharing our computational experience, we will be aptly equipped to determine the most appropriate approach (model, formulation method, numerical procedure, software implementation, etc.) for a given problem. By including a wide variety of problems, from the didactic planar slider-crank mechanism to complex biomechanical models, our hope is that the library will become a useful resource for all members of our global research community, facilitating collaboration and cross-pollination of ideas. A prototype library has been designed with a comprehensive classification scheme that allows users to quickly identify benchmark problems related to their area of specialization. The capability to upload new simulation results for existing problems, propose new benchmarks, and search the database have been incorporated. The prototype is available at http://real.uwaterloo.ca/benchmarks/. Our hope is that the library will become a tool that supports the research of multibody dynamicists worldwide, helping to advance the state-of-the-art in our ever-growing field.

Keywords: *computational benchmark, library, mechanical system, multibody dynamics, performance comparison, repository*

1 Introduction

Since the time of Sir Isaac Newton, dynamics has been a topic of relevance to an increasing number of scientific and engineering disciplines, and is used to study the motion of objects from the very small (e.g., molecular dynamics) to the very large (e.g., celestial mechanics). While Newton's three laws of motion described the dynamics of particles (moving at nonrelativistic speeds) [1], Euler and Lagrange introduced their masterpieces *Nova methodus motum corporum rigidarum determinandi* [2] and *Mécanique Analytique* [3] on the systematic formalism of constrained multibody systems, by which they are acknowledged as the pioneers of a sophisticated branch of applied science called *multibody dynamics*.

Multibody dynamics has emerged as a challenging branch of mechanics dealing with the motion of interconnected rigid or flexible bodies and the forces that are responsible for this motion. In a systematic treatment of a multibody system, the motion of bodies (*kinematics*) and the dynamic effect of forces acting between components of the system (*kinetics*) must both be considered. Depending on the problem, the kinematic analysis may be done separately or performed implicitly during a dynamic analysis. Multibody dynamics has been applied to the study of a wide variety of systems, including machines, mechanisms, vehicles, robots, and humans. With this variety of applications comes a variety of approaches to the systematic modeling and simulation of multibody systems, a diversity that motivates the need to maintain close interaction and collaboration among the members of the multibody community.

The objective of this work is to establish an online library of computational benchmark problems designed and solved by multibody dynamics researchers. This library will serve as a comprehensive reference for current and future generations of researchers and students. By sharing our computational experience, we will be aptly equipped to determine the most appropriate approach (model, formulation method, numerical procedure, software implementation, etc.) for a given problem. Benchmarks have already been established for evaluating computational performance in several areas (e.g., kinematics and dynamics [4, 5], contact dynamics [6], vehicle dynamics [7, 8], and flexible multibody systems [9]), but these problems have not been collected in a single location. By including a wide variety of problems, from the didactic planar slider-crank mechanism to complex spatial biomechanical models, our hope is that the library will become a useful resource for all members of our global research community, facilitating collaboration and cross-pollination of ideas.

The *Library of Computational Benchmark Problems* provides a collaborative means of comparing and evaluating different multibody system formalisms, solution approaches, and computational issues, along with examining the performance of various software packages. A general framework for classifying multibody systems and simulation procedures—from modeling approaches to numerical methods—will be discussed in Section 2. The principles of evaluating the efficacy of a multibody dynamic model and the obtained numerical results play an important role in achieving the main goal of the benchmark library. To this end, some analytical tools for assessing accuracy, stability, and computational cost will be presented in Section 3. A typical multibody system is considered in Section 4, and the process of categorizing, modeling, simulating, and submitting numerical results will be demonstrated. We also provide guidelines to help maintain a unified scheme for representing models and discussing results through the online library.

2 Systematic classification of multibody benchmark problems

Since many diverse engineering problems can be studied as multibody systems, designing a comprehensive classification scheme for multibody systems is a challenging task. In this section, we present some general strategies for classifying multibody systems. An effective classification scheme will allow users to search the library effectively. The library has been designed to permit modifications to the classification scheme in the future so it can be adapted to meet the ever-growing variety of problems studied by multibody dynamicists.

2.1 Physical characteristics and applications

Physical features, including material behavior (rigid vs. flexible components), topology (open- vs. closed-loop), dimension (planar vs. spatial systems), geometry (micro vs. macro scale), and system size (small- vs. large-scale) are the key criteria used to classify multibody systems. All benchmark problems are classified using these criteria in addition to those discussed below.

A wide variety of applications, from aerospace and vehicle engineering to marine and railroad systems, from biomechanical and sport applications to particle and molecular mechanics, can be studied as multibody problems. Furthermore, engineering sciences like robotics [10, 11] and mechanism and machinery design [12, 13] benefit from the diverse approaches developed in the field of multibody dynamics. By including applications from many diverse fields, we hope to encourage collaboration and cross-pollination of ideas between disciplines.

