

# Realistic Haptic Feedback for Material Removal in Medical Simulations

Maximilian Kaluschke<sup>1</sup>

Mario Lorenz<sup>2,3</sup>

René Weller<sup>1</sup>

Niels Hammer<sup>3,4,5</sup>

Gabriel Zachmann<sup>1</sup>

Luigi Pelliccia<sup>2</sup>

**Abstract**—We present a novel haptic rendering method to simulate material removal in medical simulations at haptic rates. The core of our method is a new massively-parallel continuous collision detection algorithm in combination with a stable and flexible 6-DOF collision response scheme that combines penalty- and constraint-based force computation. Moreover, a volumetric object representation of the 3D objects allows us to derive a realistic local material model from experimental human cadaveric data, as well as support real-time continuous material removal. We have applied our algorithm to a hip replacement simulator and two dentistry-related simulations for root-canal opening and caries removal. The results show realistic continuous forces and torques at haptic rates.

## I. INTRODUCTION

In our increasingly aging society, the prevalence of surgical interventions for joint-related disease steadily rises. In Germany alone, 238k hip and 190k knee joint endoprostheses were implanted in 2017, making total hip and knee arthroplasty the 8th and 15th most common surgical procedures [8].

However, medical procedures such as orthopedic surgery are highly dependent on the surgeons' skill set and experience. Therefore, trainees need to acquire sound practical experience before performing complex surgery. Unfortunately, learning such skills on cadaveric models is expensive. However, in artificial models the physical properties often differ from the real procedures, thus limiting the learning gain dramatically. For instance, many orthopedic procedures are trained on simple dummies that exhibit very different physical properties. Dentistry students traditionally practice on plastic teeth with vastly different material properties compared to dentin and tooth enamel.

Medical Virtual Reality (VR) simulations may potentially bridge this gap, providing a suitable solution for preparing medical staff for interventional procedures. Among the most complex types of procedures to simulate involve removal of material, such as bone or tooth

enamel. The simulation of material removal is computationally complicated since pre-computed data structures have to be re-computed or updated. Especially the support of haptic feedback is challenging because of the required high frequency of ideally 1 kHz or more.

However, in many procedures, the ability to experience and learn the *feeling* of a surgical intervention is paramount to the students' ability to learn. Henn et al. have shown that practice in a virtual setting can significantly improve students' motor skills compared to traditional methods [12].

In this paper, we present a novel method for material removal simulations that supports a wide variety of materials and medical procedures, such as reaming, milling or drilling of bones or other biological hard tissues, which are required in hip- and knee arthroplasty but also for the root canal opening of teeth. The main idea of our approach is to combine a volumetric object representation with a novel massively-parallel continuous collision detection algorithm. This enables the adjustment of local material properties to fit experimental data from biomechanical tests.

## II. RELATED WORK

Convincing medical simulations to help users train complex tasks has been an important goal in the field of computer science. The emergence of virtual reality hardware and software has helped to revolutionize medical simulations, as it became easy to generate immersive visuals.

However, haptic rendering remains a challenging task in this domain. There has been some research focusing on simulating soft tissue cutting to offer virtual operations on the skin or internal organs. Most solutions so far used a surface representation, i.e., work directly on the triangle mesh and modify its topology by snapping close vertices to a cutting tool or subdividing cut triangles [23], [16], [13]. Most of these methods were not designed for real-time operation, and as such, have limited use for training. Other methods split the force computation or physics and haptic feedback (e.g., using interpolation) in separate threads to achieve real-time performance [22], [24]. Some cutting scenarios, such as scissors cutting a plate of material have been solved using analytical models [10], [5], [17], [9]. Unfortunately, these solutions are very specific and consequently cannot be generalized

<sup>1</sup>University of Bremen, Institute for Computer Graphics and Virtual Reality {mxkl,weller,zach}@cs.uni-bremen.de

<sup>2</sup>Chemnitz University of Technology, Professorship Machine Tool Design and Forming Technology

<sup>3</sup>University Hospital Leipzig, Department of Orthopedics

<sup>4</sup>Fraunhofer Institute for Machine Tools and Forming Technology IWU

<sup>5</sup>Medical University of Graz, Department of Macroscopic and Clinical Anatomy

to other scenarios. Zheng et al. used neural networks to simulate soft-tissue displacement using known mesh manipulation to simulate cutting [20].

