

Models and Algorithms for the Collision of Rigid and Deformable Bodies

Anton Deguet, Ammar Joukhadar, Christian Laugier

► **To cite this version:**

Anton Deguet, Ammar Joukhadar, Christian Laugier. Models and Algorithms for the Collision of Rigid and Deformable Bodies. RR-3568, INRIA. 1998. <inria-00073113>

HAL Id: inria-00073113

<https://hal.inria.fr/inria-00073113>

Submitted on 24 May 2006

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.

***Models and Algorithms for the Collision of
Rigid and Deformable Bodies***

Anton Deguet and Ammar Joukhadar and Christian Laugier

N° 3568

Décembre 1998

———— THÈME 3 ————



***rapport
de recherche***

Models and Algorithms for the Collision of Rigid and Deformable Bodies

Anton Deguet and Ammar Joukhadar and Christian Laugier

Thème 3 — Interaction homme-machine,
images, données, connaissances
Projet Sharp

Rapport de recherche n° 3568 — Décembre 1998 — 24 pages

Abstract: In this report, we describe models and algorithms designed to produce efficient and physically consistent dynamic simulations. These models and algorithms have been implemented in a unique framework, modeling both deformations and contacts through visco-elastic relations. Since this model of interaction (known as “penalty based”) is much debated, we present a compared study of two contact models; “penalty” and “impulse”.

Indeed, the “penalty” based model is supposed to have two major drawbacks : - determining the visco-elastic parameters and - choosing the computation time step. We present a solution for both problems based on physical concepts. Finally, we will present results comparing real data, “impulse” based simulation and “penalty” based simulation.

Key-words: collision, interactions, deformable objects, rigid body mechanics, dynamic modelling, parameter identification, animation, simulation, virtual reality

(Résumé : tsvp)

Sharp is a common project between Inria Rhône-Alpes and the Gravir laboratory (Informatique Graphique, Vision et Robotique) of the Imag institute (institut d’Informatique et de MATHématiques Appliquées de Grenoble).

Modèles et algorithmes pour la collision d'objets rigides et déformables

Résumé : Dans ce rapport, nous décrivons des modèles et algorithmes conçus pour réaliser des simulations dynamiques rapides et physiquement cohérentes. Ces modèles et algorithmes ont été implantés au sein d'un simulateur unique, intégrant à la fois les déformations et les collisions grâce à des relations visco-élastiques. Dans le mesure où ce modèle d'interaction (connu sous le nom de "pénalité") est très contesté, nous présentons une étude comparée de deux modèles de collision; "pénalité" et "impulse".

En effet, les modèles dits de "pénalité" est supposé avoir deux difficultés majeures : - déterminer les paramètres visco-élastiques et - choisir un pas de temps pour une intégration numérique stable. Nous présentons des solutions basées sur des propriétés physiques pour ces deux problèmes. Finalement, nous montrons des résultats qui permettent de comparer des données réelles et les résultats de simulations à partir de deux modèles de collision : "pénalité" et "impulse".

Mots-clé : collision, interactions, objets déformables, mécanique du solide, modèle dynamique, identification de paramètres, animation, simulation, réalité virtuelle

Contents

| | | |
|----------|---|-----------|
| 1 | Introduction | 5 |
| 2 | Related Works | 6 |
| 2.1 | Impulse based Models | 6 |
| 2.2 | Penalty based Models | 8 |
| 2.3 | Discussion | 11 |
| 3 | Modeling rigid and deformable objects | 12 |
| 3.1 | Discretization | 12 |
| 3.2 | Identification | 13 |
| 3.3 | Interactions | 15 |
| 4 | Motion and deformation generation | 15 |
| 4.1 | Introduction | 15 |
| 4.2 | Adaptative time step and error estimation | 17 |
| 5 | Experiments | 19 |
| 5.1 | Protocol | 19 |
| 5.2 | Results | 20 |
| 5.3 | Discussion | 22 |
| 6 | Conclusions | 22 |

List of Figures

| | | |
|---|--|----|
| 1 | <i>Regular visco-elastic and non linear visco-elastic model</i> | 10 |
| 2 | <i>A soft collision with a sampling that is too low</i> | 11 |
| 3 | <i>Discretization of a convex polyhedron</i> | 13 |
| 4 | <i>Model of a human thigh with an echographic probe.</i> | 14 |
| 5 | <i>Simulation of the behavior of the knee ligament.</i> | 14 |
| 6 | <i>Without a suitable time step the numerical solution may diverge</i> | 16 |
| 7 | <i>A good strategy to avoid divergence and reduce the computational time is to use an adaptive time step τ which allows the system to be integrated using a minimum number of samples.</i> | 17 |
| 8 | <i>The insulator cap and the stereo button</i> | 20 |
| 9 | <i>The insulator cap falls and rebounds</i> | 21 |

1 Introduction

At present, almost all planners used in robotics are based on geometric models. Such models cannot represent all aspects of reality but, when combined with basic dynamic control strategies and used in well structured and known environments, they allow one to solve many problems. The phenomena that geometric models are unable to represent - deformations, complex interactions with dynamic and static friction, inertia, collisions, etc. - are nonetheless necessary for the solution of complex problems such as manipulating a rigid or a deformable object using a dextrous hand, guiding an off-road vehicle on a rugged terrain [3] or estimating pose statistics of parts in an assembly line [16].

Since all of the above mentioned phenomena are obviously dynamic, i.e. related to forces encountered in the system being modeled, the most natural approach is to adapt and integrate the Dynamic laws in dynamic simulators. However, models and algorithms for dynamic simulation cannot always be deduced from Dynamic laws. In fact, Dynamic laws are only observations which describe real phenomena in physical values but do not explain what causes them. Thus, the general approach consists in considering a Dynamics law (i.e. an observation) and trying to conceive a model and an algorithm able to produce consistent simulations. The challenge is therefore to integrate different models in a unified framework and find algorithms with acceptable complexity so as to deal with diverse situations in a reasonable computation time.

For robotics applications, one may distinguish five dynamic problems useful for dynamic simulation :

1. Ballistic movements which can be treated using the Newton-Euler law (Isaac Newton, 1642-1727 and Leonhard Euler, 1707-1783) or Lagrangian formalism (Louis de Lagrange, 1736-1813).
2. Constraints due to mechanical articulations which can be represented at a macroscopic level by geometrical constraints,
3. Normal responses for collisions which are difficult to model and for which there are different physical laws ; based on geometrical constraints, impulsive forces or penalty forces.
4. Friction forces that are efficiently computed by the means of the Coulomb law (Charles de Coulomb, 1736-1806).
5. Deformations, modeled by finite elements or mass-connectors networks, which must be combined with the four above phenomena and add difficulties.

