Umeå University
Department of Computing Science
SE-901 87 Umeå, Sweden

UMINF-06.04
ISSN 0348-0542
Revised: March 5 2006

A REGULARIZED TIME STEPPER FOR MULTIBODY SYSTEMS *

CLAUDE LACOURSIÈRE†

Abstract. A Lagrangean framework for physical systems consisting of rigid bodies, kinematic constraints of various types and dissipative forces, is presented. This is used to analyze the relationship between strong penalty forces and constraints as well as to provide a discretization which behaves nicely in the singular limit where the penalty forces become infinitely strong. The framework is also used to analyze the residual errors of approximate methods used to solve constrained systems and the typical regularization terms which are introduce to solve the stepping equations more efficiently. It is shown that the approximate solution of the constrained system is in fact the exact solution of a penalty system but in some cases, the penalty system is unstable because the stiffness and damping terms are not guaranteed to be positive[1].

Key words. 90C33, 90C20, 90C55, 90C90, 65Z05, 65L80, 65L80, 65L20

AMS subject classifications. Complementarity problems, Quadratic programming, Methods of successive quadratic programming type, Applications of mathematical programming, Applications to physics, Methods for differential-algebraic equations, Stability and convergence of numerical methods.

*This research was conducted using the resources of High Performance Computing Center North (HPC2N), and supported in part by the "Objective 1 Norra Norrlands" EU grant awarded to HPC2N/VRlab at Umeå University, and in part by the *Swedish Foundation for Strategic Research* under the frame programme grant SSF-A3 02:128, and in part by the CMLabs research and development programme.

†HPC2N/VRlab and the Department of Computing Science, Umeå University, S-901 87 Umeå, Sweden, and CMLabs Simulations Inc., 420 Notre-Dame Ouest, Suite 505, Montréal, Qc, H2Y 1V3, CANADA, Email:claude@hpc2n.umu.se

[1]Presented at the International Workshop on PDE Methods in Computer Graphics March 31 - April 1, 2005 University of Copenhagen, Denmark

1. Introduction. Lagrangean dynamics is the most general framework for the analysis of any physical system as the equations of motion of any complicated system can be derived from work-energy principles, irrespective of the coordinate system that is chosen. This is used here to model rigid multibody systems subject to a variety of constraints, including both holonomic and non-holonomic constraints, as well as both ideal (workless) and non-ideal constraints, including, in particular, contact and dry friction constraints.

The equations of motion are derived in descriptor form which has gained popularity in interactive physics simulations but is still shunned in robotics. It is argued below that it is nevertheless possible to generate efficient numerical methods to integrate the equations of motion and that in addition, the theoretical framework allows to include very stiff forces in a stable and efficient way.

The theoretical relation between kinematic constraints and the singular limits of strong forces is well known [10][13], but this fact has not been exploited in connection with the derivation of stable stepping schemes so far. The representation of constraints as the singular limit of very strong potential actually regularizes the equations of motion as they become Differential Algebraic Equations (DAEs) of index 1 instead of index 3, which are much harder to solve. It can be argued that this drastically changes the nature of the physical system but on the other hand, a small amount of compliance and dissipation is always present in real systems. Thus, not only is the regularized formulation easier to process numerically, it is in fact more realistic.

In addition, the regularized formulation allows the mapping of the perturbation terms and residual errors which are found in common solution methods back to an approximate physical system which is solved exactly by the numerical method. Though this analysis is far from complete, it provides insight into common stability problems and potential remedies.

The paper is organized as follows. Section 2 covers the basics of Lagrangean dynamics applied to rigid body systems and presents different classes of constraints. Section 3 introduces the regularization framework and this is then discretized in section 4. Solution methods and correspondence between numerical approximations and regularized physical problems is covered in section 5. Results of computational experiments are covered for two simple illustrative examples in section 6, and some future avenues are sketched in section 7.

2. Lagrangean dynamics. Key results of Lagrangean dynamics are summarized here with emphasis on rigid multibody systems with several types of constraints.

2.1. Rigid body kinematics. Consider a collection of n rigid bodies, each labeled with an index i. A body has its center of mass at position $x^{(i)} \in \mathbb{R}^3$ and orientation defined by the quaternion $e^{(i)}$. Quaternions are chosen to parametrize the group of rigid rotations in $\mathbb{R}^3$ because this representation is free of singularity, unlike the case for the more common Euler angles. The coordinates are agglomerated into a generalized coordinate vector $q^{(i)} = (x^{(i)^T}, e^{(i)^T})^T$ where $(\cdot)^{(i)^T}$ is the transpose of the vector or matrix with index i. The orthogonal matrix $R(e^{(i)}) = R^{(i)} \in \mathbb{R}^{3\times3}$, transforms vectors from the frame of reference attached to body i into the global

inertial frame of reference. The convention for the transformation matrix R is:

$$R(e) = \begin{bmatrix} (2e_0^2 - 1) + 2e_1^2 & 2(e_1e_2 - e_0e_3) + & 2(e_0e_2 + e_1e_3) \\ 2(e_0e_3 + e_2e_1) & (2e_0 - 1) + 2e_2^2 & 2(e_2e_3 - e_0e_1) \\ 2(e_1e_3 - e_0e_2) & 2(e_0e_1 + e_3e_2) & (2e_0^2 - 1) + 2e_3^2 \end{bmatrix}. \tag{2.1}$$

Given a vector y' expressed in the frame of body i, the global inertial frame representation of this vector is given by: $y = x^{(i)} + R^{(i)}y'$, and an axial vector n' attached to the body frame has the representation $n = R^{(i)}n'$ in the global inertial frame. The velocity of rigid body i is represented with a 6 dimensional vector as:

$$\begin{aligned} v^{(i)} &= \begin{bmatrix} \dot{x}^{(i)} \\ \omega^{(i)} \end{bmatrix} = \bar{T}(q^{(i)})\dot{q}^{(i)}, \text{with } \bar{T}(q^{(i)}) = \text{diag}(I, \tilde{T}(e^{(i)})), \text{ and} \\ \tilde{T}(e) &= \frac{1}{2}\begin{bmatrix} -e_1 & -e_2 & -e_3 \\ e_0 & -e_3 & e_2 \\ e_3 & e_0 & -e_1 \\ -e_2 & e_1 & -e_0 \end{bmatrix}. \end{aligned} \tag{2.2}$$

Here, $\dot{x}^{(i)}$ is the velocity of the center of mass and $\omega^{(i)}$ is the angular velocity vector, expressed in the global inertial frame.

2.2. Mass Properties. The mass of rigid body i is $m^{(i)}$, a positive scalar, and the inertia tensor in the body frame, written as $\mathcal{I}_0^{(i)}$, is a 3×3 symmetric positive definite matrix. The inertial frame representation of the inertia tensor is written as $\mathcal{I}^{(i)} = R^{(i)}\mathcal{I}_0^{(i)}R^{(i)^T}$. The inertia matrix of the rigid body is then written as: $M^{(i)} = \begin{bmatrix} m^{(i)}I & 0 \\ 0 & \mathcal{I}^{(i)} \end{bmatrix}$, where I is the 3×3 identity matrix.

The net external generalized force acting on a rigid body is written as $F^{(i)}$, which is a 6 dimensional vector combining the force acting at the center of mass and the torque.

The quantities associated to each rigid body are agglomerated in state vectors as follows. The generalized coordinate vector is written as: $q = ({q^{(1)}}^T, {q^{(2)}}^T, \ldots, {q^{(n)}}^T)^T$, and similarly for the generalized velocity vector: $v = ({v^{(1)}}^T, {v^{(2)}}^T, \ldots, {v^{(n)}}^T)^T$, and the generalized force vector: $F = ({F^{(1)}}^T, {F^{(2)}}^T, \ldots, {F^{(n)}}^T)^T$. We have the relationship:

$$v = T(q)v, \text{with } T(q) = \text{diag}(\bar{T}(q^{(1)}), \bar{T}(q^{(2)}), \ldots, \bar{T}(q^{(n)})). \tag{2.3}$$

The mass matrix of the system is $M = \text{diag}(M^{(1)}, M^{(2)}, \ldots, M^{(n)})$.

2.3. Multibody systems. Using the previous definitions for the generalized coordinates $q(t)$, generalized velocities $v(t)$, and mass matrix $M(q(t))$, we now derive the equations of motion for the multibody system using a Lagrangean formulation.

The kinetic energy for the system is written as $K(q, v) = \frac{1}{2}v^T M(q)v$, and the mass matrix $M(q)$ depends on the coordinates. We also assume a potential energy real scalar function, $U(q)$ which is both velocity and time independent. This restriction is not necessary but it simplifies the notation.

2.4. Unconstrained closed system. For the case at hand, the Lagrangean is the scalar function: $\mathcal{L}(q,\dot{q}) = K(q,\dot{q}) - U(q)$, and this leads to the Euler-Lagrange equations of motion:

$$\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{q}^T} - \frac{\partial \mathcal{L}}{\partial q^T} = 0, \tag{2.4}$$

through Hamilton's least action principle [4],[10]. For the case at hand, after we take the constraint $\|e^{(i)}\| = 1$ and the relation $\dot{q} = T(q)v$ into account, these equations read:

$$\begin{aligned} \dot{q} &= T(q)v, \quad \text{with } |e^{(i)}| = 1 \\ M\dot{v} + \dot{M}v + T^T(q)\frac{\partial U}{\partial q^T} &= 0. \end{aligned} \tag{2.5}$$

We write $F_M = -\dot{M}v$ for the gyroscopic forces.

