Evaluation of Collision Handling Techniques in 2D collisions

Nitisha Warkari, Mugdha Jamsandekar
{nitisha.warkari, mugdhajams}@gatech.edu
Masters in Computer Science,
Georgia Institute of Technology.

Abstract

In this paper we study the animation in two dimensions of disks moving at a constant velocity in a frictionless environment. We analyze the effect of collisions between disks and between a disk and an edge. There are two approaches to animation – Periodic Interference Test (PIT) which computes the velocities of the disks at each frame and Predicted Instant of Collision (PIC) which computes the exact time at which collision occurs, advances the animation to this point and then calculates the new velocities exactly at the point of collision. This paper describes these two approaches and their associated advantages and disadvantages.

1. Introduction

Collision detection is an essential component of animation and plays a major role in application like part in physical simulations, video games and computational geometry. This involves algorithms which compute the point in time and space at which two given solids intersect, and calculate the new velocities and trajectories of the two solids after the impact.

There are two approaches for detecting collisions – *Periodic Interference Test* and *Predicted Instant of collision.* The Periodic Interference Test approach is the simple of the two. It checks for intersection at the beginning of each frame. If the solids are found to be intersecting, their new velocities and trajectories are calculated. The Predicted Instant of Collision approach computes the exact time at which the collision occurs. If the collision occurs at a time t_{col} before the next frame, the animation is advanced to t_{col}. The PIC approach is found to be much more accurate as compared to the PIT approach. The PIC approach is thus popularly used in application like a Billiards game while precision is critical. Application like video game design which typically tradeoff accuracy for computational speed however prefer the simpler PIT approach.

The paper is organized as follows. Section 2 describes the basic physics and vector mathematics involved in an elastic collision between two solid bodies. Sections 3 describes the PIT and PIC techniques of collision detection and their associated advantages and disadvantages in detail. Section 4 illustrates with four examples how and why the PIT approach is flawed. Section 5 describes the physical simulation designed by us which uses the PIC approach for collision detection.

.

2. The physics behind collisions

In this paper we describe elastic collision in a frictionless environment that preserves both energy and momentum.

2.1 Collision between two disks

Consider two balls B_1 and B_2 having initial velocities U_1 and U_2. After the two balls collide, their normal velocities get exchanged. The normal N at the point of contact is given by the equation

$$N = (C_1C_2)N_u$$

where N_u is the unit normal vector at the point of contact and C_1 and C_2 are the centers of the two balls at the time of collision.

The normal components U_{1n} and U_{2n} of balls B_1 and B_2 are :

$U_{1n} =$ dot$(U_1. N)N$

$U_{2n} =$ dot$(U_2. N)N$

The new velocities V_1 and V_2 of the balls B_1 and B_2 after their collision are :

$V_1 = U_1 – U_{1n} + U_{2n}$

$V_2 = U_2 – U_{2n} + U_{1n}$

2.2 Collision with a static edge

When a ball collides with a vertical wall, its normal velocity is reversed while its tangential velocity remains unchanged.

If the initial velocity of the ball is $V = V_t + V_n$. The new velocity after the collision would be $V_{new} = V_t - V_u$

When the ball collides against a vertical edge, the y component of the velocity remains unchanged while its x component gets reversed. The new velocity now becomes $V_{new} = V_y – V_x$.

When the ball collides against a horizontal edge, the x component of the velocity remains unchanged while its y component gets reversed. The new velocity now becomes $V_{new} = V_x – V_y$.

3. Two Approaches to animation – PIT and PIC

There are two approaches to animation the Periodic Interference Test (PIT) and Predicted Instant of Collision (PIC). PIT is the simpler and less accurate of the two approaches. PIC generates a more realistic animation. The following sections describe these two approaches in detail.

3.1 Periodic Interference Test (PIT)

In PIT, we advance the physical simulation by a small time step say t, then check if any objects are colliding or intersecting. At each simulation step, a list of all intersecting bodies is created, and the positions and trajectories of these objects are somehow "fixed" to account for the collision. Here, we typically miss the actual instant of collision, and only catch the collision after it has actually happened.

3.1.1 Advantages of PIT

- The collision detection algorithm need not be aware of the different physical variables. A simple list of physical bodies is fed to the algorithm, and the program returns a list of intersecting bodies.
- This collision detection algorithm doesn't need to consider friction, elastic or non-elastic collisions and deformable bodies (which in normal cases, is considered in PIC).
- PIT is in effect, one dimension simpler than PIC since PIC must deal with the time variable, which is absent in PIT.