2.2 Modeling approaches and analysis schemes

Modeling techniques can have a substantial effect on the performance of the simulation of a multibody system; however, there is no fundamental rule to determine which technique to apply for a particular class of multibody problem. As such, applying several modeling approaches for each benchmark problem and comparing the simulation results and performance is one of the primary goals of the benchmark library. Users will be able to assess the performance and applicability of various modeling techniques for each benchmark problem, ultimately guiding the selection of modeling approaches for new problems.

The governing equations of motion can be formulated using several recognized approaches: Newton–Euler equations, Lagrangian equations, Hamilton's canonical equations, Kane's method, and Gibbs–Appell equations. Depending on the type of constraints (holonomic or nonholonomic), topology (open- or closed-loop), and material behavior (rigid or flexible bodies), there may be some inclination to use a particular method to model a benchmark problem. For instance, recursive formulations with relative coordinates are more stable and computationally more efficient for open-loop flexible multibody systems [14–17]; for systems with nonholonomic constraints, Kane's approach has proven to be quite attractive [18–20].

The representation of the topology of multibody systems using linear graphs [21–25] and bond graphs [26–29] lead to reliable, systematic procedures for formulating the governing equations of motion. Capability in generating the equations of motion automatically make these graph-based procedures appropriate for use in software. Cascading and clustering, where the system is segmented into a series of unconstrained subsystems with kinematic constraints, are recent approaches applied in multibody dynamics [30].

The configuration of a multibody system must be specified using a thoughtful selection of generalized coordinates. In fact, efficient formulation and simulation procedures are highly dependent on the selected coordinates. Equations expressed in terms of absolute (Cartesian) coordinates are easily formulated (also known as a body-coordinate formulation), but, in general, very large systems of equations are generated, and they contain many algebraic equations representing the constraints between different connecting bodies. Relative (joint) coordinates correspond to the degrees of freedom of joints in the system, and lead to dynamic equations expressed in terms of the minimum number of coordinates for open-loop systems (also known as the joint-coordinate formulation); however, more topological accounting must be considered in this formulation. Relative coordinates are particularly suitable for multibody systems with tree-like topologies [31]. Point (natural) coordinates can be used to represent a multibody system using a collection of interconnected points [32]. Rotational coordinates corresponding to the orientation of bodies in the system can be avoided using this approach, thereby reducing or even eliminating the need for computationally-expensive transcendental function calls.

Based on the nature of the problem, different analysis schemes—static, kinematic, forward dynamic, or inverse dynamic—may be required. A static equilibrium analysis is of great importance in many multibody problems, and generally requires solving a system of nonlinear algebraic equations. As the complexity of the model increases, so too does the system of equations that must be solved to determine the equilibrium position. Vibration, sensitivity, and contact analyses must be considered when flexible elements or contact/impact mechanics are involved [33–38]. The detection of contact points and the estimation of contact forces are pivotal issues in these multibody problems. In velocity and acceleration analyses, the velocity and acceleration of each body in the multibody system are determined when the positions (and also the velocities, in the case of an acceleration analysis) of the input bodies are given [39]. Such analyses are fairly straightforward, as they involve solving systems of linear equations and the solution is unique.

2.3 Simulation methods and computational issues

Depending on the system topology, selected coordinates, modeling technique, and formulation, the governing dynamic equations may be in the form of either ordinary differential equations (ODEs) or differential-algebraic equations (DAEs). Much research has focused on the numerical simulation of multibody systems, encompassing different modeling approaches, formulations of the dynamic equations, software implementations, and numerical integration techniques [32, 39, 40].

Broadly speaking, there are two ways of formulating the equations of motion: symbolically and numerically. The symbolic manipulation of dynamic equations has been a subject of great interest in recent decades, particularly for the purposes of real-time simulation, optimization, and parameter identification. In certain contexts, a symbolic approach may be superior to a numeric approach, since the dynamic equations are generated only once, not at each time step of the simulation. Maplesoft has been a pioneer in developing symbolic computation engines in the fields of mathematics, science, and engineering. The demand for software that generates and manipulates the governing equations for dynamic systems motivated the company to introduce MapleSim, a multi-domain modeling and simulation tool that can be used in the design and analysis of a wide range of engineering systems [7]. MapleSim applies graph-theoretic and symbolic computing principles to generate systems of equations that can be simulated efficiently. There are also some other software packages capable of symbolic modeling in the field of multibody system dynamics, e.g. ROBOTRAN, AutoSim, CarSim, and Dymola.

