Computer Modeling in Engineering & Sciences

Tech Science Press

DOI: 10.32604/cmes.2021.012034

ARTICLE

A Three-Stage Cutting Simulation System Based on Mass-Spring Model

Xiaorui Zhang[1,2,*], Jiali Duan[1], Wei Sun[2], Tong Xu[1] and Sunil Kumar Jha[3]

[1]Jiangsu Engineering Center of Network Monitoring, Engineering Research Center of Digital Forensics, Ministry of Education, School of Computer and Software, Nanjing University of Information Science & Technology, Nanjing, 210044, China

[2]Jiangsu Collaborative Innovation Center of Atmospheric Environment and Equipment Technology (CICAEET), Nanjing University of Information Science & Technology, Nanjing, 210044, China

[3]Faculty of Information Technology, University of Information Technology and Management, Rzeszow, 35-225, Poland

[*]Corresponding Author: Xiaorui Zhang. Email: zxr365@126.com

Received: 11 June 2020 Accepted: 10 December 2020

ABSTRACT

The cutting simulation of soft tissue is important in virtual surgery. It includes three major challenges in computation: Soft tissue simulation, collision detection, and handling, as well as soft tissue models. In order to address the earlier challenges, we propose a virtual cutting system based on the mass-spring model. In this system, MSM is utilized to simulate the soft tissue model. Residual stress is introduced to the model for simulating the shrinking effect of soft tissue in cutting. Second, a cylinder-based collision detection method is used to supervise the collision between surgical tools and soft tissue. Third, we simulate the cutting operation with a three-stage cutting method with swept volume, Bézier curve, and an algorithm named shortest distance nodes matching method. In order to verify the system performance, we carry out three validation experiments on the proposed system: Cutting accuracy experiment, collision detection validation, and practical cutting evaluation. Experiments indicate that our system can well perform the shrinking effect of soft tissue in cutting. The system has fast and accurate collision detection. Moreover, the system can reconstruct smooth incisions vividly.

KEYWORDS

Mass-spring model; collision algorithm; virtual surgery; soft tissue simulation

1 Introduction

Preventable adverse events in surgery, such as unskilled or incorrect operation, continue to occur at a rate [1] between 3% and 17%. Therefore, it is crucial to promote surgery success. The success of surgery greatly depends on the skill of surgeons. Traditionally, in surgery training, we usually perform surgeons through books, films, and animal carcasses. With the development of computer science, the virtual reality (VR) technology would be a preferable choice for surgery training, which can simulate surgery in a virtual environment and display it on a computer screen. As a new multi-disciplinary research area, virtual surgery [2] combines the VR technology with

computer graphics technology, which promotes the development of medical training by using the less real tissues and reducing potential operational risk.

Cutting is one of the most common operations in surgery. Virtual surgery training can improve the proficiency of surgeons in cutting skills. A VR based cutting surgery system enables surgeons to manually cut the virtual tissues, whose core idea is establishing the tissue cutting model with physical models based on medical images, like computed tomography (CT) scans [3] and magnetic resonance imaging (MRI) [4]. The virtual cutting system has three important components [5]: The simulation of the soft tissue, the collision detection between surgery tool and soft tissue, and soft tissue models.

Soft tissue simulation is the basis of virtual cutting surgery. Among the current wide used soft tissue model, the physically-based meshless model [6] is weak in tissue properties description. Once the soft tissue undergoes a topological change in cutting, it will involve many complex biomechanical phenomena, such as the shrinking effect [7] of soft tissue around the wound. Therefore, the existing meshless cutting model still needs to be improved in this respect. Pan et al. [8] proposed an interactive dissection approach based on metaballs for virtual organs in virtual surgery. The model consists of inner metaballs and surface mesh. During cutting, the MLS-driven meshless method can facilitate adaptive topology modification and cutting surface reconstruction. However, its simulation of soft tissue properties during cutting is relatively poor, and the soft tissue shrinking effect is not included. Jin et al. [9] presented a meshless total Lagrangian adaptive dynamic relaxation algorithm to predict the steady-state deformation of soft tissue caused by surgical cutting. The algorithm relies on spatial discretization in the form of a cloud of nodes. However, the method is only a 2D soft tissue cutting method, which cannot well describe the physical properties of 3D soft tissue. Zhang et al. [10] presented a survey of the state-of-the-art deformable models studied in the literature concerning soft tissue deformable modeling for interactive surgical simulation. It introduces the challenges of surgical simulation, followed by discussions and analyses on the deformable models, which are classified into three categories: The heuristic modeling methodology, continuum-mechanical methodology, and other methodologies. And also, there are some other different kinds of soft tissue simulation model, such as FEM. The finite element model (FEM) is composed of continuous volume elements, which has high precision but consumes a lot of computing resources. FEM is suitable for simulating organs with volume characteristics, especially for simulating the internal structure of organs. However, in surgical simulation, its interactive performance is relatively poor due to a large amount of computational cost. Yeung et al. [11] simulated cutting and deformation on a linear elastic model based on FEM, using an augmented matrix to provide rapid updating. However, this study mainly studies linear materials, and nonlinear material updating still needs to be accelerated. Townsend et al. [12] used the updated Lagrangian finite element formula to simulate the nonlinear material model. This method can be applied to simulate various biological soft tissue materials, providing a theoretical model, but its practicability has not been verified. Zhang et al. [13] proposed a new chainmail algorithm for real-time modeling of soft tissue deformation under force interaction. Different from the traditional chain mail box boundary region, according to the concept of principal strain in continuum mechanics, an ellipsoid boundary region is defined to control the movement of chain elements. On the basis of the ellipsoid boundary region, a new position adjustment rule is proposed, which is further combined with the dynamics of the temporary domain model for the dynamic simulation of soft tissue deformation. The tactile interaction with soft tissue is realized by force input, soft tissue deformation and force feedback. What's more, Frisken-Gibson [14] presented a set of algorithms that allow interactive manipulation of linked volumes that have

more than an order of magnitude more elements and considerably more flexibility than existing methods. Implementation details, results from timing tests, and measurements of material behavior are presented.