For the simulation of bone milling or dentistry, it is sufficient to use solid objects. However, reducing such simulations simply to surface representation to simulate e.g. bone milling is challenging, because the interior material distribution can hardly be considered without any error. Moreover, they require a re-computation of the surface mesh in case of removed material. Some methods still rely on surface triangles mesh representations and simulate the cut volume by mesh deformation [19], [11], [3]. These do not work for concave objects or require a lot of extra work in those cases. Hence, a more suitable material representation is volumetric. Such methods are typically using a voxel representation [18] of the cuttable material [2], [14], [1].

However, all mentioned voxel-based methods use a penalty-based simulation. This is prone to force discontinuities which makes it unsuitable for complicated simulations with extremely large forces like hip surgery. We will present a constraint-based method that was inspired by the god-object method [25] and uses a second instance of the tool. However, we allow for arbitrary tools, independent of triangle count, and additionally compute penalty torques based on linear constraints.

### III. OUR APPROACH

The basic idea of our approach is to combine a volumetric object representation (polydisperse sphere packings according to [21]) to represent local material properties with a novel combined constrained- and penalty-based simulation method to compute the forces and torques. Furthermore, a new massively-parallel continuous collision detection method that is independent of the actual triangle count and relies only on the volumetric object representation helps us prevent errors typically appearing with voxel-based approaches like the *pop-through* effect with thin objects. Moreover, we use a bounding volume hierarchy (BVH) for the tool’s sphere packing for further acceleration.

The virtual tool has two instances, one that is free-moving and another one that is constraint to the surface of the bones, similar to the god-object method [25]. However, we use the sphere packing representation, which allows the simulation to easily support material removal

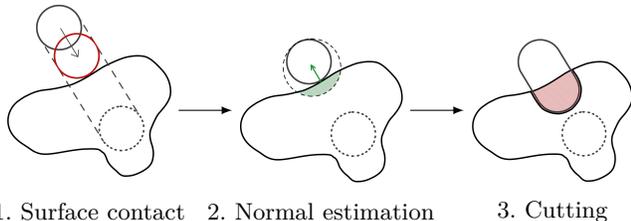


Fig. 1: Overview of our novel multi-pass GPU collision detection algorithm. Illustrations are highly exaggerated, each simulation frame only does tiny, incremental changes.

at run-time. The constraint-based approach solves the aforementioned problem of pop-through by providing a well-defined force direction at all times.

#### A. Collision Detection and Material Removal

The computational bottleneck of haptic rendering is usually the collision detection. We present a novel massively-parallel continuous collision detection approach that utilizes the GPU and supports material removal. It is based on a sphere packing representation of all objects, which includes the bones as well as the haptic tool. The sphere packings and their corresponding BVHs are pre-computed once for each collidable object and loaded during the start of the simulation. The sphere packing is updated at run-time to reflect changes due to material removal.

Our approach consists of three basic steps:

- 1) Finding the first point of contact between the tool and bone.
- 2) Estimating surface material properties at this point.
- 3) Removal of material along the path of the tool.

1) *Pass 1: Surface Contact:* To determine the contact point on the bone’s surface, we first perform a continuous collision detection: We search the tool’s spheres to check for an intersection along the path from the last free pose in the last frame and the end-effector’s pose in the current frame. We call the free state *proxy*, this is either the previous contact-free object pose or a pose that is constraint to the cut-able object’s surface.

