

# A Computer-Based Real-Time Simulation of Interventional Radiology

F. Wang, L. Duratti, E. Samur, U. Spaelter and H. Bleuler

**Abstract**—Interventional radiology is a minimally invasive procedure where thin instruments, guidewires and catheters or stents are steered through the patient’s vascular system under X-ray imaging for treatment of vascular diseases. The complexity of these procedures makes training in order to master hand-eye coordination, instrument manipulation and procedure protocols for each radiologist mandatory. In this paper, we present a computer-based real-time simulation of interventional radiology procedures, which deploys a very efficient physics-based thread model to simulate the elastic behavior of guidewires and catheters. A fast collision detection scheme provides continuous collision response, which reveals more details of arterial walls than a centerline approach. Furthermore rendering techniques for realistic X-ray effect have been implemented. Our simulation structure is updated at a haptic rate of 500 Hz, thus contributing to physical realism.

## I. INTRODUCTION

Interventional Radiology (IR) is a minimally invasive surgery (MIS) procedure to improve blood flow through arteries that have been deteriorated, such as in the coronary arteries that supply blood to the heart. In IR, X-ray imaging allows physicians to visualize vessels and instruments while treating the lesions. These interventions are performed using two main categories of instruments, guidewires and catheters. Guidewires, used for catheter navigation, are very flexible cylindrical instruments that are steered to the desired point in the patient’s vascular system. Catheters, tubular instruments for the actual treatment, can afterward be inserted and driven over the guidewire directly to the intervention location.

Due to difficult hand-eye coordination, complicated looping and bending of the instruments and the risk of vessel injury, these techniques need to be performed by highly trained and experienced specialists. Some especially risky procedures, such as carotid stent placement are even expected to require specific certification and training. Computer-based training of this procedure with VR-visualization and haptic feedback offers high flexibility and scenario repeatability without risk to the patient. Furthermore it allows session recording and has therefore distinct advantages over traditional training methods on animals or cadavers.

An IR procedure starts with a palpation to find an entry point in the vascular system. Once a suitable point is found, in most cases in the groin area, a needle is used to enter the vascular system and an entry guide, the sheath, is put in place. Guidewires and catheters can be inserted through the

sheath. A guidewire is inserted and driven in the vascular system to the desired anatomical point. Due to their inherent flexibility, guidewires can more easily be steered inside the lumen. Furthermore, their flexible tips help selecting a direction by rotation at branches. Once the guidewire is in place, a catheter can be inserted and driven over the guidewire (i.e. the guidewire is inside the tubular catheter).

This paper presents a computer-based real-time simulation of interventional radiology to integrate with a haptic instrument for training. It consists of a realistic physics-based elastic thread model, a fast collision detection and techniques used in graphics rendering. The dynamic thread model follows Newton’s laws and considers main properties of the elastic thread, such as bending and twisting, as well as contact forces due to the interaction with the arterial walls. The efficient collision detection structure makes it possible to update at haptic rate and provide continuous reaction force. In addition various rendering techniques are proposed to achieve a realistic X-ray effect. The simulation environment allows principal of IR such as guidewire navigation and stent placement including realistic visualisation as well as force rendering for haptic feedback.

Organization of the paper is as follows: literature survey in section II, modeling of the catheter/guidewire dynamics in section III, the structure of collision detection in section IV, graphics and implementation with results in section V, conclusion and future plan in section VI.

## II. STATE OF THE ART

Recently a few studies have focused on simulation modeling and visualization of interventional radiology. Some of these studies [1], [2], [3], [4] have already lead to products. *NeuroCath* [1] is an interventional neuroradiology pretreatment planning system to provide integrated functionalities for both training and patient-specific pretreatment planning. In this system the catheter is simulated using a linear finite element model. The interventional cardiology training system *ICTS* [2] is a PC-based simulator that incorporates synthetic fluoroscopy, real-time three-dimensional interactive anatomic display, and coronary catheterization and angiography using actual catheters. The catheter simulation is based on a multi-body system in the *ICTS*. The Catheter Instruction System *CathI* [3] is another computer-based training system which uses original instruments providing force-feedback for guidewire and catheter handling and offers a morphologically correct coronary artery model. The *Procedicus VIST* [4] is a real-time physics based system that consists of a haptic interface device [5], a computer and two displays, one for the fluoroscopic image and one for the instructional system.

This research has been supported in part by the NCCR Co-Me (Computer-Aided and Image Guided Medical Interventions) of the Swiss National Science Foundation.

