

PHYSICS BASED MODELING AND SIMULATION OF COLLISION
RESPONSE

ABDUL RAB

A thesis submitted in fulfilment of the
requirements for the award of the degree of
Master of Science (Computer Science)

Faculty of Computer Science and Information System
Universiti Teknologi Malaysia

OCTOBER 2004

Dedicated to my beloved daughter Laila-Tul-Badar

ACKNOWLEDGEMENT

First of all I offer my humble thanks to Almighty Allah SWT, who made the things possible, which were impossible for me. Many Darood on our Holy prophet (peace be upon him) being a trivial ummati.

In preparing this thesis, I was in contact with many people, researchers, and academicians. They have contributed towards my understanding and thoughts. In particular, I wish to express my sincere appreciation to my main thesis supervisor, Professor Madya Daut Bin Daman, for encouragement, guidance, critics and friendship. I am also very thankful to my co-supervisors Associate Professor Dr. Siti Maryiam Bt. Hj. Samsuddin and Associate Professor Dr. Dzulkifli Mohamad for their guidance, advices and motivation. Without their continued support and interest, this thesis would not have been the same as presented here.

I want to express my profound gratitude to Prof Ghazali B. Sulong , Pn. Norhaida bte Mohd. Suaib, En Abdullah Bade and En. Mohamad Shafry B. Mohamad. Rahim for their guidance, encouragement and valuable suggestions.

My fellow postgraduate students Hafiz Faisal Zafar, Anjum Iqbal, Abdul Majid, Rashid Khokhar, Yusrin Bin Mohd. Yusof, Saandilian Devadas, Chu Kai Chuan and Lai Chui Yen should also be recognized for their valuable suggestions and support.

I must express my sincerest and heartiest thanks to my wife Dr. Shazia Naz and my other family members for their encouraging attitude unfailing patience and excellent co-operation through out my research work.

ABSTRACT

To make computer games more realistic, physics based modeling and simulation is becoming a popular area of game development. Objects, in games, need to move and interact realistically with one another. Quite often gamers ignore the classic Newtonian physics of real world, and play their own version of reality. It takes game away from reality. Physical parameters like gravity, velocity changes, inertia as well as the collision with the other objects and collision repose must be considered while developing a computer game. The use of physics simulation is an important step that increases the overall level of realism; however this introduces a significant increase in complexity. The goal of this research work is to model and simulate the collision response between the regular shaped rigid or plastic objects with relative physics. In order to achieve this objective a mathematical model has been developed. Impulse based method has been implemented to approximate rigid body collision and their responses. In impulse-based method, an impulsive force is applied to each colliding object at the point or points of impact. This force can be calculated such that it changes the velocity of each object instantly, not allowing them to penetrate one another. Seven cases of collision with different parameters are discussed here and are compared with Chris Hecker's work. It is found that the model developed is superior to Chris model because this model works for both head-on and oblique collision while Chris model is only good for head-on collision. Results also show that this model is as good for the spheres of the same masses as it is for the spheres of different masses. Simultaneously it works for head on and oblique collisions. It is covenant with situations either the spheres are moving in the same direction or in the opposite direction. It has been modeled how much the spheres rebound away from each other after collision. Simulation of collision response also supports the mathematical calculations. The research work presented in this thesis shows a successful attempt of inclusion of physics in computer games.