Accurate collision detection [15,16] is the key to a virtual cutting system. Extended finite element methods [17–19] can well handle topological changes in the model, which allow the model to simulate incision independently regardless of the background model mesh. However, the simulation of collisions and interactions remains unsolved. Paulus et al. [20] combined a remeshing algorithm with a FEM mesh to simulate cutting surgery, tissue avulsion, and microfracture. However, the algorithm did not take collision detection into consideration. Qian et al. [21] proposed an energized soft tissue cutting model, which is adapted to the interaction between the energized cutting tool and the soft tissue in virtual surgery. The model divides soft tissue into three categories and simulates their physical properties respectively. During cutting, the topology of the mesh does not change and the topology is controlled by a heat transfer model. However, the model does not involve the collision detection problem.

The main task of a virtual cutting system is to accomplish the incorporation of cutting operations and soft tissue models. The effect and smoothness of the incision are important measuring indicators of the degree of the integration. Kibsgaard et al. [22] presented an implementing academic method for cutting in deformable objects simulated by mass-spring model (MSM) combined with a volumetric tetrahedral mesh. The cutting algorithm allows arbitrary cuts in the deformable objects. However, its cutting model is a plastic-like cube instead of soft tissue, and the incision rendering effects are unsmooth since there are lots of triangular meshes. Zhang et al. [23] proposed a cutting model based on an improved MSM. A virtual spring is introduced to compensate for the weakness of the conventional MSM, and a collision detection algorithm is adopted to determine the springs that intersect with the cutting face. However, the cutting simulation is based on a mesh plane instead of soft tissue. Meanwhile, there are sharp protrusions in the mesh plane, and the incision is relatively rough. Courtecuisse et al. [24] addressed the problem of cutting the heterogeneous structures and propose a method to update the preconditioner according to the topological modifications. Quesada et al. [25] studied an example of corneal surgery. The main objective of this numerical example is to reproduce the cut on the cornea with reasonable accuracy, compatible with the physical sensation obtained by a surgeon. Thus, they used a finite element model of the cornea.

To address the aforementioned limitations, we propose a three-stage cutting simulation system based on MSM. Firstly, we construct the soft model with MSM and introduce the residual stress to the model to simulate the shrinking effect of soft tissue in cutting operation. Then a cylinder-based collision detection method is utilized to monitor the collision between the cylindrical surgical tool and the soft tissue. The cutting process occurs after a collision is detected. The cutting process is divided into three stages. In the first stage, masses and springs in the range of the area that swept by a generated swept volume are deleted. In the second stage, the wound is drawn through the Bézier curve [26]. In the third stage, an algorithm based on shortest distance mass marching is designed to reconstruct the soft tissue mesh around the wound. The proposed system can realize relatively accurate soft tissue deformation compared with other same type of MSM models under the premise of precise collision detection. Also, the system and can well simulates the shrinking effect of soft tissue in cutting operation, and present smooth and realistic incision. The main contributions of this work are summarized as follows:

1) In order to simulate the shrinking effect of soft tissue undercutting operation, we propose an MSM-based soft tissue model, which is improved by residual stress. The residual stress is added to connect nodes after the spring between nodes are deleted, which is the key to simulate a shrinking effect.
2) We designed a cylinder-based collision detection algorithm for cylindrical surgical tools, whose core idea is monitoring the collision by calculating certain angles between tools and tissue surface. It is necessary because the interaction area of cylindrical surgical tools is usually not a point, but a line or a surface.
3) In order to obtain a smooth and vivid incision, we propose a three-stage cutting algorithm. In the first stage, the swept volume is used to quickly build the reconstruction area of the incision. In the second stage, the Bézier curve is used to draw the incision. In the third stage, the shortest distance mass matching method is designed to reconstruct the mesh around the incision.

The remainder of this paper is organized as follows. Section 2 illustrates the incision system. Section 3 introduces the system environment for the cutting operation and preferences of soft tissue. Section 4 validates the performance of the proposed system. In the last section, a summary of our work is concluded and future works are discussed.

2 System Design

The three-stage cutting simulation system consists of three major components: The construction of the MSM-based soft tissue model with residual stress, a cylinder-based collision detection algorithm, and the three-stage cutting simulation. The following sections describe in detail the major components of the cutting simulation system.

2.1 Model Construction

In real surgery, soft tissue has a shrinking effect at the moment when it is cut open. In order to simulate such a phenomenon, we introduce residual stress to MSM to construct the soft tissue model. According to the medical image, the soft tissue model with triangular topological structure is constructed based on MSM.

First, we establish a soft tissue model with MSM. The status of mass nodes N_i around the incision can be described by Eq. (1).

$$\begin{cases} dx_i(t)/dt = v_i(t) \\ dv_i(t)/dt = F_i/m \end{cases} \tag{1}$$

where x_i is the coordinates of N_i, v_i is the velocity of N_i, F_i is the sum of force acting on N_i, m is the mass of N_i. F_i can be computed by Eq. (2).

$$\begin{cases} F_i = \sum f_{spring}(N_i, N_j) + \sum f_{damping}(N_i) + G(N_i) \\ f_{spring}(N_i, N_j) = K(L_0 - \|N_iN_j\|)\dfrac{N_iN_j}{\|N_iN_j\|} \\ f_{damping}(N_i) = -Dv_i(t) \end{cases} \tag{2}$$

where f_{spring} is the spring force between node N_i and its surrounding nodes N_j, L_0 is the distance between N_i and N_j, K is the elastic coefficient of spring, $f_{damping}$ is the damping force, D is the damping coefficient, and G is gravity.

Then we introduce the residual stress to MSM to simulate a shrinking effect, and the residual stress σ_r is calculated by Eq. (3).

$$\sigma_r = E\varepsilon_r \tag{3}$$

where E is Young's modulus, ε_r is a residual strain, which is assumed to be 0.1 and uniformly distributed in the soft tissue. The residual stress will be added to connect mass nodes instead of springs when the springs are deleted by the swept volume which is used during the cutting simulation. After connecting mass nodes with residual stress, F_i can be rewritten by Eq. (4).

$$F_i = \sum f_{spring}(N_i, N_j) + \sum f_{damping}(N_i) + G(N_i) + \sigma_r \tag{4}$$

2.2 Collision Detection

There are many surgical tools with long and narrow shape in surgery, such as scalpels. The interactive area of such tools is often a surface or a line. In virtual surgery simulation, the collision detection of such tools is different from normal point-based collision detection. Therefore, in this work, we simulate surgical tools with long and narrow shape as a cylinder and design a cylinder-based collision detection algorithm. The schematic diagram of the collision detection method is shown in Fig. 1.