The contact distance  $d_{t_i, b_j}$  for a tool sphere  $t_i$  and bone sphere  $b_j$  along the linear movement  $\vec{m}$  of the tool sphere  $t_i$  is given by:

$$d_{t_i, b_j} = \frac{2(\vec{m} \cdot \vec{\Delta}_c) - \sqrt{4(\vec{m} \cdot \vec{\Delta}_c)^2 - 4\vec{m}^2(\vec{\Delta}_c^2 - (r_{t_i} + r_{b_j})^2)}}{2\vec{m}^2} \quad (1)$$

A collision occurs in case of  $4(\vec{m} \cdot \vec{\Delta}_c)^2 - 4\vec{m}^2(\vec{\Delta}_c^2 - (r_{t_i} + r_{b_j})^2) \geq 0$ . We take the global minimum  $d_{min}$  of all positive distances  $d_{t_i, b_j}$  (for numerical inaccuracy we allow up to  $-\varepsilon$  negative values), which delivers the contact point  $p_c$  along  $\vec{m}$ . The tool is translated by  $d_{min}$ , effectively bringing the closest tool spheres in the following passes into contact with the bone.

This step can easily be parallelized by checking all spheres of the bone simultaneously against the tool’s BVH.

2) *Pass 2: Normal Estimation:* To generate continuous contact normals we do not simply rely on a single contact point but allow virtually for a small overlap according to the contact sphere’s radius.

To accomplish this, we assume a small  $\varepsilon$ -ball around the first contact point  $p_c$  and consider all intersecting spheres within this ball to be colliding.

This means, we search for all tool spheres  $t_i$  within the radius around the contact sphere and all overlapping bone spheres  $b_j$ . We use these sphere pairs to compute

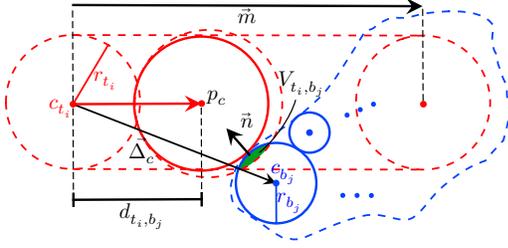


Fig. 2: Simplified visualization of passes 1 and 2. Continuous collision detection of each sphere along the movement vector  $\vec{m}$  and normal estimation based on all touching spheres.

a contact normal. The contribution to the normal for a particular sphere pair is weighted by the intersection volume  $V_{t_i, b_j}$  and the bone sphere's density  $\rho_{b_j}$ :

$$\vec{n} = \sum_{t_i} \sum_{b_j} \frac{c_{t_i} - c_{b_j}}{|c_{t_i} - c_{b_j}|} V_{t_i, b_j} \rho_{b_j} \quad (2)$$

where

$$V_{t_i, b_j} = \frac{\pi(r_{t_i} + r_{b_j} - \vec{\Delta}'_c)(\vec{\Delta}'_c)^2 + 2\vec{\Delta}'_c(r_{t_i} + r_{b_j}) - 3(r_{b_j} - r_{t_i})}{12\vec{\Delta}'_c}$$

with  $\vec{\Delta}'_c = c_{b_j} - c_{t_i}$  after  $c_{t_i}$  was modified in pass 1 (see III-A.2) and  $\rho_{b_j} \geq 0$  is the density of bone sphere  $b_j$ . To compute the final contact normal we normalize  $\vec{n}$ .

Similarly, we determine the surface density  $\rho$  and  $\hat{\rho}$ :

$$\rho = \sum_{b_i} V_{b_i} \rho_{b_i} \quad \hat{\rho} = \frac{\rho}{\sum_{b_i} V_{b_i}} \quad (3)$$

Note that  $\rho$  is not normalized by the sum of volumes; in this way, the contact volume is factored into the cutting speed reduction (see (4)). This method models reaming behavior very well, however, it might not be suitable for drilling simulations. In the case of drilling, it is more appropriate to use the normalized  $\hat{\rho}$  or to decrease the effect of the volume.