ABSTRAK

Bagi menghasilkan permainan berkomputer yang realistik, permodelan dan simulasi berasaskan fizik telah bidang yang popular dalam pembangunan permainan berkomputer. Objek dalam permainan perlu bergerak dan berinteraksi realistik antara satu sama lain. Kebanyakan pemain tidak menghiraukan *Newton* Fizik dalam dunia nyata, mereka hanya mahukan hiburan dan bermain untuk kesukaan pada permainan tersebut. Ini merupakan versi realiti, maka ia telah membawa permainan berkomputer tersebut jauh daripada realiti. Parameter fizikal seperti graviti, perubahan halaju, dan inerti sama seperti perlanggaran antara objek dan tindakbalas perlanggaran perlu di ambil kira dalam pembangunan permainan berkomputer. Penggunaan simulasi fizik didalam permainan berkomputer adalah langkah yang penting dalam meningkatkan tahap realistiknya, tetapi ia telah mengakibatkan permainan berkomputer tersebut menjadi lebih kompleks. Matlamat penyelidikan ini adalah memodelkan and menjalankan simulasi tindak-balas perlanggaran diantara objek tegar atau objek plastik dengan Fizik yang berkaitan. Bagi mencapai matlamat tersebut, model matematik telah dibangunkan. Kaedah *impulse* digunakan bagi menganggarkan perlanggaran dan kesan perlanggaran objek tegar. Kesan daya ini dianggap bahawa penukaran halaju objek pada masa yang sama tanpa membenarkan perlanggaran ke atas objek. Tujuh kes perlanggaran dan parameter yang berlainan dibincangkan dan dibandingkan dengan model yang dijalankan oleh Chris Hecker. Didapati bahawa model yang telah dibangunkan lebih baik daripada model Chris Hecker kerana model ini boleh digunakan bagi perlanggaran *head-on* dan oblik sementara model Chris Hecker hanya bagus bagi perlanggaran *head-on*. Keputusan menunjukkan model ini bagus untuk sfera yang mempunyai jisim yang sama dan yang berbeza-beza. Model ini juga boleh digunakan bagi perlanggaran *head-on* dan oblik yang berlaku secara serentak. Ia juga sesuai bagi keadaan dimana sfera bergerak pada hala tuju sama atau arah berlawanan. Ia telah dimodelkan untuk menunjukkan bagaimana sfera bertindak balas selepas perlanggaran berlaku. Simulasi tindak balas ini juga menyokong pengiraan matematik. Kajian tesis ini membuktikan kejayaan dalam penambahan fizik dalam permainan komputer.

TABLE OF CONTENTS

CHAPTER	TITLE	PAGE
1	INTRODUCTION	
	1.1 Research Background	01
	1.2 Research Motivation	05
	1.3 Important Issues	06
	1.4 Problem Statement	06
	1.5 Objective of Research	07
	1.6 Scope of Research	07
	1.7 Research Contribution	09
	1.7.1 Collision Response Modeling	09
	1.7.2 Simulation of Collision Response	10
	1.8 Thesis Overview	10
2	LITERATURE REVIEW	
	2.1 Physically Based Simulation	12
	2.2 Overview of Physically Based Modeling	13
	2.3 Collision Detection	17
	2.3.1 Point of Contact	18
	2.3.2 A Sphere to Sphere Sweep Test	18
	2.4 Application of Physical Based Simulation	20
	2.5 Computer Games	20
	2.5.1 Control Model	21

2.5.2	Stability	22
2.5.3	Visualization	23
2.6	Collision Response	23
2.6.1	Penalty Method	23
2.6.2	Constraint Based Method	24
2.6.3	Impulse Based Method	24
2.6.4	Comparison of These Three Methods	25

3 FUNDAMENTALS OF IMPACT MODELING

3.1	Impulse	27
3.2	Momentum	28
3.3	The Principle of Conservation of Momentum	28
3.4	Elastic and Inelastic Collisions	29
3.4.1	Elastic Collision	29
3.4.2	Inelastic Collision	29
3.5	Impulse Momentum Principle	30
3.6	Newton's Law of Restitution Coefficient Model	31

4 METHODOLOGY

4.1	Framework	33
4.2	Mathematical Modeling of Collision Response	35
4.2.1	Derivation of Response when the Spheres are Moving in the Same Direction	35
4.2.2	Derivation of Response when the Spheres are Moving in the Opposite Direction	38
4.3	Simulation of the Collision Response	36
4.3.1	Simulation Domain	39
4.3.2	Flow Chart	39

5	IMPLEMENTATION	
5.1	The Basic Algorithm	42
5.2	Pseudo Code	45
6	RESULTS AND DISCUSSION	
6.1	Input Parameters	47
6.2	Output parameters	48
6.3	Results	49
6.3.1	Case 1	49
6.3.2	Case 2	52
6.3.3	Case 3	54
6.3.4	Case 4	57
6.3.5	Case 5	60
6.3.6	Case 6	63
6.3.7	Case 7	66
6.4	Comparison with Existing Work	69
6.5	Discussion	70
7	CONCLUSIONS AND RECOMMENDATIONS	
7.1	Conclusion	73
7.2	Recommendations for Future Work	74
	REFERENCES	76
	APPENDIX A	80