In this paper, we will present a study of different models for collisions and a system designed to simulate the behavior of deformable and rigid colliding bodies in a unified framework consistent with the previously cited Dynamic laws.

2 Related Works

From its inception, Mechanics has been the main contribution to dynamic simulation [18, 6, 17]. More recently, we can distinguish three principal fields concerned with dynamic simulation in real time. The first one is the dynamic control of kinematic chains and deals mainly with the problem of constraints introduced by articulations [4, 5]. The second one is computer graphics. In this field, complex geometric models introduce so many different control parameters that physically correct behavior is hard to obtain even for an experienced operator. Thus, in order to reduce the number of degrees of freedom, simplified dynamic models are used. The most frequent approach consists in considering collisions and contacts as geometrical constraints, and thereby obtains a uniform treatment of both interactions and articulations [1].

Finally, in robotics, dynamic simulations appear to be useful for planning or teleoperation. For this, the different models and algorithms developed for kinematic chains may be applied, but the representation of contacts by geometrical constraints leads to physically incorrect approximations [14]. Furthermore, the constraints-based model induces a global representation of all constraints by a system of non-linear equations which is difficult to maintain and solve (NP-hard).

In opposition to this global and geometrical representation of contact, we present two models based on physical approaches : “impulse” and “penalty”.

Notations In the following sections, we will assume that it is possible to establish an orthonormal frame associated with the collision. The z axis will be aligned with the direction of contact (determined by the relative velocity of the objects) and called the normal direction. The x and y axis define the tangential plan of the collision. Let x , v and a be the position, velocity and acceleration vectors.

2.1 Impulse based Models

Introduction For a neophyte it is easy to imagine that two rigid objects colliding at high speed will rebound immediately. Thus, the collision itself can be considered as instantaneous, and contact forces as impulsive.

This assumption of “instantaneosity” is the base of the theory of impact, also called stereomechanics in [6]. Since the collision is considered instantaneous, it is not necessary to detail what really happens during it, and the phenomenon must be described by a relation between the state variables before ($t_i = t_c^-$) and after the collision ($t_f = t_c^+$). In addition, positions remain constant and thus the only change concerns velocities.

The simplest relation is given by the Newton impulse model which establishes the relation : $v_z(t_f) = -ev_z(t_i)$ where e is the coefficient of restitution. Since $e \in [0, 1]$, the Newton relation shows that the object cannot rebound faster that it arrived. If $e = 1$ the collision is perfectly elastic and plastic if $e = 0$. This model is far too simple for it is impossible to integrate the Coulomb law for friction because there is no expression of normal forces.

Combining Impulse and Coulomb The Poisson law (Siméon Denis Poisson, 1781-1840) establishes that $p_z(t_f) = (e - 1)(p_z(t_{mc}))$ where p is the impulse, defined by $p = \int F dt$ or $\Delta m \vec{v}$ (variation of the linear momentum). The main difference with the previous law is that even if the phenomenon is instantaneous, it has to be divided into two phases, compression and restitution, separated by a maximum compression point at t_{mc} .

It is important to note that even if this law requires a partial study of what happens during the collision, it is still not a continuous representation of the whole phenomenon relative to time. It can even be shown that time cannot be used to evaluate $p_z(t_{mc}) = \int_{t_i}^{t_{mc}} F(t) dt$. Because the collision is instantaneous, we should evaluate

$$\lim_{t_{mc} \rightarrow t_i} \int_{t_i}^{t_{mc}} F(t) dt$$

and in order to get a non null value of p_{mc} , $F(t)$ must take an infinite value which is not realistic [2]. Nevertheless, while an expression of forces as a function of time cannot be used, the Coulomb law requires some expression of the evolution of forces.

Practical implementation A possible solution is to describe the evolution of the collision according to a parameter other than time. According to B. Mirtich, this parameter γ must be continuous and increase monotonically. The following is an abridged presentation of his approach, for further details, see [15].

As $F = m\ddot{x} = m\dot{v}$, $p = \int F(t) dt$ can be rewritten to get $p = \int_1^2 m \dot{v} = m(v_2 - v_1) = m\Delta v$. Introducing γ , we obtain $p(\gamma) = m(v(\gamma) - v(0))$. For simplicity, the matrix M is introduced, which yields

$$Mp(\gamma) = v(\gamma) - v(0) \tag{1}$$

and by differentiation with respect to γ ,

$$M \frac{d}{d\gamma} p(\gamma) = \frac{d}{d\gamma} v(\gamma) \tag{2}$$

Equation (2) is justified by the fact that M is constant, nonsingular, symmetric and positive definite.

The problem of choosing γ is still not resolved. B. Mirtich proposes integrating the ODE (2) with respect to p_z . With $\gamma = p_z$, he demonstrates that the vector $\frac{d}{d\gamma} p(\gamma)$ is a function of μ_c , v_x and v_y where μ_c is the coefficient of friction when the tangential velocity is nonzero ($v_x \neq 0 \neq v_y$). This yields a differential equation with respect to p_z ,

$$\frac{d}{d\gamma} v(\gamma) = M \begin{pmatrix} -\mu_c v_x / \sqrt{v_x^2 + v_y^2} \\ -\mu_c v_y / \sqrt{v_x^2 + v_y^2} \\ 1 \end{pmatrix} \tag{3}$$

As the parameterization is a projection among the z axis, it is logical to have the third element of the right-hand-side equal to 1.

Collision integration The integration of the equation (3) allows the computation of the evolution of velocity during the collision. When the normal velocity $v_z = 0$, the maximum compression is reached (γ_{mc}). It is then possible to evaluate p_{mc} and, by multiplying it by $(e - 1)$, find the value of total impulse $p(\gamma_f)$. To determine $v(\gamma_f)$, equation (3) is integrated until the total impulse is equal to $p(\gamma_f)$.

In a recent publication [19], Stronge shows that in some situations the Poisson law may introduce an energy gain in the system. To avoid this, he proposes a new definition for e , as a ratio between the work before and after the collision:

$$W_z(t_f) = (1 - e^2)W_z(t_m c) \quad (4)$$

B. Mirtich introduced this criterion in the collision integration. The formulas are more complicated and the parameterization is different. As all details and results cannot be treated in this brief presentation, we encourage the reader to consult [14].