3) *Pass 3: Material Removal*: Obviously, in case of a collision, we have to remove a certain amount of material of the bone. Let  $p_e$  be the end position of the contact sphere assuming that it was not stopped at the bone's surface.  $p_e$  is calculated based on a cutting speed  $v_d$ , for which we chose a base speed  $v_b$  and reduce it based on the current surface density estimate  $\rho$  (see (3)).

$$v_d = \frac{v_b}{\rho} \quad (4)$$

Given a frame time  $\Delta_t$ , we can compute the end position  $p_e$ :

$$p_e = p_c + \vec{m} \cdot \frac{v_d}{\Delta_t} \quad (5)$$

The swept spheres define capsules along the movement direction  $\vec{m}$ , with centerline  $p_e - p_c$  and radius  $r_{t_i}$ . Inside those capsules, we remove all bone material. To do that, we compute the shortest line  $\vec{d}_c$  for each colliding bone

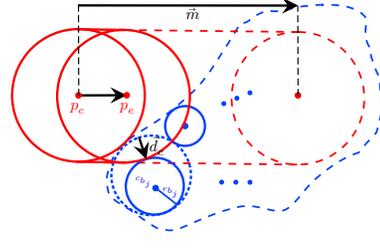


Fig. 3: Simplified visualization of pass 3. The tool motion creates a capsule that removes material. Small overlapping material spheres are shrunk and moved to resolve the overlap.

sphere from the spheres' centers to the capsules center line and shrink accordingly:

$$\forall b_j : c_{b_j} \leftarrow c_{b_j} - \frac{\vec{d}_c}{2} \quad r_{b_j} \leftarrow r_{b_j} - \frac{|\vec{d}_c|}{2} \quad (6)$$

Since the difference vector  $\vec{d}_c$  is used twice, once to shrink the radius and second to move the center, we factor both equations by  $\frac{1}{2}$ . This way we eliminate overlap, while not moving the sphere outside its original space (see Fig. 3).

### B. Force and Torque Computation

We use the contact points and normals of the collision detection to compute the forces and torques for the haptic rendering. This is done with a novel hybrid method that combines constraint- and penalty-based approaches: The linear force is calculated from the first contact point of the tool with the bone, this defines a hard constraint of the bone surface on the tool. The result is a collision-free position on the bone's surface. We attach a dampened spring of stiffness  $k$  and damping factor  $\zeta$  along the displacement vector from the end effector's position  $p_{HIP}$  to the surface position  $p_{proxy}$ .

$$F = (p_{proxy} - p_{HIP})k - v_{HIP} \cdot \zeta \quad (7)$$

The torque is calculated using a fast penalty-based approach, with the linear constraints as input. In each frame, the sphere  $s$  that is responsible for the linear constraint at  $p_s$  with linear force  $F_s$  generates a torque

$$\tau = (p_s - p_{COM}) \times F_s \quad (8)$$

where  $p_{COM}$  is the tool's center of mass. The center of mass  $p_{COM}$  of a set of spheres is in general calculated by a weighted sum over all spheres  $s_i$

$$p_{COM} = \frac{\sum_{s_i} c_{s_i} V_{s_i} \rho_{s_i}}{\sum_{s_i} V_{s_i} \rho_{s_i}} \quad (9)$$

where  $c_{s_i}$ ,  $V_{s_i}$  and  $\rho_{s_i}$  are the center, volume and material density of sphere  $s_i$  (i.e. of the bone).

A major advantage of our continuous collision detection is that we do not have to pre-compute normals for the inside of the object that would eventually become obsolete due to removed material and would have to be recomputed like it typically appears in voxel-based approaches.

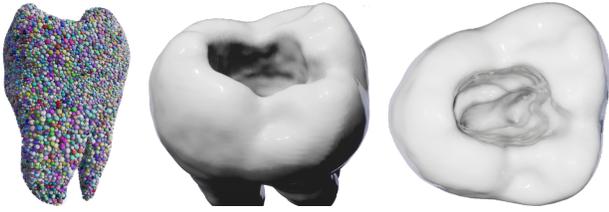


Fig. 4: Cut-able tooth representation. *Left*: Sphere packing of the tooth with 36k spheres. *Center*: Mesh rendering after creating an access opening by removing material. *Right*: Same modified mesh viewed from the top.