LIST OF TABLES

TABLE	TITLE	PAGE
2.1	Comparison of Penalty, Constraint-based and Impulse-based methods	25
6.1	Input and Output data for Case 1	49
6.2	Input and Output data for Case 2	52
6.3	Input and Output data for Case 3	55
6.4	Input and Output data for Case 4	58
6.5	Input and Output data for Case 5	61
6.6	Input and Output data for Case 6	63
6.7	Input and Output data for Case 7	66

LIST OF FIGURES

FIGURE NO.	TITLE	PAGE
1.1	Examples of convex and non convex 3D objects	08
2.1	Collision between two spheres	19
2.2	Control model	21
3.1	A free falling ball experiencing an Impulsive force from the hard surface	28
3.2	Two extreme cases (i) completely Elastic & (ii) completely inelastic	30
4.1	Framework	34
4.2	Two spheres are colliding while moving in the same direction	35
4.3	Two spheres are colliding while moving in the opposite direction	38
4.4	Flow chart of the system	38
5.1	Two colliding spheres	42
6.1	The sequence of the simulations with respect to time for case 1	50
6.2	Velocity time graph for case 1	51
6.3	Distance time graph for case 1	51
6.4	The sequence of the simulations with respect to time for case 2	53
6.5	Velocity time graph for case 2	54
6.6	Distance time graph for case 2	54
6.7	The sequence of the simulations with respect to time for case 3	55-56
6.8	Velocity time graph for case 3	57

6.9	Distance time graph for case 3	57
6.10	The sequence of the simulations with respect to time for case 4	58-59
6.11	Velocity time graph for case 4	60
6.12	Distance time graph for case 4	60
6.13	The sequence of the simulations with respect to time for case 5	61-62
6.14	Velocity time graph for case 5	62
6.15	Distance time graph for case 5	63
6.16	The sequence of the simulations with respect to time for case 6	64
6.17	Velocity time graph for case 6	65
6.18	Distance time graph for case 6	65
6.19	The sequence of the simulations with respect to time for case 7	67
6.20	Velocity time graph for case 7	68
6.21	Distance time graph for case 7	68

CHAPTER 1

INTRODUCTION

1.1 Research Background

What are computer games? While, at first this question seems simple to answer. The computer games are games on computers. But this does not give a whole lot of information. It foregoes to mention that games are a combination of stories, art, music, sound effects, animation, and programming techniques in a dynamic, interactive form of electronic entertainment [1]. They exist on various platforms, PCs, consoles (video games), and networks (multiplayer games).

The past decade has seen an enormous progress in computing power and graphics hardware and along with that a trend towards more and more "realistic" games. Each year, the computer gamer sees an improvement in graphics quality and better simulation. Racing cars look more like real cars, sound more like real cars, and handle more like real cars. Trees are now rendered leaf by leaf. In first-person shooters today, one cannot only blast the enemy, but also doors, windows, walls, and whatever objects happen to be in the room, and they will burst into shrapnel in different ways depend on what material they are made of. Even the hair of characters in some games is specially rendered to drift naturally in the wind.

There is no doubt that realism is a very big thing in computer games today. Game developers advertise it as their main selling point, and video game console manufacturers battle it out over whomever has the best graphics hardware to render

the most number of polygons in the shortest time. Gamers and game magazines alike talk wildly about it, or complain about the lack of it. However, "realism" is not an easily definable concept. Just what is considered "realistic" in a computer game can appear to be contradictory. Gamers consistently ignore many obviously unrealistic aspects of a computer game. For example, one can crash a car in a racing game and still continue the race, when in real life the impact would have totally wrecked the car, let alone seriously injuring the driver [2].

How can we understand what makes a game "realistic" and why some games are realistic while others are not? Realism consists of many factors – sound, graphics, environment, interface, playability, etc [3].

Realism in computer games is achieved in many different ways as mentioned above. Perhaps the most direct and cognitively closest is in the graphical quality of games. The first thing people notice about a computer game is how "real" the graphics look. The importance of perception to us is captured in the old adage "seeing is believing" [3]. However, perception alone is not sufficient to lead us to feel that the game world is "real". The bottom line is that how we interact with the world. We cannot perceive a virtual game world as being "real" unless it reacts to us in a "realistic" way.

Leaving us with the question: what influences graphical realism? For this, four areas can be explored. Namely: geometry, surfaces, lighting and animation.

