Fast and Stable Guidewire Simulator for Minimally Invasive Vascular Surgery

Zhan-Jie Gao, Xiao-Liang Xie, Gui-Bin Bian, Jian-Long Hao, Zhen-Qiu Feng and Zeng-Guang Hou

Abstract—In recent years, minimally invasive vascular surgery is widely applied in treatment of cardiovascular diseases, and the manipulation of the guidewire is the essential skill for this surgery. Lots of time and money have to be taken to achieve the skill. In this paper, we present a multithreading guidewire simulator which can help the apprentice to gain the skill and modeling the guidewire is the core technique of the simulator. The guidewire is modeled by a fast and stable method based on the Cosserat theory of elastic rods. The method describes the behavior of the guidewire with the Lagrange equations of motion and it uses the penalty method to maintain constraints. We further propose a simplified solving procedure for the guidewire model. Finally, some experiments are conducted to evaluate the effectiveness of this model.

I. INTRODUCTION

Coronary heart disease is the first cause of cardiovascular disease death [1]. Percutaneous coronary intervention (PCI) is a minimally invasive surgery which is widely used to treat coronary heart disease and the most essential skill in the surgery is the guidewire manipulation. During the PCI, a flexible guidewire should be threaded to the position of the stenosis with the help of live angiography and all of the manipulations are done out of the body. Under current circumstances, the instruments threaded in a complex 3D vessel system only can be observed on the 2D screen. Therefore, the surgery can only be done by an experienced specialist. Interactive virtual reality based simulator is qualified for training inexperienced surgeons. With the simulator, the trainees can practice the surgery anytime, and the cost of time and money for training on animals or cadavers can be saved.

The guidewire simulation is one of the core tasks in interactive virtual reality based simulator. Lots of methods are put forward to simulate the guidewire. Luboz et al. [2] presented a mass-spring model to simulate different kinds of guidewires and catheters by changing the bending coefficient of the model. Cotin et al. [3] used a multi-body system to simulate the guidewire. Since the mass-spring model is a linearly elastic model, it cannot simulate the large nonlinear geometric deformation. In [4], Alderliesten et al. presented an algorithm by extending his early work to simulate the guidewire, and a highly accurate simulation could be achieved, but the simulation was not real time.

Some nonlinear elastic models have been proposed to simulate the surgical threads. In 2002, the Cosserat theory of elastic rods was first showed to simulate strands [5]. Based on this model, Bertails et al. [6] used the Super-Helix to predict the motion of the hair. Later, Tang et al. [7, 8] and Wang et al. [9] used the Cosserat theory of elastic rods to simulate guidewire. Spillmann and Teschner [10] introduced the unit quaternions to represent the directors in the theory, so that the directors can be directly used without updating material frames in a natural way. Whereafter, Duratti et al. [11] and Mao et al. [12] integrated the unit quaternions into their models to simulate the guidewire. Our approach is inspired by the work of [10] and [13]. In this paper, a multithreading guidewire simulator is presented for minimally invasive vascular surgery and a fast and stable method based on Cosserat of elastic rods is used to model the guidewire. In the method, the Lagrangian equations of the motion are used to describe the behavior of the guidewire. We further propose a simplified solving procedure for this model. Thus, we can use the numerical method to integrate the resulting equations more easily.

The organization of this paper is as follows. An overview of the system structure is showed in section II. Section III describes the guidewire physical model. Section IV shows the vascular model used in our simulator and the collision detection algorithm is presented in section V. Section VI evaluates the experiments results and the conclusions are showed in section VII.

