Implementation of Collision detection for Deformable Objects

Kshitij Gautam*
University of Illinois at Chicago
Kgauta3@uic.edu

Rohan Nasina*
University of Illinois at Chicago
Rnasin2@uic.edu

1. ABSTRACT
In this paper we present a real time collision detection in a three dimensional scene between a cloth mesh and a three dimensional sphere. The methods are implemented in paper [1], [2], [3]. We have considered efficient approach for implementing cloth simulation and collision detection from [1], [3], [4] and [5]. We have developed this project using Three.js [17] library.

2. Author Keywords
Computer Graphics, Three dimensional Graphic, Realism, Computational Geometry, Object Modeling, Collision Detection, Cloth Simulation, Three.js & Web Graphics Library.

3. ACM Classification Keywords
Human Computer Graphics, Computer Graphics.

4. INTRODUCTION
In this paper we simulate collision detection and create realistic cloth simulation in a 3 dimensional space. We implement a cloth using a grid of particles connected together with a lattice of springs. A detailed algorithm is explained in Section 6. We, demonstrate collision realistic collision detection with a 3 dimensional sphere orbiting around an axis in a circular motion that intersects cloth hanging on the pole. We incorporate similar technique in [1] [2] [3] to calculate collision detection and response. To improve the realism in our implementation, we load textures on the plane that depicts actual reality of soccer ground, to mimic the environment by adding fog to the three dimensional space. Finally, to finish the scene we add more realistic modules that makes it as a soccer field.

The remainder of the paper is structures as follows. In section 5 an overview of the literature review and critical analyses the state-of-the-art methodologies present for the challenges of the paper. It also provides, brief review of the history of collision detection, cloth simulation and animation the effects of collision detection. We also summarize algorithms and techniques to improve collision detection performances and cloth simulation graphics performance. In section 6 we describe a brief overview of the framework, we describe implementation of cloth simulation, collision detection and morphing. In Section 7 we conclude this paper with the possible future scope of this research and conclusion.

Copy right is held by the authors:

- ACM copyright: ACM holds the copyright on the work. This is the historical approach.
- License: The author(s) retain copyright, but ACM receives an exclusive publication license.
- Open Access: The author(s) wish to pay for the work to be open access. The additional fee must be paid to ACM.

5. Literature Review
In recent years, researchers have been developing more realistic cloth simulation for deformable objects [1]. In paper [1] authors focuses on research on cloth simulation on deformable objects using collision detection technique, further author discusses applications in surgical simulations. In paper [3] authors classify implementing cloth simulation techniques in three categories a) geometrical techniques b) hybrid techniques c) physical technique. In [4] authors demonstrate efficient techniques for cloth simulation, both in the area of numerical simulation and the area of collision detection and response. Authors implement constraint-based collision response scheme in order to implement fast and efficient technique. In [5] authors implement integration techniques for cloth simulation, also this paper presents a quantitative comparison for the basic integration algorithms implemented for cloth simulation and raises important recommendations for efficient algorithms depending on problem. As many research has been done implementing various cloth simulation algorithms and techniques and improving efficiency of previous developed technique. In paper [6] authors examine bottle-necks in most cloth simulation techniques, and describes a technique to avoid numerical instability. Also, this paper discusses a cloth simulation technique that is stable and jumps large steps. In [7] authors implement novel approach for fast collision detection for deformable objects. This approach is similar to the approach we have implemented. Also, in [7] authors implement two pass rendering model which takes no pre-computation and considers low bandwidth to and from the graphics card and calculates a viable colliding set using visibility queries. In papers [1] [2] [4] [5] authors discuss various collision detection techniques and cloth simulation algorithm, however an efficient approach is provided in [8]. Authors talk about increase in performance of