Differential-algebraic equations representing the dynamic behavior of a multibody system can be formulated systematically by describing the configuration of each body in the system using a set of translational and rotational coordinates, incorporating the constraint equations for each kinematic joint, and applying the Lagrange multiplier formulation to describe reaction forces [32]. A potential problem with this approach, also called the body-coordinate formalism, is its computational efficiency. Velocity transformations can be used to express the dynamic equations in terms of a subset of the generalized coordinates, which may improve the computational efficiency. When algebraic constraints are present, stabilization procedures (e.g., Baumgarte stabilization) may be applied. Penalty formulations, the Lagrange multiplier formulation, the augmented Lagrangian formulation, and recursive methods are among the formulation procedures that have been widely used by researchers in the field of multibody dynamics. The systematic treatment of multibody dynamic formalisms has been thoroughly studied [39]. Four formalisms have been examined by Cuadrado et al. [40]: an augmented Lagrangian formulation with projections to index-1 and index-3 systems, a new modified state-space formulation based on projection matrices, and a fully recursive formulation. In addition, a comparative study was performed, considering topology changes, singularities, inequality constraints, numerical stiffness, and redundant constraints to evaluate performance in terms of speed and accuracy.

In flexible multibody dynamics, elastic deformations are not negligible compared to rigid-body motions. Some of the formulation methods developed for flexible multibody systems include finite element techniques, boundary element methods, assumed mode (component) methods, floating-frame-of-reference formulations, linear theory of elastodynamics, finite segment methods, and large rotation vector formulations [31, 41–43]. A wide variety of flexible multibody systems have been developed by scientists and engineers in many fields and for diverse applications.

The last step in the simulation of mechanical systems is numerical integration. Depending on the nature of the problem and the modeling techniques and formulation procedures employed, different numerical methods may be required to ensure an efficient and stable simulation. Different numerical schemes have been developed for integrating the ODEs and DAEs governing the dynamic behavior of multibody systems (e.g., implicit and explicit, single- and multi-step, forward and backward schemes). Stiffness of the obtained differential equations, absolute and relative tolerances, mass matrix estimations, singularity avoidance, and the number of arithmetic operations all contribute to the performance of the simulation process. Some of the numerical recipes and advances developed in recent decades have been used in software packages specialized in multibody dynamic analysis. A comparative study of various modeling, formulation, computing, and simulation criteria applied in different multibody software has been provided by Schiehlen [44]; an inventory of the capabilities of different multibody software packages, from topology representation to postprocessing methods, was provided in this work. González et al. [45] reviewed several benchmark problems examined by researchers in various fields, including a road vehicle, a five-link wheel suspension, a four-bar mechanism, a human body model, a heavy truck suspension, and a flexible slider-crank mechanism. Computational efficiency, which is one of the key issues in the evaluation of benchmark problems, was measured to study the performance of various simulation procedures applied in each benchmark problem.

3 Principles in the evaluation of benchmark problems

Since analytical solutions to multibody problems are rare, numerical procedures are the predominant tools used to simulate mechanical systems. Several numerical integrators, designed for both ODEs and DAEs, and with different degrees of complexity, have been thoroughly studied by García de Jalón and Bayo [39], who also provide accuracy comparisons for different applications. Accuracy, stability, convergence, computational effort, and efficiency of the computational algorithms must be evaluated for specific multibody problems. Although there have been substantial improvements in computer implementations of numerical integrators in recent decades, there is no guarantee that a particular integrator will have stable and accurate simulation results for arbitrary inputs to the nonlinear ODEs or DAEs governing the behavior of a given multibody problem.

Nonlinearities, singularities, ill-conditioned matrices (e.g., in the constraint Jacobian matrix), topology changes, time-step estimation procedures, and tolerances can be sources of instability and inaccuracy in the simulation results. Stability analysis tools can be classified based on the analytical, numerical, and experimental information analyzed from the dynamic systems. The Lyapunov direct method (for general dynamical systems), the characteristic exponent method (for linear ODEs with constant coefficients), Floquet theory (for linear ODEs with periodic coefficients), the Ibrahim time domain method, and the complex exponential method (Prony's method) are some of the well-established stability analysis schemes applicable to the field of multibody system dynamics [46].

Numerical integrator stability is one of the key aspects that should be inspected during the simulation of multibody systems. In particular, the computational error should be bounded during the integration of the equations of motion. It is important to note that the nature of the system also affects the stability of the simulation. García de Jalón and Bayo [39] introduced *conditionally stable*, *stiffly stable*, and *A-stable (unconditionally stable)* algorithms to examine various numerical integration schemes that are applied widely in the numerical analysis of multibody system simulation. Stability analysis is also essential when there are flexible components in a multibody system. The two-stage Radau IIA scheme, the energy decaying scheme, and the generalized-α scheme can be unconditionally stable integration algorithms, under some specific conditions, suitable for complex flexible multibody systems [47]. For such systems, a linearized stability analysis, applying Prony's curve-fitting method, has shown promising performance [46].