Discussion The main problem solved by this approach is the taking into account of friction forces using the Coulomb law in an instantaneous representation of collision. In a complete simulator, such collisions will be considered one after the other because two instantaneous phenomena can be considered as non simultaneous.

Since the Poisson law has been utilized for many years, the restitution coefficient, e , has already been identified for different materials. However, it has recently been shown that the factor e also depends on the speed of collision and this relation may be approximated by a linear function, $e = 1 - \alpha v_i$ [7].

The foremost problem of Impulse-based collision models appears when collisions between two objects are frequent and ballistic phases are short. When you drop a rigid parallelepiped on the ground, stability should be reached after a few rebounds, but this is not the case for a pure Impulse based simulation. Indeed, a resting contact is modeled through a train of collision and the frequency of impact tends to increase. In this specific case of resting contact, both objects can be considered as static in relation to each other. A local evaluation can be performed to detect this static contact and thereby switch to an adequate model.

For non rigid bodies, contact phases are often long in comparison to ballistic phases. However, the contact cannot be viewed as resting because both objects may still move or bend.

2.2 Penalty based Models

Introduction The goal of penalty models is to give a continuous representation of the collision phenomenon at a macroscopic level. It is obvious that a microscopic study should lead to a model of molecular structure, which does not seem feasible in real time. From a macroscopic point of view, the only thing that one can measure is the deformation of colliding bodies.

It is important to underline that the consideration of deformations does not imply that this model is restricted to highly deformable objects, almost all materials are deformed during a

collision.

Empirically, this deformation is close to a linear function of forces. Vice versa, the reactive force that first reduces the relative speed of the two objects and finally repulses them from each other, is proportional to the deformation. As this deformation is almost impossible to model, a common hypothesis is that the value of the interpenetration of the two volumes is equivalent to the deformation. Thus, $F = -\lambda x$ where x is a geometrical measure of the interpenetration. This representation by the Hooke law is suitable for perfect elastic collision. A viscous damping term is introduced in order to model plastic phenomenon and $F = -\lambda x - \mu v$. This model is quite similar to that of Kelvin-Voigt used for vibrational mechanics.

Three main problems are raised by this approach. The first one is related to sticking forces due to the viscous term of the force, the second to the representation of time and the third to the identification of λ and μ .

Sticking forces In figure (1), the collision force is plotted as a function of the penetration value. During the ballistic phase P_b , $F = 0$. Then, the force increases until it reaches maximum compression and reduces. (1.a) shows that at the end of the collision, the F is negative because of the damping term. This should not occur and tends to keep the two objects stuck together. K. H. Hunt and F. R. E. Crossley [7] propose a new formula to compute F that would respect the shape of (1.b); $F = -\lambda x - \mu xv$. This solution also suppresses the discontinuity of the repulsion force at the beginning of the collision.

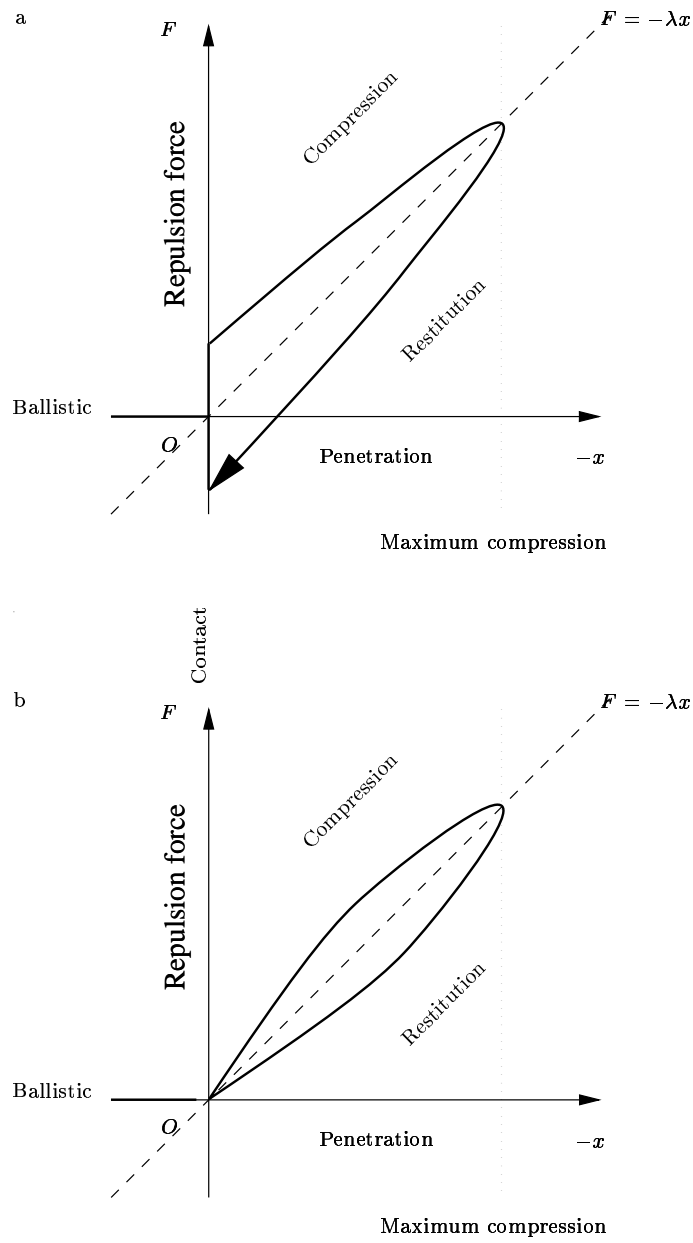
A continuous model In Mechanics, observations are made in a continuous world. Therefore, for implementation, a sampling is almost always introduced for the integration of movements over time. The value of this sampling is hard to determine. If it is too high, the computation cost will be prohibitive and if it is too low, strange phenomena may appear. Let A be a small particle falling on a thin plank (figure 2.a). At the second step (2.b), the algorithm has just detected the interpenetration and is now computing the repulsion force for the next time interval. If the coefficients λ and μ are small, the force will also be small and the speed will not significantly decrease. The particle can then go through the plank (2.c).