Additionally, we avoid the pop-through effect with thin objects and/or high velocities that typically occur with pure penalty-based approaches such as VPS [18], which is often used for material cutting simulations.

Finally, we want to be able to guarantee stable & continuous forces, so that there is no possibility of the haptic device to become uncontrollable. This is especially important for our use-case, where an industrial robot renders forces of up to 140 N (see IV-B).

Our method is especially suited for drilling, milling and reaming applications: continuous collision detection enables us to cut along the movement path which results in a continuously cut geometry. Penalty-based methods typically evaluate the material removal at distinct states of the simulation, which will result in scraps of material along the path.

### C. Friction

An important parameter in the behavior of the surface-bound proxy is the frictional force. We decided to implement a simple Coulomb friction model [7], because of its performance. Assuming that we have already computed the contact normal  $\hat{n}$  and force  $\vec{F}$  as described above. With these values, we can compute the normal force  $\vec{F}_n$  and the tangential force  $\vec{F}_t$

$$\vec{F}_n = \hat{n}(\vec{F} \bullet \hat{n}) \quad (10)$$

$$\vec{F}_t = \vec{F} - \vec{F}_n \quad (11)$$

We additionally keep track of the proxy's friction state: either it is *dynamic*, or *static* (when there is no collision, it is reset to dynamic, as it is moving). We define the frictional force as:

$$\vec{F}_f = \vec{F}_t \mu \quad (12)$$

Depending on the state we choose one of two friction coefficients  $\mu_d$  and  $\mu_s$  with  $\mu_d < \mu_s$  that fit the surface material, replace  $\mu$  in (12) by the appropriate coefficient:

$$\mu = \begin{cases} \mu_d, & \text{if proxy in dynamic state} \\ \mu_s, & \text{otherwise} \end{cases} \quad (13)$$

In case  $F_t > F_f$ , the tangential force is larger than the friction, so we stay in dynamic state and allow for lateral movement of the proxy. In the other case, the proxy changes to static friction and will not move laterally over the surface until the lateral force is high enough to overcome the friction.

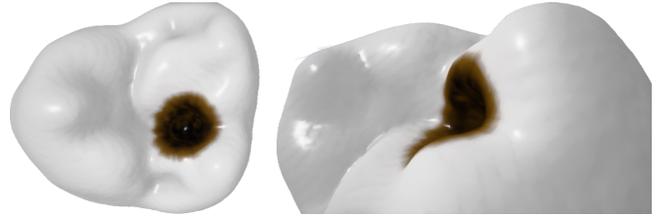


Fig. 5: Visualization of the tooth with carious parts.

## IV. USE CASES

We applied our novel material removal method to two medical applications with very different requirements regarding the amount of force and torque, as well as the drilling behavior. These two examples show the variability of our method.

### A. Use Case: Dentistry

In this use case, we have developed a training simulator for dental students in cooperation with MIRU lab from Mahidol University. In this scenario we use two Phantom Omni devices for the haptic feedback (one for controlling the bur and one for the mirror), given they are affordable, precise and the forces occurring during surgery are well within the specifications of the devices. However, the devices support 6-DOF for input but only 3-DOF for the force feedback. Hence, torques were not rendered in this use case. We have considered two important scenarios for dentistry: root canal access opening and caries removal.

1) *Root Canal Access Opening*: Students can cut a virtual tooth to create an opening to the pulp (see Fig. 4). This forms an important step in the root canal treatment and a major contributor in the tooth's stability following the surgery. For example, in the case of thinned enamel walls, the tooth may be at risk of breaking.

We modeled the tooth to have a uniform density for each of the different tooth material layers (enamel, dentin, pulp). The enamel was chosen to be about 1.5 times as hard as the dentin, with the pulp giving no resistance. This ratio was reported to be close to reality, according to preliminary feedback of a dentist.

