Pacific Graphics (2019)
C. Theobalt, J. Lee, and G. Wetzstein (Editors)

Short Paper

Towards biomechanically and visually plausible volumetric cutting simulation of deformable bodies

Yinling Qian[1] Wenbin Huang[1] Weixin Si[1†] Xiangyun Liao[1] Qiong Wang[1] Pheng-Ann Heng[2,3]

[1]Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, China
[2]T Stone Robotics Institute, The Chinese University of Hong Kong, China
[3]Department of Computer Science and Engineering, The Chinese University of Hong Kong, China

Abstract

Due to the simplicity and high efficiency, composited finite element method(CFEM) based virtual cutting attracted much attention in the field of virtual surgery in recent years. Even great progress has been made in volumetric cutting of deformable bodies, there are still several open problems restricting its applications in practical surgical simulator. First among them is cutting fracture modelling. Recent methods would produce cutting surface immediately after an intersection between the cutting plane and the object. But in real cutting, biological tissue would first deform under the external force induced by scalpel and then fracture occurs when the stress exceeds a threshold. Secondly, it's computation-intensive to reconstruct cutting surface highly consistent with the scalpel trajectory, since reconstructed cutting surface in CFEM-based virtual cutting simulation is grid-dependent and the accuracy of cutting surface is proportional to the grid resolution. This paper propose a virtual cutting method based on CFEM which can effectively simulate cutting fracture in a biomechanically and visually plausible way and generate cutting surface which is consistent with the scalpel trajectory with a low resolution finite element grid. We model this realistic cutting as a deformation-fracture repeating process. In deformation stage, the object will deform along with the scalpel motion, while in the fracture stage cutting happens and a cutting surface will be generated from the scalpel trajectory. A delayed fracturing criteria is proposed to determine when and how the cutting fracture occurs and an influence domain adaptation method is employed to generate accurate cutting surface in both procedures of deformation and fracture. Experiments show that our method can realistically simulate volumetric cutting of deformable bodies and efficiently generate accurate cutting surface thus facilitating interactive applications.

CCS Concepts

• ***Human-centered computing*** → ***Virtual reality;*** • ***Computing methodologies*** → ***Physical simulation; Shape modeling;***

1. Introduction

Volumetric cutting simulation on deformable bodies is an compulsory component of surgical simulator. Last two decades witnessed a rapid development in surgical simulator, also including the methods for cutting simulation. These methods adopt spatial discretization of volumetric meshes or points, namely mesh-based methods [DGW11, WDW13, JZYP18] and mesh-free methods [SOG09, KBT17, HFG*18, SLLW18]. Most of the mesh-based methods adopt finite element method (FEM) [WWD15]. FEM subdivides the simulation domain into basic elements, such as tetrahedron and hexahedron, and divides the original problem into a set of element equations. The element equations are simple equations that locally approximate the original complex equations to be studied. With this representation, FEM can accurately represent complex geometry while capture local effects. Due to this advantage, mesh-based methods are widely used in surgical simulators. By compositing fine finite elements into coarse finite elements and adopting interpolation scheme to compute mechanical properties for fine elements, composite finite element method (CFEM) [NKJF09] highly improves the simulation efficiency while retaining fine simulation details. Several recent works based on hexahedral elements integrate CFEM into cutting simulation framework and simulate virtual cutting in real-time [WDW13,JZYP18]. Fig. 1 depicts the three geometric models of CFEM-based cutting with one-level composition. The composite elements are used for physically-based modelling and interpolation method is employed to scatter the deformation of composite elements to fine finite voxels. A surface mesh can be reconstructed from the dual grid cube which is constructed by voxel centers and the links between voxels using the dual contouring approach [JLSW02]. Even great progress has been made in virtual cutting simulation, there are several open problems for CFEM-based cutting methods. First, previous work mainly focus on improving the simulation efficiency and stability, while less attention

† Corresponding author.