3.1.2 Disadvantages of PIT

PIT causes problems in the "fixing" step, where intersections (which aren't physically correct) need to be corrected.

3.2 Predicted Instant of Collision (PIC)

In PIC, we write a collision detection algorithm which will be able to predict very precisely the trajectories of the physical bodies. The instants of collision are calculated with high precision. If this time is less than the time of the start of next frame, the motion at that instant is shown. Here, the physical bodies never actually interpenetrate. Here, we calculate the instants of collision before we update the configuration of the physical bodies and hence, this method is more accurate than PIT.

3.2.1 Advantages of PIC:

- *Increased fidelity and stability* - PIC uses a large number of variables to determine when the collision exactly happens. PIC is used in applications like billiards which require high precision in their animations.
- *Handling resting contact* - Some objects are in resting contact, that is, in collision, but neither bouncing off, nor interpenetrating, such as the initial configuration of balls in billiards. In all cases, resting contact requires special treatment. This is difficult to handle in PIT.

3.2.2 Complications and challenges in PIC.

- *A large number of physical variables* - The collision detection algorithm needs to consider many physical variables.
- PIC should ideally consider friction, elastic or non-elastic collisions and deformable bodies.
- *Acceleration component* - When disks do not move at constant velocity, we need to consider the acceleration component. It is not impossible to include it when the path followed by the object is a linear motion. We need to use more complicated motion equations like which would be more computationally intensive. The position of an object needs to be constantly monitored for every pair of objects thus increasing the resultant complexity.
- *Non linear motion* - In case of rotation or any other non-linear motion, we'll have to consider non-linear paths like conics. This would further complicate the equations and hence, calculations. Accordingly, the speed of computation would reduce even further degrading the performance of the system.

3.3 Where PIT is preferred over PIC

Some applications like video games involve a lot of complex, rapidly changing animation. Video games thus strive to generate believable simulations within stringent time constraints. The issue which needs to be dealt with here is time and not accuracy. Such applications typically resolve collision detection using simplified rules such as such as PIT.

4. Cases where PIT fails

We study the collision between two balls to analyze four cases in which the approximation method used the PIT method would result in erroneous results.

4.2 Missed collision

It may happen that the balls collide and move out of each other's interference zone within a frame interval. In this case the collision goes undetected and the balls continue with their original velocities. This is illustrated by the following figures.

