Implementation of effective procedures for unilateral contact modeling in multibody dynamics

L. Bottasso, L. Trainelli

► To cite this version:

L. Bottasso, L. Trainelli. Implementation of effective procedures for unilateral contact modeling in multibody dynamics. *Mechanics Research Communications*, 2001, 28 (3), pp.233 - 246. 10.1016/S0093-6413(01)00168-9 . hal-01389913

HAL Id: hal-01389913

<https://hal.science/hal-01389913>

Submitted on 30 Oct 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution| 4.0 International License

IMPLEMENTATION OF EFFECTIVE PROCEDURES FOR UNILATERAL CONTACT MODELING IN MULTIBODY DYNAMICS

C.L. Bottasso, L. Trainelli

Dipartimento di Ingegneria Aerospaziale, Politecnico di Milano, Campus Bovisa, Via La Masa 34, 20158, Milano, Italy

The paper discusses the treatment of general contact/impact problems in multibody mechanical systems, with particular attention to the implementation of such procedures in commercial codes. We describe a unilateral contact model based on a flexible parameterization of the contacting surfaces, a geometrical minimum distance problem and a constitutive model of the interaction forces. The resulting numerical procedures are demonstrated by means of examples.

Introduction

The modeling of contact/impact processes represents a key topic in advanced multibody simulation, and is crucial for an effective use of this technology in many industrial problems. Apart from mechanisms with variable topology that represent the majority of applications, intermittent contact can also be caused by the presence of clearance at the joints, possibly due to wear or imperfections. The applicability of multibody dynamics to the virtual prototyping of an ever increasing range of problems in multiple areas of engineering makes the modeling of unilateral contacts an important asset of any software tool that aims at being general.

Several commercial multibody codes are available, and some of them are widely used by industry. While research codes are invaluable for testing new algorithms and procedures, they usually can not compete with industrial software for number of implemented elements and features, graphic capabilities, maturity of the user interface and integration with other CAD/CAM/CAE tools. Furthermore, most commercial codes are expandable and customizable through general user-defined subroutines that allow to add new features.

However, to date the most widely used multibody packages offer only primitive contact/impact modeling capabilities. Therefore, it would seem worthwhile to explore the idea of implement-

ing a contact/impact model in a commercial package, rather than in a research code. In fact such a tool would be applicable to a vast class of impact problems, either representing normal operative situations or non-regular working conditions which nevertheless have to be analyzed in the design process to ensure safety and/or efficiency.

A numerical procedure with unilateral contact modeling capabilities needs to address the following problems: a) give a mathematical representation of the geometry of the contacting parts; b) solve the minimal distance problem between the curves or surfaces where contact occurs, in order to evaluate the position of the candidate contact points; c) apply, at the contact points, the interaction forces defined by a model that is appropriate for the phenomenon being investigated.

Scope of this work is not the development of new methodologies for dealing with unilateral contacts, but the critical analysis of various possible alternatives proposed in the literature, with the specific goal of implementing such models in existing commercial software through user-definable features, i.e. without accessing the program source code. While a mathematical model should be primarily judged for its appropriate description of the phenomenon being investigated and for its performance, it seems that its ease of integration into existing procedures has not been previously considered. In our analysis, we will assume that user defined routines only permit the coding of non-linear holonomic constraints and of forces that are function of the system state.

The paper is organized in the following manner. First, we discuss the two main categories of unilateral contact schemes, the impulsive and the continuous force models, and we point out their strengths and deficiencies in view of what has been pointed out above. Next we consider the problem of description and parameterization of the contacting curves and surfaces and of the determination of the minimum distance between them. These are standard problems thoroughly covered in the literature, and are therefore only briefly sketched here. We then discuss the practical implementation of the selected unilateral contact model in a commercial package, pointing out a few general "tricks" that allow to bypass common limitations of the existing codes. Finally, we demonstrate the implemented procedures with the help of numerical simulations of multibody systems characterized by intermittent motion using two popular codes, ADAMS¹ and LMS DADS².

Impulsive and Continuous Force Models

The various approaches to the modeling of unilateral contact conditions fall into two main categories. The first considers an impact as an impulsive phenomenon of null duration [4, 5]. The configuration of the system is "frozen" during impact, and an appropriate model is used for relating the states of the system immediately before and immediately after the event. There are two alternative flavors to this theory: Newton's method and Poisson's method [10]. The former relates the relative normal velocities of the contacting bodies through the use of an appropriate restitution coefficient. The latter divides the impact in two phases: an initial compression phase brings the normal relative velocity of the bodies to zero through

¹ADAMS is a trademark of Mechanical Dynamics, Inc.