Geometry is everything that makes up the world, particularly walls, floors, etc. Besides largely defining the environment, geometry also affects realism by its representation and limitations. For instance, realism might be affected by the restriction of geometry not being able to use curved surfaces and level of detail is a way of handling geometry to speed up calculations that might influence realism. Therefore, it is important to look at various aspects of geometry regarding realism.

Where geometry makes up the world, surfaces dress it up. By applying images to the surfaces defined by the geometry, one creates the illusion of materials. Because these images are 2D, where surfaces in the real world rarely are, it is

important to look at things like displacement mapping. In addition, how surfaces react to light is important, so there needs to be looked at shading and radiosity. One must conclude that surfaces are important to realism as well.

Lighting is often named as the most important influence to realism [4]. This means that one cannot look at realism without looking at lighting. Specifically things like reflection, transparency and refraction are important, as do (dynamic) light and shadows and various blurry effects like anti-aliasing, soft shadows and distance fields.

Finally, after lighting, animation is usually the second thing noted as very important to realism. Especially in a real-time environment like computer games, animation is often a deciding factor in realism. Issues like skeletal animation, vertex manipulation, and physics simulation can influence how realistic something seems. Therefore, these things have to be looked closer at.

Computer games already use a variety of techniques to improve realism. Due to the real-time nature of computer games, they have to use various tricks to achieve realistic images and still have interactive frame rates. One of these tricks is to pre-calculate a lot. This includes geometry, lighting and visibility.

Having pre-calculated light adds a lot of realism to real-time environments in the sense of a single image being realistic. However, when a player moves through such an image, one can feel the unchanging nature of the world. This is where dynamic lights come in. While these lights are more expensive in real-time calculations and often of less quality, they add a certain dynamic to light in general, conveying the feeling that light can change at any time. There are generally two ways to have dynamic lights, either add extra lights real-time, or influence the pre-calculated values [5].

Tied closely together with light are shadows. Since they depend on light, they are often pre-calculated together with lights. However, with dynamic lights and dynamic objects, shadows have to be calculated real-time. While leaving them away can look unrealistic, adding them can be a quite demanding calculation. This is why

usually tricks are used, like placing a sprite of some blurry spot on the ground below the object, representing a shadow. However, real-time shadow calculations are increasingly used.

In geometry, the way to increase realism is usually by keeping the areas small. This way, less calculating time is needed which can be used to add detail. As processor power increases, the areas can become larger and the details more abundant. Another way to increase realism through geometry is in the way it is used. By letting architects design the areas for example, the feeling that the area might actually exist is increased. Another way to do that is by giving the areas a consistent feeling. This can be realized by using a general color palette, a similar style in texturing, or other artistic tricks like it.

Finally, computer games use a range of special effects to increase realism. These special effects are only applicable in certain locations and much of their benefit is often under discussion. One of the most discussed ones are lens flares, since they look good but one generally does not look through a camera to see the real world. Other things usually considered being effecting are fog, particles, and reflection. While this last one is considered an effect. Reflections (and to a lesser extent transparency and refraction) are a large part of the real world. Currently, games have certain surfaces marked as reflective which give them some mirror-like properties, but the general blurry reflective-ness of real-world surfaces must either be mimicked in textures or is not done at all.

Of course, the final consideration is that games have to be entertaining and realism often has to give way to increase the fun of a product. Another consideration is that games have to make money. So certain aspects of realism might not be implemented because of deadline, funding, or other marketing issues. Luckily, real-time 3D technology is evolving at a high rate, so more and more becomes possible to achieve realism [3].

1.2 Research Motivation

The game industry has become multibillion dollar industry in a relatively brief period. It has grown more rapidly than experts predicted and it continues to push current computer technology beyond capacity. During this continuing period of accelerated growth, little time is invested in serious academic research. To date, advances in game technology have been evolutionary. One game is improving on other often due to the result of “borrowed research” from other discipline that has been applied to the game environment. The use of physics simulation is an important step that increases overall level of realism, but it is a significant increase in complexity [6]. For computer games and interactive use, accuracy is really not the primary concern. The important goals for games are believability (including stability) and speed of execution. This feature explains the basic elements of an approach to physically based modeling which is well-suited for interactive use [7].