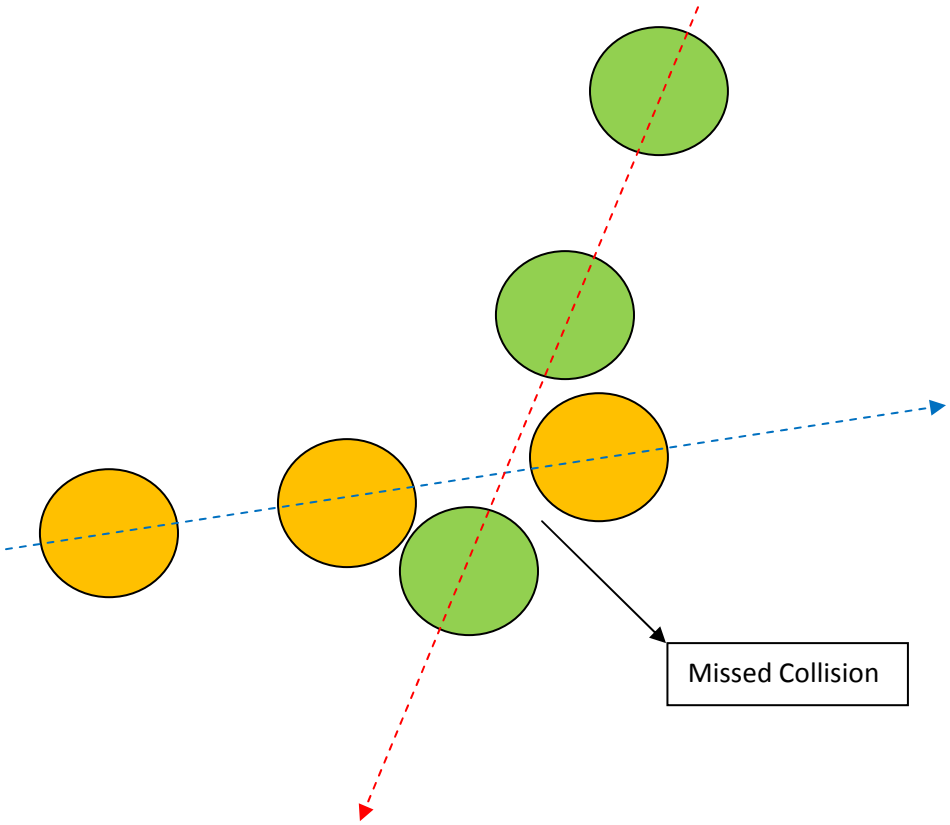

Fig. 1 Missed Collision due to PIT

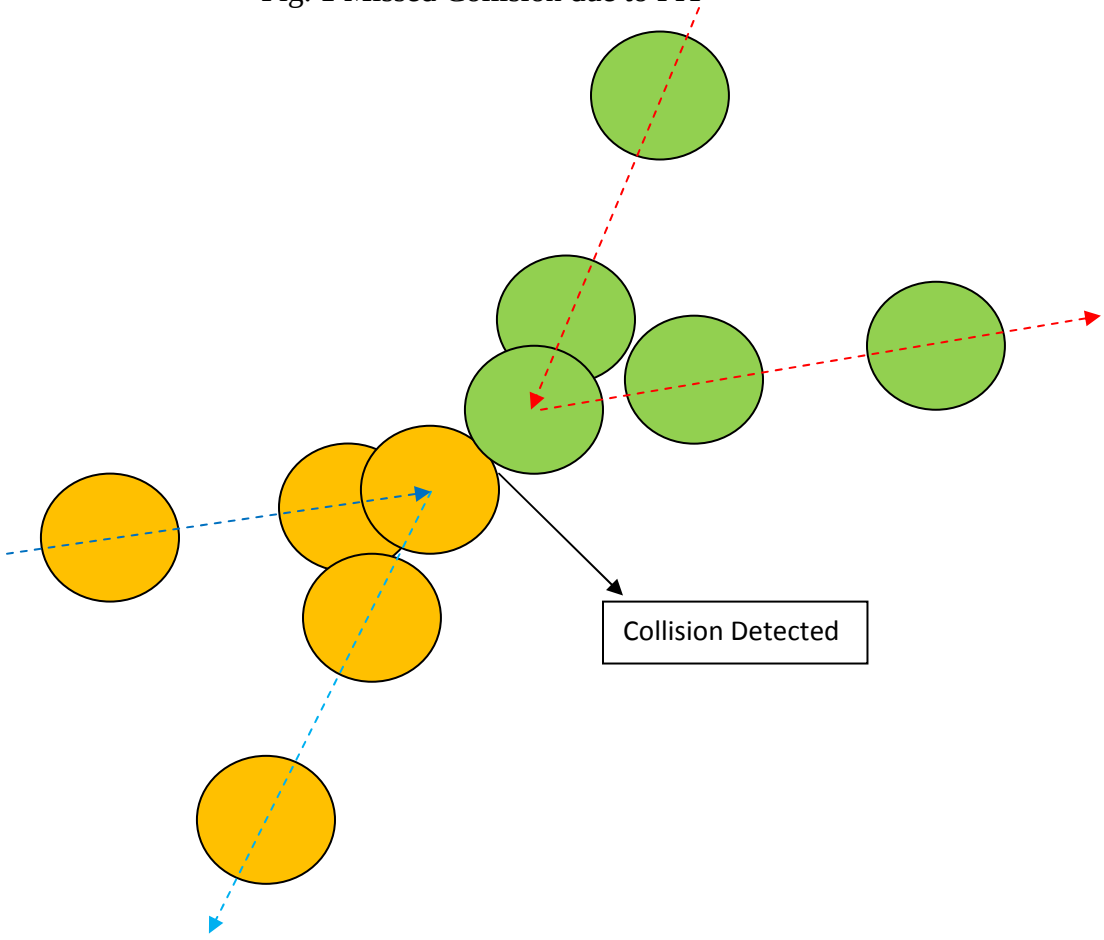

Fig. 2 Correct Collision detection using PIC

4.2 Penetrating balls

The collision between two balls is typically determined late in PIT, at the time when they interfere instead of at the time when they collide. The calculated trajectories would thus be incorrect.