DOI: 10.2312/pg.20191335

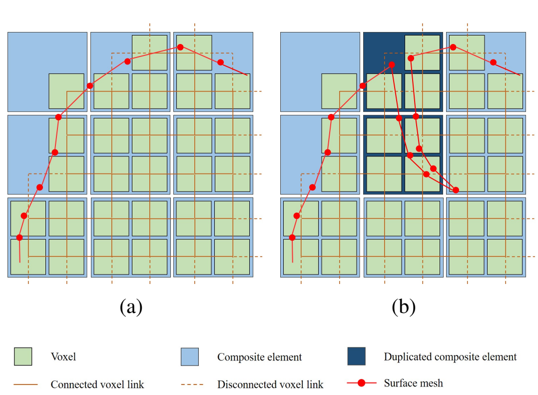


Figure 1: *CFEM-based virtual cutting method. The left is a object before cutting and the right is the object after cutting.*

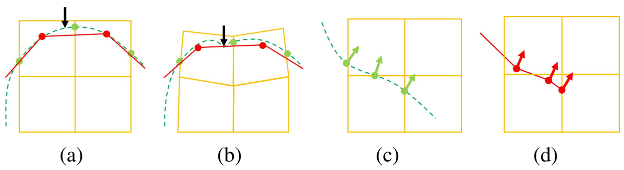


Figure 2: *2D view of cut surface inconsistency from CFEM-based virtual cutting due to limited voxel resolution. The green points are intersection points on link grid cubes and the red points are reconstructed vertices via dual contouring approach The green curves are desired visual effects while the red segments are surface reconstructed from dual contouring approach. (a-b) show the inconsistency in deformation stage and (c-d) show the inconsistency in cutting surface reconstruction.*

is paid to the cutting fracture mechanics. This severely restricts the realism of interaction between cutting tools and the virtual tissue. In recent work, the scalpel is modelled as infinitely sharp and can penetrate the object immediately [WWD15]. Actually, the scalpel would first press to deform the object and then penetrate the object with a large enough stress. Secondly, for CFEM, the accuracy of cutting surface is resolution-dependent, which cannot achieve the consistent representation while compute in a lightweight manner. Inconsistency would be visible in both the deformation and fracture process as illustrated in Fig. 2. In deformation, an indentation segment should appear along with the pressing process of the scalpel. However, since the link grid is resolution limited, the local triangle facets will be pressed down as the collision response rather than an accurate indentation segment. Then in fracture, intersection points will be added onto the link edges which are disconnected, and then the fracture ending point will be the intersection point on the last link edges (as illustrated in Fig.2(d)).

Real cutting on deformable objects exhibit a characteristic pattern [CDL07]: cutting is formed by repeating units including a deformation stage followed by a sudden fracture. However, it's non-trivial to combine deformation simulation into existing methods to produce virtual cutting in this manner. Besides, We need to generate deformed mesh with the surface representation in existing method and determine when and how the fracture occurs. To achieve our goal, we propose a geometry-based fracture model which can be seamlessly integrated into current CFEM-based virtual cutting. Moreover, an influence domain adaptation method is designed to generate surface mesh which is consistent with the scalpel trajectory in both deformation and fracture stage. The main contributions of this work can be summarized as:

- A biomechanics-motivated virtual cutting model which is represented by deformation-fracture repeating units and a delayed fracturing criteria to determining cutting fracture;
- A influence domain adaptation method for generating deformed surface mesh even with a low resolution fine element grid in high efficiency

2. Related work

Wu *et al* [WWD15] provide a comprehensive survey on cutting methods of deformable objects. Here we only review the most related work.

Early virtual cutting simulation are mainly based on FEM with tetrahedral meshes [CDA00, SHGS06, SDF07]. Tetrahedral mesh are used for both physically-based modelling and visualization. When cutting occurs, a cutting surface should be generated in consistency with scalpel trajectory on the tetrahedral mesh. Unfortunately, this might create ill-shaped elements and lead to numerical instability. Various methods are proposed to avoid the creation of ill-shaped elements, including deleting existing elements [CDA00], splitting along existing elements faces [LT07], snapping vertices [BGTG03], refining the tetrahedral mesh [LJD07], or combing snapping and refinement [SHGS06]. Wang *et al* [WJST14] further designed a virtual node algorithm to improve the stability and support multiple cuts per tetrahedron face. The extended FEM (XFEM) multiplies shape functions of cut elements by enrichment functions to generate physically correct deformation behavior near the cutting surface while maintaining the stability of deformation [KMB*09, KBT17].