This model can be augmented in several different ways which we cover next. The connection to Newtonian physics is established by identifying the force vectors acting on the system as $F = -T^T(q)\partial U/\partial q^T$. Not all force models can be represented as potentials, the notable exception being dissipative drag forces, as well as dry friction forces. This will be covered in sections 2.9 and 2.8 below.

2.5. Kinematic constraints. Kinematic constraints restrict the range of the generalized coordinates, $q(t)$. The canonical example is a vector function of the coordinate $\Phi(q) : \mathbb{R}^N \mapsto \mathbb{R}^m$, and the constrained system must now satisfy the condition $\Phi(q(t)) = 0$ on the entire trajectory. A simple example of this is $\Phi(q(t)) = q_i(t) = 0$, which fixes one of the coordinates of one of the rigid bodies at the origin for all time. The time derivative is written as $\dot{\Phi}(q(t)) = (\partial\Phi/\partial q)T(q)v = Gv$ to simplify the notation. The matrix $G \in \mathbb{R}^{m\times n}$ is the Jacobian of the function Φ. The vector of m-constraint equations leads to a real, m-dimensional vector of Lagrange multipliers $\lambda \in \mathbb{R}^m$, with one entry for each constraint. The new Lagrangean becomes $\mathcal{L}(q,v,\lambda) = K(q,v) - U(q) - \lambda^T\Phi(q)$, where the $\lambda(t)$ is an unknown vector to be determined, and the new equations of motion, which are obtained by finding the saddle point condition for the quantity $A(t_0,t_1,q,v,\lambda(t)) = \int_{t_0}^{t_1} ds\mathcal{L}(q(s),v(s),\lambda(s))$ over all possible trajectories $q(s), v(s), \lambda(s)$. The saddle-point condition is a minimum over the trajectory $q(s), v(s)$, and a maximum over the functions $\lambda(s)$. The resulting equations of motion are:

$$\begin{aligned} \dot{q} &= T(q)v \\ M\dot{v} + T^T(q)\frac{\partial U}{\partial q^T} - G^T\lambda &= F_M \\ \Phi(q) &= 0. \end{aligned} \tag{2.6}$$

Details of the derivation are found in both Goldstein [4] and Lanczos [10]. The equations of motions are now DAEs of index 3 (for a definition, see Ref. [6]). These equations are much more difficult to integrate than standard Ordinary Differential Equations (ODEs) and it is difficult in general to stay on the constraint surface $\Phi(q) = 0$ or to stabilize the motion so that $|\Phi(q)|$ does not drift too far from 0.

Though the laws of physics should provide an explanation for all observable motion without the need for kinematic constraints, these are necessary for modeling macroscopic motion. Kinematic constraints represent the net effect on the motion of physics which occurs at very small time and length scales and which would be inefficient to model in full.

Note that the force due to the constraint is $F_c = G^T\lambda$ and since $\Phi(q) = 0$, then, $\dot{\Phi}(q) = 0 = Gv$. Therefore, the work done by the constraint is $W_c = F_c^T v = \lambda^T G v = 0$, i.e., this type of constraint is workless or *ideal*. This will also hold for velocity and contact constraints.

2.6. Velocity constraints. In addition to kinematic constraints, it is often useful to describe drivers which impose a given velocity on a joint coordinate for instance. These constraints are non-holonomic, i.e., they cannot be used to reduce the number of coordinates in a global way. The analysis is restricted to the case of Pfaffian form constraints: $\Upsilon(q, v) = J(q)v - w_0(t) = 0$, where $\Upsilon(q, v)$ is an r-dimensional real vector, $J(q)$ is an $r \times n$ dimensional real matrix function of the coordinates q, and w_0 is an r-dimensional real, constant vector. These constraints introduce an r dimensional vector of unknown Lagrange multipliers, $\rho(t)$. However, unlike the case of holonomic constraints, these cannot be set in a variational framework and so the Lagrangean is still: $\mathcal{L}(q, v, \lambda, \rho) = K(q, v) - U(q) - \lambda^T\Phi$ but the equation of motion which are consistent with the constraints and the d'Alembert principle are:

$$\begin{aligned} \dot{q} &= T(q)v \\ M\dot{v} + T^T(q)\frac{\partial U}{\partial q^T} - G^T\lambda - J^T\rho &= F_M \\ \Phi(q) &= 0 \\ J(q)v - w_0(t) &= 0. \end{aligned} \tag{2.7}$$

Details of the derivation are found in Goldstein [4], Lanczos [10], and Layton [11].

2.7. Contact Constraints. Inequality or contact constraints are written implicitly as $\Xi(q, t) \geq 0$ and we write $\dot{\Xi} = Nv + \partial\Xi/\partial t$ where N is the Jacobian of Ξ. In the case of a non-penetration contact constraint, the rows of N are the normals of the contact points. The non-penetrating contact constraint Ξ is made up of a number of n_c contact constraints: $\Xi = (\xi^{(1)}, \xi^{(2)}, \ldots \xi^{(n_c)})^T$, and each of the $\xi^{(j)}(q)$ is a simple signed distance function, which depends on the coordinates of at most two of the rigid bodies in the system. Corresponding to the contact constraints, we have a vector of Lagrange multipliers ν which is n_c-dimensional. The vector ν can only be positive for the case where the contacts are not adhesive. Also, the vector ν is non-zero only when there is a contact, i.e., $\nu^{(j)} > 0 \rightarrow \xi^{(j)} = 0$ and likewise, the Lagrange multipliers vanish when there is no contact, i.e., $\xi^{(j)} > 0 \rightarrow \nu^{(j)} = 0$. This is a complementarity condition. The details of how to compute the contact constraints given rigid body geometry is not covered here but there is ample and adequate literature on this topic. The reader may consult Ref. [3] for a recent survey of techniques and a thorough bibliography.

Using the same method as above, the following equations of motion are found:

$$\begin{aligned} M\dot{v} + T^T(q)\frac{\partial U}{\partial q^T} - G^T\lambda - J^T\rho - N^T\nu &= F_M \\ \Phi(q) &= 0 \\ J(q)v - w_0(t) &= 0 \\ 0 \leq \Xi(q) \perp \nu &\geq 0, \end{aligned} \tag{2.8}$$

where the inequality conditions are understood component wise, and the perpendicular operator means that if $0 \leq x, \perp y \geq 0$, then, for each component $0 \leq x_i, y_i \geq 0$ and $x_i y_i = 0$. The last condition means that if $x_i > 0$, then, $y_i = 0$ and vice versa.

This is now a non-smooth problem because of the inequality conditions. Such a system is subject to impacts where the generalized velocities suddenly change values though the positions and orientation coordinates are always smooth functions of time.

2.8. Dry Friction. This is the first example of a non-ideal constraint, i.e., one that performs non-zero work on the system. Though it is possible to derive the equations of motion for such systems using the Lagrangean formalism, we work directly from the previous equations of motion.

Dry friction requires a mix of a velocity constraint and an effort constraint, i.e., a restriction on the value of the Lagrange multipliers. This is so that we can model the switch between static and kinematic friction. The gradient of a signed distance function, $\xi^{(j)}$, introduced above, defines a vector $\bar{n}^{(j)}$ normal to the tangent contact plane. This is spanned by a set of basis vectors $\{d^{(j,1)}, d^{(j,2)}, \ldots, d^{(j,n_d)}\}$. The simplest case, we use just two orthogonal vectors but some models require the use of a large number of non-orthogonal vectors to reduce anisotropy [1]. Using these, we can compute the projection $D^{(j)}$ such that the relative tangential velocity in the contact plane is given by $\sigma^{(j)} = D^{(j)}v$ and the tangential contact force is given by $D^{(j)T}\beta^{(j)}$.

Static friction corresponds to zero contact velocity, i.e., $D^{(j)}v = 0$. For kinematic friction, the magnitude of the tangential contact force is given by the product of the kinetic friction coefficient and the magnitude of the normal force at the contact point, $\|\beta^{(j)}\| = \mu_k^{(j)}\nu^{(j)}$, and the direction is anti-parallel to the tangential contact velocity. We only cover the case of isotropic friction here and we use only one friction coefficient for both cases. Two simplifications are possible.

First, one can linearize the norm operator. A vector in the tangent plane can be written as: $x^{(j)} = \sum_k \alpha_k d^{(j,k)}$ with $\alpha_k \geq 0$, with only few non-zero α_k. We introduce $E^{(i)} = (1, 1, \ldots, 1)^T$ such that $\|x^{(j)}\| \simeq {E^{(j)}}^T x^{(j)} = \sum_k \alpha_k$. A linearized approximation of the Coulomb friction model consists of the following set of complementarity conditions:

$$\begin{aligned} 0 \leq D^{(j)}v + E^{(j)}\sigma^{(j)} \perp \beta^{(j)} &\geq 0 \\ 0 \leq \mu^{(j)}\nu^{(j)} - E^{(j)T}\beta^{(j)} \perp \sigma^{(j)} &\geq 0. \end{aligned} \tag{2.9}$$

The second equation tells us that as long as the magnitude of the friction force $|\beta^{(j)}|$ is less than the product of the friction coefficient and the normal force, the sliding velocity $\sigma^{(j)}$ vanishes and the first equation imposes that constraint as $D^{(j)}v = 0$. When the magnitude of the tangential force reaches the maximum, we get a non-zero

sliding velocity and the first block of complementarity conditions picks a direction for the tangential force which is nearly anti-parallel to the sliding.