II. SYSTEM OVERVIEW

The system structure of the simulator is described in this section. The simulator consists of two subsystems, namely the physical simulation subsystem and the rendering subsystem. The physical simulation subsystem consists of collision detection and the updating of physical state. The scene rendering and the interactive control are implemented in the rendering subsystem whose time cost is very high in simulation. Through the multithreading technique, two threads are used to perform the two subsystems. Thus a high update rate can be achieved in both subsystems. The flow chart is presented in Figs. 1 and 2 shows the appearance of our simulator. In Fig. 2, the force feedback device is used to manipulate the guidewire, such as push, pull and rotation.
Begin
Initialize the system
Create guide wire data
Create physical simulation thread
Collission detection and response
Guide wire date update
Render date update

Finish?
YES
Finish?
NO
Render scene

End

Child thread

Main thread

YES
NO

Begin

Create guide
wire data

Create physical simulation thread

Fig. 1. Main thread is for rendering subsystem and child thread is for physical simulation subsystem.

Fig. 2. System overview of the simulator for minimally invasive vascular surgery.

III. PHYSICAL MODEL OF GUIDEWIRE

In this section, we will present the guidewire physical model based on the Cosserat theory of elastic rods. More detailed information can be found in [14]. First of all, we denote \( y_\tau(t,\tau) \) as the temporal derivative \( \frac{\partial y}{\partial \tau} \), and \( y'(\tau, t) \) is denoted as the spatial derivative \( \frac{\partial y}{\partial t} \).

A. Representation of the guidewire

The guidewire is inextensible, and its length is denoted as \( L \). Based on the Cosserat theory, the guidewire is modeled by a chain of spatial control nodes and a chain of right-hand orthonormal bases. Thus the guidewire is discretized into \( N \) spatial nodes \( G_i = G_i(x, y, z, t), i \in [1, N] \). Besides, \( N-1 \) centerline elements \( G_{i+1} - G_i \) will be achieved. \( N-1 \) directors \( (c_1(j), c_2(j), c_3(j)), j \in [1, N-1] \) which are the right-hand orthonormal bases are used to describe the centerline elements, as illustrated in Fig. 3. \( c_3 \) is a unit vector and parallel to \( G' : \frac{G'}{\|G'\|} = c_3 \), where \( G' \) can be obtained by \( G'_i \approx \frac{G_{i+1} - G_i}{l_i} \). \( l_i \) is denoted as the resting length of \( G_{i+1} - G_i \).

\( c_1 \) and \( c_2 \) are denoted as two rates of change in bending of the guidewire, and the \( c_3 \) represents the changing rate in torsion. In differential geometry, a vector \( w = (w_1, w_2, w_3) \), called Darboux vector, satisfies:

\[
\begin{align*}
\dot{c}_k &= w \times c_k & w_k &= w \cdot c_k, k \in \{1, 2, 3\} \quad (1) \\
w_1 \text{ and } w_2 \text{ denote bending strains in two directions, and } w_3 \text{ is the torsional strain. The temporal derivatives of the directors can be described as } \dot{c}_k &= \omega \times c_k \text{ by angular velocity } \omega \text{ and directors.}
\end{align*}
\]

For the purpose of directly using the directors, the unit quaternions \( Q_j = (q_1(j), q_2(j), q_3(j), q_4(j))^T \) are used to denote the directors, instead of the Euler angles which have singularity problem. The strain rates \( w_k \) in the reference frame and the angular velocity components \( \omega_k \) in the reference frame can be described as:

\[
\begin{align*}
w_k &= 2Q' \cdot (A_k Q), & \omega_k &= 2Q \cdot (A_k^0 Q) \quad (2)
\end{align*}
\]

with the constraint \( \|Q\| = 1 \), where \( A_k \) and \( A_k^0 \) are both 4 by 4 skew-symmetric matrices. \( Q' \) can be represented as \( Q' \approx Q_{i+1} - Q_i \), where \( i_j = \frac{\|l_j + \|l_j\|}{2} \) is the rest length of the orientation.