Fixing parameters One might suggest that the previous problem could be solved by increasing λ or μ . This would only be a trick and, from a physical point of view, definitively not a correct solution. Indeed, these parameters are supposed to represent the physical properties of the materials. To determine the parameters, a useful result is given by [7] which establishes a relation with the Poisson's coefficient of restitution ($\sim \alpha$, see section 2.1)

:

$$\mu = \frac{3}{2}\alpha\lambda \quad (5)$$

Intuitively, the damping term of F causes energy loss and its importance is relative to the value of λ , thus, the relation between e ($\sim \alpha$) and μ/λ is not surprising.

Figure 1: *Regular visco-elastic and non linear visco-elastic model*

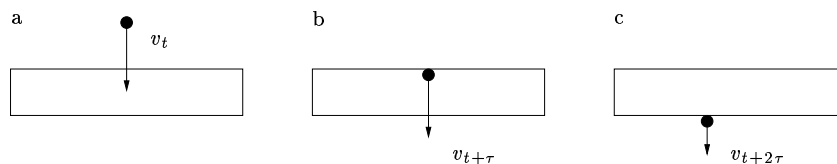


Figure 2: *A soft collision with a sampling that is too low*

Since α can be deduced from e and v_i , which are both known, λ is the only unknown. We present an ad hoc solution for λ in section 5.

Discussion The penalty model, based on a non linear visco-elastic force, is simple to implement and gives an explicit expression of the normal forces of repulsion which leads to an easy integration of friction forces with the Coulomb law. Contrary to what is usually affirmed, there is an automatic and physically logical means of determining the collision parameters λ and μ .

The remaining essential question is that of the time step required to obtain a consistent simulation.

Terminology The “penalty” based approach can also be viewed as an approximation method to ensure non penetration between two objects. In this case, the collision is modeled by a geometrical constraint, assuming that rigid objects cannot interpenetrate. The problem is then to solve a dynamic system $D(x) = 0$ under constraints $C(x) \leq 0$ which is NP-hard [1]. The approximate solution consists in minimizing $D(x) + \lambda C(x)$ where $\lambda C(x)$ appears to be a penalty function and λ is the weight of the cost function $C(x)$. Since $C(x)$ is a geometrical constraint, it is a distance or a volume that expresses the interpenetration. Finally, the cost function is $-\lambda x$.

Thus, both approaches lead to the same formalism but what appears as a physical parameter in one case is empirical in the other. The utilization of the model is also different. Indeed, the principle is to find the adequate collision response in one computation step. Therefore, this approach is definitely not continuous. The word “penalty” is employed to describe similar but distinct things, a numeric approximative method and a physical model.

In conclusion, one can assert that the “penalty” numerical method is not optimal for approximating repulsive forces in a dynamic constraint problem but this does not mean that the physical model is not valid.

2.3 Discussion

Both models are based on physical laws and use a local representation of contacts, which is also logical from a physical point of view. The difference lies in the physical variables used for these laws : force versus impulse. For the “penalty based” model, the simulation is continuous through time and provides the normal response as a force, whereas the “impulse

based” model instantaneously modifies the speeds of colliding objects. However, both detail what happens during the collision through a collision integration. This is essential since it has been proven that the tangential velocity change during the collision. Thus both models are physically more precise than a contact model based on unilateral constraints (which assumes a constant tangential velocity).

Finally, the impulse approach appears appropriate for simulations in which the duration of contact is brief as compared to the period of the colliding objects’ natural frequencies (rigid bodies) whereas the “penalty model” seems better adapted to simulations in which contacts are repeated and relatively long (deformable bodies).

As previously noted, almost all solids can be considered as deformable. Theoretically, when the rigidity of a deformable object tends to the infinite, the behavior of both models should be analogous. We present comparison results in section 5.

3 Modeling rigid and deformable objects

The main goal of our simulator is to integrate both rigid and deformable objects. To simulate deformations, we have retained a visco-elastic model which has been used for many mechanical applications [6, 20], as opposed to the finite element method for simplicity and rapidity. We give a summarized description of a simulator described in [8].

3.1 Discretization

In order to be modeled, an object has to be discretized into particles. The level of this discretization mainly depends on the deformability characteristics of the object. Usually, a very fine and uniform discretization of objects is used for modeling deformable objects. But this straightforward solution generates prohibitive execution times and numerical problems. Non-uniform discretization techniques, like the one used in [12], may also generate inconsistent behavior if the inertia properties of the objects are not preserved.

In order to solve such problems, we have developed a technique that subdivides each object into a set of components with the same inertia properties of the object (inertia matrix and inertia center) and in which each component is not subject to large deformations. The discretization is realized using a wave propagation technique (figure 3) applied from the inertia center of the object to model. The number and size of these components obviously depends on the deformability properties of the object to be modeled (i.e. deformable parts will be more deeply subdivided, and almost rigid parts will be represented by a few components). If the object is rigid, the resulting set of particles will form only one primitive. When it is deformable, each particle is considered as a primitive and appropriate connectors will be added between them.

Three kinds of connectors have been implemented, a linear connector which connects two particles and tends to maintain a resting distance, a torsion connector which connects three particles and tends to maintain an unsigned resting angle, and a joint connector which connects four particles to maintain a signed angle.

For a volumic model, it is necessary to maintain the angles between the edges of the network. A possible solution based on linear connectors consists in connecting all particles to get a set of triangles (like a roof's frame). This classical approach presents a major drawback : torques will not depend on the angle, but on the distances between particles. We have introduced torsion and joint connectors to solve this problem.

The second modeling step consists in giving a "volumetric" dimension to the objects, by adding facets delimited by the external particles of the previous network. These facets are necessary to process collision and contact phenomena.

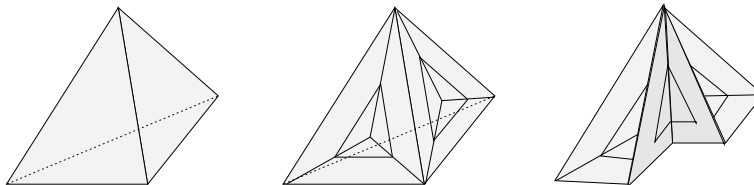


Figure 3: *Discretization of a convex polyhedron*

3.2 Identification

The parameters λ and μ associated with each connectors and the mass of particles have to be identified in order to obtain physically correct behavior.

For the computation of masses, we developed an algorithm which ensure that the mass and inertia properties are respected globally and locally. At a local level, the minimal element of discretization is tetrahedron that will not deform a lot during the simulation [9].

