Proceedings of the 2005 IEEE
International Conference on Robotics and Automation
Barcelona, Spain, April 2005

Effective Constrained Dynamic Simulation Using Implicit Constraint Enforcement

Min Hong, Min-Hyung Choi†, Sunhwa Jung†, Samuel Welch* and John Trapp*

*Bioinformatics, †Computer Science and Engineering, *Mechanical Engineering*
University of Colorado at Denver and Health Sciences Center
4200 E. 9th Avenue Campus Box C-245, Denver, CO 80262, USA
*†*Campus Box 104, P.O. Box 173364 Denver, CO 80217, USA*

Min.Hong@UCHSC.edu, Min.Choi@cudenver.edu, sjung@ouray.cudenver.edu,
sam@carbon.cudenver.edu, jtrapp@carbon.cudenver.edu

***Abstract* - Stable and effective enforcement of hard constraints is one of the crucial components in controlling physics-based dynamic simulation systems. The conventional explicit Baumgarte constraint stabilization confines the time step to be within a stability limit and requires users to pick problem-dependent coefficients to achieve fast convergence or to prevent oscillations. The recently proposed post-stabilization method has shown a successful constraint drift reduction but it does not guarantee the physically correct behavior of motion and requires additional computational cost to decrease the constraint errors. This paper presents our new implicit constraint enforcement technique that is stable over large time steps and does not require problem dependent stabilization parameters. This new implicit constraint enforcement method uses the future time step to estimate the correct magnitude of the constraint forces, resulting in better stability over bigger time steps. More importantly, the proposed method generates physically conforming constraint forces while minimizing the constraint drifts, resulting in physically correct motion. Its asymptotic computational complexity is same as the explicit Baumgarte method. It can be easily integrated into various constrained dynamic systems including rigid body or deformable structure applications. This paper describes a formulation of implicit constraint enforcement and an accumulated constraint error and dynamic behavior analysis for comparison with existing methods.**

Index Terms – constraint, constraint drift reduction, implicit constraint, dynamic simulation, physically-based modeling

I. Introduction

While unconstrained dynamic simulation has made big strides over the past decade, controlling the behavior of dynamic simulation in a stable and effective manner, both in intuitively defining and accurately enforcing the control terms, still poses a significant challenge. Geometric constraint enforcement is an essential technique to achieve controlled behavior of objects such as robot joints and non-penetration conditions for collision problems. Maintaining a hard constraint under a tight error bound is critical since a small, but accumulating constraint drift could result in instabilities in dynamic systems. Constraint-based control using Lagrange multipliers is well studied and successfully used for rigid body dynamic systems [4]. However, the Lagrange Multiplier method, associated with ordinary differential equations, includes a fundamental limitation of an accumulated constraint drift especially without external stabilization terms due to the discretization. Therefore the accumulated constraint drift becomes apparent and causes severe problems in constrained dynamic simulations that require long term stability. Several breakthroughs [5, 6] have been proposed, both in stabilizing the constraint and minimizing drift, but parameter tweaking is required at the cost of substantial performance degradation. For highly interactive applications that place a high priority on prompt feedback, the additional computational cost to reduce the constraint errors might be detrimental. To combine the controllability using the constraints with such applications, a less expensive but still accurate and stable constraint enforcement method is important. Another very important but often overlooked aspect of constrained dynamics is to enforce the constraints while maintaining the physically correct dynamic behavior. Constraint drift reduction methods without considering the physically correct behavior often relocate the objects in the direction that simply minimizes the constraint error but could result in the loss of energy and consequently unnaturally damped and non-conforming behavior.

The main contribution of this paper is an implicit constraint enforcement technique that shows good stability over large time steps and quick convergence. In addition, our proposed method preserves the natural dynamic behavior of simulations and demonstrates long term stability. It is particularly effective to enforce a hard constraint over a large time step. Unlike the Baumgarte method that employs second order stabilization terms, our new constraint method does not require ad hoc parameters for stabilization and provides the same asymptotic computational cost as the Baumgarte method.

The rest of this paper is organized as follows: a brief overview of constraint enforcement and related work are described in section 2. Section 3 includes the description of the implicit constraint enforcement scheme. In section 4 we provide comparison between our method and previous approaches both in constraint error minimization and the preservation of dynamic behavior. Section 5 presents the experimental results of our method with various applications.