F. Wang, L. Duratti, E. Samur, U. Spaelter and H. Bleuler are with Laboratoire de Systèmes Robotiques, Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland (e-mail: fei.wang@epfl.ch)

In addition to these complete systems, some studies have focused on visualization and modeling [6], [7], [8]. Duriez et al. [6] have proposed a real-time model for deformation of devices such as catheters and guidewires during navigation inside complex vascular networks. The simulation is based on a static finite element representation and an optimization strategy based on substructure decomposition. Based on this approach, Lenoir et al. [7] have used a composite model to realistically simulate a catheter/guidewire system where simulation of both devices is performed by a physics-based composite model of wire-like structures. This model can dynamically change its material properties to locally describe a combination of both devices. Manivannan et al. [8] have proposed a method for rendering X-ray images in real-time on a PC with consumer level graphics hardware with improved quality of the images. They have implemented volume rendering techniques and studied the characteristics of actual X-ray images to develop a method that can provide a new level of realism. Although these studies have proposed different kinds of real-time models, realistic visualization and catheter modeling integrated into a complete system, taking into account all the details is still a challenging task in terms of computational cost and performance.

### III. MODEL DYNAMICS

We have developed a physics-based thread model for medical simulation. The model takes into account Newton's laws and considers the main properties of a real thread, such as stretching, compressing, bending and twisting, the effect of gravity as well as contact forces due to self-collision and interaction with the surgery tools. Based on this model realistic knot tying has been realized at haptic rendering rate [9]. The same technique has also been used by [10] for motion planning of deformable linear objects manipulation using a cooperating robot arm. We modify this thread model to simulate the elastic behavior of catheters and guidewires.

The topological structure of the thread is represented by a list of  $K+1$  nodes with 4 DOF  $i \in \{0, 1, \dots, K\}$  at positions  $\{\mathbf{x}_i = (x_i^1, x_i^2, x_i^3)\}$  and at torsion angles  $\{q_i\}$  connected by  $K$  links. A mass  $m_i$  and a polar momentum of inertia  $I_i$  are assigned to each node  $i$ . A group of (massless) linear and angular springs attached to each link produce stretching, compressing, bending and twisting between and on the nodes. The dynamics of each node  $i$  is determined by:

$$\begin{aligned} \tau_i &= I_i \cdot \ddot{q}_i(t) \\ \mathbf{F}_i &= m_i \cdot \ddot{\mathbf{x}}_i(t) \end{aligned} \quad (1)$$

Torsion force  $\tau_i$  applied along the thread is due to stretch and friction:

$$\tau_i = \tau_i^s + \tau_i^f \quad (2)$$

$\mathbf{F}_i$  is the sum of external forces applied on mass  $m_i$ :

$$\mathbf{F}_i = \mathbf{F}_i^s + \mathbf{F}_i^b + \mathbf{F}_i^t + \mathbf{F}_i^r + \mathbf{F}_i^c + m_i \mathbf{g} \quad (3)$$

where  $\mathbf{F}_i^s$  is the force resulting from stretching and compression of the spring links connected to node  $i$ . The *bending force*  $\mathbf{F}_i^b$  computed from two angular springs with minimum

bending radius at nodes  $i-1$  and  $i+1$ .  $\mathbf{F}_i^t$  is the force resulting from twisting of the link which is proportional to the stiffness constant of the angular spring. *Dissipation* enters the system in the form of friction forces,  $\mathbf{F}_i^r$ . The *contact* force with the environment is  $\mathbf{F}_i^c$ , and  $m_i \mathbf{g}$  is the gravitation force. The reaction force between two colliding links acts on the four neighboring nodes.

The thread dynamics are computed at each time step while the haptic tools are manipulated. Let  $C$  denote the finite set of all nodes controlled by the operator. For example, if one moves the thread by pushing one end,  $C$  contains only this end node. The poses of such control nodes are determined by measurement with the VR haptic devices, so only the poses of the other nodes  $\bar{C} = \{0, 1, 2, \dots, K+1\} - C$  need to be computed by the algorithm. This is performed in the following steps:

- 1) Read the position and orientation of all the control nodes  $C$  from the haptic device.
- 2) Compute the total external force and torque applied on each node. This force can be fed back to control nodes through the haptic device.
- 3) Read the time step  $dt$  between the two consecutive loops and integrate the position and torsion angle at all nodes.