For λ and μ , it is possible to find some information about the compression and/or traction properties of certain materials. This can be used to determine the parameters of the linear connectors. Experimental data for torsion and joint connectors are more difficult to obtain since their influences cannot be isolated nor decoupled.

In order to solve this problem, we introduced an identification process based on a genetic algorithm [9]. This feature has been tested with success on a concrete case for a medical application [11]. In order to detect a venous thrombosis, the surgeon slides an echographic sensor along the skin of the thigh. If he/she applies too much pressure the clot can be released. Our simulator is used to model the behavior of a human thigh under pressure (figure 4). The model has also been used to simulate the behavior of an animal's Anterior Cruciate Ligament (figure 5). Here as well, results are totally satisfying since the margin of error does not exceed 5%. Moreover, surgeons agreed that the simulation was visually realistic [13].

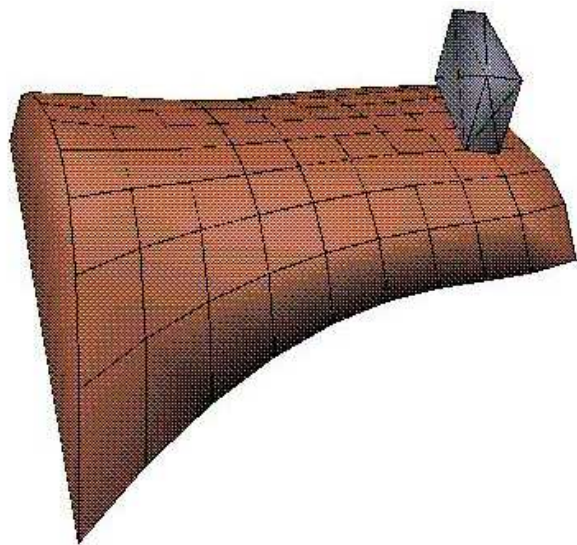


Figure 4: *Model of a human thigh with an echographic probe.*

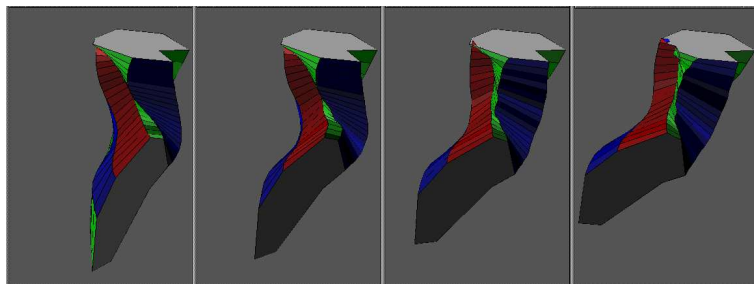


Figure 5: *Simulation of the behavior of the knee ligament.*

3.3 Interactions

For deformable bodies, interactions cannot be considered as instantaneous since they will induce a deformation which has to be analyzed with respect to time. For this reason, a penalty based model seems appropriate. We use the non linear visco-elastic model and the results presented in section 2.2 to determine the parameters of the reaction force.

The interpenetration is defined as the intersecting volume between the two objects. This requires a new efficient algorithm able to deal with negative distance for convex or non convex polyhedra. This algorithm is presented in [10].

Finally, both models of deformation and interaction are based on visco-elastic relations and this approach leads to a logical and coherent framework.

Nevertheless, none of the presented works deal with the problem raised by the utilization of a continuous model in a simulator based on a discrete representation of time. We present a physically based method to solve this problem in section 4.

4 Motion and deformation generation

4.1 Introduction

The motions and the deformations of each object are the result of the motion of its primitives. In our case, these primitives are rigid objects or particles which can be considered as a particular case of rigid objects.

Since each primitive is rigid, its motion can be computed using the Newton-Euler differential equations. As the primitives are linked by visco-elastics connectors we obtain following system of differential equations :

$$\vec{F}_i = m_i \vec{a}_i \quad (6)$$

$$\vec{N}_i = I_i \vec{\theta}_i + \vec{\theta}_i \wedge (I_i \cdot \vec{\theta}_i) \quad (7)$$

$$\vec{F}_i = f(\vec{x}_j, \vec{v}_j, \vec{\theta}_j, \vec{\theta}_j) + \vec{F}_i^I + \vec{F}_i^C | i, j \in [1..n] \quad (8)$$

$$\vec{N}_i = g(\vec{F}_i) \quad (9)$$

where n is the number of primitives, \vec{x}_i , \vec{v}_i , \vec{a}_i , the position, velocity and acceleration of the primitive i , $\vec{\theta}_i$, $\vec{\theta}_i$, $\vec{\theta}_i$ the orientation, rotation velocity and rotation acceleration around its inertia center. m_i and I_i are the mass and the inertia matrix of the primitive i . The firsts two equations are those of Newton-Euler. The last equation only converts forces into torques. The third equation is very significant since it represents the forces due to the connectors (relations to primitives j) through the function f . Since we consider linear and torsion/joint connectors, both position and orientation are required to compute f . The third equation also includes all external external phenomena; interaction forces \vec{F}_i^I and control forces \vec{F}_i^C . Unfortunately, this system is *neither continuous nor linear*, because it involves collision forces and control forces that can be applied at any time by the user on the environment.

Consequently, it is difficult to solve this system analytically or by an implicit¹ numerical method. A classical means of solving such a system is to use an explicit method based on a second order limited development with a time step τ :

$$\vec{v}_{t+\tau} = \vec{v}_t + \tau \vec{a}_t + O\left(\frac{\tau^2}{2} \vec{a}_t\right) \quad (10)$$

$$\vec{x}_{t+\tau} = \vec{x}_t + \tau \vec{v}_t + \frac{\tau^2}{2} \vec{a}_t + O\left(\frac{\tau^3}{6} \vec{a}_t\right) \quad (11)$$

$$\vec{\theta}_{t+\tau} = \vec{\theta}_t + I^{-1}(\vec{N}_t - \vec{\theta}_t \wedge (I \cdot \vec{\theta}_t))\tau \quad (12)$$

Some well known drawbacks of such methods for dynamical simulation are as follows:

- Such process is very sensitive to the chosen time step. If it is too large, this may lead to numerical divergence generated by the error term (figure 6). In physical terms, this means that the simulation goes too fast and reaches an incoherent position (abnormal compression or traction) because all modifications of forces (\vec{a}_t) related to \vec{F}_i^I and \vec{F}_i^C are neglected during τ .