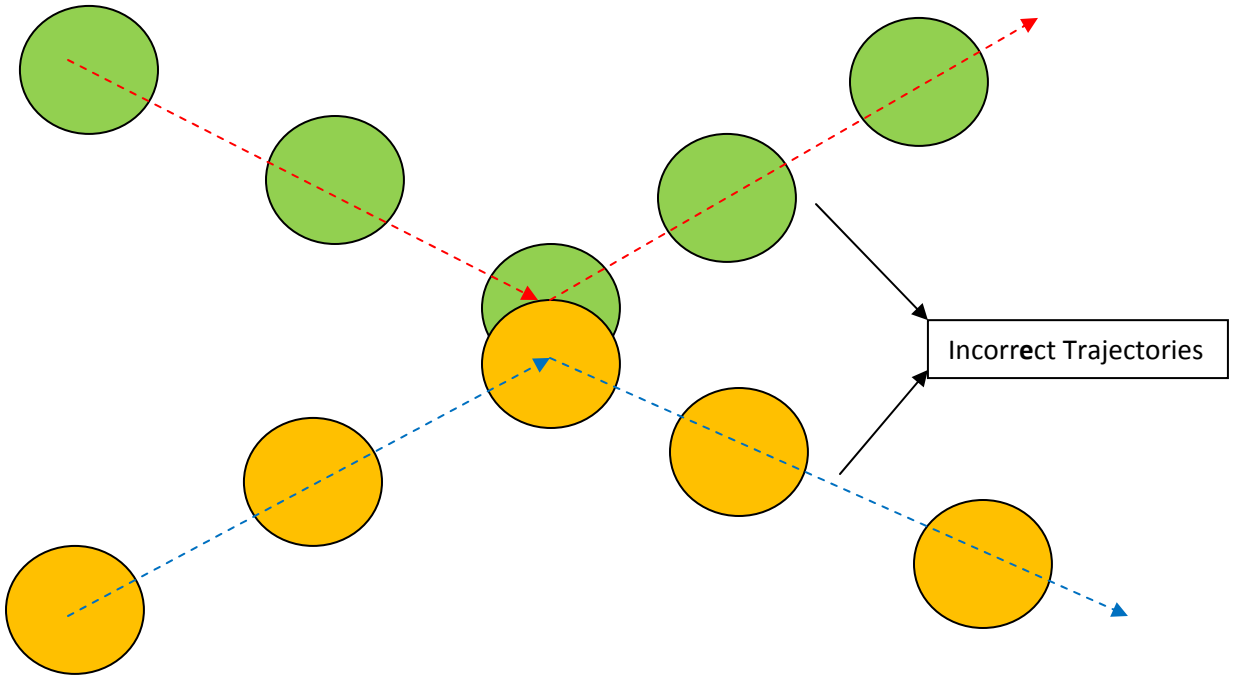

Fig. 3 Incorrect trajectories calculated due to PIT

PIC on the other hand calculates the exact instant of collision and can accurately calculate the velocities and trajectories of the two balls at the exact time of collision. The correct expected trajectory is calculated by PIC as shown below.