Computational cost, including the number of arithmetic operations, the number of machine instructions, the number of algorithmic unknowns, the execution time, and complexity analyses are tools for assessing the efficiency of simulation codes. Given a desired accuracy, the execution time required to solve a benchmark problem can be considered as a measure of overall performance [5, 45]. In the work of González et al. [45], the accuracy was measured as the maximum error between the reference solution and the obtained solution; in [5], the authors introduced hardware and software performance ratios to measure the performance of formulations and implementations.

In many situations, such as hardware-in-the-loop, control engineering, and optimization applications, it is imperative that the numerical simulations run in real time. Real-time performance can be facilitated by a thoughtful selection of formulation and integration schemes, which demands an analysis of algorithms in terms of both time and storage requirements. Anderson [48] presented an $\mathcal{O}(n)$ formulation for the simulation of tree-like rigid multibody systems. In another work [49], Anderson et al. presented a recursive $\mathcal{O}(n)$ algorithm for dynamic simulation of closed-loop multibody systems. An $\mathcal{O}(\log(n))$ algorithm has been introduced by Critchley and Anderson [50] for the dynamics of general multibody systems. Bauchau [51] proposed a parallel computation scheme, based on the computational domain decomposition and the Lagrange multiplier technique, to enforce kinematic constraints through a hybrid strategy for the simulation of flexible multibody systems. Noticeable efforts in developing high-performance algorithms in multibody dynamic simulation can be found in the official Computational Dynamics Laboratory website of Rensselaer Polytechnic Institute [52].

The order of convergence is another metric by which the efficiency of an integration scheme can be assessed [47]. The order of convergence measures the reduction of error relative to the change in the integration step size. Structure and properties of some well-established one-step and multi-step numerical integrators (e.g., implicit Runge–Kutta methods and backward differentiation formulas) have been studied by Hairer and Wanner [53].

In the absence of experimental data, the results from a second software package can be used to validate simulation results. Adams, MapleSim, SIMPACK, SAMCEF, Mecano, RecurDyn, LMS Virtual.Lab Motion, NEWEUL, ROBOTRAN, SPACAR, MBDyn, and DYMORE are some of the multibody dynamic software packages that can be used for model validation.

4 The Library of Computational Benchmark Problems

The benchmark library is currently available at `http://real.uwaterloo.ca/benchmarks/`, and has been designed with a comprehensive classification scheme that will allow users to quickly identify benchmark problems related to their area of interest. A screenshot of the online benchmark library is shown in Figure 1.

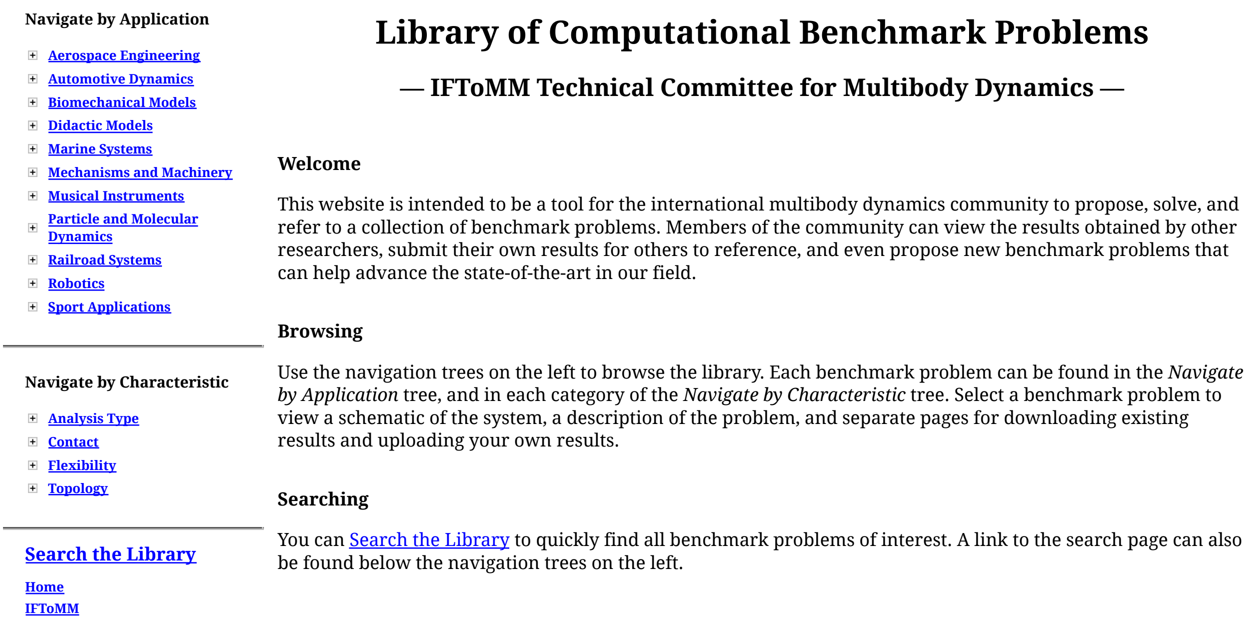


Figure 1. A screenshot of the multibody benchmark library welcome page.