On the other hand, if τ is too small, the execution time will be prohibitive.

Therefore, it is difficult to choose a time step which ensures numerical stability, because it depends on many factors like collision characteristic or deformability properties. In practice, the time step is usually chosen empirically after several attempts to execute the simulation process and the validity of the simulation is visually determined by an operator.

- The error term cannot be easily estimated, because the system has a non continuous behavior. This is a major drawback for robotics applications in real time.

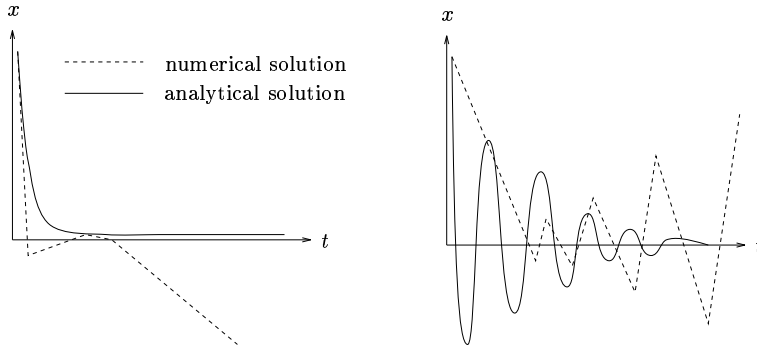


Figure 6: *Without a suitable time step the numerical solution may diverge*

¹If $x_{t+\tau} = f(x_\gamma) \mid \gamma \leq t$ the approach is explicit (incremental), else it is implicit.

4.2 Adaptative time step and error estimation

Problem and approach The equation (10) shows that the numerical error increases when the time step τ and/or the derivative of the acceleration, \vec{a} , increase. Since the value of \vec{a} cannot be controlled, the only way to decrease the error term is to decrease the time step, and consequently to increase the computational time. It is judicious to remark that when the external forces are null, the error will be negligible and a large sampling is not required. Thus, we used an adaptive time step τ (figure 7), where the sampling rate varies according to the variation of $|\dot{a}|$ in order to keep the error term less than a given threshold while using a few samples as possible.

The main problem comes from the fact that the value of \dot{a} is not easy to determine since we

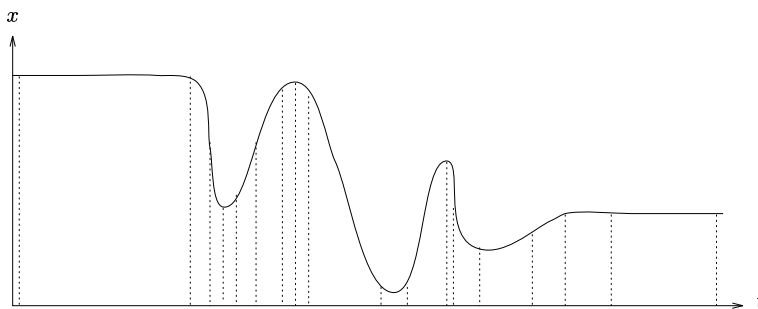


Figure 7: A good strategy to avoid divergence and reduce the computational time is to use an adaptive time step τ which allows the system to be integrated using a minimum number of samples.

have many simultaneous collisions and we assume that a user can add discontinuous forces through a virtual controller or a force feedback device.

In physical terms, the error can also be considered as a jump in time which induces an improbable state. Therefore, a Physical law must have been transgressed and this law could be used to determine the amplitude of the error. As we saw before, a too large time step can produce abnormal forces which, after an integration step, will cause a variation of mechanical energy. This variation is a transgression to the main property of the mechanical energy (which remains constant for an isolated system) and seems to be a good criteria to detect the error.

Let E_m be the mechanical energy of the physical system S considered ($E_m = E_p + E_k$, where E_p is the potential energy and E_k is the kinetic energy). E_m depends on \vec{x} and \vec{v} :

$$E_m = E_p + E_k = - \int_x^{x+\Delta x} \vec{F} \cdot d\vec{x} + \frac{1}{2} m \Delta v^2$$

Three cases have to be considered when reasoning about the variation of E_m :

- S in an isolated system without energy dissipation. This situation arises when an object (or a set of objects) is not subjected to an external force, nor friction phenomenon. In this case, E_m is constant.
- S is an isolated system with energy dissipation. This situation arises when a friction force appears between objects. In this case E_m decreases.
- S is a non-isolated system. This situation arises when S is subjected to an external forces (control force or collision). In this case E_m increases.

Time step determination The time step determination algorithm is basically applied during the time intervals where the object is considered as an isolated system. The extension of the domain of validity of this algorithm is discussed below. As previously explained, the mechanical energy E_m is **constant** when the system is isolated : $\Delta E_m = 0$. If ΔE_m increases during the numerical integration, this means that this “non physical” variation is the result of the error term. At this point, the idea is to monitor the value of $|\Delta E_m|$ at each time step and to choose the greatest time step which satisfies $|\Delta E_m| < \epsilon_e$ for a small ϵ_e (tolerated error). The following algorithm gives a practical implementation of this idea:

Let \vec{x}_t be the position of the object.
Let \vec{v}_t be its velocity.
Let \vec{F}_t be the sum of the forces applied on it.
LOOP
 Calculate $x_{t+\tau}$ and $v_{t+\tau}$
 Calculate ΔE .
 IF $|\Delta E| > \epsilon_e$ THEN
 LOOP
 $\tau = \frac{\tau}{2}$
 Calculate $x_{t+\tau}$ and $v_{t+\tau}$
 Calculate ΔE
 UNTIL $|\Delta E| < \epsilon_e$
 ELSE
 $\tau = \frac{3}{2}\tau$
UNTIL ∞

Justification $|\Delta E_m| > \epsilon_e$ means that the variation of energy is too large, and consequently the time step is too large; $|\Delta E_m| < \epsilon_e$ means that the probability that the state of the system is “correct” is very high. An intuitive justification of this assumption relies on the fact that two different states of the system having approximatively the same numerical value for E_m cannot be associated with two consecutive (and close) time step, because of the continuity of the object motions. For instance, when two objects penetrate each other, the mechanical energy increases continuously until they separate. Thus the two objects would have to cover

in a single time step a distance which would be greater than the sum of their dimensions in order to reach an incorrect state which verifies the energy conservation.