II. Related Work

Constraint dynamics schemes have been widely studied in robotics and computer graphics to control and achieve desirable behavior. Constrained dynamics using penalty forces [9, 10] and Lagrange multipliers are well studied [1, 2, 8] both in rigid and deformable body simulation. Despite the limited utility, a single-fixed-node constraint has been successfully implemented with mass modification [7]. The seminal work of Baumgarte [1] included second order feedback terms to stabilize the constraints and has been used successfully for many applications [2, 3, 4]. But it has several computational complexity related issues. Specifically, in Baumgarte's explicit approach, the time step must be within a stability bound [15]. In addition, we have to choose problem specific parameters to make the numerical system converge to the solution quickly. Quite often, it is not easy to find the parameters a priori. Ascher and Chin [5] explained the difficulties of the parameter choice for Baumgarte stabilization and introduced improved constraint stabilization techniques. Provot [11] used simple distance reduction using dynamic inverse constraints to simulate cloth. However, abrupt change of state of objects ignores physical consequences and it may cause loss of energy or incorrect dynamic behavior of objects. In addition, the ordering of the node displacement is critical when the internal meshing structures are complex and collision is involved. Cline and Pai [6] proposed a post-stabilization approach for rigid body simulation. This approach compensates the error of integration each time step to correct the constraint error. But post-stabilization requires a linear system solution to find the error correction and it does not preserve the correct dynamic motion of objects when it reduces constraint drifts.

III. Implicit constraint enforcement

One of the most popular methods for integrating a geometric constraint into a dynamic system is to use Lagrange multipliers and constraint forces. The constraint-based formulation using Lagrange multipliers results in a mixed system of ordinary differential equations and algebraic expressions. We write this system of equations using *3n* generalized coordinates, *q,* where *n* is the number of discrete masses, and the generalized coordinates are simply the Cartesian coordinates of the discrete masses.

$$q = \left[x_1 y_1 z_1 x_2 y_2 z_2 \cdots x_n y_n z_n\right]^T \qquad (1)$$

Let $\Phi(q,t)$ be the constraint vector made up of *m* components each representing an algebraic constraint. The constraint vector is represented mathematically as

$$\Phi(q,t) = \left[\Phi^1(q,t)\ \Phi^2(q,t) \cdots\ \Phi^m(q,t)\right]^T \qquad (2)$$

where the Φ^i are the individual scalar algebraic constraint equations. We write the system of equations

$$M\ddot{q} + \Phi_q^T \lambda = F^A \qquad (3)$$
$$\Phi(q,t) = 0$$

where F^A are applied, gravitational and spring forces acting on the discrete masses, M is a *3n*x*3n* diagonal matrix containing discrete nodal masses, λ is a *m*x*1* vector containing the Lagrange multipliers and Φ_q is the *m*x*3n* Jacobian matrix. A typical solution strategy is to differentiate the constraint equation creating a system of ODE's that is easier to solve than the mixed algebra-ODE system described above. Baumgarte [1] described a technique in which the differentiated constraint equations are augmented with terms that are analogous to feedback applied to a classical second order system.

$$\ddot{\Phi} + 2\alpha\dot{\Phi} + \beta^2\Phi = 0 \qquad (4)$$

the parameters α and β determine the character of the transient as $\Phi \to 0$. Equation (4) may be expressed in a more convenient form by differentiating the constraint vector as follows

$$\dot{\Phi}(q,t) = \frac{d\Phi(q,t)}{dt} = \frac{\partial\Phi(q,t)}{\partial t} + \frac{\partial\Phi(q,t)}{\partial q}\frac{dq}{dt} \qquad (5)$$

or

$$\dot{\Phi}(q,t) = \Phi_t + \Phi_q \dot{q} \qquad (6)$$

Continuing in this fashion we arrive at the result

$$\Phi_q \ddot{q} = \hat{\gamma} \qquad (7)$$

where

$$\hat{\gamma} = -(\Phi_q \dot{q})_q \dot{q} - 2\Phi_{qt}\dot{q} - \Phi_{tt} - 2\alpha(\Phi_q \dot{q} + \Phi_t) - \beta^2\Phi \qquad (8)$$

and the subscripts q and t indicate partial differentiation with respect to q and t, respectively. $\ddot{q}$ may be isolated form equation (3)

$$\ddot{q} = M^{-1}F^A - M^{-1}\Phi_q^T \lambda \qquad (9)$$

this may be substituted into equation (8) to obtain the linear system that must be solved for the Lagrange multipliers

$$\Phi_q(q,t)M^{-1}\Phi_q^T(q,t)\lambda = -\hat{\gamma} + \Phi_q(q,t)M^{-1}F^A(q,t) \qquad (10)$$