Hexahedral based FEM cutting has attracted more attention in recent work because it can avoid the problem of ill-shaped elements and the discretization can be generated more easily from volumetric image data or polygonal surface mesh. Dick *et al* [DGW11] employ an adaptive octree grid to represent cuts at very fine scales and embed topological changes of the simulation grid into a geometric multigrid solver. The splitting cubes algorithm is adopted to construct a surface that accurately aligns with the cuts from a specific linked voxel representation of the simulation domain. Seiler *et al* [SSSH11] embed a high-resolution surface into an adaptively refined octree simulation mesh to improve the robustness. Wu *et al* [WDW13] propose an efficient collision detection method based on spatial hashing and signed distance field and a topology-aware interpolation scheme is introduced to improve the collision accuracy. Jia *et al* [JZYP18] designed a CPU-GPU parallel framework for real-time interactive cutting simulation of deformable objects. A novel multi-stage reduction algorithm is used to perform inter-object collision and object self-collision on the GPU and cutting operation is divided into four independent tasks that run in parallel on CPU.

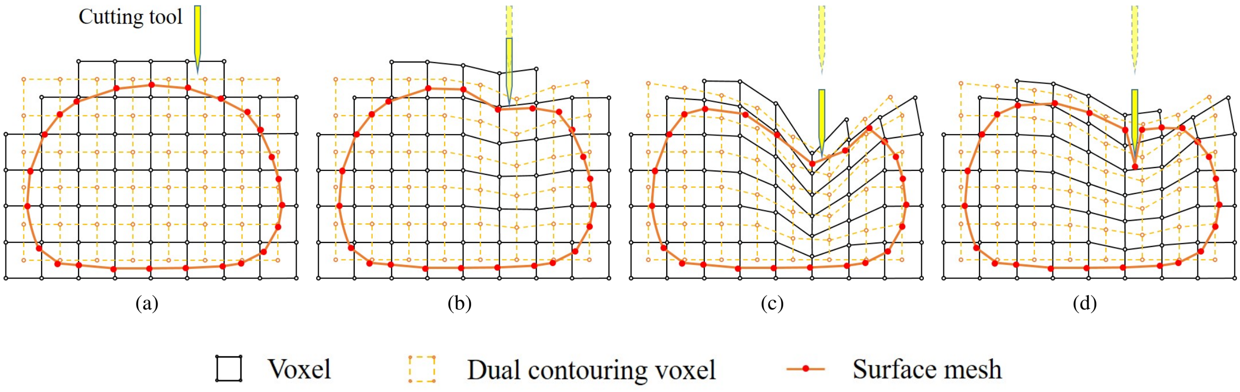


Figure 3: *Biomechanics-motivated virtual cutting model. The cutting process consists of repeating units of deformable-fracture process, and fracture only happens with a large enough stress. From left to right are: object before contacting, deformed object under a small stress, deformed object with a stress which will lead to fracture, object after cutting fracture.*

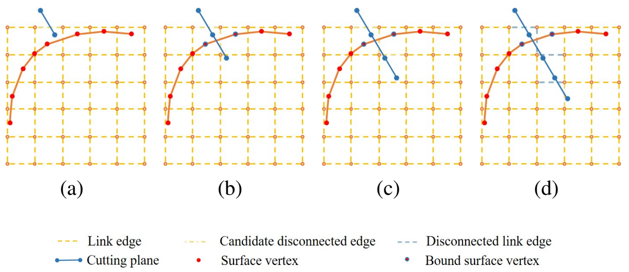


Figure 4: *Geometry based fracture determination. The collision detection is conducted between the cutting tool and reference object. The first three cutting plane only deform the object and the last one will lead to a cut in the object. Link edges are collected in the first three frames and will be used for cutting computation in the last frame.*

