

Stable Collision Response Method in Complex Interactions

J. Hernantes¹, I. Diaz¹, H. Alvarez² & D. Borro¹

¹CEIT and TECNUN (University of Navarra). Manuel Lardizabal 15, 20018 San Sebastian, Spain

²University of the Basque Country, Manuel Lardizabal 1, 20018 San Sebastian, Spain

Abstract

Haptic technology is quite recent and therefore in many cases it is difficult to simulate real contacts or interactions with a high sensation of realism. Collision response methods that calculate the force-feedback tend to cause haptic instabilities when the normal direction changes abruptly. In consequence, collision events in sharp corners are often difficult to render properly by means of haptic devices. This paper describes a collision response method that first distinguishes the nature of the corners -concave or convex- and then provides an adequate solution in each case to avoid haptic instabilities.

Categories and Subject Descriptors (according to ACM CCS): H.5.2 [Information Interfaces and Presentation]: User Interfaces — Haptic I/O, I.3.4 [Computer Graphics]: Three-Dimensional Graphics and Realism — Virtual reality

1. Introduction

Humans are able to perceive the environment using all their senses. Usually sight is the predominant sense, although some of the other senses are also needed to perform most tasks. For instance, maintainability studies need accessibility tests to verify whether each part of the mock-up is accessible or not. Obviously, a visual test is not enough to detect possible inaccessible parts. Providing users with the natural ability to use all their senses in a simulation environment is an important goal in the Virtual Reality research area. Within this context, haptic devices are used to provide us with force feedback in domains where it is needed, enhancing considerably interactivity. The process of computing and generating forces in response to user interactions with virtual objects is known as haptic rendering [SBM*95]. Three main modules can be identified in a typical haptic rendering algorithm: collision detection, collision response and control modules. This paper focuses specifically on the second module. Previous works dealt with the voxel-based collision detection [BGAM04] and control algorithms [GAGB05].

2. Description of Collision Response Algorithm

When two objects collide, the group of collided triangles of the static object is defined as the collision area. The method proposed subdivides this area into smaller areas called contact zones in order to facilitate the computation of the collision

final haptic force. Each contact zone is represented by its resultant penetration value and normal vector. The final haptic force is computed taking into account the information of each contact zone.

The subdivision of the collision area into different contact zones is achieved taking into account the proximity and distribution of the static colliding triangles in the 3D space. After that, the method subdivides each contact zone according to the angle between normal vectors of the triangles grouped in each zone. The normal of each contact zone is computed as the average normal vectors of its faces.

Next, we compute the exact penetration depth using a perpendicular contact plane for each contact zone and subset of the vertices which define the tool model and are contained inside a sphere with radius r .

The final step is to obtain a proper resultant normal vector and a penetration depth value based on the information of each contact zone. Two different strategies have been developed.

2.1. Method based on penetration weights

The idea is to compute the resultant normal direction in terms of a weighted average, where the weights depend on the computed penetration values (Figure 1a). This is a valid strategy for convex corners. However, applying it to concave corners, we observe that the normal vector which belongs to the second contact zone will have a higher influence in the

final normal because its penetration value is bigger. In consequence, an incorrect value of the normal vector is obtained (n_{wrong}). As consequence, a different criterion has been established for convex corners. The assignment of weights will be carried out in the opposite way (Figure 1b).

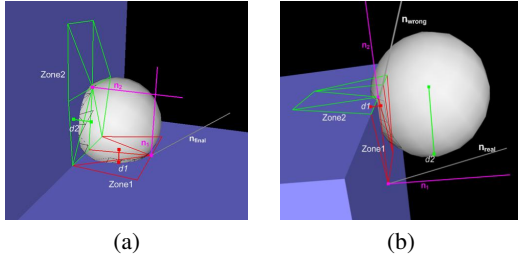


Figure 1: Convex corner (a) and concave corner (b)

2.2. Method based on temporal coherence

This method based on temporal coherence uses information from previous frames in order to compute the resultant normal vector in the current frame. In case of concave corners, the solution is the same as in the previous method.

However for convex corners, we will not allow users to collide with more than one contact zone at the same time. As a result, there will always be only one active contact zone at each time. Working in this way, the user is forced to leave the surface in contact before colliding with another one.

3. Experiments and Results

We rendered the motion of a virtual tool through a convex corner, paying particular attention to the speed rate and quality of force feedback. Haptic instabilities arise from the delay in the collision detection computation and because of abrupt changes in the haptic force value and direction.

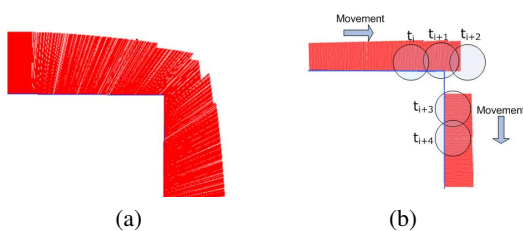


Figure 2: Force directions in a convex edge: weighted normals (a) and the temporal coherence method (b)

We can deduce whether the computed force response is valid or not analyzing the direction and module of this force. Sudden changes in the force direction can result in force discontinuities or unstable behaviour, which can produce a defective perception of the collision force.

The method based on penetration weights provides a progressive change in the force direction (Figure 2a), which avoids instabilities in the final force. In this case, the user can go around feeling a rounded corner. Red lines show the vectorial representations of the force computed by the collision response methods for each point of the surface.

The feeling using the method based on temporal coherence is similar to what happens when the user covers a real corner by a tool (Figure 2b). The user feels the same force in the t_i and t_{i+1} steps until the virtual tool stops colliding with the horizontal surface. After reaching a non-colliding state (t_{i+2}), the user could collide with the vertical surface approaching the tool to this surface (t_{i+3}) in order to go around the corner.

4. Conclusions

Two different collision response methods that deal with complex collision interactions such as concave and convex corners have been presented. Although these methods result in a very different perception of corners, both avoid abrupt changes in the haptic force direction, improving in this way overall stability of the haptic system and the perception of the geometry.

The results of experiments show that both methods manage to compute continuous forces in complex concave and convex intersection of surfaces.

The research work of H. Alvarez was supported by Gipuzkoako Foru Aldundia (FEDER).

References

- [BGAM04] BORRO D., GARCIA-ALONSO A., MATEY L.: Approximation of optimal voxel size for collision detection in maintainability simulations within massive virtual environments. *Computer Graphics Forum* 23, 01 (Mar. 2004), 13–23. 123
- [GAGB05] GARCIA-ALONSO A., GIL J., BORRO J.: Interfaces for vr applications development in design. In *Virtual Concept'05* (2005), p. 109. 123
- [SBM*95] SALISBURY K., BROCK D., MASSIE T., SWARUP N., ZILLES C.: Haptic rendering: Programming touch interaction with virtual objects. In *ACM Symposium on Interactive 3D Graphics* (1995), pp. 123–130. 123