The second option is to impose a fixed bound on the tangential forces based on an estimate of the normal force. Using two perpendicular directions, $d^{(j,1)}, d^{(j,2)}$, we impose the box bounds $-\mu^{(j)}\bar{\nu}^{(j)} \leq \beta^{(j,i)} \leq \mu^{(j)}\bar{\nu}^{(j)}$ for $i = 1, 2$, where $\bar{\nu}^{(j)}$ is an approximation of the expected normal force for the given contact point. The constraint equations expressing this model are as follows:

$$\begin{aligned} Dv - \sigma_+ + \sigma_- &= 0, \\ 0 \leq \beta - \underline{\beta} \perp \sigma_+ \geq 0, \quad 0 \leq \overline{\beta} - \beta \perp \sigma_- &\geq 0 \end{aligned} \tag{2.10}$$

where $\overline{\beta}, \underline{\beta}$ are the lower and upper bounds of β, respectively, and σ_+ and σ_- are the positive and negative components of the tangential contact velocity, respectively. This model is dissipative and bimodal like the Coulomb model, i.e., there is a static model where the contact velocity vanishes, and a kinetic mode where the friction force is fixed in magnitude and opposes the motion. However, it exhibits anisotropy and a slightly wrong transition point, especially if the values of $\underline{\beta}^{(j)}, \overline{\beta}^{(j)}$ are not adjusted. This model can be turned into an iterative scheme which converges to the correct answer by adjusting the values of the tangential friction force bounds, $\underline{\beta}^{(j)}, \overline{\beta}^{(j)}$ with current estimates on the normal forces at the given contact.

The first option described above is much more difficult to handle numerically and we therefore choose the second one.

Putting all the equations together, we obtain the following equations of motion:

$$\begin{aligned} M\dot{v} + T^T(q)\frac{\partial U}{\partial q^T} - G^T\lambda - J^T\rho - N^T\nu - D^T\beta &= F_M \\ \Phi(q) &= 0 \\ J(q)v - w_0(t) &= 0 \\ D(q)v &= \sigma \\ 0 \leq \Xi(q) \perp \nu \geq 0, \quad 0 \leq \beta - \underline{\beta} \perp \sigma \geq 0, \quad 0 \leq \overline{\beta} - \beta &\perp \sigma \leq 0, \end{aligned} \tag{2.11}$$

and the last two complementarity conditions express the fact that either the contact velocity vanishes and the tangential forces are within the given bounds, or that a bound is reached and the contact velocity has the opposite sign. Of course, using the same scheme as we did for velocity constraints, we can also impose a velocity at the contact point. In this case, we simply replace the constraint equation with $Dv - v_0 = \sigma$. Such sliding velocities are useful in modeling wheel slip for instance.

2.9. Rayleigh's dissipation function. The forces derived from potentials strictly preserve energy but real physical systems exhibit dissipation. The Lagrangean formulation of dynamics can be extended with a special form of velocity-dependent forces which are derived from a scalar function (see Goldstein [4], p 23–24). This parallels the way conservative forces are obtained from scalar potentials. The result of the analysis is that non-conservative forces can be included in the Lagrangean framework, and thus be expressed in a form independent of the coordinate system, provided they can be expressed as the velocity gradient of a scalar function $\mathcal{R}$, i.e., $F_d = -\frac{\partial \mathcal{R}}{\partial v^T}$, where F_d is the generalized force vector and v is the generalized velocity vector. The

function $\mathcal{R}$ is called the Rayleigh dissipation function when it has the special form $\frac{1}{2}v^T B(q)v$, where $B(q)$ is a symmetric, positive semi-definite, smooth, real matrix function of the coordinates. This is enough to model a number of simple phenomena as we show below.

The Euler-Lagrange equations for a system with Lagrangean $\mathcal{L}(q,v)$, and Rayleigh function $\mathcal{R}(q,v)$ is then:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial v^T}\right) - T(q)\frac{\partial L}{\partial q^T} = -\frac{\partial \mathcal{R}}{\partial v^T}. \tag{2.12}$$

This extends naturally to the case where we have kinematic or velocity constraints.

Though the Rayleigh functions can be used to model dissipative forces, they are not limited to that case, a fact we will use below to justify velocity constraints as the limit of large Rayleigh functions.

A simple example is provided by a spring-damper system where the potential function is $U(q) = \frac{k}{2}q^t q$ and the Rayleigh dissipation function is $\mathcal{R} = \frac{\gamma}{2}v^t v$. The kinetic energy for this is $K = \frac{m}{2}v^t v$ and therefore, the Euler-Lagrange equations of motion are: $m\ddot{q} + kq = -\gamma v$, which is the well-known equation of motion for a spring-damper system.

3. Regularization. In this section, we describe how to perform the transition from simple potential and dissipation functions to position, velocity, and effort constraints. This is then used to derive a discrete time stepping scheme which can handle both pure constraint and regularized cases.

3.1. Strong potentials and holonomic constraints. . Consider an m-dimensional vector function $\Phi(q)$ and the potential function $U = \frac{1}{2}\Phi^T(q)C^{-1}\Phi(q)$, where C is a constant, diagonal, real, positive matrix of order m. The diagonal entries are written as $c_i \geq 0$. The matrix C^{-1} is the potential strength. For such a choice, we have $U(q) \geq 0$. This is a natural choice as many of the simple force potentials of classical physics admit such a form. For instance, a spring force can be described with the function $\Phi(q) = \|q - q_0\|$ where q_0 is some reference point is space.

The generalized force F arising from such a potential function U is given by:

$$F_\Phi = -G^T C^{-1}\Phi, \tag{3.1}$$

where $G = (\partial\Phi/\partial q)T(q)$ as before. Looking at what happens in the limit where $c_i \to 0, \forall i$, Rubin and Ungar [13] showed that provided both the potential and it's derivative vanish initially namely, $U(q(0)) = 0$ and $G(q(0))v(0) = 0$, then, given a sequence $\{C^{(k)}\}_{k=0}^{\infty}$, such that $c_i^{(k)} \to 0$, the sequence of solutions to the equations of motions, $q_{C^{(k)}}(t), v_{C^{(k)}}(t)$, converges uniformly to $q_{C=0}(t), v_{C=0}(t)$, on an arbitrary, fixed interval, $[0, t_1]$. In addition, the functions $\lambda_{C^{(k)}}(t) = -[C^{(k)}]^{(-1)}\Phi(q_{C^{(k)}}(t))$ are continuous and converge uniformly, whilst the functions $\Phi(q_{C^{(k)}}(t))$ converge uniformly to 0 on the entire interval $[0, t_1]$, i.e., $\Phi(q_{C^{(k)}}(t)) \to 0$, for $t \in [0, t_1]$. This says that if the strong oscillating potentials are not excited initially, the do not capture all the energy of the system and in addition, the oscillations of $\Phi(q)$ remain *uniformly* bounded, i.e., the constraint error has the same bound everywhere over the integration interval, and it shrinks uniformly to 0 as C vanishes.

By introducing the artificial variables $\lambda = -C^{(-1)}\Phi(q)$, the previous system can be rewritten as a DAE of index 1:

$$\begin{aligned} M\dot{v} - G^T\lambda &= F_M \\ C\lambda &= -\Phi. \end{aligned} \tag{3.2}$$

This can of course be combined with other types of potentials, extra kinematic constraints, and velocity constraints. Of course, λ is completely redundant here but this will not be the case when the equations are discretized in section 4.1. It is clear that the equations of motion of the constrained system are recovered when $C = 0$.

Also, the artificial variable $\lambda = -C^{-1}\Phi$ is the magnitude of the constraint force. It is negative since this force acts to *pull* the trajectory back to the surface $\Phi = 0$. The direction of the constraint force is G^T which is precisely the gradient of Φ.

3.2. From strong Rayleigh functions to velocity constraints. Now consider Rayleigh functions of the special form $\mathcal{R}(q,\dot{q}) = \frac{1}{2}[J(q)v - w_0]^T B^{-1}[J(q)v - w_0]$, where B is a diagonal matrix of order r with positive entries, $b_i \geq 0, \forall i$. This leads to the force:

$$F_\Psi = -J^T B^{-1}\left[J(q)v - w_0\right]. \tag{3.3}$$

Introducing the vectors $\rho = B^{-1}\left[J(q)v - w_0\right]$ the equations of motion are the index 1 DAEs:

$$\begin{aligned} M\dot{v} - J^T\rho &= F_M \\ B\rho &= -\left[J(q)v - w_0\right] \end{aligned} \tag{3.4}$$

An interesting special case is to take $J = (\partial\Phi/\partial q)T(q) = G$. This allows the system to dissipate energy and to relax back to the constraint surface $\Phi = 0$ if it happens to wander away from it. This case leads to the constraint force F_c:

$$F_c = -G^T\left(C^{-1}\Phi + B^{-1}Gv\right), \tag{3.5}$$

which leads to the equations of motion:

$$\begin{aligned} M\dot{v} - G^T\tilde{\lambda} &= F_M \\ C\tilde{\lambda} &= -\left[\Phi + AGv\right], \end{aligned} \tag{3.6}$$

where we introduced $A = CB^{-1}$, which is also a real, diagonal matrix of order m, with non-negative entries on the diagonal.

Though it is possible to recover effort constraints from special forms of dissipation functions, this is not covered here.