3. Biomechanically and visually plausibile volumetric cutting simulation

The method proposed in this paper is based on CFEM-based virtual cutting [WDW13, JZYP18], since it can be efficient enough to achieve real-time performance and generate more accurate surfaces. In cutting, a scalpel is simplified to a line and can generate a cutting plane between two frames. Then, all the links swept by the cutting plane will be disconnected and the surface mesh will be reconstructed according to the updated dual grid cube. To further improve the visual realism of virtual cutting, we model cutting as a deformation-fracture repeating process, which is closer to real cutting of deformable bodies. It is worth noting that the object would first deform under the stress from scalpel and cutting fracture happens after the stress exceeds a threshold. Fig. 3 illustrates a deformation-fracture unit of the cutting model. Cutting consists of several units of this deformation-fractures process. To modify previous CFEM-cutting method to achieve our goal, we need to combine deformation and fracture in a unified framework. Unfortunately, this is non-trivial. Thus, we first propose a delayed fracturing criteria to seamlessly combine the deformation and fracture process. In addition, an influence domain adaptation method is proposed to reconstruct surface mesh which is highly consistent with the scalpel motion. The overall simulation loop of the proposed algorithm is given in Algorithm 1.

Algorithm 1 Virtual Cutting Simulation Loop

```
Cutting plane computation
Cut CFEM link computation
Cutting fracture determination
if fracture occurs then
    Cutting computation
end if
Influence domain adaptation
Collision detection
CFEM element updating
```

Biomechanics-motivated virtual cutting The key issue for generating biomechanically plausible cutting effects is to determine when and how the fracture occurs in the deformation state. At the same time, we need to consider the compatibility with CFEM-based cutting geometrical models. To fulfill these requirements, we propose a delayed fracturing criteria which is inspired by the fracturing criteria which is adopted for ductile fracture simulation [HJST13]. The cutting tool edge is simplified to a segment sequence, and is named as cutting front/end in current frame/previous frame. Besides, the cutting front and cutting end form the cutting plane, which is represented by a triangle mesh and will be used in tool object collision detection.

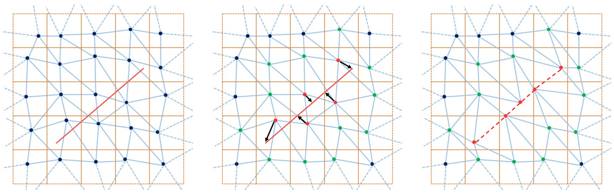
To compute the disconnected links, the geometrical models before the current deformation-fracture unit are taken as the reference state and the cutting front in each frame is also transformed to the reference space for collision detection. In the deformation stage, surface vertices will be bound to cutting front successively and will move along with the cutting front before fracturing. Fig. 4 illustrates the proposed delayed fracturing criteria for a deformation-fracture unit which consists of three deformation operation and one fracturing operation. During the deformation stage, all the link edges cut by the cutting plane are only collected for potential fracturing computation. At the same time, if the cutting plane collides with the surface mesh, the surface vertices on the cut triangle facets will be bound to the cutting front (See in Fig. 4(b)). In the subsequent deformation computation, an external force from the cutting tool will be added to these vertices and further scattered onto corresponding composite element vertices. The deformation offset of fine element voxel centers which is also the vertices of the link edge, will be used for determining whether fracture occurs. Once the maximum deformation offset exceeds a given threshold d_0, we conduct a cutting computation. Observing that real cutting cannot disconnect all the candidate disconnected edges, another threshold d_1 which is smaller than d_0, are used for checking whether the candidate link will be disconnected. Fig. 4(d) illustrates the states after a cutting operation. The candidate disconnected links remaining connected will be kept for next cutting operation and newly generated surface vertices will be bound to the cutting front for the upcoming deformation computation.

Influence domain adaptation method To generate deformation and fracture which is exactly consistent with the cutting tool, we propose a local surface reconstruction methods which can be seamlessly integrated to the dual contouring approach. The method consists of two parts. In the first step, a segment sequence which is aligned with the cutting front is generated by adjusting the surface vertices which come from the cut link grid cubes. Then, these adjusted vertices are converted to dual contouring grid to make it compatible with the surface representation of CFEM-based cutting method.