This equation along with equation (3) can be solved with a variety of explicit methods. Solution with explicit methods constrains the time step to be within a stability limit. An additional difficulty is that selection of the parameters α and β in such a way as to force satisfaction of the constraint to a lower error tolerance decreases the time increment allowed from stability considerations. In addition, selection of these parameters is ad hoc and problem dependent.

Motivated by these considerations as well as the success of the implicit integration method of [7], we decided to investigate the implicit method of constraint satisfaction. The implicit method is first order and is implemented the following way. The equation of motion along with the kinematics relationship between q and $\dot{q}$ are discretized as

$$\dot{q}(t+\Delta t) = \dot{q}(t) - \Delta t M^{-1}\Phi_q^T(q,t)\lambda + \Delta t M^{-1}F^A(q,t) \qquad (11)$$

$$q(t+\Delta t) = q(t) + \Delta t \dot{q}(t+\Delta t) \qquad (12)$$

The constraint equations written at new time thus they are treated implicitly

$$\Phi\left(q(t+\Delta t), t+\Delta t\right) = 0 \qquad (13)$$

Equation (13) is now approximated using a truncated, first-order Taylor series

$$\Phi(q,t) + \Phi_q(q,t)\left(q(t+\Delta t) - q(t)\right) + \Phi_t(q,t)\Delta t = 0 \qquad (14)$$

Substituting $\dot{q}(t+\Delta t)$ from equation (11) into equation (12) we obtain

$$\begin{aligned} q(t+\Delta t) &= q(t) + \Delta t \dot{q}(t) \\ &+ \Delta t\left\{\Delta t M^{-1}F^A(q,t) - \Delta t M^{-1}\Phi_q^T(q,t)\lambda\right\} \end{aligned} \qquad (15)$$

Substitution of this result into equation (14) thus eliminating $q(t+\Delta t)$ results in the following linear system with λ the remaining unknown.

$$\Phi_q(q,t)M^{-1}\Phi_q^T(q,t)\lambda = \frac{1}{\Delta t^2}\Phi(q,t)+\frac{1}{\Delta t}\Phi_t(q,t) + \Phi_q(q,t)\left(\frac{1}{\Delta t}\dot{q}(t)+M^{-1}F^A(q,t)\right) \qquad (16)$$

Note that the coefficient matrix for this implicit method is the same as the coefficient matrix for the Baumgarte method. This system is solved for the Lagrange multipliers then equations (11) and (12) are used to update the generalized coordinates and velocities. The matrix is usually symmetric and positive definite. We solve this system using a preconditioned conjugated gradient method with an option to switch to bi-conjugate gradients stabilized for non-symmetric cases.

This constraint-based scheme has several advantages. The scheme is simple to implement, and it requires no additional computation. The constraints are satisfied without the time lag and oscillations associated with the Baumgarte method. Perhaps the most important advantage of this scheme is that the constraint enforcement does not place a stability limit on the time step while factoring the same linear system as the Baumgarte method requires. Another important advantage of implicit constraint enforcement is its insensitivity to the initial conditions. By the time of constraint activation, initial conditions may not be perfectly met or too costly to guarantee at every instance. This is particularly important for many actual implementations and applications since the preparation for the correct initial conditions could be very time consuming and tedious. Due to the inherent numerical damping behavior of the implicit approach, it quickly converges to the solution without much oscillation, even if we do not have a perfect initial condition that meets $\boldsymbol{\Phi} = \boldsymbol{0}$, $\dot{\boldsymbol{\Phi}} = \boldsymbol{0}$. In addition, since the constraint forces are computed from the future time step with respect to the motion of objects, the constraint error reduction is performed within the context of conforming dynamic behavior, resulting in the preservation of correct dynamic behavior. This is detailed in Section IV.