²LMS DADS is a trademark of LMS International.

the application of an impulse at the contact location; then, an expansion phase applies an impulse of opposite sign. The magnitude of the second impulse is related to the magnitude of the first one through a restitution coefficient.

A second approach models the contact/impact condition as a finite duration event, and tries to describe the time history of the resulting interaction forces for the whole duration of the phenomenon [6, 8, 1]. This is achieved by introducing a suitable phenomenological model for the contact forces, usually expressed as functions of the inter-penetration, or “approach”, between the contacting bodies. The computation of the approach at each time instant of the simulation is achieved solving the same set of equations that express the minimum distance problem when the bodies are not in contact. As for all contact models, we have even in this case a complementarity problem: either the sum of the relative distance and the approach is greater than zero, and in this case the contact forces vanish, or the same sum is null, and the relative distance is equal and opposite to the approach (inter-penetration) with non-vanishing interaction forces.

These models are well established in the literature, and need not be discussed in greater detail here. Furthermore, they both can deal with contacts with friction and rolling conditions. Although the impulsive model has been used with success for multibody contact/impact simulations [10], it is quite clear that it seems most suited to the description of events of very short duration. Furthermore, its implementation requires the interruption of the numerical integration of the equations of motion when an impact is detected, followed by the solution of a linear complementarity problem that implies the manipulation of entire descriptive matrices of the whole multibody system prior to the restarting of the integration process. The resulting procedure would not be easily implemented in proprietary software packages through the standard user-accessible channels, which are typically routines that allow to add constraint conditions and external forces. In particular, the requirement to access entire system matrices (e.g. the mass matrix) can not usually be easily met. Furthermore, stopping and restarting the computation requires some form of “driving” procedure that can pose further implementation problems.

The continuous force model does not rely upon a short duration assumption. In addition, its implementation seems simpler than in the previous case. In fact, a contact force is applied if the bodies are in contact on the sole knowledge of the inter-penetration and its time rate. These quantities are already available through the solution of the minimum distance problem.

Another important difference between the two methods is related to the accurate determination of the contact event. Indeed, contact can only be captured within the accuracy of the current time step. This means that in general some time refinement will be necessary in the proximity of the events. With a continuous force model, contact will trigger sharp gradients in the computed system state variables. Assuming that the target software package is equipped with a reasonably reliable and robust time adaption scheme, the sharp gradients will automatically activate the selection of smaller steps. This will ensure the automatic determination of the contact conditions within a desired tolerance without the need to modify the code time marching procedures. On the contrary, time adaption with an impulsive force model could not rely on the same mechanisms, and some form of modification of the adaption algorithm would be necessary. This however might be not possible without accessing

the source code.

In view of these considerations, for this work we select a continuous force model based on a phenomenological law that relates the magnitude of the approach to the force of contact. In general, the interaction force will be given by the sum of elastic and dissipative components. The elastic term can usually be based on the classical Hertz model. For the determination of the second term, we follow the classical work of Hunt and Crossley [7], where the dissipative forces are proportional to the elastic ones and are energetically equivalent to the Newton restitution coefficient model. This has the advantage of determining the dissipative component through a quantity that can be easily measured experimentally.

Parameterization of Contact Curves and Surfaces

Following the methodology just outlined, we have seen that we need to provide a mathematical description of the geometry of the contacting parts. This will allow the solution of the minimum distance problem, that in turn will trigger the activation of the interaction forces between the contacting parts. Therefore, some kind of parameterization of the geometric entities is required. It is clear that the parameterization adopted should be as general and flexible as possible, in order to allow the greatest range of possible applications. One parameterization that is broadly used for its generality and flexibility is based on Non-Uniform Rational B-Splines (NURBS) [2, 9], often used for CAD solid modeling, manufacturing, graphics etc. NURBS can represent everything from a simple 2D line or circle to the most complex free-form surface.

Based on the NURBS representation, it is possible to evaluate all the geometric attributes of the entities, for example tangents, normals and point locations. These geometric attributes become the ingredients of the minimum distance problem that is solved at each time instant of the simulation. A NURBS-based parameterization of curves and surfaces can be easily coded in a library of routines. This library is then linked with the rest of the solver, and provides the needed geometric functionalities to the user defined subroutines implementing the minimal distance problem.