Not long ago, most game physics focused on flight models and contact physics was not part of the picture. Almost in every game different objects collide and interact with each other. In many interactive real-time animation systems, the entities often need to be viewed not as geometric shapes devoid of physical properties, but as real entities having properties such as mass, moment of inertia, elasticity, and friction. Their motions are constrained not only by their own physical properties, but also by collisions with other objects. If two solid objects collide in the real world, they bounce off each other, or break into pieces or deform their surfaces depending upon the nature of the object i.e. rigid or non-rigid [8]. In a computer world, there is nothing to stop geometrically modelled objects from simply floating through each other like ghosts. A collision handling system is necessary to enforce solidness, and ensure that entities behave as expected when they come into contact, i.e. they should not interpenetrate, and their behaviours subsequent to collision should be compatible with their physical properties. This involves two very distinct phases: collision detection, and collision response. Detection is a problem of kinematics, while response is a problem of dynamics [9].

Before applying any mathematics to the problem one must first derive the required parameters, which will determine how an object reacts to other on collision.

It is not enough merely to determine whether or not objects have collided but we must also determine the nature of the collision that has occurred. Physically based Simulation systems attempt to implement collision response calculations and this is a feasible option when the number of colliding objects is always expected to be fairly small.

1.3 Important Issues

Issues like skeletal animation, vertex manipulation and Physics simulation can influence how realistic something seems. Therefore these things need closer examination. The area of collision response simulation with the consideration of physical properties of materials is still much to be explored. The following issues have to be looked closer at.

1. Modeling and simulation of elastic and plastic impacts.
2. Collision response of regular convex objects in 2D (no friction no rotation).
3. Collision response of regular convex objects in 2D (no friction but rotation).
4. Collision response of concave objects in 2D and in 3D
5. Collision response of regular convex objects in 3D.
6. Response of oblique collision in 2D.
7. Issue of physical accuracy is also an important one to consider.

1.4 Problem Statement

Until recently little attention has been paid to a game's physics—how objects within games respond to each other and the forces around them. Previously computer games fudged their own systems for moving objects around. But one of the big things that is important now and will become even more important in the future is realistic physics support. “Things need to move and interact realistically with one another”.

1.5 Objectives of Research

1. To model physics-based collision response of 3D objects.
2. To simulate collision response with maximum accuracy.

1.6 Scope of Research

1. Only two rigid spheres are considered

Only rigid bodies are considered because the equations of motion for rigid bodies can be derived from Newton's laws, and are second order differential equations. Physically based modelling/simulation requires heavy computations. Linear system solvers, ordinary differential equation solvers and optimization methods are often required to solve equations from physically based modeling. So if more objects are considered then there will be more collisions and ultimately the slower will be the speed of the game.

2. Physics elements

Physics elements like masses, velocities, momentum, coefficient of restitution, nature of colliding objects and the impulses experienced by the objects upon collision are only being considered because addition of more physics parameters will slower the speed of the game.

3. Collision response of regular convex 3D objects

The mesh represents a convex object if the line connecting any two points within the object lies wholly inside the object. Figure 1.1 shows some convex and some non-convex objects. For each non-convex object a sample line is shown whose end points lie in the object, but that is not itself contained within the object. A line can have a maximum of two intersection points with the surface of a convex object.

The centre of gravity of a solid convex object always lies inside the object.

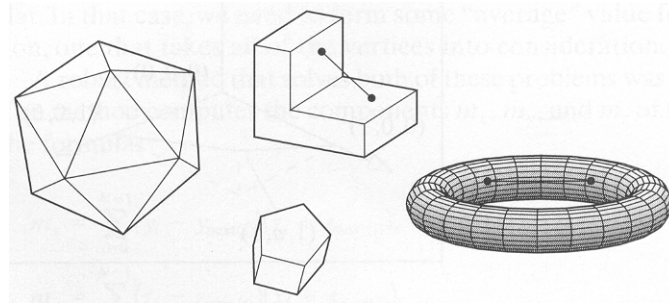


Figure 1.1: Examples of convex and non-convex 3D objects [10].

4. Only the response at the point of contact is under consideration

Our area of interest is only the point of contact of the two bodies and we are not concerned with the surface of the floor and the viscosity of the air.

5. No friction at the point of contact

It is assumed for this mathematical model that the surface of the two spheres is smooth and there is no friction at the point of contact. So, the component of momentum perpendicular to the line joining the centres of the two spheres remains constant before and after the collision.

6. Only the rectilinear motion is considered.

If we take the angular motion into account then we have to think about the density of the object, moment of inertia, radius of the body and the nature of the spin. So by considering all these parameter we would not be able to avoid the game from crawling.