Mechanism topology (open-loop or closed-loop) and analysis type (kinematic, static, forward dynamic, or inverse dynamic) are among the characteristics on which classification is based. Benchmark problems are also naturally categorized by application area, so vehicle dynamicists, for example, can easily navigate to the vehicle-related pages of the library, regardless of the topology or analysis type associated with each problem. The capability to upload new simulation results for existing problems, propose new benchmarks, and search the database have been incorporated. The ultimate objective of the library is to provide a tool that supports the research of multibody dynamicists worldwide, helping to advance the state-of-the-art in our ever-growing field.

The Library of Computational Benchmark Problems provides users with the ability to do the following:

1. Describe new benchmark problems in a systematic way so that other users can easily identify the characteristics of the problem and solution approaches employed.

2. Locate all the information required to regenerate a multibody problem or use technical data from the simulation results.

3. Communicate with other users to discuss issues and challenges regarding a particular multibody problem, be informed by other tools in evaluating the problem, and improve collaborative work on well-established benchmark problems.

4. Compare and summarize different results from a benchmark problem for improving modeling and simulation schemes.
5. Identify copyright agreements for submitted problems and novel procedures.
6. Share ideas for improving the benchmark library.

General guidelines are provided below to help contributors of the online library define new benchmark problems. A double four-bar mechanism benchmark will be used to demonstrate.

4.1 Introducing a benchmark problem

It is essential to specify the appropriate class of problem based on the classification scheme used in the library so that users can easily find benchmark problems of interest. Each benchmark problem can belong to several categories. For example, a flexible robot carrying a payload in space can be classified as an open-loop, flexible multibody system with no contact while being associated with aerospace engineering and robotics application categories. Each benchmark problem will also be associated with one or more of the analysis categories (static, kinematic, forward dynamic, and inverse dynamic). Capability to search for a benchmark problem of interest can be accessed by following the “Search the Library” link.

When describing a new benchmark problem, some key points should be mentioned in the problem definition file uploaded to the library:

1. Simplifications and assumptions considered in order to model the physical system using mathematical tools (e.g., small or large deformation assumptions in the case of flexible multibody systems, non-ideal joints, or assumptions in contact problems).
2. Technical information, including the geometry, mass properties, effect of gravity, reference coordinate systems, and constant parameters (e.g., coefficients of restitution and friction in contact problems).
3. The topology of the system should be depicted (particularly in complex spatial multibody systems) and the operation of the mechanism should be explained.
4. The inputs, outputs, simulation time, initial conditions, and problem objective should be specified.
5. Numerical results and plots required in validating the model should be presented and discussed.
6. Depending on the benchmark problem, simulation issues (such as instabilities, singularities, and computational errors) and the resolution of these issues should be discussed.

A double four-bar mechanism driven only by gravity, shown in Figure 2, has been uploaded to the library in the “Mechanisms and Machinery” branch. Note that this system is also categorized as being without contact, consisting of only rigid bodies, and having a closed-loop topology. The objective of this benchmark problem is to carry out the simulation as efficiently as possible, while maintaining an error (maximum drift of the total energy throughout the simulation) below 0.1 J. All the links are of length $1\ \mathrm{m}$ and mass $1\ \mathrm{kg}$. Consequently, when the mechanism reaches the horizontal position, the number of degrees of freedom increases from 1 to 3.