3.3. Effect of perturbation. Assuming $c_i \approx 0, \forall i$, the theory of Rubin and Ungar [13] leads to the conclusion that we will observe small oscillations about the equilibrium point. We should expect that $|\Phi(q(t))| \approx \|C\|_\infty$, where $\|C\|_\infty$ is the absolute row sum norm, $\|C\|_\infty = \max_i \sum_j |c_{ij}|$. If each c_i is sufficiently smaller than the discretization time step, this error will be insignificant. It can be argued that there is no such thing as an ideal constraint in the real world and therefore, introducing $c_i \neq 0, \forall i$, actually restores some of the compliance which occurs naturally into the

idealized model. The ability to treat the cases of very small as well as moderate values of c_i within the same framework is a great advantage from the modeling perspective. This fact is one strong motivation to use the descriptor form of the equations of motion. In a reduced coordinate formulation, it is not possible to relax the constraints at all as they are incorporated in the definition of the degrees of freedom of the system.

It is shown below 5 that the coefficient matrix of the linear system which needs to be solved is of the form:

$$\begin{bmatrix} M & -G^T \\ G & h^{-2}C \end{bmatrix}, \tag{3.7}$$

where h is the time step. The addition of the non-zero diagonal matrix C guarantees that the matrix above has full rank, even for the case when G does not. This type of diagonal perturbation is common in numerical analysis, e.g., factorization of symmetric positive semi-definite matrices [7] and interior point methods [15] to name just two. In this case however, the correspondence between the approximate solution of the original model and the exact model that is solved by the approximation is known. In addition, it is clear that if one is introducing C in the numerical method for whatever reason, one should also modify the right hand side to introduce non-zero damping and avoid unwanted oscillatory behavior.

The same comments apply to the coefficient matrix B which scales the Rayleigh functions. These allow to model active controllers which essentially impose target velocities on the system variables or a function thereof. We can thus impose target velocities either on the generalized coordinates themselves, or on the joint coordinates for instance. Relaxing these constraints amounts to introducing stiff Proportional Derivative (PD) controllers. If the integration method is stable in the limit where $b_i \to 0, \forall i$, the controllers can be arbitrarily stiff.

In the case of dry friction constraints, the perturbation δ amounts to adding viscosity at the contact points. Indeed, in this case, we will show below that the discretization leads to the equation: $Dv + \delta\beta = 0$, assuming that β is within bounds. Solving this equation for β, we find $\beta = -B^{(-1)}Dv$. For a viscous drag force, we would have $F = -\gamma Dv$, where $\gamma \geq 0$ is a scalar, and we can see that the entries b_i^{-1} are the inverse of drag coefficients. This introduces an unwanted slide velocity if $b_i \neq 0$ but this sort of creep can be handled using discrete events within the simulation, i.e., the creeping bodies can be frozen in place, a common trick of the trade. Since the exact solution for $B = 0$ can be either prohibitively expensive or just impossible to compute, due to contact degeneracy for instance, this is a lesser evil.

4. First order discretization. First order stepping schemes can look deceptively simple and hopelessly inaccurate. However, as described in detail in [5], some of the first order schemes such as the mid-point method are actually second order whereas even the almost trivial leapfrog scheme of Verlet and others have a number of special properties which make them well suited for physics simulations. The discretization presented here is closely related to the SHAKE algorithm [14][5], but using a simple linear approximation.

The reason not to use SHAKE directly is that at this time, there is no obvious extension of the basic SHAKE or RATTLE algorithms for the case of non-holonomic constraints, especially the discontinuous contact constraints we are interested in. In

addition, neither SHAKE nor RATTLE can handle relaxed constraints which is the aim of the current discretization scheme.

Only first order discretization is considered because inequality constraints arising from contacts or otherwise, introduce discontinuities. High order methods do not really make sense for discontinuous problems.

Another aspect of first order discretization is that the second derivatives of the constraints never appear and we never compute the acceleration explicitly. This is not restricted to first order methods though. There are several stepping schemes, or arbitrary order, based on discretized Lagrangeans (see reference [5], section VI.6), which are formulated without reference to accelerations. These integration schemes are derived from a formal time integration of the Lagrangean so that the stepping equations now relate finite impulses to finite changes in velocities.

Using a velocity stepping scheme has three important consequences. First, we are explicitly considering non-smooth systems which can exhibit velocity discontinuities. In consequence, the acceleration are not well-defined everywhere but since we only look at finite differences of velocities, we do not need to worry. Second, it has been shown [1] that a velocity formulation leads to solvable LCP formulation for contact problems and therefore, the foregoing discretization is compatible with that formulation. Finally, avoiding computation of the second derivative of the constraints saves computational time and programming errors, since these terms are usually quite complicated.

4.1. Singular Discretization. To set the ideas, first consider the case of a potential expressed as $U = \frac{1}{2}\Phi^T C^{-1}\Phi$ and use the auxiliary variable λ introduced in section 3.1. Using a leap-frog scheme for the velocities, the discretized equations read:

$$\begin{aligned} q_{n+1} &= q_n + hT(q_n)v_{n+1} \\ Mv_{n+1} - G(q_n)^T\lambda_{n+1} &= F_M \\ C\lambda_{n+1} &= -\Phi(q_{n'}). \end{aligned} \tag{4.1}$$

Choosing $n' = n+1$ and performing a series expansion of $\Phi(q_{n+1} \approx \Phi(q_n) + hGv_{n+1}$, we obtain the following system of equations:

$$\begin{bmatrix} I & -hT(q) & 0 \\ 0 & M & -G^T \\ 0 & G & h^{-2}C \end{bmatrix} \begin{bmatrix} q_{n+1} \\ v_{n+1} \\ h\lambda_{n+1} \end{bmatrix} = \begin{bmatrix} q_n \\ Mv_n + hF_M \\ -\frac{1}{h}\Phi(q_n) \end{bmatrix}. \tag{4.2}$$

This is very similar to the RATTLE scheme (see [5], section VII.1.4, eqn 1.21, but replacing $v_{n+1} = (q_{n+1} - q_n)/h$, and $v_n = (q_n - q_{n-1})/h$).

The case of Rayleigh function is treated similarly. We start from $\delta\rho = -\Upsilon$ and discretize this as follows:

$$B\rho_{n+1} \approx -J(q_n)v_{n+1} + w_0. \tag{4.3}$$

Since we solve for $h\rho_{n+1}$, we introduce $\tilde{C}_\Upsilon = \mathrm{diag}(\{b_i/h\})$, where b_i is the inverse drag for the i^{th} velocity constraint. Likewise for the tangential friction constraints, we define $\tilde{C}_D = \mathrm{diag}\{b_j/h\}$ where b_j is the inverse viscous drag for the j^{th} tangential friction constraint. Finally, for the contact forces, we introduce $\tilde{C}_\Xi = \mathrm{diag}\{c_k h^{-2}(1 + c_k b_k/h)^{-1}\}$ and $\tilde{B}_\Xi = \mathrm{diag}\{h^{-1}(1 + c_k b_k/h)^{-1}\}$, where c_k, b_k are the compliance and damping for the k^{th} contact respectively.

We also introduce damping $\gamma^{(l)}$ for the l^{th} the equality constraint $\Phi^{(l)}$ so the diagonal matrix $h^{-2}C$ changes to $\tilde{C}_\Phi = \text{diag}\{c_l h^{-2}(1+c_l b_l)/h)^{-1}\}$ and the scalar multiplying the vector Φ is changed to the diagonal matrix $\tilde{B} = \text{diag}\{h^{-1}(1+c_l b_l/h)^{-1}\}$. We also add other forces which are not stiff so that $F_e = T^T(q)(\partial U_e/\partial q) + F_M$. Finally, we add the contact constraints and dry friction constraints. With all these terms included, we find the stepping formula:

$$\begin{bmatrix} I & -hT & 0 & 0 & 0 & 0 \\ 0 & M & -G^T & -J^T & -N^T & -D^T \\ 0 & G & \tilde{C}_\Phi & 0 & 0 & 0 \\ 0 & J & 0 & \tilde{C}_\Upsilon & 0 & 0 \\ 0 & N & 0 & 0 & \tilde{C}_\Xi & 0 \\ 0 & D & 0 & 0 & 0 & \tilde{C}_D \end{bmatrix} \begin{bmatrix} q_{n+1} \\ v_{n+1} \\ h\lambda_{n+1} \\ h\rho_{n+1} \\ h\nu_{n+1} \\ h\beta_{n+1} \end{bmatrix} + \begin{bmatrix} -q_n \\ -Mv_n - hF_e \\ \tilde{B}_\Phi \Phi(q_n) \\ -w_0 \\ \tilde{B}_\Xi \Xi(q_n) \\ -v_0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \eta \\ \sigma \end{bmatrix} \tag{4.4}$$

$$0 \le \nu \perp \eta \ge 0, \quad 0 \le \beta - \underline{\beta} \perp \sigma \ge 0,$$
$$0 \le \overline{\beta} - \beta \perp \sigma \le 0, \quad \underline{\beta} \approx -\mu\nu_{n+1}, \quad \overline{\beta} \approx \mu\nu_{n+1}.$$

This system of equations allows to compute the positions and velocities at step $n+1$ given the system configuration at step n. As long as the extra forces F_e do not include very high frequency terms or aren't too stiff, this stepping scheme is stable.

5. Solving the stepping equations. Section4.1 concluded with a stepping equation of the form:

$$\begin{gathered} Hz + p = w_+ - w_-, \\ 0 \le z - l \perp w_+ \ge 0, \quad 0 \le u - z \perp w_- \ge 0. \end{gathered} \tag{5.1}$$

Here, H is a known square matrix, p is a known vector of appropriate dimensions, the vector z contains all the unknowns, restricted to be within the lower and upper bound vectors, l and u, respectively, and w_+, w_- are the positive and negative components of the residuals.

