An intelligent software tool for avoiding collision in complex radiotherapy set-ups

Bineesh KV [1*], Ilapakurty Ravi[2], GVS Manyam[3]

[1*]Department of Electronics, Panacea Medical Technologies Pvt ltd, Banglore
[2]Department of Electronics, Panacea Medical Technologies Pvt ltd, Banglore
[3]Department of Mechanical, Panacea Medical Technologies Pvt ltd, Banglore
bineesh.kv@panaceamedical.com[1*], iravi@panaceamedical.com[2], gvs@panaceamedical.com[3]

Abstract

Gantry- Couch/ Gantry- Patient collision is a serious problem in various medical equipments in radiotherapy. This collision may cause damage to the machine or even may cause injury to the patient. In this article, an efficient method of avoiding collision that is inherent in the radiotherapy machine is presented. The study uses an intelligent co-ordinate geometry based algorithm for avoiding collision in machines. The overall computation takes less than 0.1 seconds for sensing the collision. Also this new technique for avoiding collision is tested with two different radiotherapy machines namely Imagin and Bhabhatron-II. This is a faster and accurate method of avoiding collisions in complex iso-centric set-ups.

Keywords: Radiotherapy, Collision detection, Medical equipment, Complex radiotherapy set-ups, iso-centric.

1. Introduction

Collision between Gantry and couch or gantry and patient is a serious problem during treatment time in many radiotherapy equipments. This collision causes damages to machine or even cause injury to the patient. Thus it is necessary to avoid collisions in the machine. The main intension of the software is to avoid the collisions between patient and the machine and in between different moving parts of the machine. The article describes a new algorithm for sensing the collision during the machine movement and by stopping the motions avoid collision. The main challenge in this software design was the development of an algorithm to detect collision of objects in space.

All radiotherapy machines provide the flexibility in movement in almost 18 directions. The paper describes an intelligent mechanism to determine the chance collision. The software requires only few inputs during the run time. The machine's motion parameters like gantry angle focus to Axis distance (FAD), couch angle, couch longitudinal, lateral, vertical, image intensifier lateral, Longitudinal and vertical, etc. Since these parameters are important in the treatment, the parameters will be already available with the machines.

The main problem involving in the software design is the way of determining the collision in advance using the above mentioned parameters. One way is to create a look up table of machine positions at which there is a chance of collision. But this way is not much practical, since there is theoretically infinite numbers of positions at which machine collides. Figure 1 shows few machine positions showing collision. The second way is to derive equations based on the trigonometry of the machine setup. But in this case also equations changes with respect to machine and it may lose some kind of preciseness. Another way of avoiding collision is by using proximity sensors. That is when different machine parts come into near area, the proximity sensors fitted on it can detect the chance of collision. But this method has its own disadvantages like higher cost of proximity sensors; and also it will be very difficult to detect the machine-patient collision.

2. Materials and Methods

This article proposes an intelligent co-ordinate geometry based technique for avoiding collision in therapy machines. The software only requires the motion parameters as described before in run-time. The first step involved in this technique is to model the entire machine using the software. For this purpose complete machine dimensions like couch length, breadth, thickness, collimator radius, imager radius, height etc are required. After obtaining these values, the entire machine in the software was modeled. One of the simplest way of doing this is representing each item in the machine using simple 3D structures like rectangular boxes, cylinders etc, for example, representing the couch using a number of rectangular faces. Similarly, image intensifier is of somewhat complex cylindrical shape and represented using simple polygonal or cylindrical shape. Similarly, the whole machine was modeled. Figure 2 shows the actual image of RT- machine namely Imagin (from Panacea Medical Technologies pvt ltd) and Figure 3 shows its approximated mathematical model.Once the modelling of the machine is completed, next aim is to model the patient. The patient can be modelled perfectly as a semi-cylinder as shown in the Figure 4.