IV. Constraint Error and Dynamic Behavior Analysis

To measure the total amount of constraint drift and to compare the dynamic behavior during simulation, we tested our implicit constraint enforcement system with a robotic link connected with revolute joints. As a reference, we also implemented a same 2-link robot arm represented in a reduced coordinate system. The complete state of the robot arm is represented by angles of joints without the use of constraints so the constraint error does not exist at all. The results of the reduced coordinate approach are obtained with a higher order method (fourth order Runge-Kutta) and represent an accurate trajectory for the simple 2-link robot arm. For comparisons, all experiments are done in the reduced coordinate system as a reference, a post stabilization method [6] (one of the best ways to reduce constraint drift), and our implicit constraint method. Instead of Baumgarte stabilization method [16] that adds the second order stabilization terms to reduce the constraint drift in equation (4), the post-stabilization method [6] rectifies the position of the nodes at each integration time step to satisfy the constraint enforcement. The constraint equations include constraint error, $\Phi(q(t),t) \neq 0$, because of numerical drift. The post-stabilization method attempts to find the displacement dq which eliminates this numerical drift and forces the constraint equations to be zero. Thus we can approximate this condition by following equation:

$$\Phi(q+dq) \approx \Phi(q) + \Phi_q(q)dq = 0 \qquad (17)$$

again Φ_q is the mx$3n$ Jacobian matrix. The unknown dq can be obtained by solving the linear system in equation (17). Since usually Φ_q is not a square matrix, a pseudo-inverse method is used to solve the linear system:

$$dq = -\left(\Phi_q^{\,T}(\Phi_q\Phi_q^{\,T})^{-1}\right)\Phi(q) \qquad (18)$$

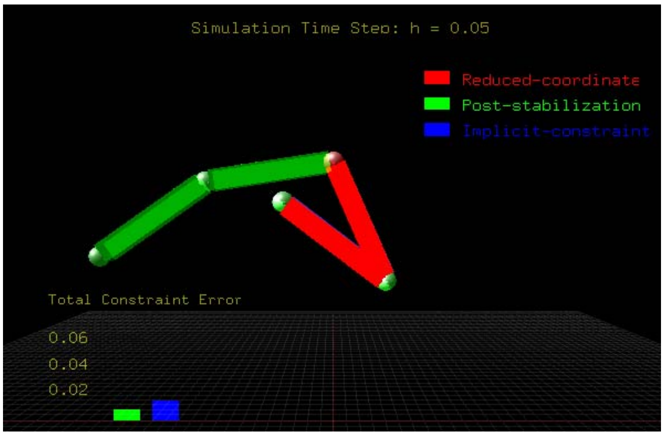

Fig. 1 A simulation of a freely falling 2-link robot arm under gravity

Figure 1 shows a snapshot of a 2-link robot arm falling down under gravity simulated using three different methods (reduced-coordinate approach, post-stabilization method, and implicit-constraint method). Initially the robot links are in horizontal position and one end is fixed. Each method is displayed in different colors and the constraint drift is shown in bar graph at the lower left corner. Note that there is no constraint error in the reduced coordinate system method. This snapshot was taken at 73 seconds of the simulation after 1460 iteration steps with the step size of 0.05. It clearly shows that the robot arm computed with the post stabilization method is substantially removed from the reference robot arm. The robot arm computed with our implicit constraint method is almost identical to the reference robot arm showing an overlapped image. The total constraint error of the implicit method, shown in the lower left corner, is a little bigger than that of the post stabilization method but the difference is mostly negligible. Supplemental animations that show the behavior of the robot arm motions in three different approaches can be found at [17].

To compare the accuracy of the constraint error reduction, we recorded the constraint error at each time step. Figure 2 shows an accumulated constraint error for the simulation of the falling 2-link robot arm shown in figure 1. We compared the accumulated constraint error of the Baumgarte method, the post stabilization method, and implicit constraint enforcement at each integration time step. The accumulated error of the Baumgarte method increases as the iteration proceeds. Although the accumulated constraint error of post-stabilization method converges to zero, this approach requires solving an extra linear system, as compared to the Baumgarte method, to achieve this result. The size of the linear system grows according to the number of constraints, and the performance penalty becomes significant as the number of constraints grows. Our implicit constraint method significantly reduces the accumulated error compared to the explicit Baumgarte method and keeps a tight error bound for a long integration time. In addition, our method does not require parameter tweaking for numerical stabilization and has approximately the same asymptotic computational cost as the explicit Baumgarte method.