The Minimum Distance Problem

Letting q be the minimal distance between two entities, we say that they are in contact if $q < 0$. Hence, at any integration time step, the distance q has to be evaluated and a contact force F must to be applied as

$$F = H(q)f(a, \dot{a}), \quad (1)$$

where $a = -q$ is the approach between the entities and $H(q)$ is a Heaviside step function, null for positive q . To decide on the application of the impact force, the minimal distance between the entities must be determined. Let us formalize the problem that has to be solved at any time step to find the point eligible for contact in both the 2D and 3D cases [10].

For the 2D case, the regular curves $C_i \subset \mathbb{R}^2$ ($i = I, II$) are defined through the mapping

$\mathbf{u}_i : A_i \mapsto \mathbb{R}^2$ as

$$C_i := \{\mathbf{u} \in \mathbb{R}^2 : \mathbf{u} \in \mathbf{u}_i(\xi_i) \forall \xi_i \in A_i \subset \mathbb{R}\}, \quad (2)$$

where ξ_i is the curvilinear coordinate of the curve and $\mathbf{u}_i \in C^1(A_i)$. The points eligible for contact are then obtained by solving the non-linear system:

$$\mathbf{n}_I(\xi_I) \cdot \mathbf{t}_{II}(\xi_{II}) = 0, \quad (3)$$

$$\mathbf{d}_{I-II}(\xi_I, \xi_{II}) \cdot \mathbf{t}_{II}(\xi_{II}) = 0, \quad (4)$$

where $\mathbf{t}$ and $\mathbf{n}$ are the vectors representing the tangent and normal to the curves, respectively. Eq. (3) forces the normal vector on C_I to be perpendicular to the tangent vector on C_{II} , while eq. (4) forces the distance vector $\mathbf{d}_{I-II}$ to be perpendicular to the tangent vector on C_{II} . The non-linear system is solved by two values ξ_I and ξ_{II} corresponding to the minimal distance points on the curves.

For the 3D case, the regular surfaces $S_i \subset \mathbb{R}^3$ ($i = I, II$) are defined through the mapping $\mathbf{u}_i : A_i \mapsto \mathbb{R}^3$, where ξ_i and η_i are curvilinear coordinates on the surfaces, and $\mathbf{u}_i \in C^1(B_i)$. The points eligible for contact are obtained by solving the following non-linear system of equations

$$\mathbf{n}_I(\xi_I, \eta_I) \cdot \mathbf{s}_{II}(\xi_{II}, \eta_{II}) = 0, \quad (5)$$

$$\mathbf{d}_{I-II}(\xi_I, \eta_I, \xi_{II}, \eta_{II}) \cdot \mathbf{s}_{II}(\xi_{II}, \eta_{II}) = 0, \quad (6)$$

$$\mathbf{n}_I(\xi_I, \eta_I) \cdot \mathbf{t}_{II}(\xi_{II}, \eta_{II}) = 0, \quad (7)$$

$$\mathbf{d}_{I-II}(\xi_I, \eta_I, \xi_{II}, \eta_{II}) \cdot \mathbf{t}_{II}(\xi_{II}, \eta_{II}) = 0. \quad (8)$$

$\mathbf{t}$ and $\mathbf{s}$ are two orthogonal $\mathbb{R}^3$ vectors lying on the tangent planes and $\mathbf{n}$ is the $\mathbb{R}^3$ vector normal to the surfaces. Eqs. (5) and (7) force the normal vector on S_I to be perpendicular to the tangent vectors on S_{II} ; eqs. (6) and (8) force the distance vector to be perpendicular to the tangent vectors on S_{II} . The non-linear system is solved by the values $\xi_I, \eta_I, \xi_{II}, \eta_{II}$ corresponding to the minimal distance points on the surfaces.

Once the minimal distance points have been determined, in both the 2D and 3D case the minimal distance value q is given by the following expression

$$q = \mathbf{d}_{I-II} \cdot \mathbf{n}_I. \quad (9)$$

Practical Implementation in General Proprietary Multibody Software

Some care needs to be exercised when implementing the minimum distance problem in a proprietary code using the typical functionalities provided by the user defined subroutines. In fact, in general the point of application of user forces on a given body will need to be specified a-priori. This is clearly in contrast with the problem here considered, since the point of application of the contact forces depends on the location of the contact point, which is a problem unknown and can not be specified a-priori.