Fig.1. *Mathematical models of Imagin with different collision positions*

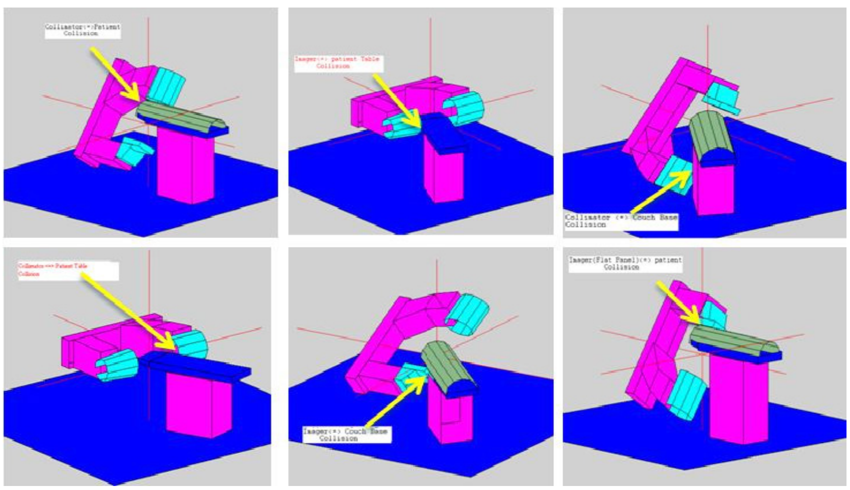


Fig.2. *Actual radiotherapy machine (Imagin) (Mechanical design team, Panacea Medical Technologies Pvt ltd)*

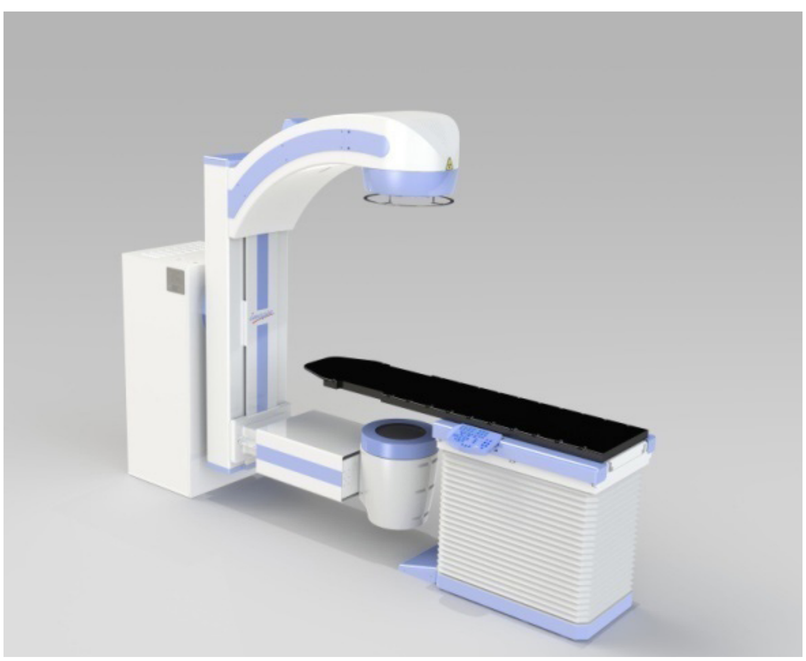
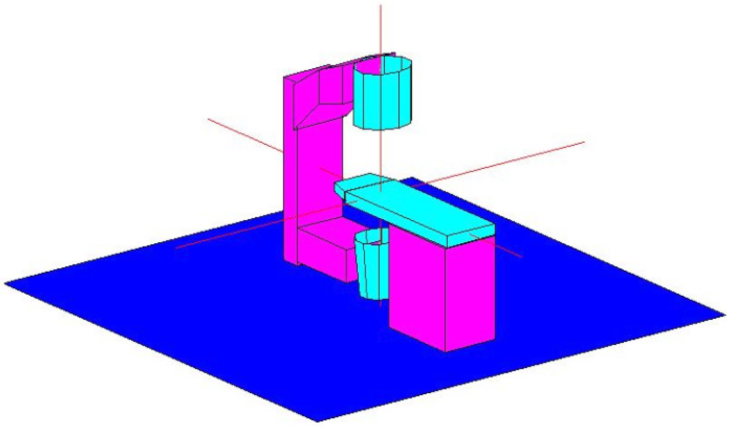