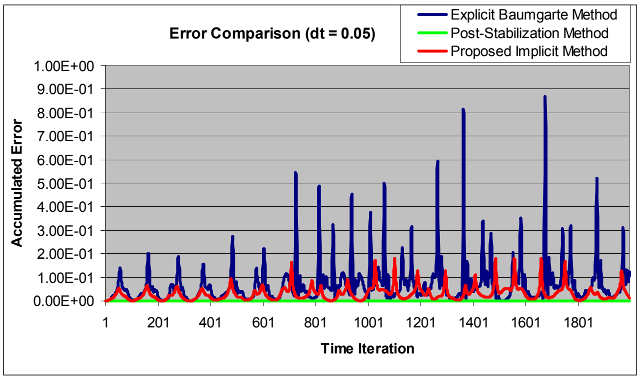

Fig. 2 Accumulated constraint error comparison for three different methods using 2-link robot arm simulation.

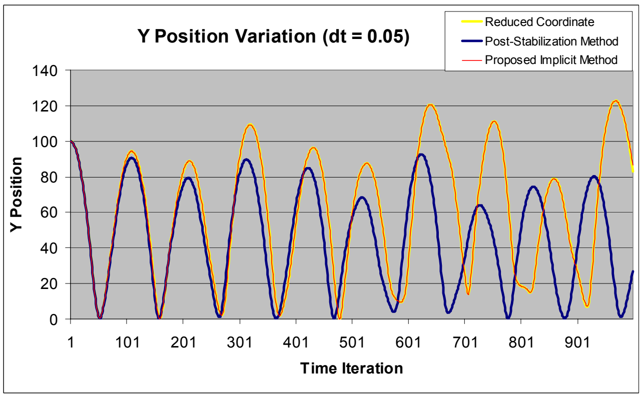

Fig. 3 Comparison of y-coordinate trajectory of the second node.

Figure 3 shows the trajectory of the second node from the simulation of the falling 2-link robot arm in figure 1. The position of the second link of the robot arm computed with the post stabilization method deviates from the reference simulation early in the simulation and shows substantial differences over the total integration time. Although the post-stabilization method preserves a tight error bound for the accumulated constraint error, this method causes the error in the physical behavior since the constraint drift reduction is performed independently from the conforming dynamic motion of the objects. Also this method requires an additional expensive linear system solution. However, as figure 3 illustrates, unlike the post-stabilization method, our implicit constraint method returns physically accurate behavior of the node movement, demonstrating almost identical behavior over the total integration time. The trajectory of the second link of the robot arm overlaps with the reference simulation as shown in the figure 3 with obscured lines.

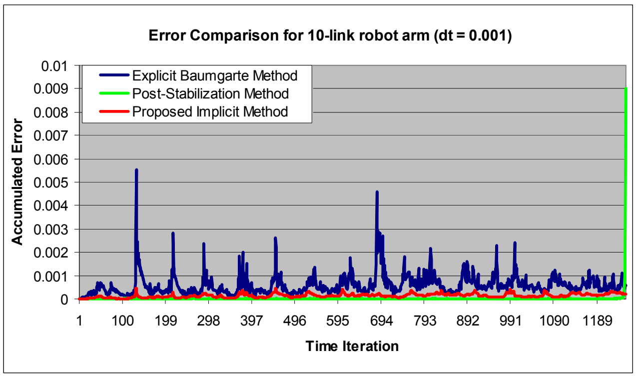

Fig. 4 Accumulated constraint error comparison for three different methods using 10-link robot arm simulation.

To test the long term stability of the constraint enforcement methods, we used a 10-link robot arm connected with ball and socket joints. Figure 4 depicts the experimental results of accumulated constraint errors for three different methods where the 10-link robot arm is falling down under gravity. Initially the robot links are in a saw tooth position and both ends are fixed. Even though the post-stabilization method well maintains constraints until around 1300 integration steps, the constraint error of the post-stabilization method suddenly grows even with a relatively small time integration step (0.001) and it causes the demolition of the dynamic simulation after producing a huge constraint error as shown at the last iteration step in Figure 4. On the other hand, our implicit constraint method shows comparable, if not better, overall constraint error with minor oscillation and it demonstrates better long term stability. Our repeated experiments confirm that the total error decreases gradually over time due to the inherent artificial numerical damping of the implicit method. The amount of damping is related to the step size, so the stability of the implicit system with a big time step is partially provided by the increased damping. This 10-link robot arm experimental result obviously reveals the limitation of the post-stabilization method in terms of long term stability. The fixed end boundary conditions creates a more complicated situation where the direction of the constraint drift reduction could be conflicting, and solving a linear system does not guarantee the optimal solution to maintain all of constraints. The post stabilization method operates within a certain error threshold and if the total error is under the limit and the constraint reduction direction is not conflicting, it can successfully minimize the drift close to zero. However, if any of those two conditions are not met, it shows a sudden failure and it could be critical in long integration time simulations.

