Computers & Graphics xxx (2017) xxx–xxx

Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

Special Section on VRIPHYS 2017

Position based simulation of solids with accurate contact handling

Q1 Mihai Frâncu*, Florica Moldoveanu

Q2 *University Politehnica Bucharest, Splaiul Independentei 313, Bucharest, Romania*

ARTICLE INFO

Article history:
Received 12 June 2017
Revised 5 September 2017
Accepted 17 September 2017
Available online xxx

Keywords:
Position Based Dynamics
Projection
Rigid body simulation
Finite element method
Frictional contact

ABSTRACT

Simulating multi-body dynamics with both rigid and flexible parts and with frictional contacts is a hard problem. We solve this by expressing the couplings between the bodies as position level constraints. The implicit treatment of the constraint directions gives us improved stability over velocity based methods. Then by employing regularization of nonlinear constraints and a convex minimization formulation, we bridge constraint-based methods to traditional force-based methods. In fact, the former are just a dual variables formulation of the latter. We solve this dual problem using position based dynamics (PBD). We show how PBD is a completely valid modeling technique and we extend it with an accurate contact and Coulomb friction model. We further show for the first time how the same solver can be used to simulate both rigid and deformable solids with two way coupling. For the soft bodies we introduce a novel form of linear finite elements expressed as constraints, that is more accurate than PBD mass-spring systems. More of our results include the energy conserving Newmark integrator and the accelerated Jacobi solver suitable for parallel architectures. Note that this paper is an extended and revised version of the conference paper published in [1].

1. Introduction

For the last decade position based dynamics (PBD) has been successfully applied to the simulation of particle systems and deformable bodies [2]. This was possible given the inherent stability of the method due to its full implicit formulation: not only the magnitude of the constraint forces are considered implicit, but also their directions [3]. This is especially true for materials with fast changing constraint gradients and transverse oscillations, e.g. cloth or threads [4].

At its heart, PBD is a nonlinear constraint projection scheme, similar to the ones used in molecular dynamics [5]. The main drawback of PBD is that it has no rigorous mathematical model for contact and friction and thus it is almost never used for rigid body simulations (with the exception of [6]). In our literature research we have not found any clear proof for the convergence of a PBD-like method with unilateral constraints and friction. Because of this, some authors choose to treat contacts as bilateral constraints [3] or approximate friction at the end of the step [7,8] without giving a sound recipe for mixing friction with the position corrections. This paper reiterates our existing work in [1] and brings some extensions to it.

We offer not only an accurate treatment of contact and friction, but we also simulate deformable bodies in a physically correct manner using the theory of continua and the finite element method (FEM). By employing the constraint regularization technique [9] at position level, we are able to show that our constraint solving problem is just the dual variables formulation of an equivalent elasticity problem. In the end, we are able to unify the simulation of rigid and deformable solids under the umbrella of PBD, using constraints as building blocks.

1.1. Related work

There has been a wealth of work published on the subject of rigid body simulation with contact and friction - for a survey see [10]. We note the advances made in the 90s by Baraff, Stewart and Anitescu. Given the drawbacks of penalty forces, Baraff introduced the acceleration based linear complementarity problem (LCP) method. This method had its problems too (related to impacts and the Painlevé paradox) that were later solved by a velocity based approach that allows discontinuities in the velocities, i.e. impulses. The new velocity time stepping (VTS) schemes [11,12] became very popular in computer graphics, games and real time simulators. We take a similar approach in this paper, but based on more recent work geared towards convex optimization [13–15], although expressing the problem as such an optimization is not mandatory.