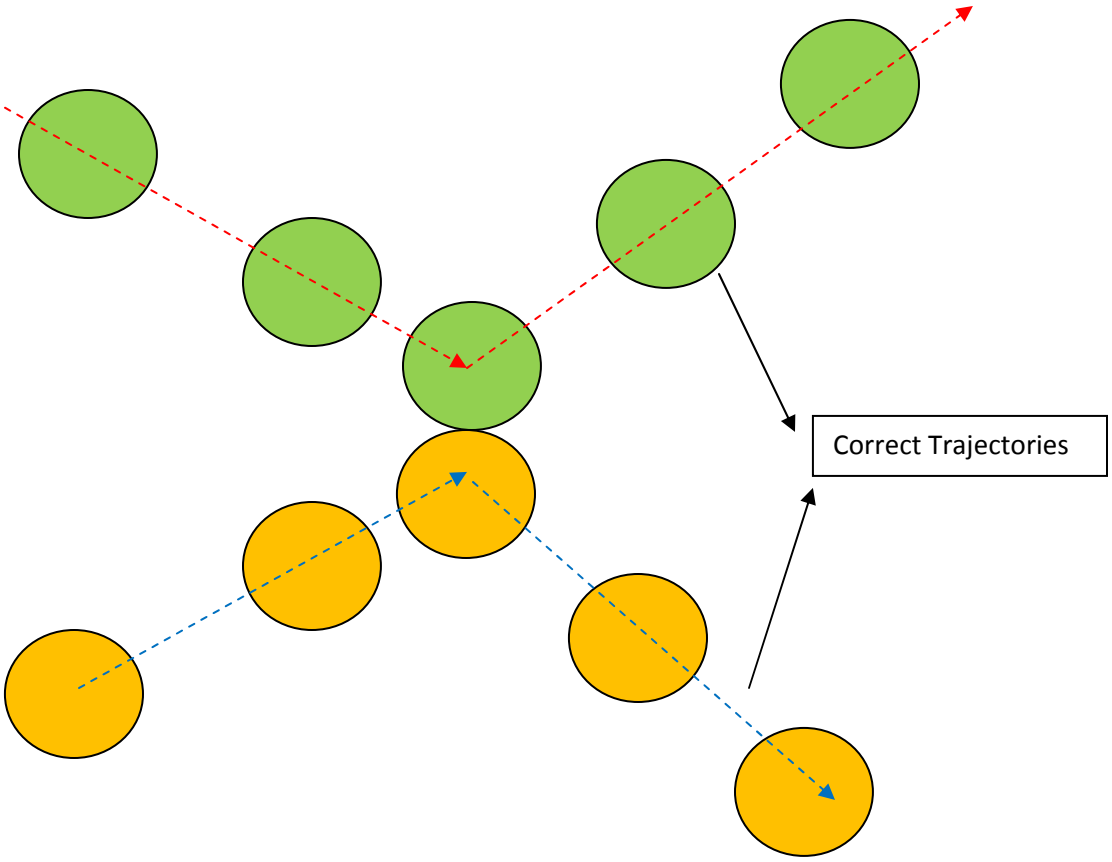

Fig. 4 Correct trajectories calculated by PIC

4.3 Multiple Collisions

In PIT, multiple collisions that should happen between two frames are not handled properly.

This can happen in billiards when a moving ball B1 collides with a ball B2 near the static boundary of the board and B2 bounces back and collides with B1 again within the same frame. These multiple collisions will not be caught in the PIT method if they occur within the same time frame. Hence, there will be an error in the calculated resultant velocity.

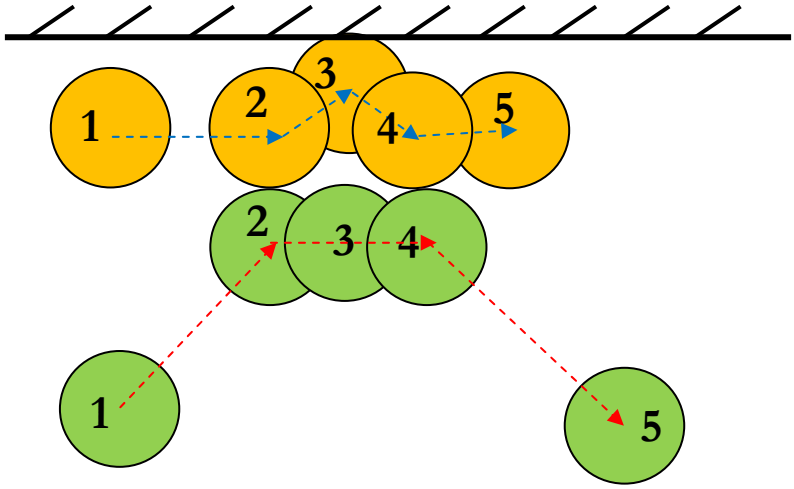

Fig. 5 Multiple collisions correctly detected in PIC

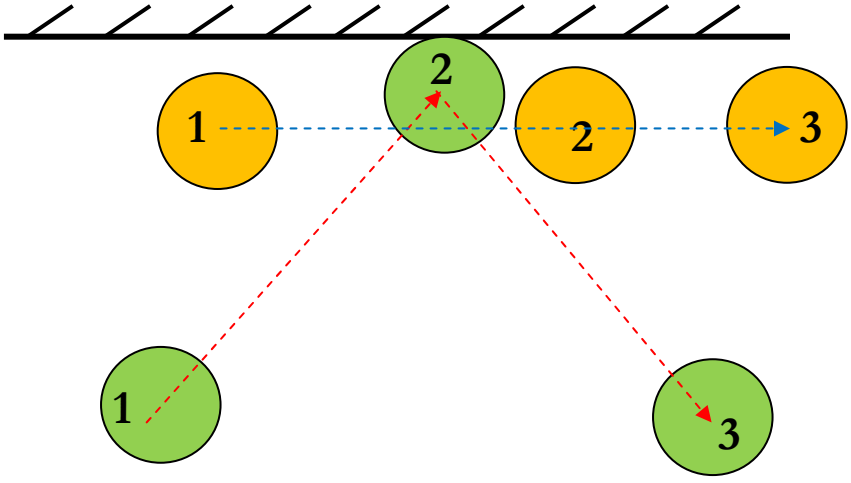

Fig. 6 Multiple collisions skipped in PIT

4.4 Wrong Direction after Bounce

In this case, we assume that one ball is static (of very high mass). Suppose that the other ball collides with the static ball and bounces off it at an instant between the frame start and end. In PIC, this collision is accurately detected and simulated. In PIT, the collision is detected at the end of frame at a time later than that of actual collision. Hence, the direction in which the ball travels after bounce is wrong.