The influence domain adaptation method is as illustrated in Fig.5. As we discussed in the fracture determination method, surface vertices bound to cutting front move along with the cutting tool. To generate an indentation curve consistent with the cutting front, the surface vertices must exactly lie on the cutting front. After a collision is detected between the cutting plane and the surface mesh, we can calculate the cutting trace on the surface mesh (See in Fig.5(a)). Also, the dual grid cubes embedding in the cutting indentation can be computed and the corresponding surface vertices can be adjusted to form an indentation on the surface consistent with the cutting front. To minimize the stretching of the surface mesh, a projection vertex is calculated for each surface vertex pending for adjustment, see in Fig. 5(b). Projecting these vertices onto the computed positions generates a desired segment sequence thus improving the mesh accuracy.

However, dual contouring grid only can be adjusted by adding interaction point on the link edges. But it is challenging to adjust the intersection points to complete the adjustment process. In addition, dual contouring approach can't generate accurate cutting fracture as shown in Fig. 2(c-d). We propose to enhance the dual contouring grid by introducing a virtual intersection point which locates inside the link grid cube. All link grid cubes are initialized with an invalid virtual intersection point. Once a surface point need to be adjusted or a fracture occurs inside a link grid cube, the virtual intersection point is activated and assigned with the desired position. If a link grid cube has a activated virtual intersection point, the position of this point is directly returned for the dual contouring computation.

Directly combining deformation and cutting computation in one framework would easily lead to inconsistency between a deformation frame and a fracture frame. To solve this problem, we adopt a surface point binding scheme to guarantee the consistency during the cutting process. In deformation, surface vertices bound to the scalpel in influence domain adaptation will move along with the scalpel. At the same time, forces with respect to the deformation offsets are added to the coarse element vertices which the surface vertices belong to, thus generating visually plausible deformation results. When cutting occurs, vertices with deformation offset larger than d_1 are released and the newly generated vertices inside the object will be bound to scalpel.



(a) (b) (c)

Figure 5: *Influence domain adaptation. (a) Cutting trace on the surface is computed for local surface adjustment. (b) Compute the projection point for each vertex. (c) Segment sequence formed by the adjusted surface vertices.*

4. Experimental results

Currently, we implemented a single thread CPU version of the proposed algorithm and run the experiments on a PC with Intel Xeon CPU E3-1200 v3@3.10 GHz with 16 GB RAM memory.

To evaluate the effectiveness of the proposed algorithm, we first test with a liver model in different configurations, including basic CFEM-based cutting [WDW13], method with delayed fracture criteria, method with both delayed fracture criteria and influenced domain adaptation. The results are given in Fig. 6. With the basic CFEM-based cutting method, cutting fracture occurs and the cutting surface will be reconstructed correspondingly once the object collides with the object in each frame (shown in the first row of Fig.6). After integrating with the delayed fracture criteria, cutting fracture only occurs when the maximum offset of voxel centers exceeds the given fracturing threshold. Then, we can get virtual cutting results which is more visually plausible than that from the original method. The results with the delayed fracture criteria is given in the second row and third row of Fig. 6. Fig. 6(e,f,h,i,j,l) shows results with only deformation, while Fig. 6(g,k) show results after fracturing. Since the resolution of fine finite element is limited, it's computation intensive to generate surface in high consistency with the cutting tool. The basic CFEM-based cutting would generate visible inconsistency when the cutting front is not exactly lying on a link edge and show inconsistency artifact which is the same as that in Fig. 2(d), see in the downright corner of Fig.6(d). The proposed influence domain adaptation method well solved this problem. The