B. Formulas of Energy

In this work, the energy consists of three kinds of energy, namely the elastic potential energy \( EP_e \), dissipation energy \( D_e \) and the constraint energy \( C_e \). \( EP_e \) is composed of stretch \( EP_e^s \) and bending energies \( EP_e^b \)

\[
\begin{align*}
EP_e^s &= \frac{1}{2} \int_0^L (K_s(\|G'\| - 1)^2) d\tau \quad (3) \\
EP_e^b &= \frac{1}{2} \int_0^L (\sum_{k=1}^3 K_{kk}(w_k - \dot{w_k})) d\tau \quad (4)
\end{align*}
\]

where \( K_s \) is defined as the stretching stiffness constant. \( K_{kk} \) can be described as \( K_{11} = K_{22} = E \frac{\pi r^2}{4} \) and \( K_{33} = G \frac{\pi r^2}{4} \), where \( E \) is the Young's modulus, while \( G \) is the shear modulus. \( \dot{w}_k \) is the intrinsic bending and torsion, and they can be used to change the shape of the guidewire. \( r \) is the radius of the guidewire. \( D_e \) consists of internal friction \( D_e^f \) and visco-elastic effects \( D_e^v \).

\[
\begin{align*}
D_e^f &= \frac{1}{2} \int_0^L (\gamma_v \nu_{rel} \cdot \nu_{rel}) d\tau \\
D_e^v &= \frac{1}{2} \int_0^L (\gamma_o \omega_0' \cdot \omega_0') d\tau
\end{align*}
\]

where \( \gamma_v \) denotes the internal friction coefficient, while \( \gamma_o \) denotes the visco-elastic coefficient. \( \nu_{rel} \) is the relative velocity in the local frame, and \( \omega_0 \) is the relative angular velocity in the reference frame. The constraint consists of two parts: \( \frac{G'}{\|G'\|} = c_3 \) and \( \|Q\| = 1 \).

\[
C_e = \frac{1}{2} \int_0^L \kappa (G' - c_3) \cdot (G' - c_3) d\tau \quad (7)
\]

5810
is used for the first part of the constraint where $\kappa$ is the spring const. For the second part, $Q = \frac{Q}{\|Q\|}$ is used to keep it in every iteration.

C. guidewire Dynamics

Under normal circumstances, the Lagrangian equation of motion

$$\frac{d}{dt} \frac{\partial K_e}{\partial \dot{\eta}} - \frac{\partial K_e}{\partial \eta} + \frac{\partial (EP_e + C_e)}{\partial \eta} + \frac{\partial D_e}{\partial \dot{\eta}} = \Gamma_e$$

(8)

can be obtained by the kinetic energy $K_e$, elastic potential energy $EP_e$, dissipation energy $D_e$ and the constraint energy $C_e$. $\Gamma_e$ are the external forces and torques, such as gravity, external forces and torques exerted by trainees, and friction. With this equation of motion achieved, the sum $\Gamma_{all}$ of inner $\Gamma_{in}$ and external $\Gamma_e$ forces and torques can be obtained. Here, the $\eta$ is the global coordinates which are $N$ spatial control points and $N - 1$ directors. Thus, $7N$ global variables should be calculated. Because $N$ is big in simulating guidewire, achieving the equation of motion will consume a large amount of computation.

In our simulator, an easy method is introduced to reduce the calculated amount. With the resulting equation of the equation (8) , the system’s form can be described as:

$$M \ddot{a} = \Gamma_{in} + \Gamma_e = \Gamma_{all}$$

(9)

where $M$ is the mass matrix, and $a = \dot{\eta}$. $\Gamma_e$ can be directly achieved from the collision detection and human interactions. From the [10], $\Gamma_{in}$ can be directly obtained by

$$\Gamma_{in} = -\frac{\partial (EP_e + C_e)}{\partial \eta} - \frac{\partial D_e}{\partial \dot{\eta}}$$

(10)