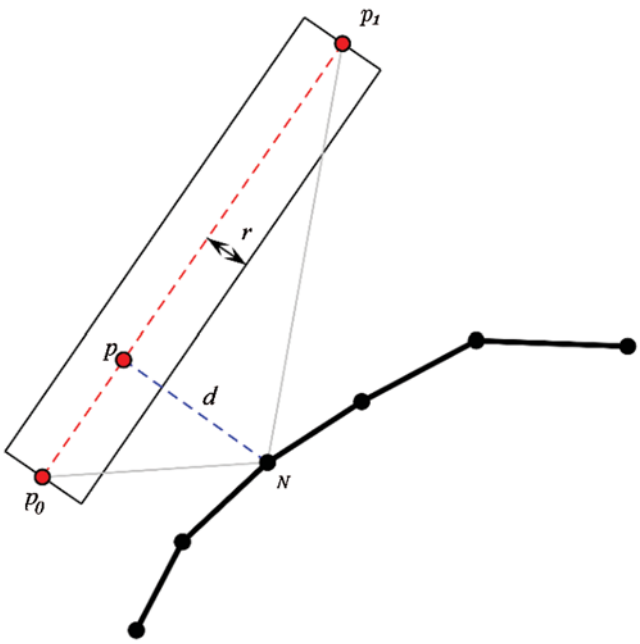


Figure 1: Schematic diagram of the collision detection method

As is shown in Fig. 1, the red dotted line is the axis of a surgical tool, assuming the ends of the axis is p_0 and p_1, N is one of the mass nodes on the soft tissue surface, p is on the axis

p_0p_1 and is the closest point to node N, d is the Euclidean distance between N and p_0p_1, r is the radius of the cylinder, which can be adjusted according to the shape of a surgical tool.

The premise of this collision detection method is that the original position of the surgical tool is on the outside of the simulated soft tissue model before the graphics rendering begins. Then in each frame of the graphics rendering process, every mass node of the soft tissue model is tracked by the system. The system will judge if the collision occurs according to the following steps.

Step 1. Calculating angle $\angle p_0p_1N$ and $\angle p_1p_0N$. If $\angle p_0p_1N \leq 90^\circ$ and $\angle p_1p_0N \leq 90^\circ$, go to Step 2, otherwise the node N is still outside the cylinder.

Step 2. Calculating the Euclidean distance d between N and p_0p_1. If $d < r$, node N is inside the cylinder, which indicates that collision occurs between soft tissue surface and surgical tools. Force feedback will be transferred from node N in the direction of $\vec{Np}$.

2.3 Three-Stage Cutting Simulation

In order to obtain smooth and vivid incision, the cutting process is divided into three stages. The goal of the first stage is to create an incision generation region. The goal of the second stage is to draw the incision. The goal of the third stage is to connect the incision with the surrounding mass nodes.

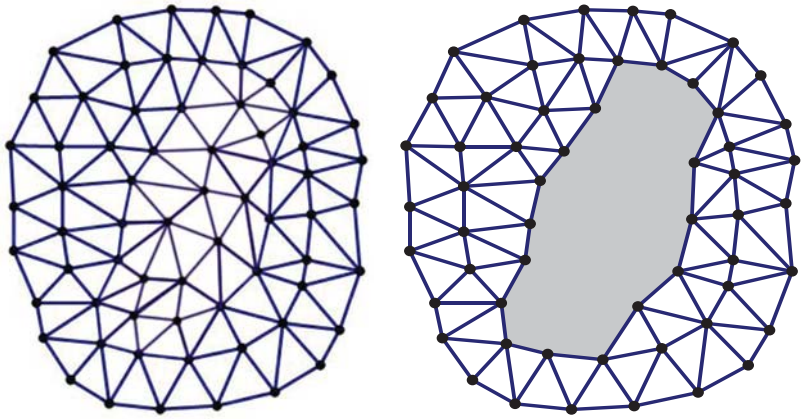
2.3.1 The First Stage

In the first stage, we use swept volume to remove the springs inside the interactive area. The first stage cutting algorithm is shown in Algorithm 1.

Algorithm 1: The first stage cutting algorithm

Procedure Surface Cutting
Generate the swept volume V_{sweep}
for all nodes and springs inside swept volume V_{sweep} do
if V_{sweep} touches nodes
Delete the nodes and springs connected with the nodes
end if
if V_{sweep} touches springs
Delete the springs
end if
end for
Mark the area generated by surface cutting procedure grey

As is shown in Algorithm 1, a swept volume V_{sweep} is generated to delete springs in the cutting region. V_{sweep} is a circular region that can be dynamically adjusted according to the shape of surgical tool. In this work, the cutting operation is accomplished by cutting springs. During the cutting procedure, delete all springs and nodes that are inside V_{sweep}. V_{sweep} moves with the surgical tool and represents the interactive area of tool and tissue surface. The soft tissue surface mesh after Stage 1 is shown in Fig. 2, where the gray area is the incision generation area in Stage 2. Fig. 2 shows the incision generation area.



(a) (b)

Figure 2: Incision generation area. (a) Before (b) After

2.3.2 The Second Stage

In the second stage, the Bézier curve is utilized to draw the incision in the incision generation area. The Bézier curve contains two endpoints and several control points that affect the shape of the curve. The degree of polynomials depends on the number and position of control points. The Bézier curve always passes through the first and last control points, so it can accurately simulate the whole incision, especially the endpoints of the incision. The endpoints of the Bézier curve changes in real-time as the control points increases, which is manifested as the process of continuous extension of the incision.