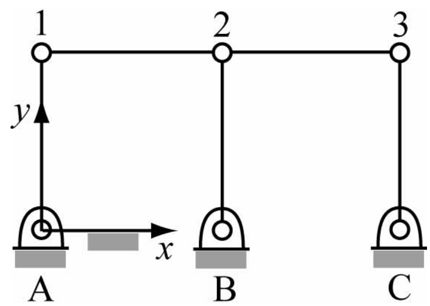
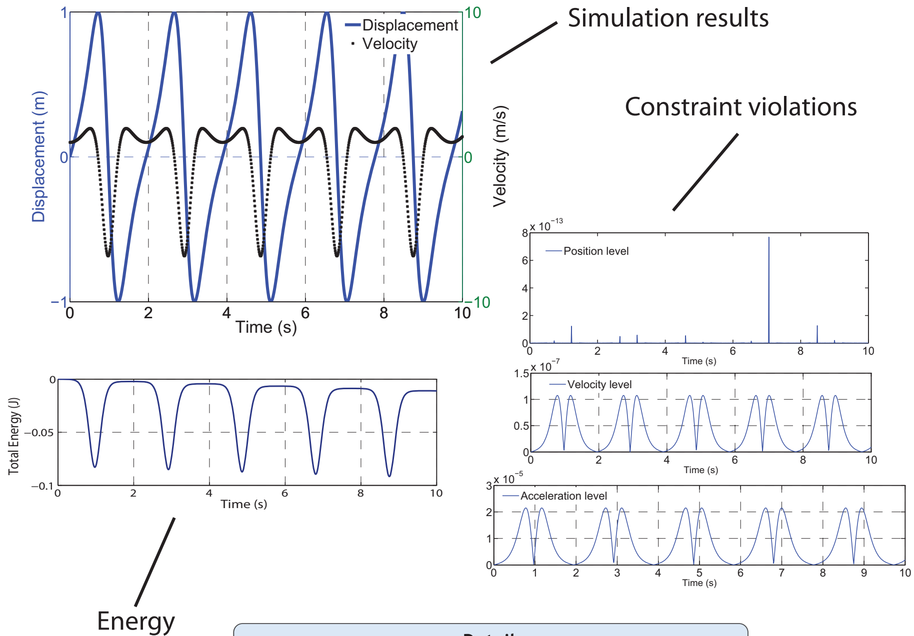


Figure 2. A double four-bar mechanism.

The gravitational acceleration acts in the $-y$-direction with a value of 9.81 $\mathrm{m/s^2}$. Initially, the position of the system is as shown in Figure 2, and the velocity of each of the three mobile pins (points 1, 2, and 3) is $1\ \mathrm{m/s}$ in the $+x$-direction. The total simulation time is 10 s. A document is provided that describes the benchmark problem in detail.



Details	
CPU time (s)	0.6
CPU / GPU	Intel Core 2 DUO E6550 @ 2.33 GHz
Operative System	Windows XP SP3
Method description	Coordinates: natural Equations of motion: index-3 augmented Lagrangian formulation with projections of velocities and accelerations Integrator: trapezoidal rule Solution: Newton-Raphson iteration Time step: 0.01 s (fixed) Penalty factor: 1.E9 Programming language: Matlab

Figure 3. Simulation results and technical details for the double four-bar mechanism.

4.2 Presenting and analyzing the results

One of the main goals of the benchmark library is to provide comparisons using different modeling and simulation procedures for each benchmark problem. To accomplish this goal, certain technical details must be specified for each set of results:

1. Time histories of all coordinates should be included as a text file to facilitate comparison with other solution techniques.

2. Based on the classification procedures mentioned above, the appropriate modeling approach, formulation, numerical procedures (e.g., type of integrator and step size), type of coordinates, and all the parameters related to each solution process should be listed on the *Details* page.

3. Since different operating systems and architectures may be used, details of the computational hardware must be provided.

4. Accuracy, execution time, and other criteria that are helpful in evaluating the model should be given.

A sample of the simulation results and technical details for the double four-bar mechanism submitted to the library are shown in Figure 3. For a particular benchmark problem, a separate *Details* page is used to record the results obtained from each solution approach, including CPU time, hardware/software information, applied formulation schemes, and numerical procedures.

5 Summary and future work

An online Library of Computational Benchmark Problems has been designed as a comprehensive reference for multibody dynamic researchers. The Library will help multibody researchers study and compare the efficacy of various modeling and simulation procedures in the treatment of multibody systems. Ultimately, trends regarding the applicability and efficiency of simulation strategies will emerge. The Library will also facilitate the dissemination of new research. Several classification schemes—from application, modeling approach, and formulation procedure to simulation method and computational issues—have been included in the design of the Library.

Some resources for well-established dynamic modeling, formulation, computer implementation, numerical integration, and computational procedures, along with procedures applied in the evaluation of multibody simulation models, have been presented here. The general framework for the online Library was introduced and guidelines for using the Library were presented. A typical benchmark problem along with some screenshots from the Library were used to demonstrate the steps required to submit a benchmark problem, simulation results, and technical information to the Library. The online Library has been designed to be flexible to changes in the classification scheme as deemed necessary, which will help ensure its longevity. Once the Library has been populated with a variety of benchmark problems and solution strategies, it will serve as an excellent resource for testing new modeling, simulation, and computational techniques.