computational power available and evolution of graphical processing unit. Authors propose a general approach to reduce the cost of collision detection between deformable objects, without taking into their geometrical and symmetrical model. Collision detection is important, however animating the exact collision is discussed in [10]. Authors describe animation and performance in collision detection while animating, authors discuss efficiency of bounding volume hierarchies by adapting techniques for building and traversing these hierarchies. Authors also discuss extended set of measurements that allows pruning of the hierarchy. Oriented inflation of bounding volumes enable us to detect proximities with a minimum of extra cost. In [11] paper introduces a full GPU implementation using fragment shader of both the simulation and rendering OFA dynamically-growing particle system. Such an implementation can render up to 1 million particles in real-time on recent hardware. The massively parallel simulation handles collision detection and reaction of particles with objects for arbitrary shape. In [12] authors present techniques for simulating the motion and the deformation of cloth, fabrics or, more generally, deformable surfaces. Authors are able to simulate any kind of surface without imposing restrictions on shape or geometrical environment. Authors also consider difficult situations with respect to deformations and collisions, like wrinkled fabric falling on the ground. In [15] authors propose an accurate collision detection algorithm for use in virtual reality applications. The algorithm works for three-dimensional graphical environments where multiple objects, represented as polyhedral (boundary representation), are undergoing arbitrary motion (translation and rotation). The algorithm can be used directly for both convex and concave objects and objects can be deformed (no rigid) during motion. The algorithm works efficiently by reducing the number of face pairs that need to be checked accurately for interference, by localizing possible collision regions using bounding box and spatial subdivision techniques

6. FRAMEWORK

Implementing collision detection using deformable objects, we have implemented three major techniques which are explained in this section.

6.1 Collision Detection

The cloth responds to collisions with objects in the environment and the ground. To implement this, the position of each particle in the system is checked against each object in the scene for every time step. Potential collisions and penetrations are treated differently. Potential collisions occur when a particle is within some small distance of an object, and has a velocity heading into that object. These potential collisions, called 'contacts', are dealt with by simply removing any component of the velocity heading into the object. This is similar to the impulse forces rigid objects experience in a collision. Even when contacts are dealt with in this manner, particles can still penetrate objects within the scene. Penetrations are dealt with by applying a penalty force to the particle. This penalty force acts in a direction normal to the objects surface and with a magnitude proportional to the distance of penetration. The result is that penetrating particles are quickly pushed out of the object. Self-collisions are not dealt with in the current simulation.

Collision detection is a resultant of conservation of momentum (p) and conservation of energy (E).

$$MV_i = MV_f \ (P)$$

$$½ \, MV_i^2 = ½ \, MV_f^2 \ (E)$$

Endpoints collisions:

$$V_{1f} = n. \ \ m_2 \, 2(V_{if}. \, N – V_{2i} \, N)/ \, m_1 + m_2$$

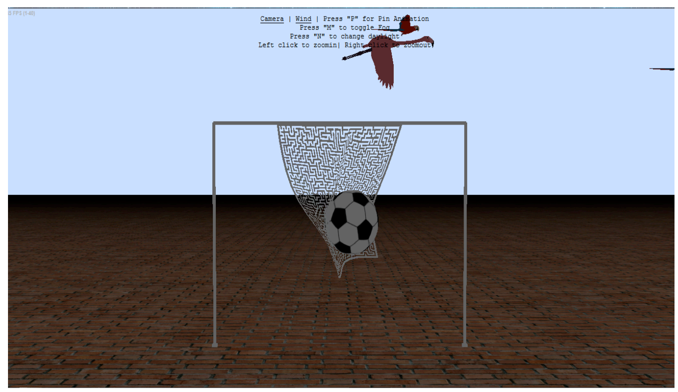

Figure 1: Collision Detection of Cloth and 3D Ball

6. 2 Cloth Simulation

This section explains cloth model. Cloth is modeled using particles connected together with a lattice of springs. In general cloth resists motion in three ways: axial stretching, diagonal shearing, and out of plane bending. Resistance to stretching is modeled by ‘structural’ springs connecting particles to their nearest vertical and horizontal neighbor in the lattice. Cloth typically does not stretch much in these axial directions, so these springs have the highest stiffness. Cloth stretches much more easily in the diagonal directions under shearing forces.

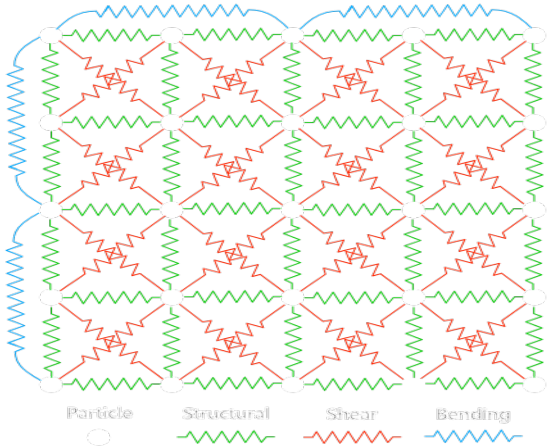

Figure 2: Cloth Simulation Model

This resistance to shearing forces is modeled using 'shear' springs connecting particles to their nearest diagonal neighbors in the lattice. Finally, cloth resists out-of-plane bending. This is modeled using 'bending' springs which connect particles two units away in the horizontal and vertical directions. This combination of structural, shear, and bending springs simulates the internal energy stored in a sheet of cloth as it experiences various modes of deformation.