Assuming there are n+1 control points, whose position $P_k = (x_k, y_k, z_k)$, where $k = 0, 1, \ldots, n$. Position vector $P(a)$ is generated according to these control points, which is used to describe the route between P_0 and P_n. $P(a)$ can be calculated with Eq. (5).

$$P(a) = \sum_{k=0}^{n} P_k B_{k,n}(a) \tag{5}$$

where $0 \le a \le 1$, and $B_{k,n}(a)$ is Bernstein polynomial [27], which can be calculated with Eq. (6).

$$B_{k,n}(a) = \frac{n!}{k!\,(n-k)!} a^k (1-a)^{n-k} \tag{6}$$

In this work, the cubic Bézier curve is used to simulate the incision, which is generated by four control points, and $n = 3$ is substituted into Eq. (6). Four functions of the Bézier curve are obtained, as in Eq. (7).

$$\begin{cases} B_{0,3}(a) = (1-a)^3 \\ B_{1,3}(a) = 3a(1-a)^2 \\ B_{2,3}(a) = 3a^2(1-a) \\ B_{3,3}(a) = a^3 \end{cases}, \quad 0 \le a \le 1 \tag{7}$$

The form of the above functions determines how the control point affects the shape of the curve. Substituting Eq. (7) into Eq. (5) shows the following properties of the curve. For $a = 0$, the curve passes through the initial control point P_0. For $a = 1$, the curve passes through the termination control point P_3. For $a = \frac{1}{3}$, the maximum value of P_1 is obtained, and when $a = \frac{2}{3}$, the maximum value of P_2 is obtained. The cubic Bernstein polynomial can be expressed in the form of a matrix in Eq. (8).

$$P(a) = \begin{bmatrix} a^3 & a^2 & a & a^0 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{bmatrix} \tag{8}$$

2.3.3 The Third Stage

In the third stage, we develop an algorithm named the shortest distance nodes matching method, which is used to connect the Bézier curve with surrounding mass nodes in the shortest distance. The schematic diagram of the shortest distance nodes matching method is shown in Fig. 3.

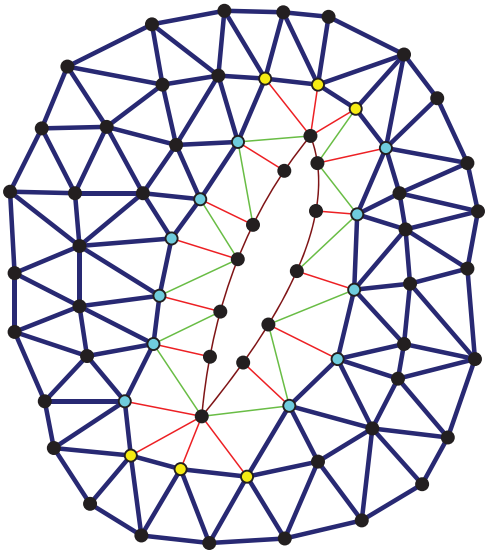


Figure 3: The schematic diagram of the shortest distance nodes matching method

The shortest distance nodes matching method has three steps: (1) Find the closest points on the Bézier curve for all surrounding mass nodes, and then connect them with red springs, as shown in Fig. 3; (2) Classify the mass nodes with the following principle: If the mass node is connected to the endpoints of the Bézier curve, the node is marked yellow and other mass nodes are marked blue; (3) Search quadrilaterals in the mesh where the blue nodes are located. If the diagonal of the quadrilateral is not connected, select the shorter diagonal and connect it with a green spring, thereby dividing the quadrilateral into triangular meshes, as shown in Fig. 3. To better illustrate the procedure, the third stage cutting algorithm is summarized in Algorithm 2.

Algorithm 2: The third stage cutting algorithm

Procedure Springs Reconstruction
for all nodes around the grey area
Connect nodes with Bézier curve with shortest distance
if the node is connected to the end of the Bézier curve
The node is marked yellow
else
The node is marked blue
end if
end for
for all quadrangles have blue nodes
if the diagonal of quadrangle is not connected
connect the shortest diagonal of quadrangle
end if
end for

3 System Realizations

After designing the soft tissue model, collision detection, and cutting algorithm in the virtual cutting system, the function of cutting simulation can be realized. Our system consists primarily of a mainframe computer and a haptic interaction facility called PHANTOM OMNI. The computer is based on Windows 10 and comes with an Intel(R) Xeon(R) CPU, E5-1650 v3@3.5 GHz processor and NVIDIA GeForce GT 720 M graphics. The simulation is carried out on VC++ 2019 and 3DS MAX 2019 with OpenGL graphics libraries. The PHANTOM OMNI is a force feedback device that allows the operators to touch and operate on the virtual object simulated by our method. The experimental environment is shown in Fig. 4.

Figure 4: Experiment environment

The virtual skin is constructed with our soft tissue in 3DS MAX 2019 according to the shape information of the real tissue on belly obtained from medical images. Parameters of the skin model are set as Tab. 1.

Table 1: Parameters of the skin model

K (N/mm)	D (Ns/m)	E (KPa)	ε_r
0.16	3	20	0.1

As is shown in Tab. 1, K is the spring elasticity coefficient, D is the damping coefficient, E is Young's modulus, and ε_r is the residual strain. After model construction, the model data will be exported as a file in obj format, which will be imported to VC++2019. Then we program the collision detection and cutting algorithm in VC++2019 before the cutting simulation, and carry out the cutting simulation on virtual skin. For the virtual scalpel contact with virtual skin, the system will monitor the collision in real-time, and simulate the cutting according to the cutting algorithm immediately. At the same time, the haptic information will be given to the operators by PHANTOM OMNI. Fig. 5 shows the cutting effect of our system.

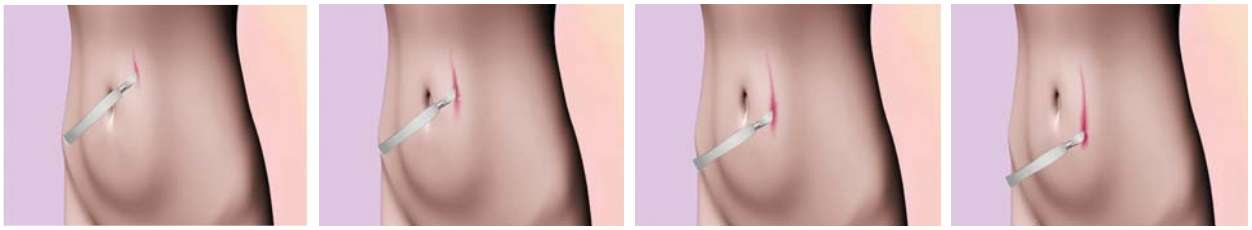


Figure 5: Cutting effects

Fig. 5 shows the cutting operation on the skin. In general, the incision effects are realistic and the rendering effect is vivid. To be specific, firstly, the deformation of soft tissue around the incision is relatively larger at the beginning of cutting, and then the incision is shrinking, which can be regarded as the shrinking effect of soft tissue at the beginning of cutting. Secondly, the incision is expanded in real-time with the trajectory of the scalpel. Finally, the incision is smooth and there are no pathological small triangles in the soft tissue mesh around the incision.