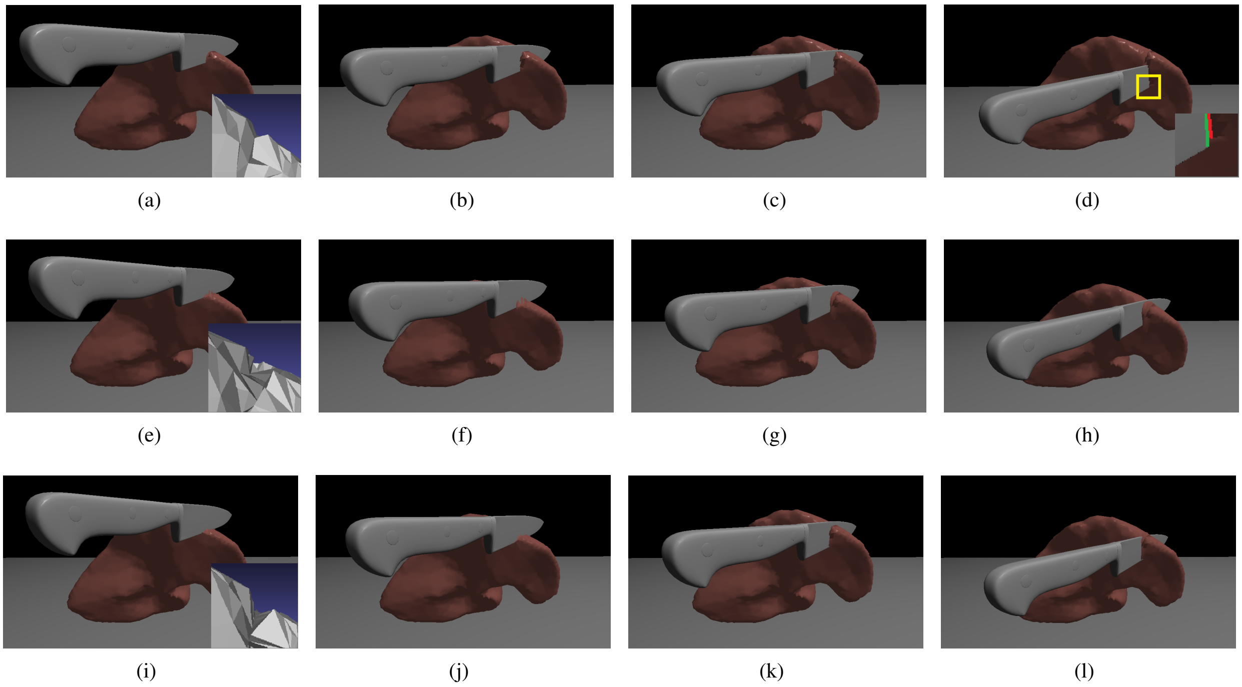


Figure 6: *Experimental results of different configurations. (a-d) show results of basic CFEM-based virtual cutting. (e-h) Results by integrating with the proposed delayed fracturing criteria. (i-l) Results with both delayed fracturing criteria and influence domain adaptation. The down-right corners of the first column show zoom-in view of mesh detail in the cutting area. The yellow box shows inconsistency between cutting tool and the cutting surface from base CFEM-based method. The green segment is the desired cutting surface while the red one is the cutting surface generated.*

downright corners in the first column in each row of Fig.6 show the top view of the surface mesh in deformation stages. Fig. 6(a) shows the result with cutting computation of basic CFEM-based method. Fig. 6(e) shows the result without influence domain adaptation which is the same as the artifact shown in Fig. 2(b). As a comparison, Fig. 6(i) shows result after influence domain adaptation and the indentation is highly consistent with the scalpel. The results demonstrate that the influence domain adaptation method can reconstruct surface mesh highly consistent with the cutting tool motion in both deformation and fracturing stages. More obvious differences can be observed in the supplementary video.

In addition, we test the proposed algorithm on several different models, including a model of bunny, bread and meat to verify the robustness of the proposed algorithm. The results are given in Fig. 7. The result show that our algorithm works well on objects with various geometric characteristics.

5. Conclusions

This paper presents a biomechanically plausible virtual cutting algorithm based on previous CFEM-based cutting methods. We model cutting as a deformation-fracture repeating process to mimic the interaction between cutting tool and a deformable object in real scene. A delayed fracturing criteria is designed to determine when and how the fracture occurs. Besides, we design an influence domain adaptation method to generate accurate surface mesh in cutting, while enhance the dual contouring approach with a virtual intersection point to support local vertex adjustment. Experiments show that the proposed algorithm can effectively improve the realism of virtual cutting effects with surface mesh highly consistent with the cutting toll trajectory.

Even the proposed geometry-based method can generate visually plausible effects, it's not physically correct thus limiting the model accuracy. We would further introduce physical stress computation to construct a more accurate fracture model. Besides, extension to high fidelity haptic feedback will be another further work direction.