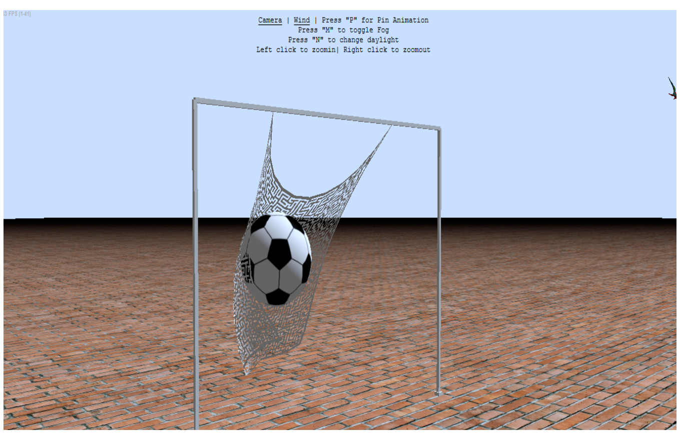

Figure 3: Cloth Simulation Implementation

6.3 Numerical Integration

During each step of the simulation, forces act on the particles in the cloth system. These forces come from external sources, such as gravity, resistance, or object collisions, as well as from internal sources (i.e. the cloth springs). Numerical integration is used to compute the velocity and position of the particles at each time step from these forces. In the basic approach to numerical integration, for each step of the simulation, forces are computed for each particle. The acceleration of each particle is calculated according to:

$F_1 + F_2 + F_3 + F_4 + F_5 + F_6 + F_7 + F_8 =$ Mass * Acceleration.

The new velocity of each particle is then approximated using the current velocity, acceleration, and length of the time step. Finally the new position for each particle is approximated using the old position, velocity, and length of the time step. Different numerical integration techniques have tradeoffs in terms of accuracy, stability, and computational efficiency. The main challenge when simulating cloth using mass-spring systems is stability. The springs in the system must be very stiff to accurately simulation cloth behavior. However it has been shown that the stiffer the springs, the smaller the time step that must be used in the numerical integration to keep the system stable. The simplest type of numerical integration uses Euler steps to calculate updated position and velocity values. This program implements basic Euler integration, Midpoint integration, and 4th Order Runge-Kutta integration. To keep the system stable with large stiffness, it's generally necessary to use 4th Order Runge-Kutta integration.

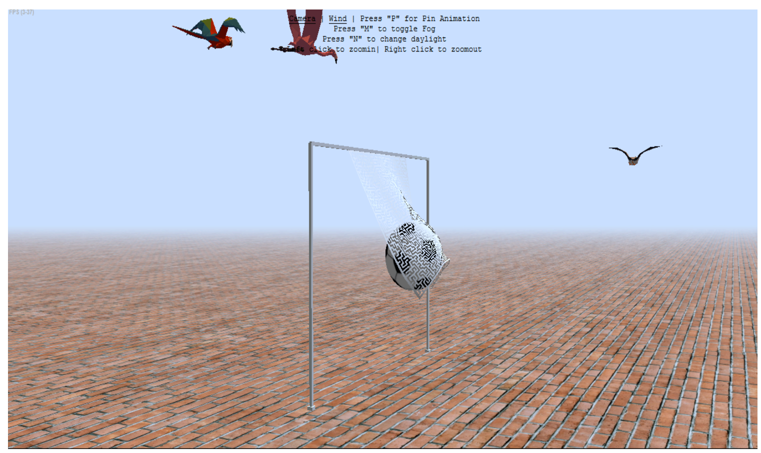

Figure 4: Morphing Animation

7. Conclusion and Future work

In conclusion we presented realistic collision detection techniques using cloth simulation and three dimensional sphere. Detailed discussion was done in Section 5 regarding various collision detection algorithms and cloth simulation techniques.

This project can be developed to create a soccer game by adding different players and adding controls to soccer ball. Also, more detailed graphics can be done to improve the visual features of goal post. A three dimensional look can be given to soccer goal post.

8. ACKNOWLEDGMENTS

We thank Prof. Angus Forbes for giving his constant encouragement for advising this project. His enthusiasm for our project and tremendous expertise is very much appreciated. We would also thank Kyle Almyrde who helped us in setting up soft wares and development environment of this project.