Fig.3. *Mathematical model of Imagin*



3. Algorithm

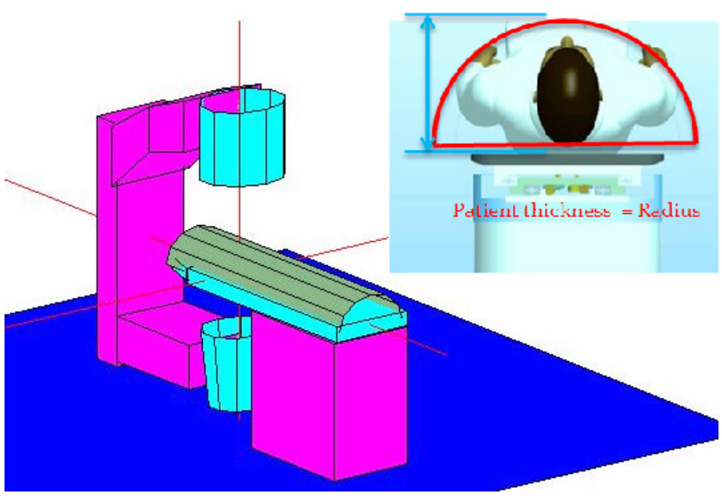
One of the challenging works in solving the problem of collision is the development of an efficient algorithm to detect collision between different part of the machine, machine and patient and machine and floor *etc.* The main parts involved in this are follows:

» Virtually, construct the machine in Iso-centric position

» Virtually, reconstruct the machine with its motion parameters

» Apply collision detection algorithm to every new machine position and detect collision

Fig.4. *Patien Modelling*



The following section of the paper describes an effective algorithm for detecting collision in the machine. We can see from figure 3 that the machine is completely modelled using rectangular, polygonal, cylindrical geometrical shapes. And each shape consists of rectangular/ polygonal faces. The main challenge is detecting collision among the different geometrical shapes in space. Each geometrical structure is made up of a number of edges and faces. For example, a box is made up of 12 edges of 6 faces. The following section describes how to find collision in between two boxes in space.

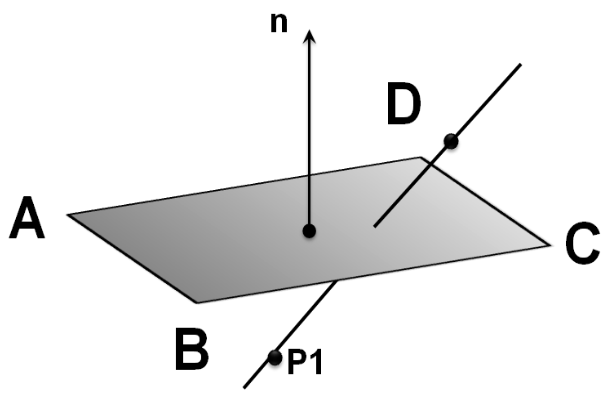
When two boxes collide with each other, any edges of the first box will intersect with any face of the second box or any faces of the first box intersect with any edge of the second box. Without any type of these intersections, the boxes will never collide. So in order to detect collision, best way is to detect these intersections.

Consider an edge in the first box and a face in the second box. Firstly, it is required to check the line containing the edge is intersected with plane containing the face. If there is no intersection, there won't be any collision; otherwise the point of intersection was found and checked whether the point of intersection is inside the face or outside. If it is inside the face there is a collision otherwise there won't be any collision. Following section describes how this can be achieved.

3.1 Face-edge intersection

Consider an edge or line segment with two end points P1, P2 and a face with four end points A, B, C, D as shown in the figure 5.

Fig.5. *Face Edge intersection (Adopted from Wikipedia)*



The equation of the two vectors representing the plane is

AB = B – A

AC = C – A

Therefore, normal to the plane will be the dot product of AB and AC

n = AB dot AC

Where n is the normal to the plane.

Equation of the vector passing through the point P1 and P2 is

P = P1- P2.

The dot product of the normal to the plane and vector P will give whether line is parallel or not parallel to the plane.

T0 = n dot P

If T0 is zero then line will be parallel to the plane and therefore, there will not be any point of intersection

But if T0 is a non-zero then the line will not be parallel and there will be a point of intersection and point of intersection can be calculated as follows.