Traditionally in computer graphics deformable bodies have been simulated using implicit integrators due to their unconditional stability properties. These have been applied not only to

* Corresponding author.
E-mail address: mihai.francu@cs.pub.ro (M. Frâncu).

https://doi.org/10.1016/j.cag.2017.09.004
0097-8493/

Please cite this article as: M. Frâncu, F. Moldoveanu, Position based simulation of solids with accurate contact handling, Computers & Graphics (2017), https://doi.org/10.1016/j.cag.2017.09.004

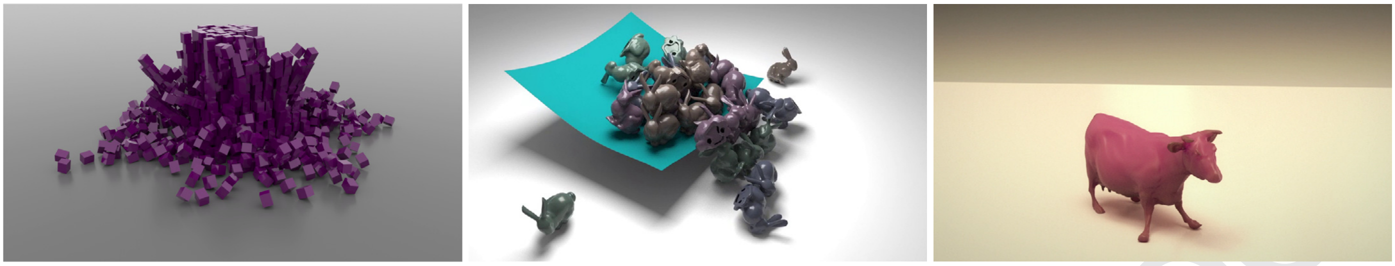

Fig. 1. From left to right: rigid boxes falling on ground, bunnies falling on a piece of cloth, a flexible cow falling on ground.

mass-spring systems, but also to simulations using finite differences or the finite element method (FEM) [16]. Recently, the popular Backward Euler integrator has been recast as an optimization problem [17] helping us to gain new insights.

While initially constraint based methods were not considered for simulating deformable bodies, this changed with the advent of PBD [7] and constraint regularization [9]. PBD was originally introduced by Jakobsen for games based on molecular dynamics methods and a nonlinear version of the Stewart–Trinkle solver for rigid bodies [8]. Goldenthal later showed how PBD stems from the fully implicit integration of a constrained system [3]. Even though in theory constraints do not allow deformation for all the degrees of freedom (often resulting in *locking* [18]), in practice, they proved quite successful for simulating a wide range of objects (e.g. cloth, hair, soft bodies - see [2] for a survey). This is due to the fact that iterative solvers are often not run to convergence and this makes the constraints soft.

The idea of a unified solver is not new and our simulator bears maybe most similarity to Autodesk Maya's Nucleus. Our results are also along the line of more recent PBD work [6,19–21] and Projective Dynamics [17].

A great job of emphasizing the role of nonlinearity for achieving stability was done in [22] - or rather the importance of using a full implicit integration of nonlinear forces. Keeping the implicit formulation intact is also the idea in [4] and the fact that dissipation (even if artificial) is key to the stability of the system is stressed in [23]. We pursue a similar approach, but rely mostly on updating the constraint directions at every iteration, without altering the mass matrix.

1.2. Contributions

We aim in this paper to show that PBD is a physically sound method. This is done in Section 2 where we introduce a fixed point iteration and prove it converges in Appendix A. Also, PBD can be used for both rigid and deformable bodies with constraints, contact and friction in a single unified solver. The advantages of this formulation include better constraint satisfaction, improved stability and out of the box two way coupling of rigid and elastic materials. In addition to [1], we explain more in depth why the method is so robust and how it can be made more conservative while maintaining its stability properties. Another new sub-section explains how our method is not bound to a minimization formulation and can also be recast as a fixed point iteration of a box LCP solver.

Our new viscoelastic model permits us to incorporate soft constraints, damping and FEM into PBD - for applications see Section 3. Another goal we had in mind was to keep the computational overhead to a minimum compared to existing methods. This is why we chose our mathematical formulation to be expressible as a matrix-free solver. We present a novel projected gradient descent algorithm for nonlinear optimization in Section 4. The algorithm is based on both the Jacobi and the Nesterov methods so it can be parallelized. In Section 5 we continue to give some more details on how to implement this solver (or a Gauss–Seidel one) for specific examples like the frictional contact constraint or the FEM tetrahedron constraint. In the end we give some code implementation notes and take a closer look at our results. As an extension to [1], we added some extra figures, plots and comments proving the nice stability and energy conservation properties of our solver.