There are non-zero residuals in the case where one of the components of the vector z reaches a bound. Some or all the bounds can be infinite.

The type of linear system is called a *Linear Complementarity Problem* (LCP). Such problems arise in quadratic programming for instance. There are several methods to process LCPs. A good survey is found in [8]. We use a block pivot method due to Kostreva which is described in [8].

To use this method, we need to solve linear systems of the form:

$$Hz = \begin{bmatrix} M & -G^T \\ G & C \end{bmatrix} \begin{bmatrix} v \\ y \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix} \tag{5.2}$$

where M is a block diagonal matrix that is symmetric and positive definite, G is a block sparse matrix and C is a diagonal non-negative matrix. Matrix M is easily inverted.

We use a Schur complement method to solve these linear systems by first computing the matrix $S = GM^{-1}G^T + C$, which is then factored using a variable bandwidth Cholesky factorizer(see ref. [2], section 10.2 for instance). This produces a lower triangular matrix L such that $S = LL^T$. We then solve the reduced system of equations:

$$\left[GM^{-1}G^T + C\right] y = g - GM^{-1}f, \tag{5.3}$$

using a forward and backward substitution to compute the value of y and finally, we compute:

$$v = M^{-1}G^T y + M^{-1}f, \tag{5.4}$$

using matrix-vector multiplication operations.

The only reason it is possible at all to use such a method to step the equations of motion forward is that matrix S is quite sparse and therefore, computing the factor L is fast. Also, adding the perturbation C on the diagonal of the matrix S guarantees it is positive definite which allows to avoid pivoting operations.

Another very common scheme is to solve the Schur complement matrix using a Gauss-Seidel method [9] and without actually forming the entire matrix. In most papers, the authors claim to solve the system for $C = 0$ but it is clear from the simulation results that this method does introduce oscillations which are due to the inaccuracy of the Gauss-Seidel process namely, the iterates never really converge to the answer but only get closer and closer, following a geometric progression, at least for the case of a linear system. In the case of LCP, even linear convergence is not guaranteed (see Ref. [12], chapter 12 for instance).

Assuming that the Gauss-Seidel process applied to solve the system $Az = b$ yields the result $\bar{x}$, we now try to identify the perturbed matrix $\bar{A} = A + \beta$ and the vector $\bar{b}$ such that $\bar{A}\bar{x} = \bar{b}$. What we want to know is whether the approximate solution solves the problem of a stable physical problem or not so we parameterize the vector b as $b = g + h$, with $h = GM^{-1}f$ and g defined as before. The type of perturbations we are considering will add a positive scalar β_i on each diagonal element of the matrix A and the elements of vector b get modified to $\bar{b}_i = b_i - \alpha_i g_i$. First compute $A\bar{x} = d = \bar{b} - \beta\bar{x}$ and then, compute $c = d - (g + h) = -\alpha g - \beta x$. Component wise, this becomes $c_i = -\alpha_i g_i - \beta_i \bar{x}_i, \alpha_i, \beta_i \geq 0$. The vector c is just the residual that is computed during the Gauss-Seidel process and there is no guarantee as to the sign of the components. Therefore, there is no guarantee that we can find positive coefficients α_i, β_i to solve each of the equations. In the case of kinematic constraints for instance, the coefficient g_i vanishes and therefore, if it so happens that c_i and $\bar{x}_i$ have the same sign, the computed solution solves a problem which has a *negative* drag coefficient, i.e., an unstable system. Likewise, if we assume there are no drag forces in the system, we have $\alpha_i = 0$ and in this case, if the residual c_i and the approximate x_i have the same sign, we have the answer to a problem which has a *negative* spring constant and that is also unstable. In the case of a position constraint, assuming there is some drag, each pair of coefficients, $\alpha_i, \beta_i \geq 0$ can be solved for if g_i and $\bar{x}_i$ have opposite signs. When that is the case, the compliance and damping values are recovered as:

$$\begin{aligned} \gamma_i &= \alpha_i/(h\beta_i), \text{ damping coefficient} \\ c_i &= h^2\beta_i/(1-\alpha_i) \text{ compliance coefficient} \end{aligned} \tag{5.5}$$

Therefore, in the best possible case, the Gauss Seidel process will add compliance and damping into the system but in the worse case, the approximate solution is in fact the exact solution of an unstable system.

6. Examples. We now consider two simple examples which are commonly used in the DAE literature namely, the simple pendulum in two dimensions, and the slider crank mechanism.

With the pendulum example, we can easily perform a comparison between the regularized approach described in this paper and a linear implicit integration method applied to the penalty formulation. This illustrates the benefits and short comings of the regularized stepping method.

The slider crank mechanism is known to suffer from constraint degeneracy in some special configurations and we show that regularization removes the problem. There are of course more sophisticated ways to address constraint degeneracy but regularization is trivial to implement and therefore quite attractive.

6.1. The 2D Pendulum. This case illustrates some interesting features of the regularized stepping scheme namely, that it does not over-dissipate energy by contrast with a linearly implicit Euler method for instance. In addition, the regularization parameter can safely be set to 0 in most cases. One important exception is mentioned in Section 6.2.

Consider a unit mass in two dimensions with position $x(t)$ and velocity $v(t)$, subject to gravitational force $f_g = (0, -10)^T$. To keep this mass at a unit distance from the origin, we construct the function $\Phi(q) = |x| - 1$, and this has Jacobian $G = (1/|x|)x^T$. The potential for this is then $U(q) = \frac{1}{2}\Phi^T C^{-1}\Phi = \frac{1}{2c}\Phi^2$, since $C = c$, as Φ, are both scalars in this case. We can add a damping force to this as we did previously so $f_d = -b^{-1}G^T Gv$. The total force on the particle is then $f(x, v) == f_g - (1/c)G\Phi - b^{-1}G^T Gv$.

If we decide to consider the force $f(x, v)$ as stiff and use an implicit method for velocity stepping, we need to solve

$$v_{n+1} = v_n + hf(x_{n+1}, v_{n+1}) \tag{6.1}$$

and using the previous discretization for the position update, namely, $x_{n+1} = x_n + hv_{n+1}$, we have to solve:

$$\begin{aligned} v_{n+1} &= v_n + hf(x_n + hv_{n+1}, v_{n+1}) \\ &\approx v_n + hf(x_n, v_n) + h^2\frac{\partial f}{\partial x}v_{n+1} + h\frac{\partial f}{\partial v}|_n(v_{n+1} - v_n), \end{aligned} \tag{6.2}$$

and this leads to the linear system:

$$\left[I_2 - h^2\frac{\partial f}{\partial x} - h\frac{\partial f}{\partial v}\right] v_{n+1} = v_n + hf(x_n, v_n) - h\frac{\partial f}{\partial v}v_n, \tag{6.3}$$

where I_2 is the 2×2 identity matrix. All partial derivatives are evaluated at time t_n.

For the regularized method, a simple implementation of the stepping equations described in Eqn 4.4 was used. For both simulations, the time-step was set to $h = 1/60 \approx 0.017$. Several values of regularization from $c = 10^{-2}$ down to $c = 10^{-20}$ and then $c = 0$ were tested. Obviously, the implicit method cannot handle $c = 0$. Damping does not appear to be very relevant to this problem so we set it to 0.

Given the strong damping properties of implicit Euler, one might expect that this method would quickly stabilize to the constraint surface $\Phi = 0$ but what might not be anticipated is how all the energy of the system is actually dissipated. In addition to this, the entire motion slows down, as is seen in Figure 6.5 where it takes more than 5 seconds for the pendulum to drop from a horizontal initial position to

a vertical position. If we use a moderate spring constant such as $c = 10^{-2}$, we get the trajectories displayed in Figure 6.1 for the implicit stepper and Figure 6.2 for the explicit case. The energy and constraint violations are shown in Figures 6.3 and 6.4, respectively.

Using a more aggressive value for the regularization parameter, i.e., $c = 10^{-8}$, we see that the implicit stepper stops oscillating entirely but the regularized stepper keeps going. This is illustrated in the trajectory plots of Figures 6.5 and 6.6, respectively, and the energy and constraint violation profiles in Figures 6.7 and 6.8, respectively. The regularized stepper can handle regularization of 0 as well as illustrated in the trajectory and constraint violations plots in Figures 6.9 and 6.10, respectively.