Suppose,

T1 = n dot (A – p1) and

V = T1/T0

There will be point of intersection when V > 0 and the point of intersection is

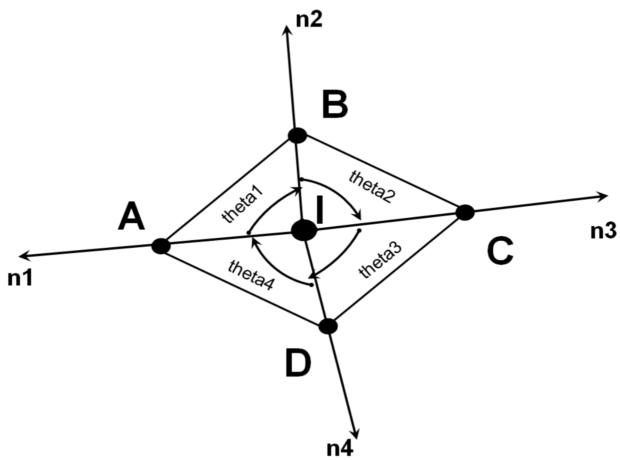
Intersection = P1 + ((P2 – P1) * V)

3.2 Checking the point of intersection is inside or outside

Thus a point of intersection between the edge of the first box and plane containing the face of the second box can be obtained. Still there is a chance that the point of intersection may be outside the face. This can be calculated as follows:

Consider a rectangular face ABCD and a point I as shown in the Figure 6. This is required to check whether, the point I is inside the rectangle ABCD or not. For this purpose, the vectors n1, n2, n3 and n4 was calculated as follows:

Fig.6. *Rectangular face with point*



n1 = A – I

n2 = B – I

n3 = C – I

n4 = D – I

The angle between n1 and n2, n2 and n3, n3 and n4, n4 and n1 was calculated as follows:

Theta1 = Cos-1 [(n1 dot n2)/ (modulus (n1) * modulus (n2))]

Similarly, *theta2, theta3 and theta4* values are calculated and added. If the sum is 360^{o}, then the point will be inside the rectangular face, else the point will be outside the face and there will not be any collision

4. Implementation and Testing

Since this collision detection algorithm does not use any in built-in functionality available in any language or platform, the algorithm can be implemented platform independently. The algorithm is implemented both in C language and in Matlab. Since the Matlab providing some kind of easiness in drawing, it is better to implement the algorithm in Matlab and if required the code can be easily converted into C.

The collision detection algorithm is implemented in C and is tested with two machines used in Radiotherapy namely Imagin and Bhabhatron-II. The algorithm taking less than 0.1 second in executing in Intel Core 2 Duo PC with 2 GB RAMS. Depending on the complexity in the machine construction, there is a chance of difference in the execution time.

5. Conclusion

In the paper, an efficient method of detecting collision that is inherent in the medical equipments used in radiotherapy is discussed. The technique only uses the computing capability of the PCs used with the radiotherapy machine. No additional hardware changes are needed. So it can be concluded that this is a most cost effective way of avoiding collision in radiotherapy machines.

6. Acknowledgment

I express my sincere gratitude to Panacea Medical Technologies Pvt ltd, Banglore for providing the Radiotherapy machines for testing the proposed control software.

7. References

1. Chiaho Hua, Jenghwa Chang, Kamil Yenice, Maria Chan, and Howard Amols (2004), A practical approach to prevent gantry-couch collision for linac - based radio surgery, Medical Physics, Vol. 31(7), July 2004.
2. I BEANGE and A NISBET (2000), Collision prevention software tool for complex three dimensional iso-centric set up, The British Journal of Radiology.
3. Eli E. Furhang, Joseph Hanley, Sou-Tung Chiu-Tsao, Sean Toner, Pei Fan, Paul Gliedman, and Louis B. Harrison(2002), Clearance assurance for stereotactic radio surgery and radiotherapy, Medical Physics, Vol 29(1), January 2002.
4. Miltiadis F. Tsiakalos, Eduard Schrebmann, Kiki Theodorou and Constantin Kappas (2001),Graphical treatment simulation and automated collision detection for conformal and stereotactic radiotherapy treatment planning, Medical Physics, Vol 28(7), July 2001.
5. http://en.wikipedia.org/wiki/Line-plane_intersection.
6. http://en.wikipedia.org/wiki/Dot_product.