Here, $7N$ global variables are used to store the inner forces and torques and they are set to zero before each iteration. Since the derivative and integral are linear operators, the calculation of $\Gamma_{in}$ is divided into five iterations namely $EP^b_e$, $EP^a_e$, $D^b_e$, $D^c_e$ and $C_e$ integration. For the $EP^b_e$ iteration, only inner torques $T^b_{in}$ produced by $EP^b_e$ should be taken into account and the formulation achieved by equation (10), is described as:

$$T^b_{in} = -\frac{\partial (EP^b_e)}{\partial \eta}$$

(11)

where $EP^b_e$ is the total bending energy and it can be written as:

$$EP^b_e = \sum_{k=1}^{N-2} e^b_p(k)$$

(12)

where $e^b_p(k)$ is the bending energy between the centerline element $k$ and $k + 1$. Here, one part of the bending energy is chosen to get the part inner torques formulation $t^b_{in}(k)$ using equation (11):

$$t^b_{in}(k) = -\frac{\partial (e^b_p(k))}{\partial \eta}$$

(13)

Here, only eight global variable are taken into account. Thus the equation (13) is achieved easily. Fortunately, every part bending energy has the same formula structure. Thus, $T^b_{in}$ can be achieved by substituting corresponding parameters into equation (13) and corresponding $t^b_{in}(k)$ should be added to the corresponding global variables of inner torques defined before. The same method is used to obtain other inner forces and torques produced by other energy formulations.

Using this method, the biggest number of the global variables taken into account is 10 and if $N$ is more than 3, it is much more easily to achieve $\Gamma_{all}$ than the original method. The part inner forces and torques formulations can be achieved in advance. Thus, $\Gamma_{in}$ can be achieved faster and the computing time is showed in Section VI when $N$ is 100.

With $\Gamma_{all}$ achieved by equation (9), the semi-implicit Euler scheme is used to integrate the system.

IV. Vascular Model

The vascular model in this simulator is complex and realistic, and it is obtained from 128 computer tomography angiography (CAT) series in DICOM datasets captured from real patients. The ITK library is used to segment the original data [15]. Then, the vascular model can be achieved by using the physical modeling function, as is showed in Fig. 4.

V. Collision Detection

The collision detection is a crucial step and it keeps the guidewire from going out of the vascular model. In order to guarantee a real-time performance of the simulator, a real-time collision detection should be contained. In this system, a real-time collision detection algorithm [16] which is based on axis-aligned bounding boxes (AABB) is introduced. The collision detection consists of broad phase and narrow phase collision detection. In the process of detection, two AABB trees are built respectively for the guidewire and vascular. In general, the broad phase collision detection is conducted. The narrow phase collision detection is executed only if the broad phase collision happens. Therefore, a quick detection can be achieved at the beginning. The deeper the guidewire inserts, the longer time the collision detection takes.

VI. Experiments and Results

Some simulations are used to evaluate the stability of the guidewire. All the simulations are implemented on a PC, which is equipped with a 3.1GHz Intel Core i5-2400 CPU...
The guidewire is inserted into the aortic arch to show the behavior in the aortic arch. In this simulation, 100 nodes are used to simulate the guidewire, while the vasculature is presented by 4,666 triangles. As the rendering and physical separate, the rendering update rate (RUR) is 59Hz, and the slowest physical simulation update rate (PUR) is 200Hz when the guide is inserted into the bottom of the vasculature. The detailed system performance corresponding to the Fig. 5 is showed in Table I. From Table I, the real-time performance of the simulator can be guaranteed.

In Fig. 6, the guidewire which has 20 nodes is inserted into a subbranch artery to show the behavior in tiny blood vessels. From Fig. 6, the guidewire model can also keep a stable performance.

### VII. Conclusions

We present a guidewire simulator for minimally invasive vascular surgery. In order to guarantee the real-time performance, a multithreading technique is applied and a high update rate is achieved in both subsystem. In the simulator, a fast and stable method based on Cosserat theory of elastic rods is used to simulate the guidewire. The method describes the motion of the guidewire by Lagrange equations of motion. With the simplified solving procedure, we can use the semi-implicit Euler scheme to integrate the resulting equations more easily. In section VI, experimental results demonstrate the stability and rapidness of our guidewire model in different situations. The real guidewire structure and the feedback forces will be included in our simulator in future works. Thus the apprentice can have a strong sense of reality.

### REFERENCES