7. Contact is non conforming

If the contact is not non-conforming then colliding surfaces will meet at more than one point. So resolving the forces at each point of contact would increase the computational cost. The greater the computational cost the slower will be the speed of the game.

1.7 Research Contribution

In this project, rigid and regular convex objects with physically plausible dynamic behaviours are investigated, which can be used in games.

1.7.1 Collision Response Modelling

1) The main contribution is in the development of a mathematical model. The model developed here deals with the contact response of two regular convex objects having non-conforming contact (sphere-sphere contact). This mathematical model is as good for spheres of the same masses as it is for spheres of different masses. A dimensionless physical quantity known as coefficient of restitution has been introduced. This coefficient is also termed as impulse ratio. It is necessary for deciding the nature of collision that is elastic or inelastic.

2) This model is as good for both head-on and oblique collisions.

3) This model is covenant with both situations; spheres are either moving in the same direction or in the opposite direction.

1.7.2 Simulation of Collision Response

(1) In this area the contribution is mainly in the realistic graphical representation of the response of the two spheres after collision, based on the mathematical model. This is the case of dynamic collision response. The simulation is, of course, original within the scope of the project. The simulation involves causing bodies and forces to interact in a realistic way based on physical rules. The bodies are not allowed to penetrate but will bounce off one another when they come into contact.

(2) Solving the equations of motion is also a point of concern because accurate results are required. This has been achieved by our simulation routine.

1.8 Thesis Overview

The plan of this thesis is presented as follows:

Chapter 2: Literature review. This chapter states some existing methods of physically based modeling and their comparison. It also contains reviews of previous work for the problem of collision response of colliding convex objects.

Chapter 3: Fundamentals of the impact. This chapter contains fundamentals used in collision response modelling. The understanding of collision response phenomenon needs prior knowledge of these fundamentals.

Chapter 4: Methodology. This chapter describes the system overview, the mathematical approach used for the modelling of collision response between colliding objects just before and after the collision. It also describes about the simulation of our mathematical model.

Chapter 5: Implementation. In this chapter a method is proposed to implement the model, basic algorithm and pseudo-code.

Chapter 6: Result and discussion. The present chapter presents the main results of this work and related discussion.

Chapter 7: Conclusion and recommendations. This chapter concludes the contribution of this work and recommends some future work.

REFERENCES

- [1] Sawyer, Ben . *The Ultimate Game Developer's SourceBook*. Arizona: Coriolis Group Books, Scottsdale. 1996.
- [2] Steven Poole. *Trigger Happy: Videogames and the Entertainment Revolution. USA, Arcade Books. 2000.*
- [3] Gek Siong Low (2001) *Understanding Realism in Computer Games through Phenomenology*. <http://hci.stanford.edu/cs378/papers/2001/low.html>.
- [4] Diefenbach, Paul J., and Normal I. Badler. (1997). Multi-Pass Pipeline Rendering: Realism For Dynamic Environments. *Proceedings of the symposium on Interactive 3D Graphics*, Providence . April 1997, USA, pp. 59- 70.
- [5] Juri A. Oudshoorn (1999). *Ray Tracing As The Future Of Computer Games*. Department of Computer Science, University of Utrecht In conjunction with Davilex Software November,1999.
- [6] Buck and E. Schomer. Interactive rigid Body Manipulation with obstacle contact. In 6th Int. Conf. in Central Europe on Computer Graphics and visualization, , 1998 pp 45-56.
- [7] Rtich, Brian & Canny, Johon, Impulsed Based Simulation Of Rigid Bodies. *Proceedings Of The 22nd Annual Conference On Computer Graphics And Interactive Techniques*, August 1-6,1995. Los Angeles, C.A.USA., 1995. pp.181-188
- [8] Bafraf, Analytic methods for dynamic simulation of non-penetrating rigid bodies. SIGGRAPH 89, pp 223-324