Dealing with external perturbations When an external force F_e (control or collision force) is applied to the system, E_m is no longer constant during a very short interval of time. After this event, we can take into account the potential energy generated by F_e and the system will returns to a state where E_m is constant. A safe and blind iteration (using a very small time step ϵ_τ) of the algorithm is necessary to evaluate the value of the new mechanical energy E_m . A good way to choose ϵ_τ , which is consistent with the algorithm described above, is to search for the time step ϵ_τ for which ΔE_m between t and $t + \tau$ is smaller than ϵ . It is always possible to find such time step because of the **continuity** of E_m . Consequently, the previous algorithm still applies where an external force is applied to the system.

Discussion The two main difficulties raised by the numerical integration of the dynamic equations were : firstly, how to choose a time step which would lead to a physically correct simulation and, secondly, what kind of compromise between computation time and physical consistency would be acceptable.

The solution presented is not a compromise because the sampling frequency is determined in order to maintain the physical accuracy. Nevertheless, since the time step is adaptative, a very small time step adapted to a given phenomena will not penalize the whole simulation and the global computation time remains acceptable.

More importantly, monitoring the time step and therefore the error through the value of the mechanical energy solves a crucial problem. Indeed, even with a constant time step, as small it might be, it is almost impossible to ensure that the sampling frequency is high enough. This practical approach leads to a high probability of avoiding the numerical divergence for both deformable and contact penalty models.

5 Experiments

5.1 Protocol

In order to test our model of contact and the algorithms developed to compute the movements, we did some experiments similar to those realized in [16]. This work has the advantage of giving numerical results.

The experiment consists in estimating the pose statistic of different small rigid parts falling on part feeders. This information is then used to optimize the motions of a robot arm which picks up the part and places them correctly (further details in [14]).

We tested our approach with two of the four presented parts; the orange insulator cap and the large stereo button (figure 8). Exact CAD models have been furnished by B. Mirtich as well as the physical parameters. Both restitution and friction coefficients were estimated to be equal to 0.3. At the moment of impact, the simulator computes $\alpha = \frac{1-e}{v_i}$. Since $\alpha = \frac{2\mu}{3\lambda}$

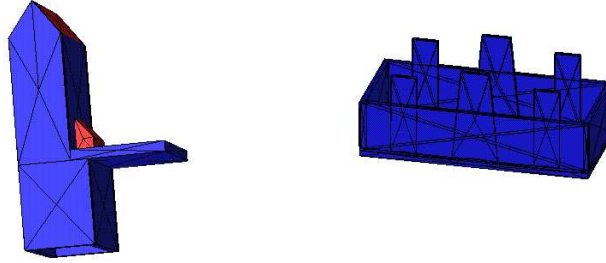


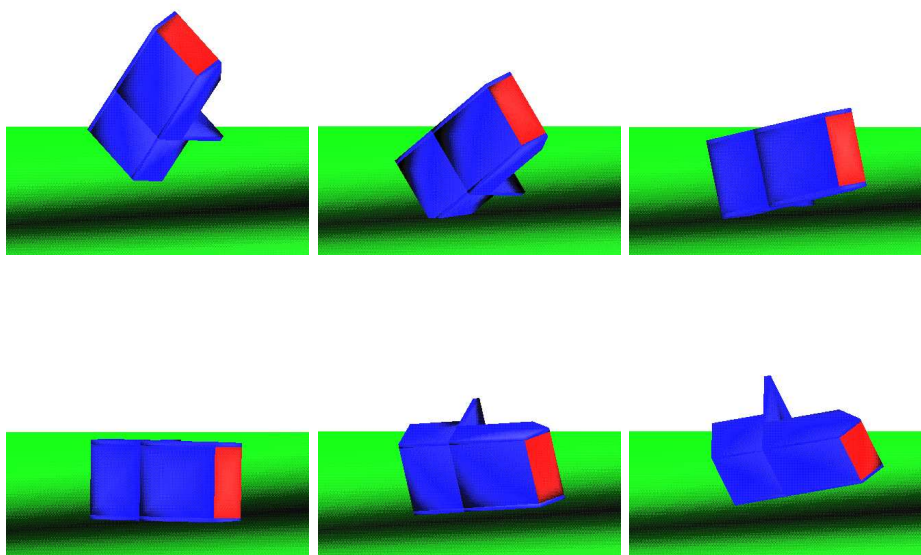
Figure 8: *The insulator cap and the stereo button*

the remaining unknown is λ . We believe that λ can be experimentally deduced, but since the parts were not available, we fixed λ by an other means. Indeed, this parameter represents the rigidity of the collision and tends to enforce a non penetration constraint. Thus, we experimentally determined λ in that the interpenetration is very small compared to the size of the object (less than 0.5% of the length). This leads to a computation at $320Khz$ at the worse moment of the collision but the average time step is close to $0.00072s$, i.e. $1.4Khz$. In most cases, we determine λ through the deformation of the material under static load. This solution is very sensitive to the shape of the cells and can only be applied with face to face contact. However, for deformable bodies, surfaces are discretized and a high percentage of them is represented by cells in face to face contact (all central faces). The two parts are non convex and therefore modeled as an assembly of rigid convex polyhedra (figure 8). For interactions, we consider the sum of the forces due to the interactions' volumes of each sub-polyhedron. This sum is then applied to the whole object (figure 9).

5.2 Results

The results are presented in tables 1 and 2. Let p_1, \dots, p_n represent the probability of each of n states, as measured in the physical test. Let a_i, \dots, a_n represent the corresponding probabilities computed by a given simulator. The error percentage for this simulator is equal to :

$$e = 100 \sum_{i=1}^n p_i |a_i - p_i|$$

Figure 9: *The insulator cap falls and rebounds*

| Pose | Impulse | Penalty | Physical tests |
|-------|---------|---------|----------------|
| 1 | 41.9 | 43.0 | 46.0 |
| 2 | 26.2 | 29.5 | 27.1 |
| 3 | 28.3 | 20.1 | 19.7 |
| 4 | 3.0 | 4.0 | 5.0 |
| 5 | 0.8 | 2.7 | 2.2 |
| 6 | 0.0 | 0.6 | 0.0 |
| error | 4.0 | 2.2 | - |

Table 1: *Orange insulator cap statistics*

| Pose | Impulse | Penalty | Physical tests |
|-------|---------|---------|----------------|
| 1 | 71.7 | 70.4 | 75.8 |
| 2 | 20.9 | 20.5 | 13.8 |
| 3 | 7.4 | 8.9 | 10.5 |
| 4 | 0.1 | 0.25 | 0.0 |
| error | 4.4 | 5.2 | - |

Table 2: *Stereo button statistics*

5.3 Discussion

These results have been obtained without any particular tuning. They are quite close to the impulse based simulations and the real which verify the accuracy of our approach. This is very encouraging since the penalty based model is not particularly well suited for stiff collisions.