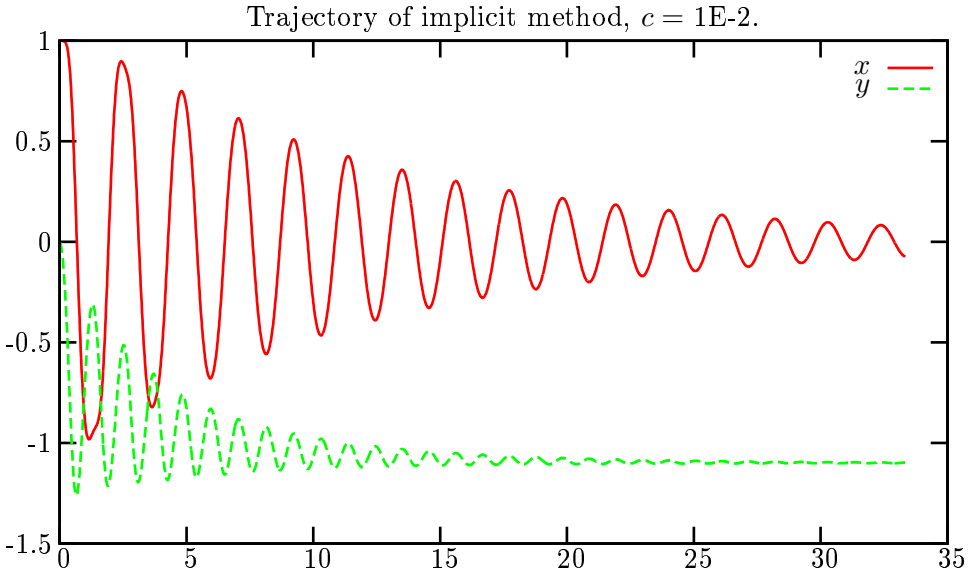

FIG. 6.1. *The trajectory for $c = 10^{-2}$ for the implicit stepper quickly dissipates to the rest position at $x = (-1, 0)$, despite the fact that damping is only applied radially.*

6.2. The Slider Crank. A slider crank mechanism consists of one arm that is attached by a hinge to the inertial frame. That hinge is driven at constant angular velocity ω_0. This driving arm is then hinged to a second section and that in turn is hinged to the load which is constrained by a prismatic joint. This is an example of a mechanical device which converts the angular motion of the motor into linear motion. A schematic is given in Figure 6.11 and screen shots of the simulation are in Figure 6.12.

This slider crank has several singularities, whenever at least two of the bodies are exactly aligned. In these configurations, the Jacobian matrices are rank deficient and it becomes difficult to compute the correct trajectory numerically. This is illustrated in Figure 6.2 where the trajectory for a very small regularization parameter, $c = 10^{-20}$, the mechanism gets stuck at the singular points. By relaxing this value to $c = 10^{-16}$ and adding some damping, the trajectory is regularized and we get smooth sinusoidal motion on the load.

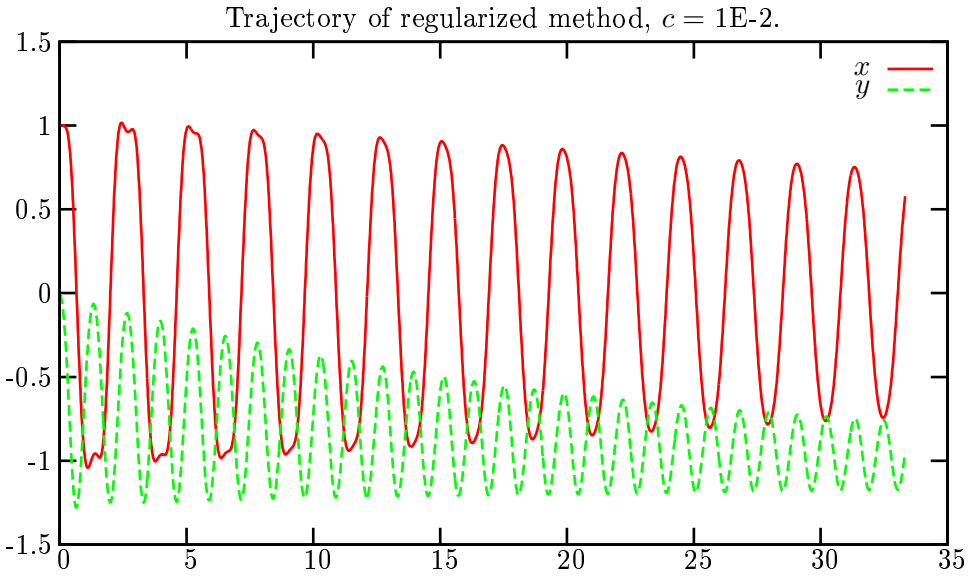

FIG. 6.2. *The trajectory for $c = 10^{-2}$ for the regularized stepper dissipates far less energy than the implicit counterpart but is still not so good.*

Of course, a full dynamics simulation is an overkill for this example but in an application, one might use such a mechanism to drive a more complicated load which would not be fully constrained and therefore, would require dynamics simulation.

7. Conclusion. A number of elements of Lagrangean mechanics were introduced to investigate the relationships between the physics model and the numerical approximation commonly used in simulations of rigid multibody systems. The analysis indicates that these perturbations actually represent a rich modeling tool as they introducing stiff compliance terms. The stability of the discretization method was not analyzed in depth but there are strong hints that it can be very stable as it is an approximation to the SHAKE algorithm, at least for the case of pure holonomic constraints, and that algorithm is known to offer both stability and geometric accuracy, i.e., to approximately preserve the invariants of the real physical solution. However, when it comes to regularized systems, there is little in the way of stability analysis at this time and the best we have is evidence from numerical experiments.

This work will be pursued by investigating other integration methods which can be proven to be globally and unconditionally stable.

8. Acknowledgments. This research was conducted using the resources of High Performance Computing Center North (HPC2N), and supported in part by the "Objective 1 Norra Norrlands" EU grant awarded to HPC2N/VRlab at Umeå University, and in part by the *Swedish Foundation for Strategic Research* under the frame program grant SSF-A3 02:128.

REFERENCES

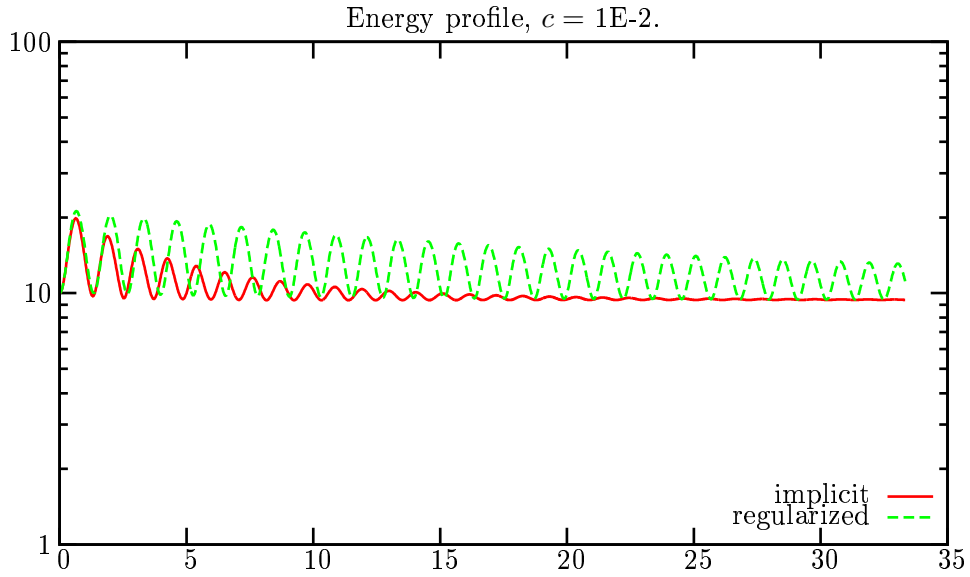

FIG. 6.3. *The energy profiles illustrates that both methods are far too dissipative when the regularization parameter is too small.*

[1] Mihai Anitescu and F.A. Potra. Formulating dynamic multi-rigid-body contact problems with friction as solvable linear complementarity problems. *Nonlinear Dynamics*, 14:231–247, 1997.

[2] I. S. Duff, A. M. Erisman, and J.K. Reid. *Direct Methods for Sparse Matrices*. Numerical Mathematics and Scientific Computation. Clarendon Press, Oxford, 1986.

[3] Christer Ericson. *Real-time Collision Detection*. Elsevier, 2004.

[4] Herbert Goldstein. *Classical Mechanics*. Addison-Wesley, Reading, MA, USA, second edition, 1980.

[5] E. Hairer, C. Lubich, and G. Wanner. *Geometric Numerical Integration*, volume 31 of *Spring Series in Computational Mathematics*. Springer-Verlag, Berlin, Heidelberg, New York, London, Paris, Tokyo, Hong Kong, 2001.

[6] Ernst Hairer and G. Wanner. *Solving Ordinary Differential Equations II: Stiff and Differential Algebraic Problems*, volume 14 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, Heidelberg, New York, London, Paris, Tokyo, Hong Kong, second revised edition edition, 1996.

[7] Nicholas J. Higham. *Accuracy and Stability of Numerical Algorithms*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, second edition, 2002.

[8] Joaquim J. Júdice. Algorithms for linear complementarity problems. In E. Spedicato, editor, *Algorithms for Continuous Optimization*, volume 434 of *NATO ASI Series C, Mathematical and Physical Sciences, Advanced Study Institute*, pages 435–475. NATO, Kluwer Academic Publishers, 1994.

[9] C. T. Kelley. *Iterative Methods for Linear and Nonlinear Equations*, volume 16 of *SIAM Frontiers*. SIAM Publ., Philadelphia, 1995.

[10] Cornelius Lanczos. *The Variational Pinciples of Mechanics*. Dover Publications, New York, fourth edition, 1986.

[11] Richard A. Layton. *Principles of Analytical System Dynamics*. Mechanical Engineering Series. Springer-Verlag, Berlin, Heidelberg, New York, London, Paris, Tokyo, Hong Kong, 1998.

[12] Katta G. Murty and Feng-Tien Yu. *Linear Complementarity, Linear and Nonlinear Programming*. Self-published: internet edition, Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, http://www-personal.engin.umich.edu/~murty/book/LCPbook/index.html, 1997.