The second step, corresponding to Eq.2 and Eq.3, updates collision detection and computes the force at each node from the novel pose  $(\mathbf{x}_i, q_i)(t+dt)$  at control node  $i \in C$  and old pose  $(\mathbf{x}_i, q_i)(t)$  at other nodes  $i \in \bar{C}$ . It computes the forces at each node using the forces defined in section III:

$$(\mathbf{F}_{i-1}, \tau_{i-1}) = \mathbf{f}(\mathbf{x}_{i+1}, \mathbf{x}_i, \mathbf{x}_{i-1}, \mathbf{x}_{i-2}, \mathbf{x}_{i-3}, q_{i+1}, q_i, q_{i-1}) \quad (4)$$

The third step integrates motion equation at each node  $i \in \bar{C}$  with Euler integration defined as  $\mathbf{p}$ :

$$(\mathbf{x}_i, q_i)(t+dt) = \mathbf{p}(\mathbf{F}_i, \tau_i)(t) \quad (5)$$

### IV. DETECTION OF COLLISION AND RESPONSE

Collision detection is a key component in VR medical simulation, with extensive research devoted to this area.

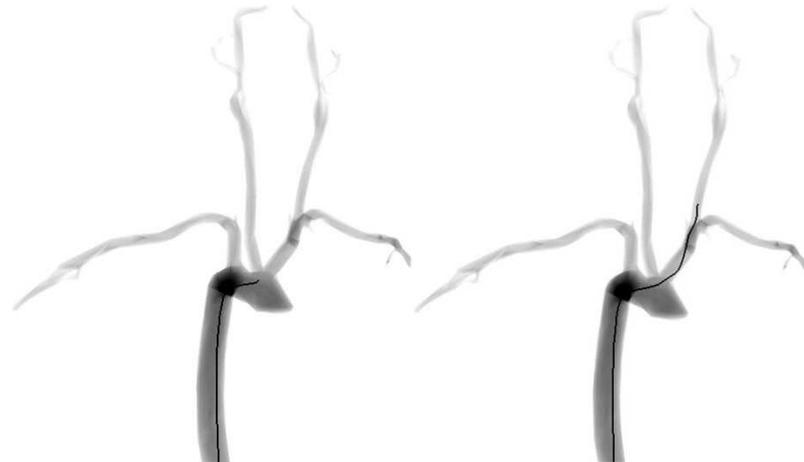


Fig. 1. A catheter sliding against the arterial walls

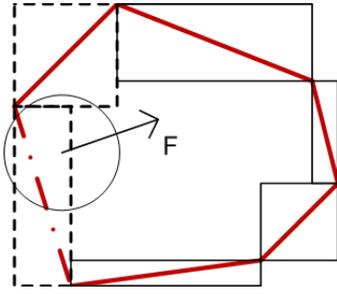


Fig. 2. AABB vs Sphere

In interventional radiology simulation, a catheter/guidewire often collides with interior arterial walls (Fig.1). Every collision must be detected at every time step, so that the reaction force (Eq. 6) can be correctly set to prevent the objects from passing through.

The bounding-volume hierarchy method (BVH) [11], [12], [13] constructs a hierarchical bounding representation by arranging the bounding volumes into a tree structure, the time complexity can be reduced logarithmically in the number of tests performed. It is best at representing geometric proximity inside an object at various levels of detail. However, geometric proximity is not invariant in a deforming object, and constant rebuilding of a BVH to maintain it can be very costly.

We unwrap the thread model volume with spheres and arterial walls with Axis-aligned Bounding boxes (AABBs) respectively. During the simulation we only update the deformation of the catheter/guidewire because the deformation of arteries is considered to be very small. As the linking sequence of our thread model is fixed, the BVH topology is pre-computed once at the beginning of the simulation and remains fixed, only the center and radius of each bounding sphere need to be updated when the model deforms.

The two resulting BVH are both binary trees. Each intermediate volume in this tree tightly bounds its two “children” and also encloses all the leaf volumes below it. To find the collision between root sphere  $S$  and root box  $B$ , we use a recursive algorithm with the following pseudo code:

```

Algorithm FindCollision ( $S$ ,  $B$ )
if ( $S \cap B = \text{no intersection}$ )
    return;
else
    if ( $S$  and  $B$  are both leaf volumes)
        Save  $S$  and  $B$  as collision pair;
        return;
    if ( $S$ 's level higher than  $B$ 's)
        Switch  $S$  and  $B$ ;
    FindCollision ( $S$ ,  $B \rightarrow \text{left\_item}$ );
    FindCollision ( $S$ ,  $B \rightarrow \text{right\_item}$ );
return;