4 System Performance

To verify the simulation accuracy of cutting, the performance of collision detection, and the practical performance of the proposed system, we carried out 3 validation experiments on our system: Model accuracy validation, collision detection validation, and practical cutting experiment.

4.1 Cutting Accuracy

In order to show the cutting accuracy of our soft tissue model, we compare the cutting of soft tissue simulated by our model with cutting simulated by other models and cutting on real soft tissue. The cutting accuracy is evaluated by the distance between the selected sample points on both sides of the incision. Over time, distance between same sample points is measured in cutting on real soft tissue and on virtual soft tissue. The virtual curve is according to the real one that represents the high accuracy of cutting. Samples of the real tissue are picked from the department of Cardiothoracic surgery of the First Affiliated Hospital of Nanjing Medical University in Nanjing. Compare the time-distance curve of real tissue and virtual tissue simulated with our model and another model. Fig. 6 shows the time-distance curves.

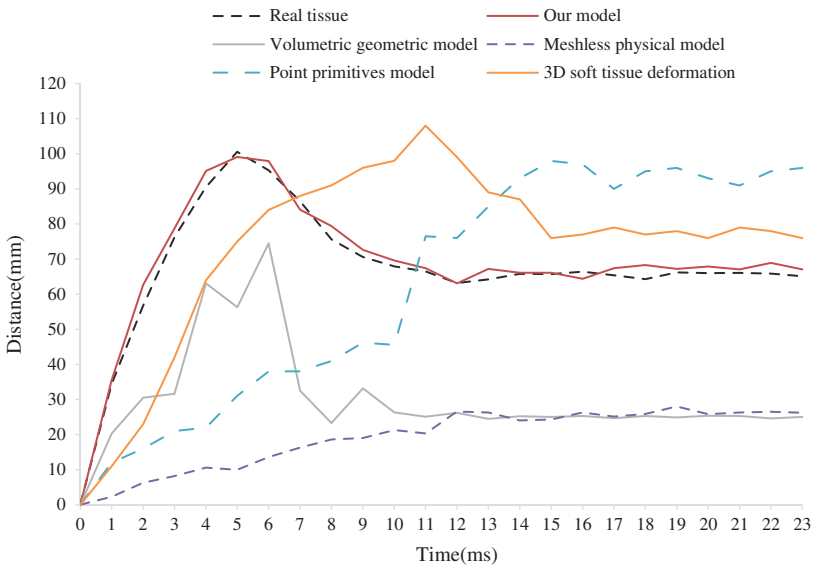


Figure 6: Time-distance curves of incision simulated by different models

According to Fig. 6, over time, in the real curve, because of the soft tissue shrinking effect, the distance at first rapidly increases, and then the distance is gradually reduced. Finally, the distance tends to be stable. The time-distance curve of tissue simulated by our model is according to the real curve. Therefore, the proposed cutting system can better present the shrinking effect of soft tissue in cutting. Overall, the incision simulated with our model has an accurate cutting presentation, and the proposed cutting system can well simulate cutting operation.

4.2 Collision Detection Validation

In order to validate the precision and real-time performance of the proposed collision detection method, multiple cutting angles are utilized. Compare the performance of the collision detection algorithm of our system and other systems under different cutting angles. The cutting angles used in this experiment are 30, 40, 50, 60, and 70.

Firstly, the accuracy of the interactive area is compared. For the convenience of comparison, only the XY coordinates of the collision point are considered. Since the interactive area of real cutting and our system is not a point, but a line, we record the center of the line as the collision point for convenience. The top view of the cutting scene is studied. XY coordinates based on the top view has been built, and the collision point of the real cutting is taken as the origin of coordinates, as is shown in Fig. 7. Record the coordinates of the collision point of our cutting system and cutting system in [28] under different cutting angles in Fig. 7.

As is shown in Fig. 7, no matter what the cutting angle is, the collision point of our system is basically concentrated near the origin of coordinates. Some collision points in [32] deviate from the origin of coordinates in some cases, therefore, out cutting system has more accuracy collision detection than other systems.