Implementation Through User Defined Constraint Subroutines

Two main approaches seem to be possible. In the first method, one treats the equations defining the minimum distance problem as a set of non-linear holonomic constraints that are added to the set of constraint conditions of the multibody system. This approach is similar to the way one would attack the problem in case of implementation in a research code. However, the problem needs to be slightly reformulated in order to be practically used.

First, one needs to introduce massless dummy parts that are constrained to move on the curves or surfaces that describe the geometry of the contacting bodies. The dummy parts will coincide at each instant with the candidate contact points, and will therefore allow the application of the contact forces. Clearly, these additional parts must be of null mass in order not to modify the dynamics of the system. For example, one can constrain the dummy parts to move on the contact curves or surfaces with axes $\mathbf{g}_{1D}$ and $\mathbf{g}_{2D}$ belonging to the tangent plane and axis $\mathbf{g}_{3D}$ aligned with the local normal. These constraint relations are once again holonomic constraint conditions that can be implemented using appropriate user constraint routines. The minimum distance problem is then defined using the axis system associated with the dummy parts and the interaction forces are applied to the same parts, rather than to the contacting bodies.

A total number of 14 scalar constraint equations is written for the 2D case, that determine the 6 degrees of freedom of each dummy part in addition to the ξ_I and ξ_{II} parametric coordinates. For the 3D case, we have 16 scalar constraints, corresponding to the 6 degrees of freedom of each dummy part and to the ξ_I , η_I , ξ_{II} , η_{II} parametric coordinates. More precisely, we have in the 2D case:

$$\mathbf{g}_{2D} \cdot \mathbf{t} = 0, \quad (10)$$

$$\mathbf{g}_{3D} \cdot \mathbf{t} = 0, \quad (11)$$

$$\mathbf{g}_{3D} \cdot \mathbf{b} = 0, \quad (12)$$

$$\mathbf{n}_I \cdot \mathbf{t}_{II} = 0, \quad (13)$$

$$\mathbf{d}_{I-II} \cdot \mathbf{t}_{II} = 0, \quad (14)$$

$$\mathbf{x}_D = \mathbf{x}_{DP} + \mathbf{x}_P, \quad (15)$$

where $\mathbf{g}_{iD}$ ($i = 1, 3$) are the dummy part unit vectors, and:

- eq. (10) states that the $\mathbf{g}_{2D}$ axis of a dummy part must be orthogonal to the curve tangent;
- eq. (11) states that the $\mathbf{g}_{3D}$ axis of a dummy part must be orthogonal to the curve tangent;
- eq. (12) states that the $\mathbf{g}_{3D}$ axis of a dummy part must be orthogonal to bi-normal $\mathbf{b}$ to the curve;
- eq. (13) states that the normal to curve I must be orthogonal to the tangent to curve II ;

- eq. (14) states that the relative distance between curve *I* and *II* must be orthogonal to the tangent to curve *II*;
- eq. (15) translates the point on curve constraint for a dummy part and its associated curve.

Eqs. (10-12) must be repeated for both curve *I* and *II*.

For the 3D case we have the same eqs. (10,11) and (13-15), plus the additional conditions:

$$\mathbf{g}_{3D} \cdot \mathbf{s} = 0, \quad (16)$$

$$\mathbf{n}_I \cdot \mathbf{s}_{II} = 0, \quad (17)$$

$$\mathbf{d}_{I-II} \cdot \mathbf{s}_{II} = 0, \quad (18)$$

where eqs. (16-18) correspond to eqs. (10), (13), and (14), respectively, and impose the conditions for the second tangent to the associated surface. Eq. (16) must be repeated for both surface *I* and surface *II*.

A second implementation detail that needs to be considered is related to the fact that constraint equations can usually only be expressed in terms of state variables of the model bodies. However, for the implementation of the minimum distance problem just outlined, the constraint relations also depend on the parametric coordinates which represent additional algebraic unknowns.

The problem can be easily solved introducing two (four in the 3D case) additional massless parts, called "trackers" in the following, associated with the parametric coordinates. The trackers are constrained to the ground by translational joints, in order to have one single degree of freedom left. The value assumed by this displacement at each time instant represents the value of a parametric coordinate along a curve or surface, this way identifying the contact point location.