```

At the end of each loop sub tests are done for each collision pair to check the sphere against the triangle. Fig.2

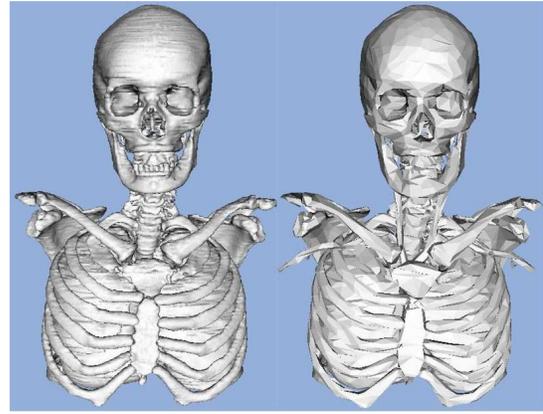


Fig. 3. Original Model vs Reduced Model

illustrates a 2D case of this primitive test. The dotted rectangles show the intersecting bounding boxes against the sphere, and the dash-dotted line shows the intersecting edge of vessel walls. The intersection depth is used to set the reaction force as follows:

$$\mathbf{F}_i^c = \kappa^c e^{\kappa^o d} \cdot \mathbf{o}_i \quad (6)$$

where  $d$  is the depth of intersection and  $\mathbf{o}_i$  is a unite vector indicating the direction of the reaction force. Hardness constant  $\kappa^c$  and  $\kappa^o$  are found by trial and error, where the contact force increases exponentially to stop the intersection.

Although slightly less efficient than the grid method, BVH is very efficient on average, especially when we only enwrap the arterial walls but not the volume it encloses.

## V. IMPLEMENTATION

The raw data set was taken from the Visible Human Project of the *Computer Vision Laboratory (BIWI), ETH Zurich, Switzerland*. We reduce the triangles count from the original 20M to 20K (Fig.3) and use VBO (Vertex Buffer

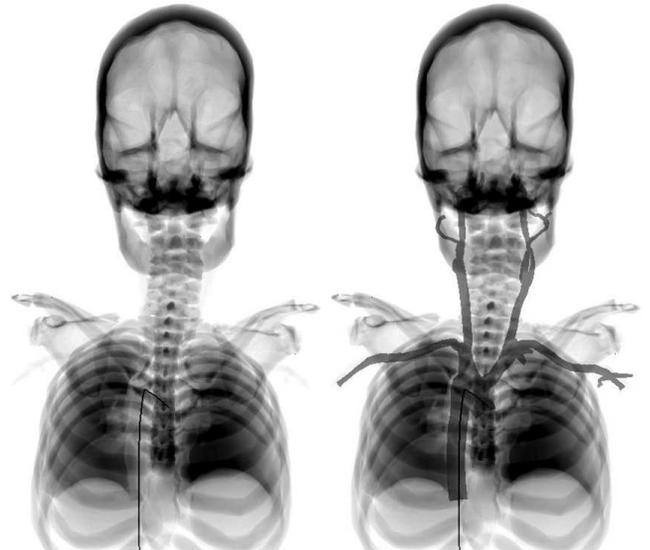


Fig. 4. Injection of radiology contrast

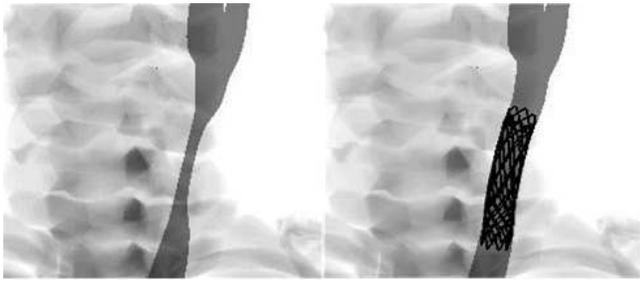


Fig. 5. Detection of blockage (left) and a stent placement(right)

Object) to speed up the rendering. RGB encoding depth [14] is applied to achieve the realistic X-ray effect.

To simulate the injection of radiology contrast (Fig. 4), we extract the centerline (position and radius) of the artery and save them into a tree like structure. When an injection happens, we first locate the corresponding position in the tree, then propagate the shadows along the tree leaves gradually at a speed of 0.35m/s.

A blockage is simulated by simply reducing the radius of the center line at corresponding position. Fig. 5 shows the procedure to detect a blockage by injecting radiology contrast and fix it by placing a stent.