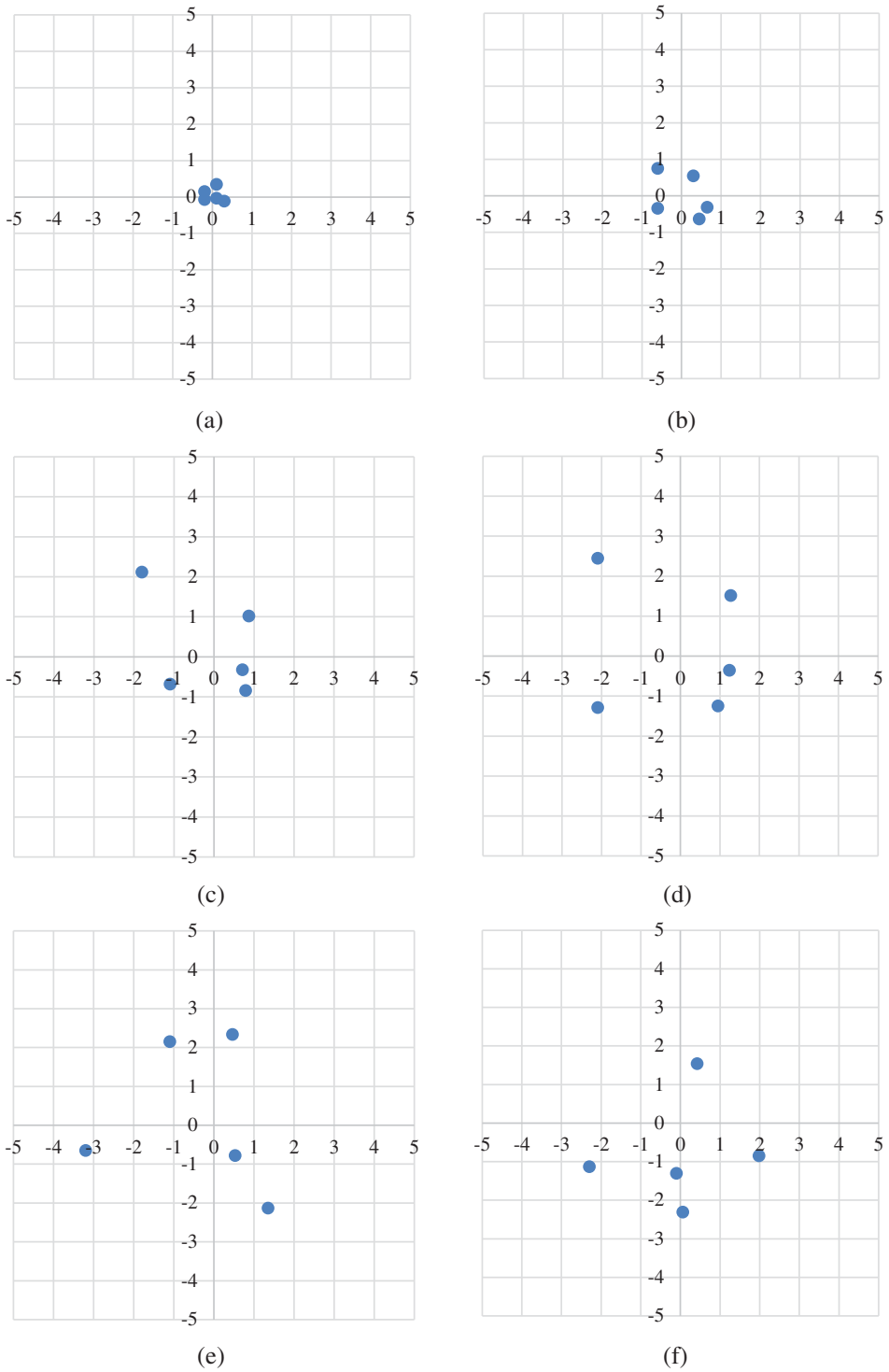


Figure 7: Collision point comparison of different models. (a) Real tissue, (b) Our model, (c) Volumetric geometric model [29], (d) Meshless physical model [30], (e) Point primitives model [2], (f) 3D soft tissue deformation [31]

Then, the real-time performance of our collision detection algorithm is validated, which is also carried out under the above five different cutting angles. Time is recorded at the beginning of the experiment. The scalpel was simultaneously cut in our system and other systems in [33]. The actual collision time is 2000 ms after timing begins. The collision time of our system and other systems along with real collision time is summarized in Tab. 2.

Table 2: Collision time comparison of different models

Model	Collision time under different cutting angles (ms)					Mean relative error (%)
	30	40	50	60	70	
Real collision time	15.64					–
Our model	15.67	15.59	15.63	15.69	15.66	0.05
Bounding volume hierarchy (BVH)	18.92	18.95	18.90	18.84	18.93	20.90
Space partition	21.37	21.39	21.30	21.24	21.35	36.38
Discrete collision detection (DCD)	23.46	23.42	23.53	23.57	23.21	49.86
Continuous collision detection (CCD)	19.63	19.53	19.86	19.48	19.72	25.60

According to Tab. 2, the mean relative error is the relative error of the collision time between virtual cutting system and real cutting at different cutting angles, which can be calculated by Eq. (9)

$$\eta_{average} = 100\% * \frac{1}{5}\sum_{t=1}^{5}\frac{t_i - t_0}{t_0} \tag{9}$$

where $\eta_{average}$ is the mean relative error, t_i is the collision time under a current cutting angle, t_0 is the real collision time and equals to 2000 ms, and i represents 5 different cutting scenes with different cutting angles. The mean relative error between the collision time of our system and real cutting is smaller, which indicates that the collision detection algorithm in the present study is better than other cutting systems in real time performance.

4.3 Practical Cutting Evaluation

To verify the practical performance of the proposed system, a comprehensive practical cutting evaluation is carried out. 20 residents from the department of Cardiothoracic surgery of the First Affiliated Hospital of Nanjing Medical University were invited to implement cutting operation on the virtual soft tissue with GHOST SDK. Four cutting systems are evaluated in the comprehensive practical cutting evaluation, including our cutting system, cutting system in [32–34]. After carrying our cutting operation on different cutting systems, residents are asked to grade the systems with the following 7 indicators: soft tissue texture, force feedback performance, cutting effect, incision shape, real-time performance, training effectiveness, and system stability. The full score of the above indicators is 10.

In virtual surgery system evaluation, different indicators have different importance degrees. Therefore, we use a weighting method to help analyze the evaluation results. The weighting

method is named the Delphi method [35], which is based on the knowledge and the experience of experts. We chose the Delphi method to determine the weights of indicators because we already invited 20 residents to score the proposed systems along with other cutting system. The procedure of the Delphi method is shown in Fig. 8.

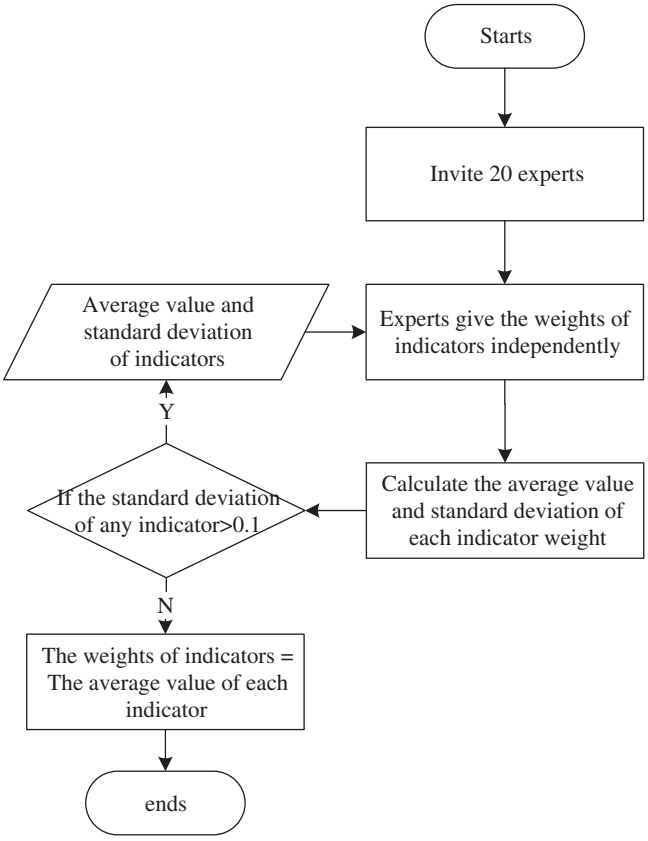


Figure 8: Procedure of the Delphi method

As is shown in Fig. 8, 20 experts in the professional field who have both deep theoretical mastery and ample work experience are invited to give the weights of the 7 indicators. In the method, the average value and standard deviation of all indicator weights given by experts are calculated. If the opinions of experts tend to be basically coincident, the average value of weight of each indicator is considered as the weights of indicators.