The students had the option to switch between a large spherical bur and a smaller cylindrical bur, with the spherical bur cutting at a slower rate. This was automatically reflected in our simulation, since the spherical bur has a larger cutting volume, resulting in a larger speed reduction (see (4)).

2) *Caries Removal*: Another task that dental students practice is the removal of caries. Caries is visually identifiable by color: various shades of brown to nearly black color. Physically, caries spots are equally hard or softer than regular tooth material, and the strength varies with the depth of the caries. Also, the friction of caries is much higher compared to the normal tooth surface and it varies similarly depending on caries depth.

We modeled the appropriate densities on top of the model from the previous section. Caries densities range from 0.6 to 0.8 (as modifiers of the tooth material).

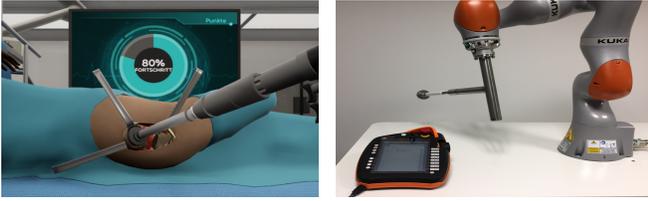


Fig. 6: Hip surgery simulator with haptic feedback. *Left*: In-game screenshot. *Right*: KUKA robot that renders the haptic feedback.

Additionally, we extended our visualization to support colors based on sphere tags for caries and densities (see Fig. 5). Spheres with caries tag are colored depending on the density modifier, normal spheres are colored in healthy tooth color. A value of 0.8 corresponds to light brown, 0.6 to brown-black, and all values between are interpolated.

### B. Use Case: Hip Replacement

In our second use case, we have implemented a hip replacement simulator for the training of orthopedic surgeons (see Fig. 6). The surgeon prepares the hip-socket for the acetabular component of the artificial joint by removing old cartilage and bone and adjusting the inner shape to ideally match the prosthetic hip socket inside. This step is crucial for the overall outcome of the procedure because a bad fit would risk future dislocation that can result in further complications for the patient.

Since the resulting shape of the hip socket is so important, we aimed to enable the user to see the exact reaming result. Our visualization renders an implicit surface that encapsulates all spheres according to the metaballs approach [4]. We have implemented a visualization that shows the difference between the users' outcome to an ideally reamed hip (see Fig. 7). This enables the instructor and the student to easily evaluate the quality of the reaming result. The ideal reaming result was created according to the feedback of an experienced orthopedic surgeon.

The view and exposure of the acetabulum in a real surgery is concealed by the overlying soft tissues during the milling process. Consequently, the surgeon needs to operate based on haptic input, making haptics the ideal technology for this application. Another challenge was

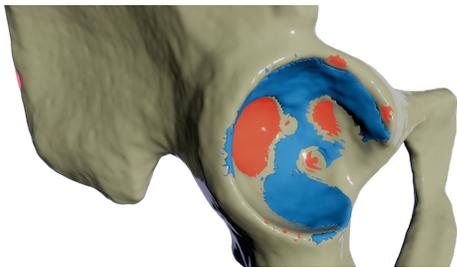


Fig. 7: A possible reaming result for the hip bone. Blue parts need to be removed. Red parts should have been kept intact. The implicit surface is generated by using a metaball-like function over the spheres.

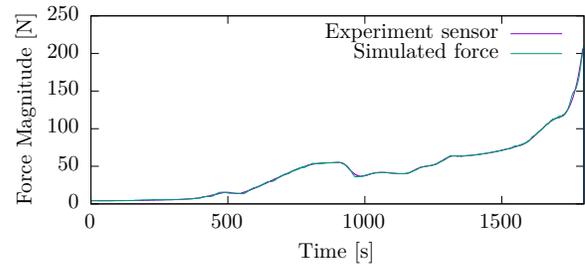


Fig. 8: Density optimization results. The force is recorded from a real hip being reamed. The hip density distribution is optimized to minimize the difference between both functions. The mean square error (MSE) is 0.9.

the demand for high forces (up to 200 N) occurring in the real surgery. To realize this we decided to use an industrial robot as a haptic device. The KUKA LBR robot provides full 6-DOF feedback.