Acknowledgments

The authors gratefully acknowledge the NSERC/Toyota/Maplesoft Industrial Research Chair program for financial support of this research.

References

[1] I. Newton. *Philosophiæ Naturalis Principia Mathematica ("Mathematical Principles of Natural Philosophy")*. Edmond Halley, London, 1687.

[2] L. Euler. Nova methodus motum corporum rigidarum determinandi. *Novi Commentarii Academiae Scientiarum Petropolitanae*, 20:208–233, 1776.

[3] J. L. Lagrange. *Mécanique Analytique*. L'Académie Royal des Sciences, Paris, 1788.

[4] Kinematic models for design digital library, Cornell University Library. `http://kmoddl.library.cornell.edu/`, April 2013.

[5] M. González, F. González, A. Luaces and J. Cuadrado. A collaborative benchmarking framework for multibody system dynamics. *Engineering with Computers*, 26(1):1–9, 2010.

[6] Multibody and non-smooth dynamics, Center of Mechanics, Institute of Mechanical Systems, ETH Zurich. `http://www.zfm.ethz.ch/e/dynamics/`, April 2013.

[7] MapleSim model gallery, Application examples created in MapleSim, Maplesoft. `http://www.maplesoft.com/products/maplesim/ModelGallery/`, April 2013.

[8] SIMPACK, Multi-body simulation software, Model database. `http://www.simpack.com/model_database.html`, April 2013.

[9] Dymore Solutions, Simulation tools for flexible multibody systems. `http://www.dymoresolutions.com/index.html`, April 2013.

[10] A. Jain. *Robot and Multibody Dynamics: Analysis and Algorithms*. Springer, New York, 2011.

[11] H. Gattringer and J. Gerstmayr. *Multibody System Dynamics, Robotics and Control*. Springer, 2012.

[12] H. Rahnejat. *Multi-Body Dynamics: Vehicles, Machines, and Mechanisms*. Professional Engineering Publishing, Bury St. Edmunds and London, and Society of Automotive Engineers (SAE) co-publishers, 1998.

[13] S. K. Saha and W. O. Schiehlen. Recursive kinematics and dynamics for parallel structured closed-loop multibody systems. *Mechanics of Structures and Machines*, 29(2):143–175, 2001.

[14] M. Xie. *Flexible Multibody System Dynamics: Theory and Applications*. CRC Press, 1994.

[15] F. M. L. Amirouche and M. Xie. An explicit matrix formulation of the dynamical equations for flexible multibody systems: A recursive approach. *Computers & Structures*, 46(2):311–321, 1993.

[16] S.-S. Kim and E. J. Haug. A recursive formulation for flexible multibody dynamics, Part II: closed loop systems. *Computer Methods in Applied Mechanics and Engineering*, 74(3):251–269, 1989.

[17] S.-S. Kim and E. J. Haug. A recursive formulation for flexible multibody dynamics, Part I: open-loop systems. *Computer Methods in Applied Mechanics and Engineering*, 71(3):293–314, 1989.

[18] T. R. Kane and D. A. Levinson. *Dynamics: Theory and Applications*. McGraw-Hill, New York, 1985.

[19] R. Masoudi and M. Mahzoon. Maneuvering and vibrations control of a free-floating space robot with flexible arms. *Journal of Dynamic Systems, Measurement, and Control*, 133(5):051001.1–051001.8, 2011.

[20] T. R. Kane. Dynamics of nonholonomic systems. *Journal of Applied Mechanics*, 28(4):574–578, 1961.

[21] J. J. McPhee. On the use of linear graph theory in multibody system dynamics. *Nonlinear Dynamics*, 9(1–2):73–90, 1996.

[22] J. J. McPhee, M. G. Ishac and G. C. Andrews. Wittenburg's formulation of multibody dynamics equations from a graph-theoretic perspective. *Mechanism and Machine Theory*, 31(2):201–213, 1996.

[23] J. J. McPhee. A unified graph-theoretic approach to formulating multibody dynamics equations in absolute or joint coordinates. *Journal of the Franklin Institute*, 334(3):431–445, 1997.

[24] P. Shi and J. McPhee. Dynamics of flexible multibody systems using virtual work and linear graph theory. *Multibody System Dynamics*, 4(4):355–381, 2000.

[25] A. Jain. Graph theoretic foundations of multibody dynamics. *Multibody System Dynamics*, 26(3):307–333, 2011.

[26] D. Karnopp. The energetic structure of multi-body dynamic systems. *Journal of the Franklin Institute*, 306(2):165–181, 1978.

[27] A. A. Zeid and J. L. Overholt. Singularly perturbed bond graph models for simulation of multibody systems. *Journal of Dynamic Systems, Measurement, and Control*, 117(3):401–410, 1995.