6. Acknowledgement

This work was supported in part by the National Natural Science Foundation of China (U1813204, 61802385, 61802386, 61902386, 81601576), in part by the HK RGC TRS project T42-409/18-R, in part by the CUHK T Stone Robotics Institute, and in part by the Science and Technology Plan Project of Guangzhou (No.201704020141).

References

[BGTG03] Bielser D., Glardon P., Teschner M., Gross M.: A state machine for real-time cutting of tetrahedral meshes. In *11th Pacific Conference onComputer Graphics and Applications, 2003. Proceedings.* (2003), IEEE, pp. 377–386. 2

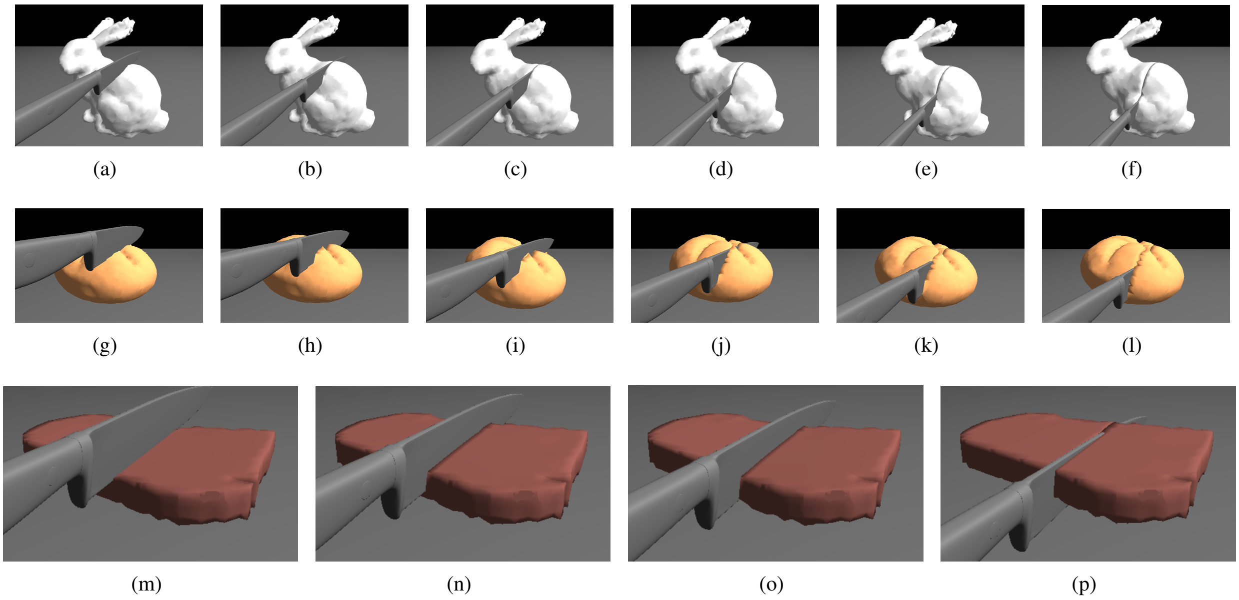


Figure 7: *More experimental results on models with different geometrical characteristics. Each row shows three units of deformation-fracturing process. Images in column 1, 3 and 5 are results under deformation and columns 2, 4 and 6 show results after cutting.*

[CDA00] COTIN S., DELINGETTE H., AYACHE N.: A hybrid elastic model for real-time cutting, deformations, and force feedback for surgery training and simulation. *The Visual Computer 16*, 8 (2000), 437–452. 2

[CDL07] CHANTHASOPEEPHAN T., DESAI J. P., LAU A. C.: Modeling soft-tissue deformation prior to cutting for surgical simulation: finite element analysis and study of cutting parameters. *IEEE transactions on biomedical engineering 54*, 3 (2007), 349–359. 2

[DGW11] DICK C., GEORGII J., WESTERMANN R.: A hexahedral multigrid approach for simulating cuts in deformable objects. *IEEE Transactions on Visualization and Computer Graphics 17*, 11 (2011), 1663–1675. 1, 2