9. REFERENCES

1. Matthias Teschner, Stefan Kimmerle, Bruno Heidelberger, Gabriel Zachmann, Laks Raghupati, et al.. Collision Detection for Deformable Objects. Copmuter Graphics Forum,
2. Efficient Simulation of Inextensible Cloth Rony Goldenthal1,2 David Harmon1 Raanan Fattal3 Michel Bercovier2 Eitan Grinspun11Columbia University 2The Hebrew University of Jerusalem 3University of California, Berkeley
3. Computer Graphics Techniques for Modeling Cloth Hing N. Ng and Richard L. Grimsdale University of Sussex.
4. Implementing Fast Cloth Simulation with Collision Response Pascal VOLINO, Nadia MAGNENAT THALMANNMIRA Lab, C.U.I., University of Geneva - CH-1211, Switzerland and Email : pascal|thalmann]@cui.unige.chPhone: -41 (22) 705 77 63 Fax: -41 (22) 705 77 80
5. Comparing Efficiency of Integration Methods for Cloth Simulation Pascal VOLINO, Nadia MAGNENAT-THALMANNMIRA Lab, C.U.I., University of Geneva - CH-1211, Switzerl and Web: http://miralabwww.unige.chEmail: [pascal|thalmann]@cui.unige.chPhone: -41 (22) 705 77 63 Fax: -41 (22) 705 77 80
6. Large Steps in Cloth Simulation David Baraff Andrew Witk in Robotics Institute Carnegie Mellon University
7. CULLIDE: Interactive Collision Detection Between Complex Models in Large Environments using Graphics Hardware Naga K. Govindaraju, Stephane Redon, Ming C. Lin and Dinesh Manocha Department of Computer Science, University of North Carolina at Chapel Hill, U.S.A.http://gamma.cs.unc.edu/CULLIDE
8. BucketTree: Improving Collision Detection between Deformable Objects Mather, B.D. Making up titles for conference papers. *Ext. Abstracts CHI 2000*, ACM Press (2000), 1-2.
9. Dynamic Simulation Of Legged Robots using a {hysics Engined. Belter, P. Skrzypczy´nski, K. WalasInstitute of Control and Information Engineering, Poznan University of Technology,Poznan, 60-965, PolandE-mail: dominik.belter@put.poznan.plwww.put.poznan.plP. Fankhauser, C. Gehring, M. Hutter, M. A. Hoepflinger, R. Siegwart Autonomous Systems Lab, ETH Zurich Zurich, 8092, SwitzerlandE-mail: pfankhauser@ethz.ch.
10. Hierarchical Techniques in Collision Detection for Cloth Animation J. Mezger, S. Kimmerle, O. Etzmuß,WSI/GRISUniversity of T¨ubingenSand 14,D-72076 T¨ubingen, Germany mezger, kimmerle, etzmuss@gris.uni-tuebingen.de
11. Hardware-based Simulation and Collision Detection for Large Particle Systems A. Kolb and L. Latta† and C. Rezk-Salama Computer Graphics and Multimedia Systems Group, University of Siegen, Germany†2L Digital, Mannheim, Germany.
12. Extending Graphics Hardware For Occlusion Queries In OpenGLDirk Bartz Michael Meißner Tobias H¨uttner Computer Graphics Lab University of T¨ubingen
13. Versatile and Efficient Techniques for Simulating Cloth and Other Deformable Objects Pascal VOLINO (*), Martin COURCHESNE (**) (*)and Nadia Magnenat THALMANN (*) (**).
14. Jean-Christophe Lombardo, Marie-Paule Cani, Fabrice Neyret. Real-time Collision Detectionfor Virtual Surgery. Computer Animation (CA'99), May 1999, Geneva, Switzerland. IEEEComputer Society, pp.82-90, <http://ieeexplore.ieee.org/xpls/abs all.jsp?arnumber=781201>.<10.1109/CA.1999.781201>. <inria-00537512>
15. A Real-Time Algorithm for Accurate Collision Detection for Deformable Polyhedral Objects Abstract. Yoshifumi Kitamura*Andrew Smith²Haruo Takemura³andFumio Kishinop ATR Communication Systems Research Laboratories2-2 Hikaridai, Seika-cho, Soraku-gunKyoto 617-02 Japankitamura@eie.eng.osaka –u.ac.jp
16. Cloth Simulation Tutorial: http://www.gorkin.com/Cloth/cloth.html
17. Three.js www.threejs.org