Having introduced all the constraint conditions just discussed, we have that all the massless parts are completely determined: the dummy parts are constrained to slide on the curves or surfaces so as to be located at each time instant at minimum distance, while the tracker locations correspond to the values of the parametric coordinates at the candidate contact points. It is now possible, based on the solution of the minimum distance problem, to trigger the application of a contact force which can be easily programmed once again through user defined force routines.

Some multibody codes have features that allow the hiding of all the details of the proposed implementation from the user, therefore easing model preparation and verification. In particular, it is usually possible to render the generation of the dummy parts, trackers and constraint equations, fully automatic without user intervention. This will make for a cleaner interface and simpler model construction.

This approach was implemented in the ADAMS software package.

Implementation Through User Defined Force Subroutines

The previously discussed approach gives a rigorous implementation of the minimum distance problem within a proprietary code using user defined constraint routines. As a matter of fact, the non-linear holonomic constraints that define the minimum distance problem are appended to the other constraint conditions of the multibody system and solved together with all the other equations. However, this comes to the price of introducing the dummy parts that move along the contacting curves or surfaces identifying the candidate contact points, and also requires the introduction of the trackers that represent the algebraic unknowns associated to the parametric coordinates. This somewhat increases the number of degrees of freedom of the system. Furthermore, the solver needs to be able to deal with massless bodies, a feature that not all commercial packages will support.

A possibly simpler strategy implements the minimum distance problem within a user defined force routine, therefore avoiding the introduction of holonomic constraint equations. In practice, each time the user force routine is called by the solver, one solves within that routine the minimum distance problem given by eqs. (3) and (4) in 2D or by eqs. (5–8) in 3D. Once the non-linear problem has been solved, for example using a quasi-Newton method, one can evaluate the minimum distance q and, in case of contact, apply the corresponding contact force.

Since the contact points are known from the values of the parametric coordinates that satisfy the minimum distance problem, one can at this point evaluate the transport of the contact force to a known pre-determined point on the body, e.g. the center of mass. This way, one avoids completely the need to define massless dummy parts and furthermore one does not need to introduce trackers for letting the solver deal with algebraic unknowns, since these algebraic unknowns are dealt with internally by the user force routine. Consequently, the implementation leads to a smaller number of unknowns and equations. The price to pay for this is that a (small) non-linear problem must be solved at each call of the user force routine, and therefore we have a non-linear problem nested within the non-linear time marching problem. However, the convergence of the inner loop is usually extremely fast, since we can use the converged values at the previous call as a starting point for the iteration.

This second approach was implemented in LMS DADS and in ADAMS.

Numerical Examples

Circuit Breakers

The first problem is concerned with the simulation of the closing operation of a circuit breaker. The system is composed of several rigid components and its configuration is depicted in Figure 1. The input energy is provided by a closing spring, that is responsible for both the quick activation time of the mechanism and for the recharging of the opening spring. Ten unilateral contact conditions are present in this example, that exhibits a very complex behavior with short duration impacts as well as prolonged contacts, together with considerable interaction forces.

The problem was analyzed using the ADAMS software package, equipped with the unilateral contact modeling capabilities described in this paper. Figure 2 gives a few snapshots of the system configuration during the simulation. Figure 3 presents a plot of the time history of the contact forces for the various unilateral constraints in last part of the simulation. A detailed description of the functionality of the system is beyond the scopes of the present work, however we can notice the typical behavior that is expected in the contact/impact analysis of these mechanisms, i.e. large interaction forces characterized by very sharp gradients that account for the changes in system topology. Close inspection of the plot shows multiple repeated contacts followed by separation of the bodies, or sticking conditions when the bodies remain together for prolonged periods of time.

A second circuit breaker is analyzed next. The mode of operation of the breaker is similar to the previous case, although the topology, geometry and details of the mechanism are different. In this case, we consider the opening of the breaker, induced by a pre-charged spring. During this operation, large intermittent contact forces arise on a number of stops placed both on the fixed frame of the breaker and on the various moving links, in order to restrain their relative motion within predefined values.

This problem was analyzed using the LMS DADS software. The overall configuration of the system is given in Figure 4. Figure 5 shows an animation of the breaker during the simulation (some parts of the mechanism are not represented for clarity). Even in this case we have several violent impacts with the development of large contact forces, as depicted in Figure 6. Figure 7 shows the time history of the minimum distance between a link and a fixed stop, illustrating the repeated contacts that take place in a few milliseconds.