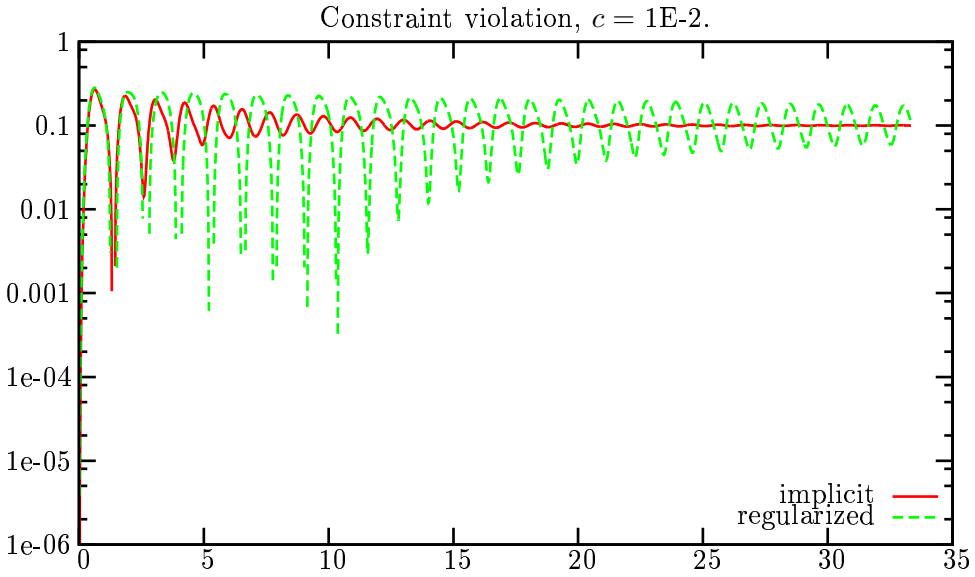

FIG. 6.4. *Constraint violation is not particularly good for such a large regularization parameter.*

[13] Hanan Rubin and Peter Ungar. Motion under a strong constraining force. *Communications on Pure and Applied Mathematics*, X:65–87, 1957.

[14] J.-P Ryckaert, G. Ciccotti, and H. J. C. Bendersen. Numerical integration of the cartesian equations of motion of a system with constraints: Molecular dynamics of n-alkanes. *J. Comp. Phys.*, 23:327–341, 1977. shake.

[15] Steven J. Wright. *Primal-Dual Interior-Point Methods*. SIAM Publ., Philadelphia, 1997.

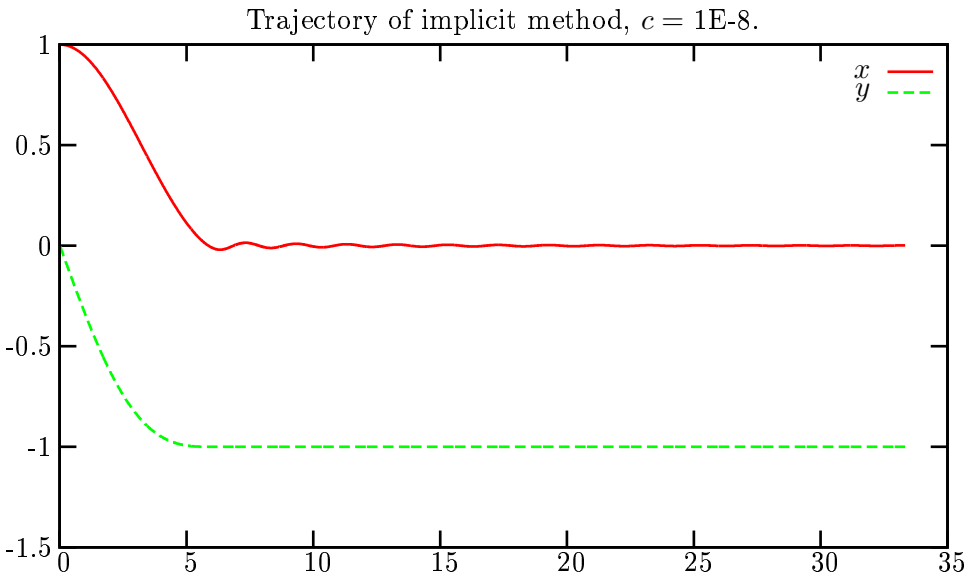

FIG. 6.5. *Here, with a small regularization constant, the implicit stepper stops oscillating entirely, despite the fact that the implicit method only applies to the radial part of the force. This is clearly not desirable.*

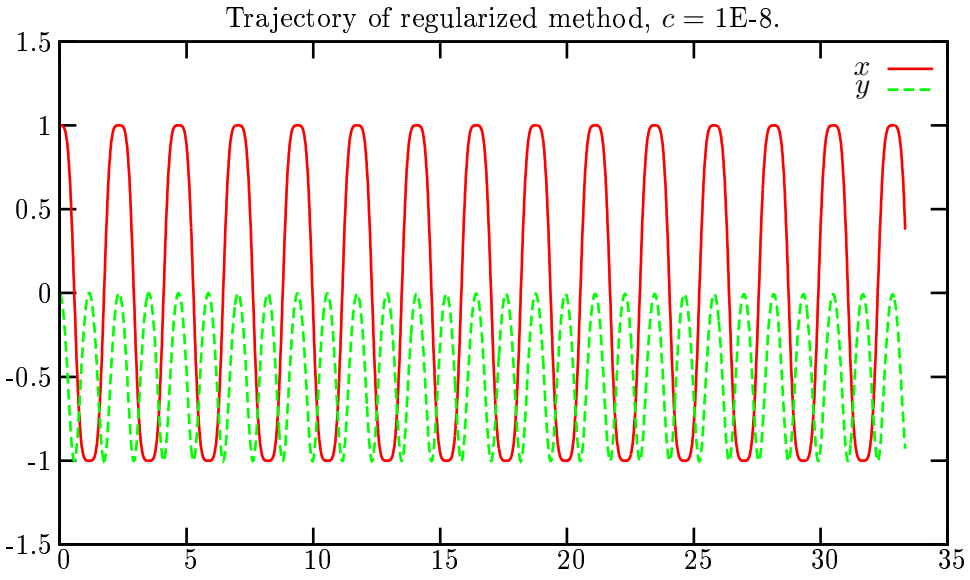

FIG. 6.6. *The regularized stepper keeps the pendulum oscillating indefinitely with a fairly small regularization parameter.*

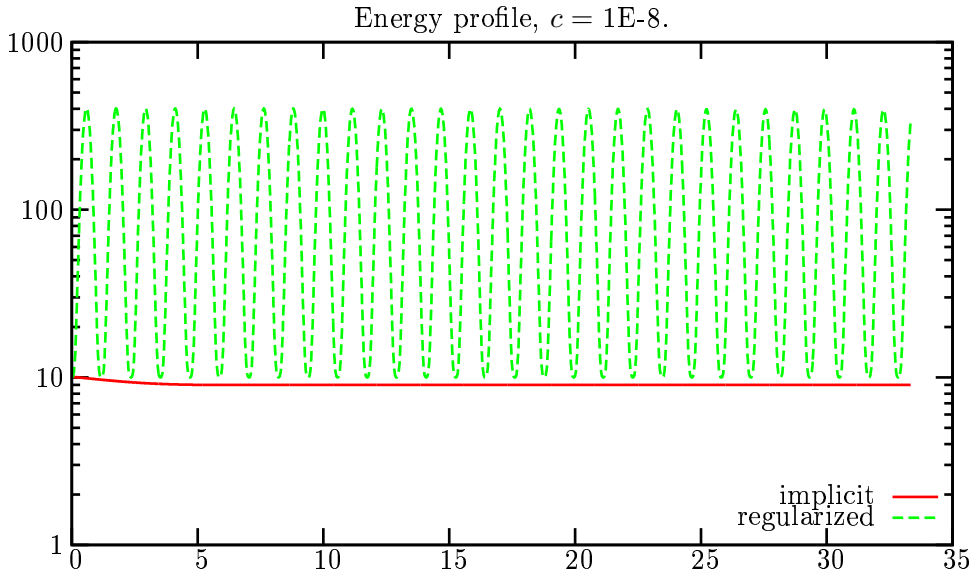

FIG. 6.7. *Here, we can see that all the energy is dissipated by the implicit method but the regularized stepper keeps all of it. The oscillations are not very welcome however and damping does not really help this.*

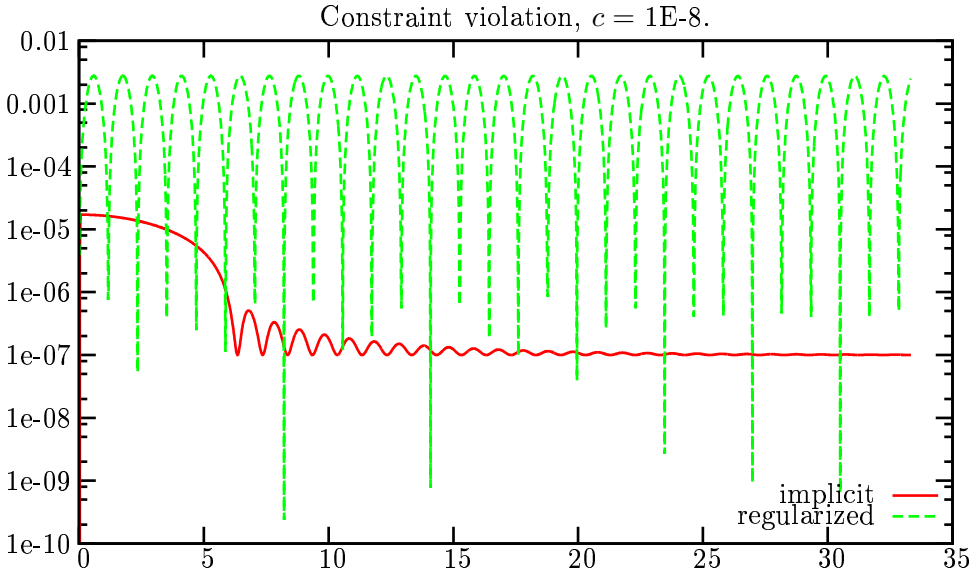

FIG. 6.8. *The constraint violation quickly decays to zero for the implicit stepper but this at the cost of stopping all motion. The regularized stepper violates the constraints by small amounts and never quite stabilizes on the surface* $\Phi = 0$.