[HFG*18] HU Y., FANG Y., GE Z., QU Z., ZHU Y., PRADHANA A., JIANG C.: A moving least squares material point method with displacement discontinuity and two-way rigid body coupling. *ACM Transactions on Graphics (TOG) 37*, 4 (2018), 150. 1

[HJST13] HEGEMANN J., JIANG C., SCHROEDER C., TERAN J. M.: A level set method for ductile fracture. In *Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2013), ACM, pp. 193–201. 3

[JLSW02] JU T., LOSASSO F., SCHAEFER S., WARREN J.: Dual contouring of hermite data. In *ACM transactions on graphics (TOG)* (2002), vol. 21, ACM, pp. 339–346. 1

[JZYP18] JIA S., ZHANG W., YU X., PAN Z.: Cpu–gpu parallel framework for real-time interactive cutting of adaptive octree-based deformable objects. In *Computer Graphics Forum* (2018), vol. 37, Wiley Online Library, pp. 45–59. 1, 2, 3

[KBT17] KOSCHIER D., BENDER J., THUEREY N.: Robust extended finite elements for complex cutting of deformables. *ACM Transactions on Graphics (TOG) 36*, 4 (2017), 55. 1, 2

[KMB*09] KAUFMANN P., MARTIN S., BOTSCH M., GRINSPUN E., GROSS M.: Enrichment textures for detailed cutting of shells. In *ACM Transactions on Graphics (TOG)* (2009), vol. 28, ACM, p. 50. 2

[LJD07] LIM Y.-J., JIN W., DE S.: On some recent advances in multimodal surgery simulation: A hybrid approach to surgical cutting and the use of video images for enhanced realism. *Presence: Teleoperators and Virtual Environments 16*, 6 (2007), 563–583. 2

[LT07] LINDBLAD A., TURKIYYAH G.: A physically-based framework for real-time haptic cutting and interaction with 3d continuum models. In *Proceedings of the 2007 ACM symposium on Solid and physical modeling* (2007), ACM, pp. 421–429. 2

[NKJF09] NESME M., KRY P. G., JEŘÁBKOVÁ L., FAURE F.: Preserving topology and elasticity for embedded deformable models. In *ACM Transactions on Graphics (TOG)* (2009), vol. 28, ACM, p. 52. 1

[SDF07] SIFAKIS E., DER K. G., FEDKIW R.: Arbitrary cutting of deformable tetrahedralized objects. In *Proceedings of the 2007 ACM SIGGRAPH/Eurographics symposium on Computer animation* (2007), Eurographics Association, pp. 73–80. 2

[SHGS06] STEINEMANN D., HARDERS M., GROSS M., SZEKELY G.: Hybrid cutting of deformable solids. In *IEEE Virtual Reality Conference (VR 2006)* (2006), IEEE, pp. 35–42. 2

[SLLW18] SI W., LU J., LIAO X., WANG Q.: Towards interactive progressive cutting of deformable bodies via phyxel-associated surface mesh approach for virtual surgery. *IEEE Access 6* (2018), 32286–32299. 1

[SOG09] STEINEMANN D., OTADUY M. A., GROSS M.: Splitting meshless deforming objects with explicit surface tracking. *Graphical Models 71*, 6 (2009), 209–220. 1

[SSSH11] SEILER M., STEINEMANN D., SPILLMANN J., HARDERS M.: Robust interactive cutting based on an adaptive octree simulation mesh. *The Visual Computer 27*, 6-8 (2011), 519–529. 2

[WDW13] WU J., DICK C., WESTERMANN R.: Efficient collision detection for composite finite element simulation of cuts in deformable bodies. *The Visual Computer 29*, 6-8 (2013), 739–749. 1, 2, 3, 4

[WJST14] WANG Y., JIANG C., SCHROEDER C., TERAN J.: An adaptive virtual node algorithm with robust mesh cutting. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2014), Eurographics Association, pp. 77–85. 2

[WWD15] WU J., WESTERMANN R., DICK C.: A survey of physically based simulation of cuts in deformable bodies. In *Computer Graphics Forum* (2015), vol. 34, Wiley Online Library, pp. 161–187. 1, 2