Recharging Mechanism

The example deals with the transient simulation of a recharging mechanism. The model is composed of several rigid body components, in particular a wheel connected to a torsional spring, and an oscillating mechanism that engages the wheel geometry through a tooth. When the oscillating part of the mechanism moves in one direction, the tooth pushes on the wheel profile and makes it turn. When the oscillating part starts moving in the opposite direction, the tooth slides on the wheel as the wheel itself is engaged by another arresting tooth that prevents it from spinning in the opposite direction.

The problem involves multiple impacts of the pushing and arresting teeth with the wheel, and sliding of the same teeth on the wheel profile. The complex shapes of the wheel and of the teeth are easily handled through the NURBS parameterization. Figure 8 gives a few snapshots of the system configuration throughout the simulation, conducted using the ADAMS software.

Conclusions

In this work we have considered the problem of implementing unilateral contact modeling capabilities in commercial multibody codes, comparing two well known methods, namely the impulsive and the continuous force models. We have concluded that the latter is better suited

for the implementation through standard user defined subroutines, i.e. without accessing the program source code, since it only necessitates of the definition of non-linear holonomic constraints and of forces that depend on the system state.

We have discussed the details of two possible implementations of the described algorithms. In the first approach, one appends the non-linear holonomic constraints that define the minimum distance problem to the constraint equations already present in the multibody system. For this to work, one must introduce massless dummy parts that are constrained to move on the contact curves or surfaces and that identify the candidate contact points at each instant of the simulation. Furthermore, one has also to introduce massless trackers that define the values of the algebraic unknowns entering the problem. In the second approach, one avoids the above mentioned complications by solving the non-linear minimum distance problem within each call to a user defined force routine.

We have successfully implemented the described procedures in two popular commercial packages, namely ADAMS and LMS DADS. These additions to the codes allow to model intermittent motion of rigid and flexible multibody systems. For ensuring a wide spectrum of applicability to the method, we have equipped the procedures with a library implementing a parameterization of the contacting parts based on NURBS. Furthermore, we have been able to provide a user-friendly interface that hides all the unnecessary details that could increase the user's workload.

After debugging on a set of simple test cases, we have tested the procedures with a number of problems of industrial relevance. The numerical simulations demonstrate the capabilities of the proposed approach. In particular, the tests conducted have shown that the time adaption algorithms of the target codes can deal effectively even with the strong requirements imposed by contact/impact analysis of multibody systems. The two proposed implementations are both rigorous by treating exactly the minimum distance problem, therefore they yield virtually identical solutions to the problems here analyzed.

Acknowledgments

This research was supported by ABB Ricerca under contract CRPOLIMI 17089/1/AS/99 with Politecnico di Milano. This support is gratefully acknowledged.

References

- [1] A. Cardona and M. Géradin, *Comp. Methods Appl. Mech. Engrg.* **103**, 115 (1993).
- [2] G.E. Farin, *Curves and Surfaces for Computer Aided Geometric Design*, Academic Press, Inc., Boston, MA (1992).
- [3] C. Glocker and F. Pfeiffer, *Nonlinear Dym.* **7**, 471 (1995).
- [4] T.R. Kane, *J. Appl. Mech.* **15**, 718 (1962).
- [5] Y.A. Khulief and A.A. Shabana, *ASME J. Mech., Transm., Autom. Design* **108**, 38 (1986).

- [6] Y.A. Khulief and A.A. Shabana, Mech. Mach. Theory **22**, 213 (1987).
- [7] K.A. Hunt and F.R.E. Crossley, J. Appl. Mech. **11**, 440 (1975).
- [8] H.M. Lankarani and P.E. Nikravesh, J. Mech. Design **112**, 369 (1990).
- [9] L. Piegl and W. Tiller, The NURBS Book, Springer-Verlag, Berlin (1997).
- [10] F. Pfeiffer and C. Glocker, Multibody Dynamics with Unilateral Contacts, John Wiley & Sons, Inc., New York, NY (1996).

Figure 1: View of the first circuit breaker.

a)

b)

c)

d)

e)

f)

Figure 2: Snapshots of the circuit breaker configuration during the simulation.

Figure 3: Time history of the contact forces during the last milliseconds of the simulation.

Figure 4: View of the second circuit breaker.

a)

b)

c)

d)

e)

f)

Figure 5: Snapshots of the second circuit breaker configuration during the simulation.