Collision Detection Software for Linac-based Cranial Radiosurgery

Introduction: During stereotactic arc therapy for cranial lesions there is a significant risk of the Linac gantry colliding with the treatment table and/or headframe, resulting in equipment damage or patient injury. To prevent this from occurring in our clinic we manually check for collisions at the machine before the patient is treated. The problem with this approach is that when a collision is detected at the machine, the finalized treatment plan has to be re-planned. This disrupts the clinical workflow and can result in a rescheduling of a patient's treatment. To reduce these instances we developed a collision prediction software program that will predict whether or not an arc will collide with the table, given specific plan parameters.

Necessary Inputs: In order for the software program to be easily implemented, only a limited amount of initial table and gantry information should be used. For the gantry measurements, only gantry-to-isocenter distance, gantry radius, cone-to-isocenter distance, and cone radius were used. For the table, we developed a code to take a handful of dimensional measurements and construct a discretized virtual table with any desired resolution. The entire commissioning procedure necessary to implement the program takes between 30 minutes to an hour for each table type that you want to analyze.

Prediction Software: Once the initial inputs are made, the table points that were created are shifted by the patient-specific isocenter coordinates, and then rotated by a particular table rotation. Each point is then converted into cylindrical coordinates with the z-axis along the axis of gantry rotation and the origin at the plan isocenter. By only analyzing these cylindrical coordinates of the specific table points we can effectively predict when and where a collision will occur by testing whether it will fall into the gantry's path. The calculation time using this method is greatly reduced from that of other analytical collision prediction programs, as noted in Furhang *et. al*. (1). Each arc analysis takes 3 seconds on a 3.2 GHz Intel processor with 1.0 GB of RAM.

Accuracy: To test the accuracy of the software we tested 28 different patient plans with a total of 161 arcs. These arcs were a mixture of treatments using cones and treatments using dynamic MLCs. One plan was a static field IMRS treatment. The error in the software prediction is the actual minimum distance between the couch/headframe and the gantry, minus the predicted minimum distance for a given arc. This is shown for all 161 arcs in Figure 1. Ideally, all points on this graph would lie on the x-axis, meaning that the software correctly predicted the actual measurements for all cases. Since this is not true for our software, we would still like to know how well it is performing. For all points lying to the right of the black line (which represents an actual collision) and above the x-axis (region 1), the software reports a conservative minimum distance to the treatment planner. Points lying in region 2 indicate that the software gives a non-conservative value for the minimum distance, but there is no collision to worry about. Region 3 indicates that the collision software correctly predicted an actual collision. Region 4 represents the situation where the software does not predict a collision, but an actual collision does occur. Since there are no points lying in region 4, the program works as was intended.

Significance and Future Directions: This software program has been used successfully in our clinic for a year. Since its implementation there have been no cases where a patient's plan had to be redone as a result of a manual collision check. As a result, the treatment planners are more confident in selecting arcs when planning a particular patient. The workflow is also not disrupted, which is very valuable in a busy clinic. As of right now the software can only check arcs with particular isocenter, table angle, and gantry start and stop angles (Input/Output screen shown in Figure 2, sample visualization shown in Figure 3). A potential improvement to the software would be to only input a lesion isocenter and then generate a full map of available table angles and gantry angles to guide the treatment planner in the planning process. This software could also be used with CT data to simulate potential collisions with the patient's body.