V. Applications

The new constraint enforcement scheme was applied to a simple rigid body simulation in the examples of error analysis and dynamic behavior comparison in section IV. In this section, we present applications of the proposed method in deformable object simulation. The deformable model used here is a simple mass-spring-damper model meshed in a triangle or tetrahedral structure. To show the general applicability of the proposed method, two examples are used: the non-penetration constraint between deformable models and the prevention of excessive elongation of cloth. The non-penetration contact constraint for highly flexible and thin structures, such as cloth, requires both accuracy and stability since even a small error in the contact condition could result in visually disturbing artefacts that show an obvious protrusion in the opposite side.

A. Non-penetration constraints

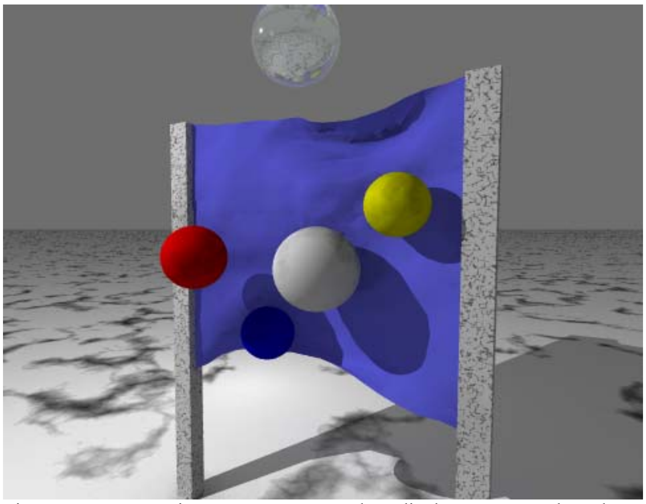

Fig. 5 Non-penetration contacts are used to display a scene where heavy balls are violently hitting the cloth secured by two posts.

Non-penetration constraints must be enforced when a piece of cloth is reaching a resting contact. For a cloth/solid case, it may involve only a node while a cloth/cloth contact case involves four nodes: a node and a triangle or edge-edge. Instead of faking the physics by instantly moving the node to the triangle surfaces, we put a non-penetration constraint on the four nodes and let the constraint bring them to the colliding surface. From three points of the triangle, a plane equation $Ax + By + Cz + D = 0$ is derived and the node coordinates are plugged in as x, y, z to make a geometric constraint relating 4 nodes. Even if the initial conditions of $\Phi = 0$, $\dot{\Phi} = 0$ are not met, the non-penetration implicit constraint enforcement brings the node and triangle to the stable contact configuration quickly. One important issue is to release the constraint in a timely manner to prevent nodes from remaining in a contact configuration even if they are trying to separate. We use both velocity directions and a Voronoi region to determine the constraint release conditions. If the two parties' relative velocity is positive (to separating direction) then the constraint is deactivated. In addition, the constraint is released as well when the node is moving away from the Voronoi region of the counterpart triangle. This constraint release scheme allows free movement on the triangle while preventing further penetration. The Edge/edge case can be modeled similarly along the edge direction. To simulate the thickness of a fabric, we can use an arbitrary constant in the constraint equation. Constraint based non-penetration is independent from a particular collision detection algorithm, so it can be used in conjunction with existing repulsion based or hybrid collision schemes. In addition, the constraint can be set to the velocities of the colliding nodes if we know the collision is imminent and if we want to limit the involved nodes motion up to the point where collision could occur. The outline of the collision and resolution approach is

1. Detect collision and identify the involved four nodes (node-triangle, or edge-edge). In cases where a complex multi-collisions occur on a same node, detailed collision resolution directions are identified by the collision resolution algorithm
2. Apply the non-penetration constraint between a node and plane (or edge and edge) so that the node can be freely moved on the plane (or on the edge). The implicit constraint will bring the node and surface (or edge and edge) to the contact configuration quickly, even if the initial positions and velocities are not valid.
3. If the node goes outside of the Voronoi region of the triangle, or the constraint force is generated to attract the two parties together, (when they are trying to move in separating direction) the constraint is released.