[28] D. Karnopp. Understanding multibody dynamics using bond graph representations. *Journal of the Franklin Institute*, 334(4):631–642, 1997.

[29] I. I. Kosenko, M. S. Loginova, Y. P. Obraztsov and M. S. Stavrovskaya. Multibody systems dynamics: Modelica implementation and bond graph representation. In *5th International Modelica Conference*, pages 213–223. Vienna, Austria, 2006.

[30] J. Huang, Y. H. Chen and K. Guo. A novel approach to multibody system modeling: cascading and clustering. *Journal of Aerospace Engineering*, 2012.

[31] F. M. L. Amirouche. *Fundamentals of Multibody Dynamics: Theory and Applications*. Birkhäuser, Boston, 2006.

[32] P. E. Nikravesh. An overview of several formulations for multibody dynamics. *Product Engineering*, pages 189–226. Springer, Netherlands, 2005.

[33] S. Ebrahimi and P. Eberhard. Aspects of contact problems in computational multibody dynamics. In J. C. G. Orden, J. M. Goicolea and J. Cuadrado, editors, *Multibody Dynamics: Computational Methods in Applied Sciences*, volume 4, pages 23–47. Springer, Netherlands, 2007.

[34] F. Pfeiffer and C. Glocker. *Multibody Dynamics with Unilateral Contacts*. Wiley, New York, 1996.

[35] J. M. P. Dias and M. S. Pereira. Sensitivity analysis of rigid-flexible multibody systems. *Multibody System Dynamics*, 1(3):303–322, 1997.

[36] W. Jiang, X. Chen and X. Luo. Vibration and sensitivity analysis of spatial multibody systems based on constraint topology transformation. In F. Ebrahimi, editor, *Advances in Vibration Analysis Research*, pages 391–418. InTech Europe, Crotia, 2011.

[37] Y. A. Khulief and A. A. Shabana. A continuous force model for the impact analysis of flexible multibody systems. *Mechanism and Machine Theory*, 22(3):213–224, 1987.

[38] R. Masoudi and J. McPhee. A mechanistic model of compression hysteresis in compliant interfaces in multibody dynamic simulation of a piano action mechanism. *Proceedings of the 2nd Joint International Conference on Multibody System Dynamics*, Stuttgart, Germany, May 29–June 1, 2012.

[39] J. García de Jalón and E. Bayo. *Kinematic and Dynamic Simulation of Multibody Systems: The Real-Time Challenge*. Springer-Verlag, New York, 1994.

[40] J. Cuadrado, J. Cardenal and E. Bayo. Modeling and solution methods for efficient real-time simulation of multibody dynamics. *Multibody System Dynamics*, 1(3):259–280, 1997.

[41] A. A. Shabana. Flexible multibody dynamics: review of past and recent developments. *Multibody System Dynamics*, 1(2):189–222, 1997.

[42] O. A. Bauchau. *Flexible Multibody Dynamics*. Springer, Dordrecht, 2011.

[43] M. Géradin and A. Cardona. *Flexible Multibody Dynamics: A Finite Element Approach*. John Wiley, Chichester, 2001.

[44] W. Schiehlen. *Multibody Systems Handbook*. Springer-Verlag, Berlin, 1990.

[45] M. González, D. Dopico, U. Lugrís and J. Cuadrado. A benchmarking system for MBS simulation software: Problem standardization and performance measurement. *Multibody System Dynamics*, 16(2):179–190, 2006.

[46] O. A. Bauchau and J. Wang. Stability analysis of complex multibody systems. *Journal of Computational and Nonlinear Dynamics*, 1(1):71–80, 2006.

[47] A. Epple. *Methods for Increased Computational Efficiency of Multibody Simulations*. PhD Thesis, Georgia Institute of Technology, U.S.A., 2008.

[48] K. S. Anderson. An order n formulation for the motion simulation of general multi-rigid-body tree systems. *Computers & Structures*, 46(3):547–559, 1993.

[49] K. Anderson and J. Critchley. Improved 'order-N' performance algorithm for the simulation of constrained multi-rigid-body dynamic systems. *Multibody System Dynamics*, 9(2):185–225, 2003.

[50] J. H. Critchley and K. S. Anderson. A parallel logarithmic order algorithm for general multibody system dynamics. *Multibody System Dynamics*, 12(1):75–93, 2004.

[51] O. A. Bauchau. Parallel computation approaches for flexible multibody dynamics simulations. *Journal of the Franklin Institute*, 347(1):53–68, 2010.

[52] Computational Dynamics Laboratory, Rensselaer Polytechnic Institute. `http://homepages.rpi.edu/~anderk5/lab/`, April 2013.

[53] E. Hairer and G. Wanner. *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems*. Springer, Berlin, 1996.