Our conclusion is not that the penalty based model is better than the impulse based one in any case. Indeed, to simulate 995 drops for the first part and 776 for the second part, we used 3 workstations (*SiliconGraphics Indy R4600 100Mhz*) over a pair of hours whereas the impulse based simulation required 45 minutes for 2000 drops (*SiliconGraphicsIndigo₂R4400200Mhz*). As previously mentionned, we did not modify the simulator for those experiments and we believe that the execution time could be seriously reduced with some simple optimizations (it was originally developed for deformable objects).

6 Conclusions

We have presented a simulator which is able to model the movement of both rigid and deformable objects. This simulator is based on a mass-spring model for the deformation and the penalty model for the interactions. This representation is unified because both use visco-elastic relations. Nevertheless, it is usually said that this approach is physically not correct. In order to correct this, we introduced :

- A relation which allows determining the collision parameters as function of the coefficient of restitution e and the elasticity of the materials.
- A practical criterion based on the mechanical energy of the system which allows detecting numerical divergence due to wrong time step.

We implemented and tested these methods in order to validate the simulator. We reproduced previously released experiments and get quite close results. This tends to prove the physical accuracy of our system.

In future, we plan to study in detail (velocity, acceleration, momentum, energy, work) the two different models during the collision with analytical and numerical integration.

The ultimate goal is to combine different algorithms corresponding to different phenomena. In [14], two models based on geometric constraints and impulses have been integrated in order to simulate respectively articulations and impacts. We believe that it would be interesting to integrate impulse and penalty in a unique simulator to model rigid and deformable interactions.

Acknowledgment We would like to thank John Canny to have invited Anton Deguet at Berkeley during summer 96, and Brian Mirtich for his help and the data he sent us.

References

- [1] D. Baraff. *Dynamic Simulation of Non-Penetrating Rigid bodies*. PhD thesis, Cornell University, March 1992.
- [2] B. Brogliato. *Nonsmooth Impact Mechanics*. Springer-Verlag, London, 1996.
- [3] M. Cherif and C. Laugier. Motion planning of autonomous off-road vehicles under physical interaction constraints. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, Nagoya (JP), May 1995.
- [4] J. J. Craig. *Introduction to Robotics, Mechanics and Control*. Addison-Wesley, second edition, 1989.
- [5] R. Featherstone. *Robot Dynamics Algorithms*. Kluwer Academic Publishers, 1987.
- [6] W. Goldsmith. *Impact: The Theory and Physical Behaviour of Colliding Solids*. Edward Alrnold, London, 1960.
- [7] K. H. Hunt and F. R. E. Crossley. Coefficient of restitution interpreted as damping in vibroimpact. *Journal of Applied Mechanics*, pages 440–445, June 1975.
- [8] A. Joukhadar. *Simulation Dynamique et Applications Robotiques*. Thèse de doctorat, Inst. Nat. Polytechnique de Grenoble, Grenoble (FR), June 1997.
- [9] A. Joukhadar, F. Garat, and C. Laugier. Parameter identification for dynamic simulation. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, volume 3, pages 1928–1933, Albuquerque, NM (US), April 1997.
- [10] A. Joukhadar and C. Laugier. Dynamic simulation: Model, basic algorithms, and optimization. In *Proc. of the Workshop on the Algorithmic Foundations of Robotics*, Toulouse (FR), July 1996.
- [11] P. Laffont. Simulation dynamique pour le diagnostic de thromboses veineuses. Mémoire de Diplôme d'Etudes Approfondies, Université de Savoie, Chambéry (FR), 1997.
- [12] A. Luciani and al. An unified view of multiple behaviour, flexibility, plasticity and fractures: balls, bubbles and agglomerates. In *IFIP WG 5.10 on Modeling in Computer Graphics*, pages 55–74. Springer-Verlag, 1991.
- [13] S. Martelli, A. Joukhadar, S. Zaffagnini, M. Marcacci, S. Lavallee, and G. Champlébourg. A fiber-based ACL model for geometrical and mechanical simulations. Research Report 2924, Inst. Nat. de Recherche en Informatique et en Automatique, Montbonnot Saint Martin (FR), July 1996.
- [14] B. V. Mirtich. *Impulsed-based Dynamic Simulation of Rigid Body Systems*. PhD thesis, University of California at Berkeley, 1996.

- [15] B. V. Mirtich and J. F. Canny. Impulse-based dynamic simulation. In *Proc. of the Workshop on the Algorithmic Foundations of Robotics*, San Francisco, CA (US), February 1994.
- [16] B. V. Mirtich, Y. Zhuang, K. Goldberg, J. Craig, R. Zanutta, B. Carlisle, and J. F. Canny. Estimating pose statistics for robotics part feeders. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, volume 2, pages 1140–1146, Minneapolis, MN (US), April 1996.
- [17] L. A. Pars. *A Treatise on Analytical Dynamics*. Heinemann, London, 1965.
- [18] E. J. Routh. *Dynamics of a System of Rigid Bodies*. Dover, sixth edition, 1905.
- [19] W. J. Stronge. Unraveling paradoxical theories for rigid body collisions. *Journal of Applied Mechanics*, 58:1049–1055, December 1991.
- [20] S. Timoshenko and J. N. Goodier. *Theory of Elasticity*. McGraw-Hill, second edition, 1951.



Unit ´e de recherche INRIA Lorraine, Technop ˆole de Nancy-Brabois, Campus scientifique,
615 rue du Jardin Botanique, BP 101, 54600 VILLERS LÈS NANCY
Unit ´e de recherche INRIA Rennes, Irista, Campus universitaire de Beaulieu, 35042 RENNES Cedex
Unit ´e de recherche INRIA Rh ˆone-Alpes, 655, avenue de l'Europe, 38330 MONTBONNOT ST MARTIN
Unit ´e de recherche INRIA Rocquencourt, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex
Unit ´e de recherche INRIA Sophia-Antipolis, 2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex

´Editeur
INRIA, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)
<http://www.inria.fr>
ISSN 0249-6399