2. Mathematical model

2.1. Equations of motion

In this section we present the continuous equations of motion and a way to discretize them that will be the basis of our further developments. We start with the equations of motion for a general system of bodies and, at first, we also introduce bilateral constraints between the bodies: general nonlinear functions equated to zero, describing for example a bead on a wire or joints articulating rigid bodies. The resulting equations can also be derived from Hamilton's principle and the principle of virtual work by using a Lagrangian augmented by a special constraint potential: $-\boldsymbol{\gamma}^T\boldsymbol{\Psi}(\mathbf{q})$ [11]. They form a special type of *differential algebraic equations* (DAE) [24]

$$\mathbf{M}\dot{\mathbf{v}} = \mathbf{f}_{tot} + \nabla\boldsymbol{\Psi}(\mathbf{q})\boldsymbol{\gamma}, \quad (1)$$

$$\dot{\mathbf{q}} = \zeta(\mathbf{q})\mathbf{v}, \quad (2)$$

$$\boldsymbol{\Psi}(\mathbf{q}) = \mathbf{0}, \quad (3)$$

where $\mathbf{v} \in \mathbb{R}^n$ is the generalized velocity vector, n is the number of degrees of freedom of the system, $\mathbf{q} \in \mathbb{R}^{n'}$ is the generalized position vector ($n' \geq n$ is the optimal number of parameters describing position and orientation), ζ is a linear kinematic mapping between velocities and position derivatives, $\mathbf{M}$ is the mass matrix [10], $\boldsymbol{\Psi}(\mathbf{q})$ is a vector-valued bilateral constraint function, $\nabla\boldsymbol{\Psi}(\mathbf{q})$ is its gradient (i.e. the constraint directions), $\boldsymbol{\gamma} \in \mathbb{R}^m$ is a Lagrange multipliers vector enforcing the bilateral constraints in (3) (m is the number of constraints), and $\mathbf{f}_{tot}$ is the total generalized force acting on the system (external and Coriolis).

In order to discretize the equations of motion we use the Implicit Euler (IE) integrator

$$\mathbf{M}(\mathbf{v}^{l+1} - \mathbf{v}^l) = h\nabla\boldsymbol{\Psi}(\mathbf{q}^{l+1})\boldsymbol{\gamma}^{l+1} + h\mathbf{f}^l_{tot}, \quad (4)$$

$$\mathbf{q}^{l+1} = \mathbf{q}^l + h\mathbf{L}\mathbf{v}^{l+1}, \quad (5)$$

$$\boldsymbol{\Psi}(\mathbf{q}^{l+1}) = \mathbf{0}, \quad (6)$$

where l is the current simulation frame, h is the time step (considered constant), and $\mathbf{L}(\mathbf{q}^l)$ is a linear kinematic mapping [10] with $\mathbf{L}^T\mathbf{L} = \mathbf{1}$ (the identity matrix). The IE discretized equations can be brought to a minimization form

$$\mathbf{v}^{l+1} = \underset{\boldsymbol{\Psi}(\mathbf{q}^l + h\mathbf{L}\mathbf{v}) = \mathbf{0}}{\arg\min}\ \tfrac{1}{2}\mathbf{v}^T\mathbf{M}\mathbf{v} - \hat{\mathbf{f}}^T\mathbf{v}, \quad (7)$$

where $\hat{\mathbf{f}} = \mathbf{M}\mathbf{v}^l + h\mathbf{f}^l_{tot}$ and the new positions come from (5).

Please cite this article as: M. Frâncu, F. Moldoveanu, Position based simulation of solids with accurate contact handling, Computers & Graphics (2017), https://doi.org/10.1016/j.cag.2017.09.004