In the present study, the predetermined standard deviation of indicators weights is 0.1. According to Delphi method, the weight of the indicators is set as follows, tissue texture = 0.09, force feedback = 0.17, cutting effect = 0.24, incision shape = 0.13, real-time performance = 0.10, training effectiveness = 0.21, and system stability = 0.06. According to the weights of different indicators and the scores graded by the experts, the average scores of all indicators and the scores of four models calculated by the Delphi method are shown in Tab. 3.

Table 3: Single-indicator and comprehensive scores of five models

	Models				
	Ours	Element deletion	Node snapping	Mesh subdivision	Element duplication
Single-indicator scores					
Tissue texture	10	8	8	9	9
Force feedback	9	8	9	9	8
Cutting effect	9	9	7	9	9
Incision shape	10	8	7	9	9
Real-time performance	9	8	9	10	9
Training effectiveness	10	8	8	9	9
System stability	9	9	9	9	9
Comprehensive scores	9.43	8.3	7.96	9.1	8.83

According to Tab. 3 the single-indicator scores of most indicators of our model are higher than that of other models. And the comprehensive score of the cutting system in [32–35], and our system are 7.83, 7.90, 7.88, and 8.94, respectively. Therefore, the residents generally believe that our cutting system is better than the systems based on the other models.

5 Conclusions

In this paper, we propose a virtual cutting system based on the MSM, which well solves the three difficult points of cutting operation simulation in virtual surgery. First, residual stress is introduced to optimize the MSM and simulate the shrinking effect of soft tissue in cutting. Then, a cylinder-based collision method is used to detect the collision between cylindrical surgical tools and soft tissue. Finally, we reconstruct the incision with a three-stage cutting simulation method, which optimizes the incorporation of cutting operation and soft tissue model. We carry out three validation experiments to verify system performance. The first experiment verifies the cutting accuracy of the proposed system; the cutting accuracy is presented by means of a width-time curve. The experiment results show that the width-time curve of our system accords better to the real curve than other models, which indicates that our cutting system has better model accuracy. The second experiment verifies the effect of collision detection in terms of real-time performance and collision accuracy. Compare to other systems, the collision point of our system is closer to the real cutting condition. In the meanwhile, the response time of our system is shorter. The third experiment is a comprehensive evaluation experiment. 20 residents are invited to try out and score our system. The Delphi method is utilized to deal with the final scores, which can improve the reliability of scores. The results show that our system has higher scores than other systems in terms of a comprehensive score. As a result, our system can vividly simulate the virtual cutting operation and is a preferable choice for cutting surgery training.

Funding Statement: This work was supported, in part, by the National Nature Science Foundation of China under Grant Nos. 61502240, 61502096, 61304205, 61773219; in part, by the Natural Science Foundation of Jiangsu Province under Grant Nos. BK20191401 and BK20201136; in part, by the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD) fund; in part, by the Collaborative Innovation Center of Atmospheric Environment

and Equipment Technology (CICAEET) fund; NUIST Students' Platform for Innovation and Entrepreneurship Training Program.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

1. Conboy, H. M., Avrunin, G. S., Clarke, L. A., Osterweil, L. J., Goldman, J. M. et al. (2017). Cognitive support during high-consequence episodes of care in cardiovascular surgery. *2017 IEEE Conference on Cognitive and Computational Aspects of Situation Management*, pp. 1–3.
2. Cotin, S., Delingette, H., Ayache, N. (1999). Real-time elastic deformations of soft tissues for surgery simulation. *IEEE Transactions on Visualization and Computer Graphics, 5(1),* 62–73. DOI 10.1109/2945.764872.
3. Yoon, J. H., Lee, J. M., Jun, J. H., Suh, K. S., Coulon, P. et al. (2015). Feasibility of three-dimensional virtual surgical planning in living liver donors. *Abdominal Imaging, 40(3),* 510–520. DOI 10.1007/s00261-014-0231-9.
4. Taylor, J. S., Tofts, P. S., Port, R., Evelhoch, J. L., Knopp, M. et al. (1999). MR imaging of tumor microcirculation: Promise for the new millenium. *Journal of Magnetic Resonance Imaging: An Official Journal of the International Society for Magnetic Resonance in Medicine, 10(6),* 903–907. DOI 10.1002/(SICI)1522-2586(199912)10:6<903::AID-JMRI1>3.0.CO;2-A.
5. Wu, J., Westermann, R., Dick, C. (2015). A survey of physically based simulation of cuts in deformable bodies. *Computer Graphics Forum, 34(6),* 161–187. DOI 10.1111/cgf.12528.
6. Joldes, G., Bourantas, G., Zwick, B., Chowdhury, H., Wittek, A. et al. (2019). Suite of meshless algorithms for accurate computation of soft tissue deformation for surgical simulation. *Medical Image Analysis, 56,* 152–171. DOI 10.1016/j.media.2019.06.004.
7. Pan, J. J., Chang, J., Yang, X., Liang, H., Zhang, J. J. et al. (2015). Virtual reality training and assessment in laparoscopic rectum surgery. *International Journal of Medical Robotics and Computer Assisted Surgery, 11(2),* 194–209. DOI 10.1002/rcs.1582.
8. Pan, J., Yan, S., Qin, H., Hao, A. (2018). Real-time dissection of organs via hybrid coupling of geometric metaballs and physics-centric mesh-free method. *Visual Computer, 34(1),* 105–116. DOI 10.1007/s00371-016-1317-x.
9. Jin, X., Joldes, G. R., Miller, K., Yang, K. H., Wittek, A. (2014). Meshless algorithm for soft tissue cutting in surgical simulation. *Computer Methods in Biomechanics and Biomedical Engineering, 17(7),* 800–811. DOI 10.1080/10255842.2012.716829.
10. Zhang, J., Zhong, Y., Gu, C. (2017). Deformable models for surgical simulation: A survey. *IEEE Reviews in Biomedical Engineering, 11,* 143–164. DOI 10.1109/RBME.2017.2773521.
11. Yeung, Y. H., Crouch, J., Pothen, A. (2016). Interactively cutting and constraining vertices in meshes using augmented matrices. *ACM Transactions on Graphics, 35(2),* 1–17. DOI 10.1145/2856317.
12. Townsend, M. T., Sarigul-Klijn, N. (2016). Updated Lagrangian finite element formulations of various biological soft tissue non-linear material models: A comprehensive procedure and review. *Computer Methods in BiomeChaniCs and BiomediCal Engineering, 19(11),* 1137–1142. DOI 10.1080/10255842.2015.1111343.
13. Zhang, J., Zhong, Y., Gu, C. (2018). Ellipsoid bounding region-based ChainMail algorithm for soft tissue deformation in surgical simulation. *International Journal on Interactive Design and Manufacturing, 12(3),* 903–918. DOI 10.1007/s12008-017-0437-5.
14. Frisken-Gibson, S. F. (1999). Using linked volumes to model object collisions, deformation, cutting, carving, and joining. *IEEE Transactions on Visualization and Computer Graphics, 5(4),* 333–348. DOI 10.1109/2945.817350.
15. Lee, Y. H., Ahn, H., Ahn, H. B., Lee, S. Y. (2019). Visual object detection and tracking using analytical learning approach of validity level. *Intelligent Automation and Soft Computing, 25(1),* 205–215.
16. Qasim, A., Kazmi, S., Asad, R. (2019). Formal modelling of real-time self-adaptive multi-agent systems. *Intelligent Automation and Soft Computing, 25(1),* 49–63.