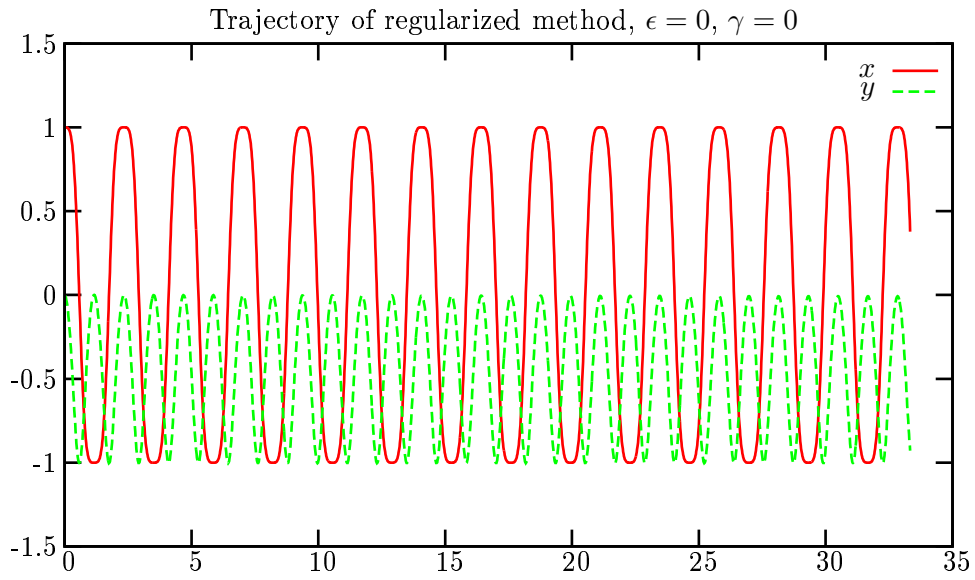

FIG. 6.9. *Setting regularization all the way to $c = 0$, we find that the regularized stepper produces a smooth oscillatory trajectory.*

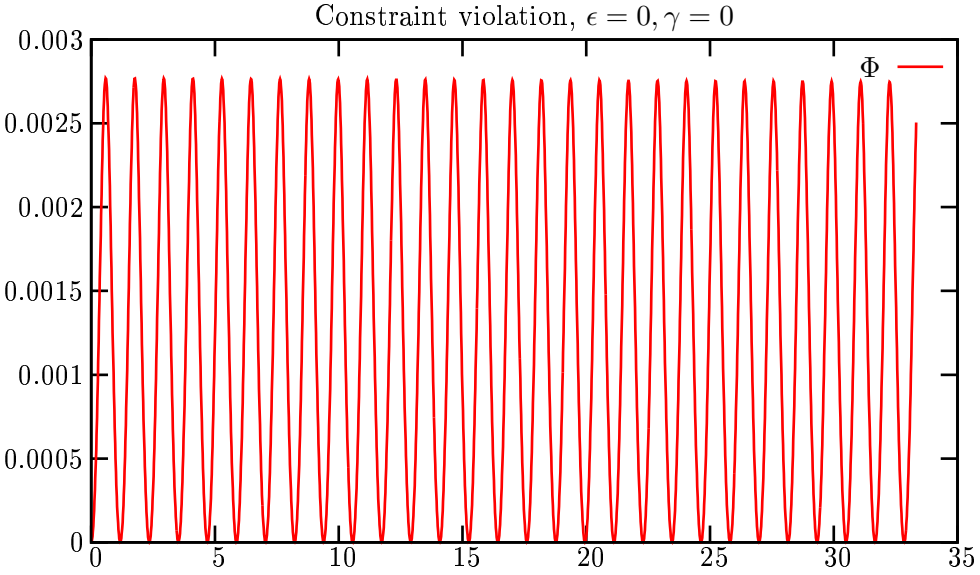

FIG. 6.10. *Setting regularization all the way to $c = 0$, the constraint violation does not quite decay to $\Phi = 0$ and the residual is now dependent on the time step.*

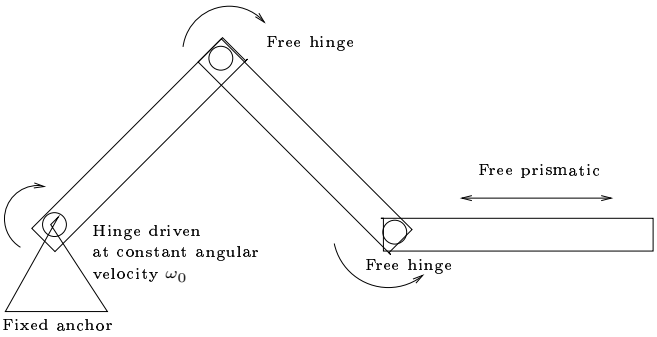

FIG. 6.11. *Schematics of a slider crank mechanism.*

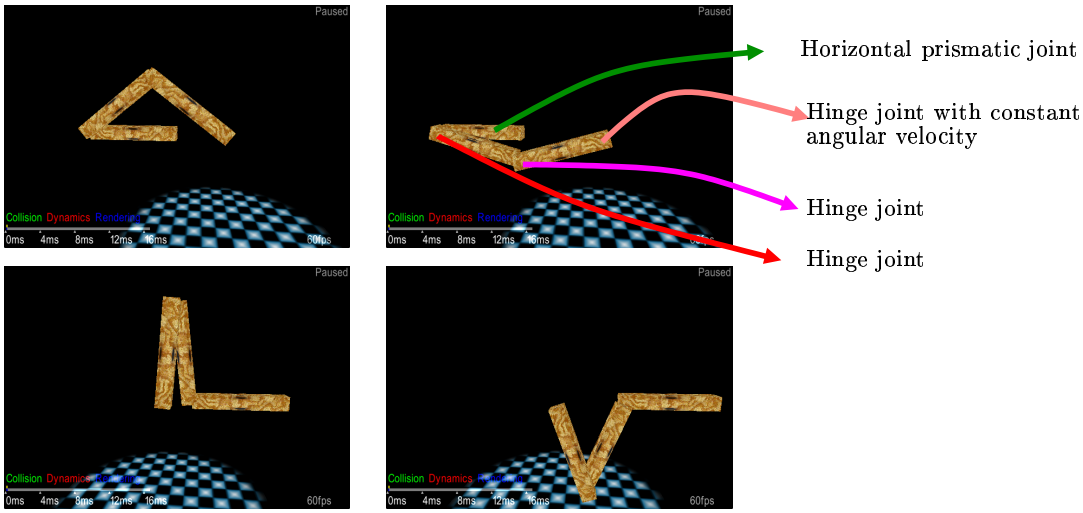

FIG. 6.12. *Screen shots of a simulation of a slider crank.*

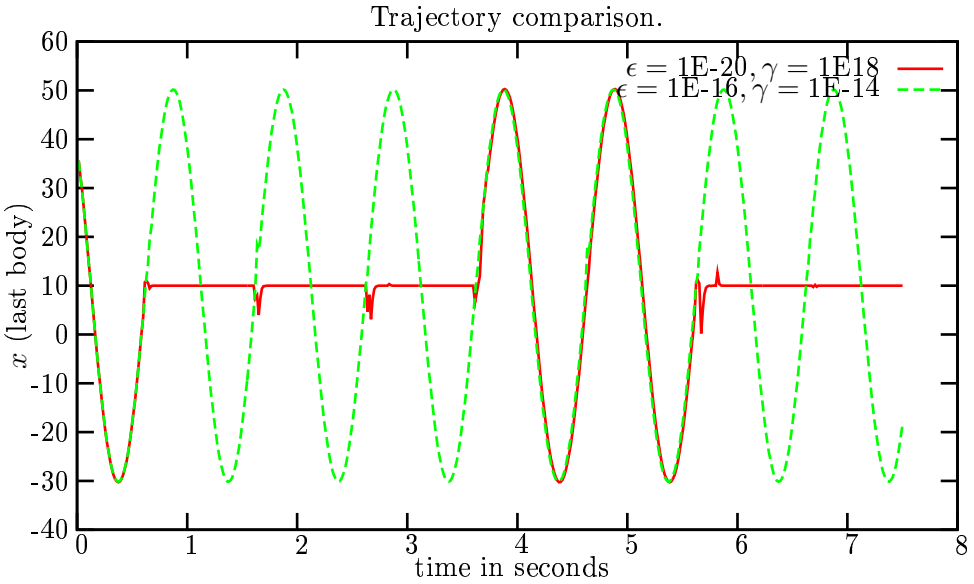

FIG. 6.13. *Trajectory of the last piece of the mechanism. When the compliance parameter is too small, the mechanism gets stuck on a singularity. By relaxing the constraints a small amount, it is possible to integrate passed the singularity.*