- [9] Aarff, D., Non-penetrating rigid body simulation, in State of the Art Reports, Eurographics '93, Barcelona, Spain, September 1993 pp 22-29
- [10] S.Hill Jr, *Computer Graphics*, 2nd ed. Upper Saddle River, N.J.: Prentice-Hall.2001.
- [11] Michael Bradley Cline. *Rigid Body Simulation with Contact and Constraints*. Master Thesis. The University of British Columbia; 2002.
- [12] Cris Hecker. Behind the Screen. *Game Developer Magazine*. March 1997:pp.13-17
- [13] Jong Wong Lee. *Procedural methods using physically based modeling*. Ph.D. Dissertation. The Gorge Washington University; 2001
- [14] R. Feynmann, R. eighton and M. Sands. *The Feynman Lectures on Physics*. Addison-Wesley.1965
- [15] H. Goldstein. *Classical Mechanics*. 2nd Ed., Addison-Wesley.1980
- [16] Baraff D. Coping With Friction For Non-Penetrating Rigid Body Simulation. *Computer Graphics*. 1991. 25(4): 31-40.
- [17] W. Armstrong and M. Green. The dynamics of articulated rigid bodies for purposes of animation. *Visual Computer*, Springer-Verlag, , 1985 pp 231-240.
- [18] D. Baraff, Linear-time dynamics using Lagrange multipliers. SIGGRAPH'96, 1996 pp. 43-54

- [19] Brian Mirtich “Impulse-based Dynamic Simulation of Rigid Body Systems”. PhD thesis at University of California, Berkley. 1996.
- [20] D. Terzopoulos and A. Witkin. Physically-Based Models With Rigid and Deformable Components. *IEEE Computer Graphics and Applications*, 1988 8(6): 41-51.
- [21] Demetri Terzopoulos, John Platt, Alan Barr and Kurt Fleischert. Elastically Deformable Models. *Computer Graphics*, 1987 Volume 21(4): 205-214.
- [22] Mirtich, Brian, & Canny, John. Impulse-Based Simulation of Rigid Bodies. *Proceedings of 1995 Symposium on Interactive 3D Graphics*, April 1995: 181-188.
- [23] Katsuaki Kawachi, Hiromasa Suzuki and Fumihiko Kimura. Simulation of Rigid Body Motion with Impulsive Friction Force. *Proceedings of IEEE International Symposium on Assembly and Task Planning*, 1997. Aug 1997:182-187.
- [24] S.Redon, A.Kheddar and S.Coquillart. Gaus’ Least Constraints Principle and Rigid body Simulations. *IEEE international Conference on Robotics and Automation*, 2002.Washington DC, USA.
- [25] David Zeltzer. Physically-Based Modeling: Past, Present and Future. *SIGGRAPH Panel Proceedings*, July 31-August 04,1989. New York, USA. ACM Press. 1989:191-209.
- [26] M. Desbrun, M. Gascuel. Animating Soft Substances with Implicit Surfaces. *Proceedings Of The 22nd Annual Conference On Computer Graphics And Interactive Techniques*, August 1-6,1995. Los Angeles, C.A.USA. ACM Press.1995:287-290.

- [27] Ming Lin, Stefan Gottschalk. Collision Detection Between Geometric Models: A Survey. *Proceedings of IMA on mathematics of surfaces* 1998.pp 55-61
- [28] Baraff D. Fast Contact Force Computation For Non-Penetrating Rigid Bodies. *Computer Graphics SIGGRAPH 1994 Proceedings: Annual Conference Series*, July 24-29, 1994, Orlando (Florida). ACM Press.1994: 23-34.
- [29] Mirticg Brian. Timewrap Rigid Body Simulation. *Proceedings Of The 27th Annual Conference On Computer Graphics And Interactive Techniques*, July 27 2000. New Orlando (Florida). ACM Press.2000:pp.193-200.
- [30] Goldstein. H. *Classical Mechanics*. 2nd Ed. N.Y. Addison-Wesley.1980.
- [31] Feynmann. R. *The Feynmann Lectures on Physics*. N.Y. Addison-Wesley. 1965.
- [32] David M.Bourg, *Physics for Game Developers*, Published by O'Reilly & Associates, Inc., USA 2002
- [33] Ramírez, Rosa; Pöschel Thorsten; Brilliantov Nikolai V. and Schwager Thomas. Coefficient Of Restitution Of Colliding Viscoelastic Spheres. *Physical Review* October 1999. Vol. 60(4): pp.4465-4472.
- [34] D.Gugan (2000) “ Inelastic Collision and Hertz Theory of Impact” *American Journal of Physics*, Vol. 68, No 10, pp. 920-924.
- [35] Kozlov, Valerii V. and Treshchev, DmitriiV., *Billiards: A Generic irtoduction to the Dynamics System with Impacts*, American Mathematical Society, 1991.