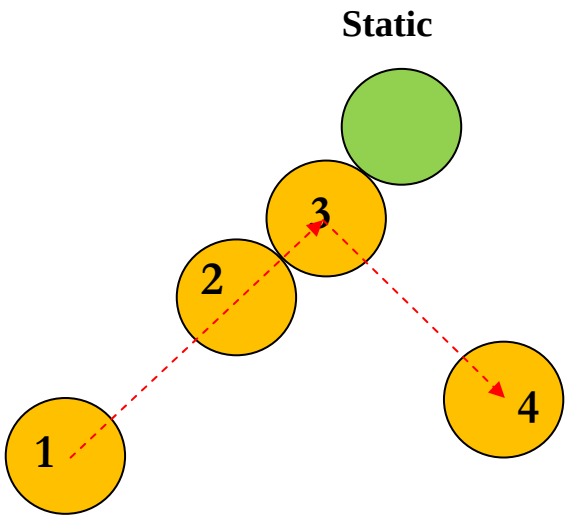

Fig. 7 Correct collision using PIC when a ball is static

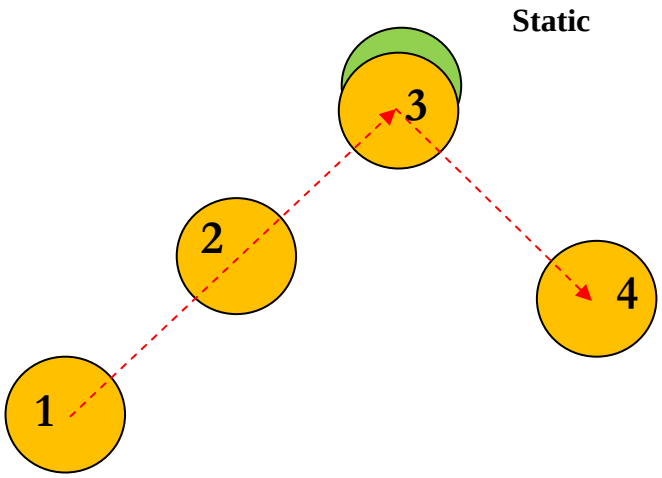

Fig. 8 Incorrect collision using PIT when a ball is static

5. Billiards game – A sample implementation

As an example of physical simulation, we have constructed a multi-player game of billiards. An initial description of the situation including a precise physical description of the billiard table and balls and initial positions of all the balls is given. Given an arrow which represents the initial velocity vector (magnitude and direction) with which the cue is hit on the white ball, we calculate the trajectories, precise motion, collisions and eventual resting places of all the balls in the board. We have designed a simulation of this game which would consist of calculating the precise impacts between the billiard balls. Since, the collisions between billiard balls are nearly elastic, and the balls roll on a low-friction surface, we have considered elastic collision, meaning that the momentum and energy is conserved and zero friction.

This particular example is numerically unstable, i.e. a small error in any calculation will cause catastrophic changes in the final position of the billiard balls. Hence, it is essential to be precise while handling the calculations. We have therefore used the Predicated Instant of Collision (PIC) method

6. Conclusion

In this paper, we have explained and evaluated the two main collision handling techniques namely 'periodic Interference Test (PIT)' and 'predicted Instant of Collision (PIC)'. We have discussed the advantages and disadvantages of these techniques and the subsequent applications for the same. We have also depicted the four main types of erroneous situations that can occur if PIT is used. Finally, it can be stated that the selection of a technique depends on the type and requirements of the application. We have designed a game of billiards using the PIC technique for collision of the billiards balls. The velocity and position of the balls at any instant is calculated using the formulae mentioned above. Hence, this can be used to learn the nature of PIT technique.

Bibliography

1) "Collision Detection – Encyclopedia", http://www.experiencefestival.com/a/Collision_detection/id/1992555

2) "Collision Detection", Wikipedia http://en.wikipedia.org/wiki/Collision_detection