B. Prevention of excessive elongation

Excessive elongation is a typical flaw of the mass-spring mechanism for cloth simulation. Provot [11] used inverse kinematics to prevent the elongation beyond 10%. In this case, ordering is important and sometimes it becomes quite complicated when there are many boundary conditions and collisions are involved. Provot [12] and Desbrun et al. [14] used an iterative method to address the problem and Bridson et al. [13] used a similar iterative technique on the velocity to prevent elongation in the next step. However, these approaches do not preserve the physical properties and result in non-conforming changes or interpolation of velocities and momentum. Non-physically based alteration of those entities may result in the loss of realism especially when we are using a large time step and the cloth is affected by big external forces. Baraff and Witkin [7] used stiff springs for structural and shear springs and soft bending springs to prevent the excessive elongation. The numerical instability was treated by using a global implicit ODE. However, using stiff structural and sheer springs all the time does not represent the true biphasic elongation of fabric well. A cloth tends to stretch easily up to a point where the resistance grows quickly. All above approaches are basically similar in terms of computational complexities because all use similar iterative sparse linear system solving to handle the elongation.

Our method is to put implicit constraints only to the elongated springs beyond 10%. This constraint based method is essentially similar to the existing methods in terms of the asymptotic computational complexity since it also requires a linear system solution to get the Lagrange multipliers. However, our approach can be applied to only the elongated springs locally, depending on the current status of each spring. The size of the linear system can be much smaller in case where the constraints are not dominantly applied. Since only the source of the stability problem is treated implicitly, overall performance can be much improved. In contrast, implicit ODE methods treat all springs in every case as potential candidates for the source of numerical instability so it solves the linear system in full size that includes every spring in the system. In our method, only when the springs are elongated beyond a threshold, they are converted to constraints, and consequently the size of the matrix becomes smaller. The superior stability of implicit enforcement on an inaccurate initial condition contributes to the stability of the system and makes the implementation very simple even if we are trying to bring an already over elongated spring to a proper length.

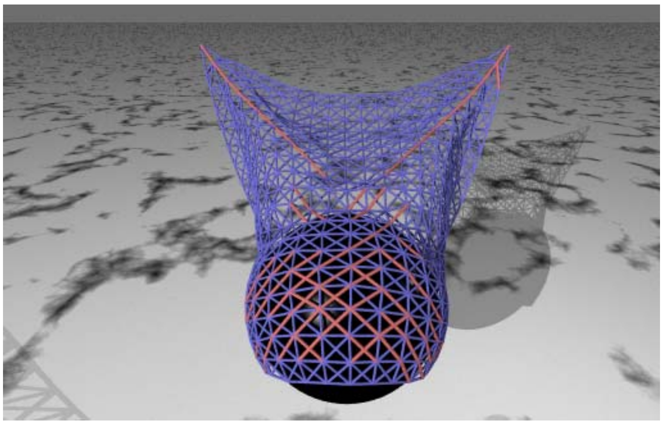

Fig. 6 The mesh view of a cloth patch in a collision between a moving ball and the freely falling cloth under gravity. The cloth is hung by two fixed point-node constraints and the implicit constraint enforcement is used to prevent the excessive elongation of cloth springs.

Constraint based elongation prevention is useful when a cloth is under a high impact force. If a heavy ball is hitting hard in the middle of a piece of cloth, the repulsion force will be generated in the normal direction of the ball surface. If we are using springs, we do not know a priori what stiffness and damping constants would guarantee the prevention of excessive elongation. Insufficient coefficients often result in over extended edges to show an over-elastic behavior. Combined with non-penetration sliding contact, the hard constraints generate a realistic cloth behavior.