17. Teschner, M., Kimmerle, S., Heidelberger, B., Zachmann, G., Raghupathi, L. et al. (2005). Collision detection for deformable objects. *Computer Graphics Forum, 24(1),* 61–81. DOI 10.1111/j.1467-8659.2005.00829.x.
18. Johnsen, S. F., Taylor, Z. A., Han, L., Hu, Y., Clarkson, M. J. et al. (2015). Detection and modelling of contacts in explicit finite-element simulation of soft tissue biomechanics. *International Journal of Computer Assisted Radiology and Surgery, 10(11),* 1873–1891. DOI 10.1007/s11548-014-1142-5.
19. Ferté, G., Massin, P., Moës, N. (2016). 3D crack propagation with cohesive elements in the extended finite element method. *Computer Methods in Applied Mechanics and Engineering, 300,* 347–374. DOI 10.1016/j.cma.2015.11.018.
20. Paulus, C. J., Untereiner, L., Courtecuisse, H., Cotin, S., Cazier, D. (2015). Virtual cutting of deformable objects based on efficient topological operations. *Visual Computer, 31(6–8),* 831–841. DOI 10.1007/s00371-015-1123-x.
21. Qian, K., Jiang, T., Wang, M., Yang, X., Zhang, J. (2016). Energized soft tissue dissection in surgery simulation. *Computer Animation and Virtual Worlds, 27(3–4),* 280–289. DOI 10.1002/cav.1691.
22. Kibsgaard, M., Thomsen, K. K., Kraus, M. (2014). Simulation of surgical cutting in deformable bodies using a game engine. *International Conference on Computer Graphics Theory and Applications*, pp. 1–6.
23. Zhang, Y., Yuan, Z., Ding, Y., Zhao, J., Duan, Z. et al. (2010). Real time simulation of tissue cutting based on GPU and CUDA for surgical training. *International Conference on Biomedical Engineering and Computer Science*, pp. 1–4.
24. Courtecuisse, H., Allard, J., Kerfriden, P., Bordas, S. P., Cotin, S. et al. (2014). Real-time simulation of contact and cutting of heterogeneous soft-tissues. *Medical Image Analysis, 18(2),* 394–410. DOI 10.1016/j.media.2013.11.001.
25. Quesada, C., González, D., Alfaro, I., Cueto, E., Chinesta, F. (2016). Computational vademecums for real-time simulation of surgical cutting in haptic environments. *International Journal for Numerical Methods in Engineering, 108(10),* 1230–1247. DOI 10.1002/nme.5252.
26. Elhoseny, M., Tharwat, A., Hassanien, A. E. (2018). Bezier curve based path planning in a dynamic field using modified genetic algorithm. *Journal of Computational Science, 25,* 339–350. DOI 10.1016/j.jocs.2017.08.004.
27. Barrientos, A. F., Jara, A., Quintana, F. A. (2017). Fully nonparametric regression for bounded data using dependent Bernstein polynomials. *Journal of the American Statistical Association, 112(518),* 806–825. DOI 10.1080/01621459.2016.1180987.
28. Delingette, H., Cotin, S., Ayache, N. (2000). A hybrid elastic model allowing real-time cutting, deformations and force-feedback for surgery training and simulation. *Visual Computer, 16(8),* 437–452. DOI 10.1007/PL00007215.
29. Shi, W., Liu, P. X., Zheng, M. (2019). A new volumetric geometric model for cutting procedures in surgical simulation. *Computer Methods and Programs in Biomedicine, 178,* 77–84. DOI 10.1016/j.cmpb.2019.06.015.
30. Shi, W., Liu, P. X., Zheng, M. (2020). Cutting procedures with improved visual effects and haptic interaction for surgical simulation systems. *Computer Methods and Programs in Biomedicine, 184,* 105270. DOI 10.1016/j.cmpb.2019.105270.
31. Jayasudha, K., Kabadi, M. G. (2020). Soft tissues deformation and removal simulation modelling for virtual surgery. *International Journal of Intelligence and Sustainable Computing, 1(1),* 83–100. DOI 10.1504/IJISC.2020.104830.
32. Nienhuys, H., Stappen, A. (2001). A surgery simulation supporting cuts and finite element deformation. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 145–152.
33. O'brien, J. F., Hodgins, J. K. (1999). Graphical modeling and animation of brittle fracture. *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 137–146.
34. Molino, N., Bao, Z., Fedkiw, R. (2004). A virtual node algorithm for changing mesh topology during simulation. *ACM Transactions on Graphics, 23(3),* 385–392. DOI 10.1145/1015706.1015734.
35. Guo, H., Wang, X., Wang, L., Chen, D. (2016). Delphi method for estimating membership function of uncertain set. *Journal of Uncertainty Analysis and Applications, 4(1),* 3. DOI 10.1186/s40467-016-0044-1.