The simulation is implemented on a DELL INSPIRON 9400 laptop with Intel Core 2 Duo 2.0GHz CPU, a NVIDIA GeForce Go 7900GS 256M GPU and 2G RAM. Currently the guidewire is simulated with 120 nodes and updated at 500Hz, the catheter is only graphically presented. Arterial walls are presented approximately by 1000 triangles for collision detection. Graphics is updated at 30Hz. As shown in Fig. 4-5, the simulation provides a realistic VR environment to help the surgeon in training of interventional radiology procedures.

## VI. CONCLUSIONS AND FUTURE WORKS

This paper presented a computer-based real-time simulation of interventional radiology, consisting of a realistic physics-based elastic thread model, a fast collision detection structure and techniques used in graphics rendering.

The dynamic thread model follows Newton's laws and considers the main properties of the catheter and guidewire, such as bending and twisting, as well as contact forces due to the interaction with the arterial walls. The simulator integrates internal and external forces within a single framework, overcomes limitations in realism of a non-physical simulation systems while supporting force feedback. The efficient collision detection structure makes it possible to update at haptic rate and provide appropriate reaction forces, which

reveals more details of arterial walls than simply letting the guidewire follow the centerline. Various techniques are applied to reduce the rendering complexity while taking full advantage of the GPU power.

In the future, we would like to improve the dynamic model, especially the bending property, to fit the elastic behavior even closer to the real catheter and guidewire. We will simulate the interaction between catheter and guidewire and finally integrate the simulation into the hardware prototype being developed at our laboratory [15].

## VII. ACKNOWLEDGMENTS

The authors gratefully acknowledge the work as a semester project of Christopher Burger and Xitact staff for their technical support.

## REFERENCES

- [1] W.L. Nowinski, Chee-Kong Chui, "Simulation of interventional neuro-radiology procedures", in *Proc. of International Workshop on Medical Imaging and Augmented Reality*, 2001, pp. 87-94.
- [2] S. Dawson, S. Cotin, D. Meglan, D. Shaffer, M. Ferrell, "Designing a computer-based simulator for interventional cardiology training", *Catheterization and Cardiovascular Interventions*, vol. 51, 2000, pp. 522-7.
- [3] U. Hoefler, T. Langen, J. Nziki, O. Schmid, F. Zeitler, J. Hesser, W. Voelker, R. Maenner, "CathI - Catheter Instruction System", in *Proc. of Computer Assisted Radiology and Surgery (CARS)*, Paris, France, 2002.
- [4] <http://www.mentice.com/>
- [5] <http://www.xitact.com/products/int/>
- [6] C. Duriez, S. Cotin, J. Lenoir, P. F. Neumann, "New Approaches to Catheter Navigation for Interventional Radiology Simulation", *Computer Aided Surgery*, vol. 11(6), 2006, pp. 300-308.
- [7] J. Lenoir, S. Cotin, C. Duriez and P. Neumann, "Interactive physically-based simulation of Catheter and Guidewire", *Computer & Graphics*, vol. 30, 2006, pp. 417-423.
- [8] M. Manivannan, S. Cotin, M. Srinivasan, S. Dawson, "Real-time PC based X-ray simulation for interventional radiology training", in *Proc. of Medicine meets virtual reality*, 2003, pp. 233-9.
- [9] F. Wang, E. Burdet, R. Vuillemin, and H. Bleuler, "Knot-tying with visual and force feedback for VR laparoscopic training", in *Proc. of 27th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (IEEE-EMBS)*, China, 2005.
- [10] M. Saha and P. Isto, "Motion Planning for Robotic Manipulation of Deformable Linear Objects", in *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, 2006, pp. 2478- 2484.
- [11] S. Quinlan, "Efficient Distance Computation between Non-convex Objects", in *Proc. of IEEE International Conference on Robotics and Automation*, 1994, pp. 3324-3329.
- [12] G. van den Bergen, "Efficient collision detection of complex deformable models using AABB trees", *Journal of Graphics Tools*, vol. 2, no. 4, 1997, pp. 1-13.
- [13] C. Ericson, *Real-Time Collision Detection*, Elsevier, 2005.
- [14] G. James, S. Green, "Real-Time Animated Translucency", GDC 2004 Slides.
- [15] D. Ilic, T. Moix, B. Fracheboud, I. Vecerina, and H. Bleuler, "A Haptic Interface for Interventional Radiology", in *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, Barcelona, Spain, 2005, pp. 2944-2948.