VI. Conclusion

This paper reports a new implicit constraint enforcement technique that is stable over a large time step and does not require problem dependent feedback parameters. Its computational cost is same as that of explicit Baumgarte method, but its accuracy is improved significantly. Unlike the post-stabilization method, our method preserves the dynamic behavior of the motion. We have shown that it can be efficiently integrated into various types of dynamic simulation. The rigid robot arm simulation shows that our implicit constraint enforcement provides both numerically accurate and physically correct behaviour in comparison with the existing methods. For cloth simulation, by replacing only the elongated springs with distance constraints, a substantial performance advantage over global implicit ODE is demonstrated. Since the new implicit constraint enforcement uses the future time step to estimate the correct amount of the constraint force, it is well suited to model hard constraints. The new implicit constraint enforcement method is also successfully used to model non-penetration constraints. Combined with an adaptive constraint activation and deactivation scheme, the implicit constraint enforcement method can substantially enhance the controllability of the dynamic simulation of deformable objects. We believe that having a powerful and stable constraint enforcement method can be a very useful tool to address various control problems in a unified and consistent manner. We have demonstrated that those diverse problems can be easily cast as a constraint problem in a simple and straightforward way and can be effectively solved without any additional cost.

For future work, the implicit constraint method will be implemented to a higher order improving the accuracy of the error minimization. In addition, a better solution to dealing with a singular linear system arising when many conflicting constraints are imposed must be found.

Acknowledgment

This research is partially supported by NSF CAREER Award (ACI-0238521).

References

[1] J. Baumgarte, "Stabilization of constraints and integrals of motion in dynamical systems," Computer Methods in Applied Mechanics and Engineering, (1): 1 – 16, 1972.

[2] R. Barzel and A. H. Barr. "A Modeling System Based on Dynamic Constraints". Computer Graphics Proceedings. Annual Conference Series. ACM Press. 22. 179-188. 1988.

[3] J. C. Platt and A. H. Barr. "Constraint Methods for Flexible Models". Computer Graphics Proceedings. Annual Conference Series. ACM Press. 22. 4. 279-288. 1988.

[4] D. Baraff. "Linear-Time Dynamics using Lagrange Multipliers", Computer Graphics Proceedings. Annual Conference Series. ACM Press, 137-146, 1996.

[5] U. R. Ascher, H. Chin and S. Reich. "Stabilization of DAEs and invariant manifolds". Numerische Methematik, 67(2), 131-149, 1994.

[6] M. B. Cline and D. K. Pai. "Post-Stabilization for Rigid Body Simulation with Contact and Constraints". In Proceedings of the IEEE International Conference on Robotics and Automation. 2003.

[7] D. Baraff and A. Witkin. "Large steps in cloth simulation". In Proc. of SIGGRAPH 1998, ACM Press / ACM SIGGRAPH, Comput. Graphics Proc., 1-12. 1998.

[8] A. Witkin, M. Gleicher and W. Welch. "Interactive Dynamics". In Proc. of SIGGRAPH 1990, ACM SIGGRAPH, Computer Graphics. 1990.

[9] A. Witkin, K. Fleischer and A. Barr. "Energy Constraints on Parameterized Models". ACM SIGGRAPH 1987 Conference Proceedings. 225-232. 1987.

[10] D. Terzopoulos, J. Platt, A. Barr and K. Fleischer. "Elastically Deformable Models". ACM SIGGRAPH 1987 Conference Proceedings. 205-214. 1987.

[11] X. Provot. "Deformation Constraints in a Mass-Spring Model to Describe Rigid Cloth Behavior". In Graphics Interface. 147-154. 1995.

[12] X. Provot. "Collision and self-collision handling in cloth model dedicated to design garment". In Graphics Interface. 177-189. 1997.

[13] R. Bridson, R. Fedkiw, and J. Anderson, "Robust treatment of collisions, contact and friction for cloth animation", In Proc. of SIGGRAPH 2002, ACM SIGGRAPH, Computer Graphics. 2002.

[14] M. Desbrun P. Schröder, and A. Barr. "Interactive animation of structured deformable objects". In Graphics Interface. 1-8. 1999.

[15] H. Jeon, M. Choi and M. Hong. "Numerical Stability and Convergence Analysis of Geometric Constraint Enforcement in Dynamic Simulation Systems". International Conference on Modeling, Simulation and Visualization Methods. 2004.

[16] D. Baraff and A. Witkin. "Physically Based Modeling In SIGGRAPH 2001 Course Notes". ACM. 2003.

[17] http://graphics.cudenver.edu/~lab/icra05.htm