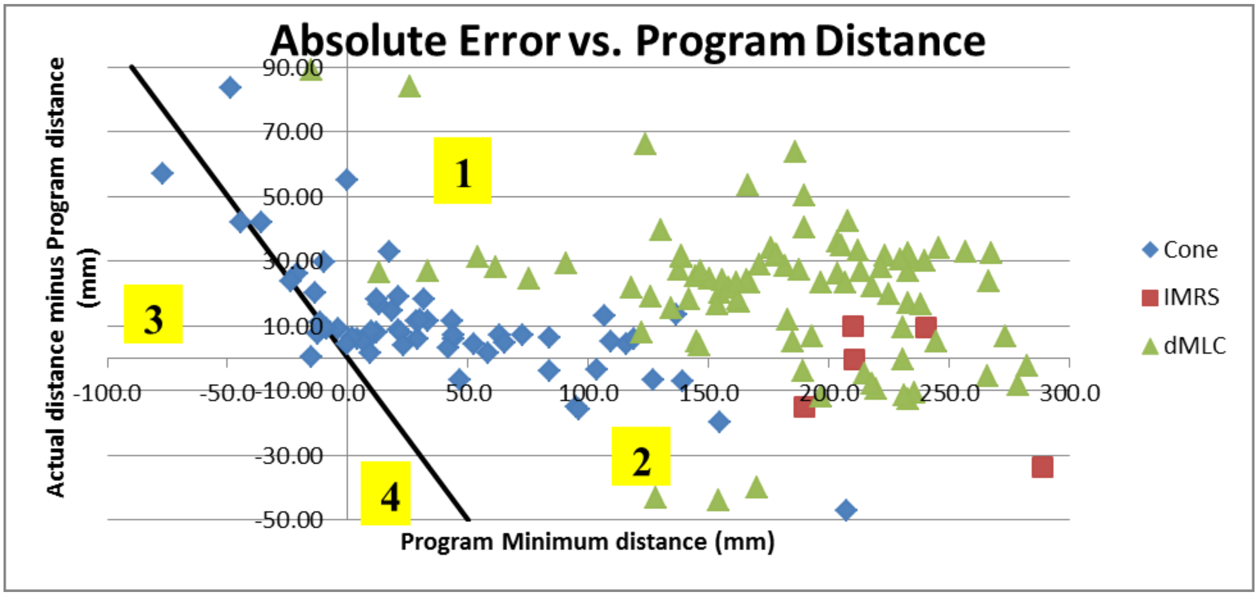

Figure 1 – Analysis of the error in the software. Points to the left of the black line indicate an actual collision.

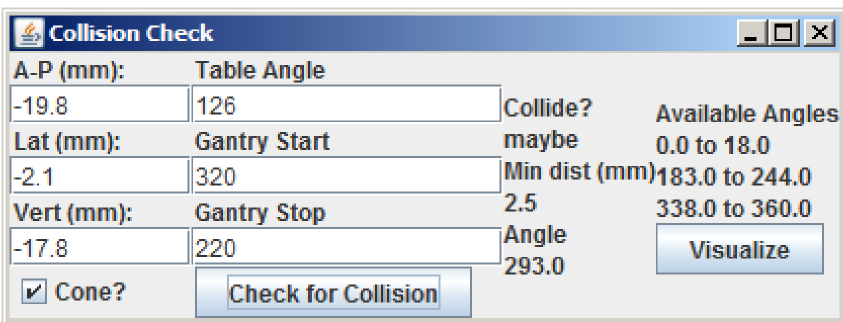

Figure 2 - An example of the Input/Output screen of the Collision Prediction program. Input the Isocenter, the table angle, and the gantry range to get the closest distance the gantry gets to the table and at what angle. Also gives available gantry angles for a particular table angle.

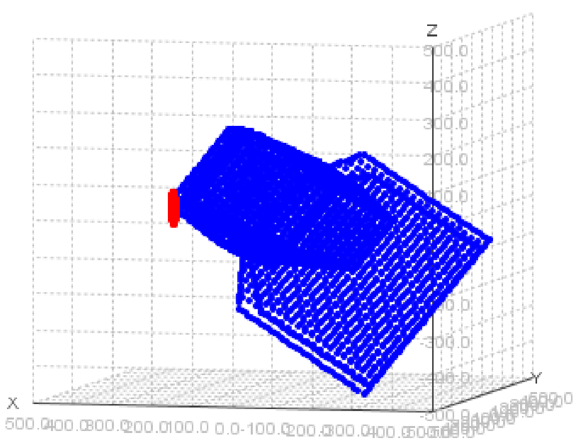

Figure 3 – A screenshot of the visualization graph used to see how close the gantry comes to the table, and where it occurs. The table is in blue and the cone is in red. This gives the treatment planner a useful way to change the plan if needed.

References:

(1) Furhang, Eli E., et. al. "Clearance assurance for stereotactic radiosurgery and radiotherapy." Med. Phys. 29 (1), pp. 45-50, January 2002