Actually, the acetabulum is structurally relatively thin in the center, when compared to the reamer head. This does not only complicate the procedure, since it is easy to over-ream and permanently damage the bone, but it also means that a penalty-based method is unsuitable here, as there would be a chance of pop-through occurring if the operator pushed too hard or too fast. In that case, the reamer would be pulled towards the hip bone, with fatal consequences for the virtual operation.

1) *Material Model Optimization*: Unlike in the dentistry use case, the material properties of a hip bone differ significantly, e.g. due to osteoporosis or age-related change. To provide realistic force feedback we have to carefully set the individual material parameters of the spheres to match real-world data. Doing this manually does not lead to satisfying results. Hence, we propose a new method to set the material parameters based on real-world data.

To obtain realistic data, we have performed a series of experiments with human cadaveric tissues to analyze the force behavior and magnitude that occurs during acetabular reaming. For this, we mounted a hip reamer onto a bi-axial testing machine that would advance the reamer towards acetabula, which were fixed to a table. The reamer was advanced with a low fixed feed rate ( $0.01 \text{ mm s}^{-1}$  and  $0.03 \text{ mm s}^{-1}$ ). The relative angle was set according to a typical path that emerges during the

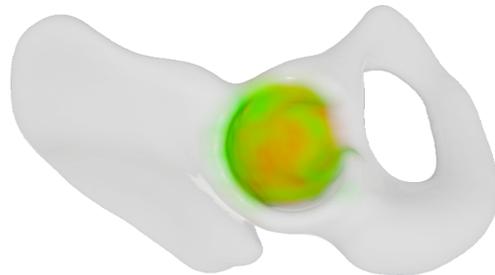


Fig. 9: Density distribution in the hip model. Colors indicate density, ranging from 0 in green to 1.5 in red. The default density is rendered in white.

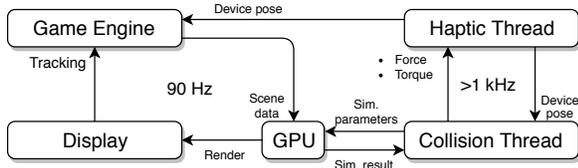


Fig. 10: Application architecture of our simulation, accessible by 3D game engines (Unity & Unreal) via a plugin.

operation. All forces applied to the reamer were recorded by the testing machine.

We use this experimental data to compute a density distribution for our sphere packing: we recreated the experimental setup virtually by advancing the tool in the same fashion towards a stationary virtual hip.

During each simulation frame, we recorded the simulation force  $F_s$  based on our algorithms output and the ideal force  $F_i$ , based on the gathered experimental data. All bone spheres  $b_i$  that contributed to  $F_s$  at this time step have a certain overlapping volume  $V_{b_i}$  (see III-A.2 for details).

We adjust the density  $\rho_{b_i}$  proportional to the force difference weighted by the normalized overlapping volume of all intersecting bone spheres  $b_j$ :

$$\rho_{b_i} \leftarrow \rho_{b_i} + (F_i - F_s) \cdot \frac{V_{b_i}}{\sum_{b_j} V_{b_j}} \quad (14)$$

The resulting density distribution yields a virtually identical force curve of the virtual recreation compared to the experimental data (MSE of 0.9) (see Fig. 8). Fig. 9 shows the generated density distribution: The middle portion (red) of the hip socket is harder than the surrounding tissue (bright green). This reflects the anatomy well, as the hip socket is surrounded by soft cartilage, whereas the middle has more exposed bone. Since the optimization is limited to a specific area, we cannot obtain a complete material model. Material properties for the white volume are undefined, thus we default to a density of 1. As future development, we plan to extend the material model to the whole acetabulum.

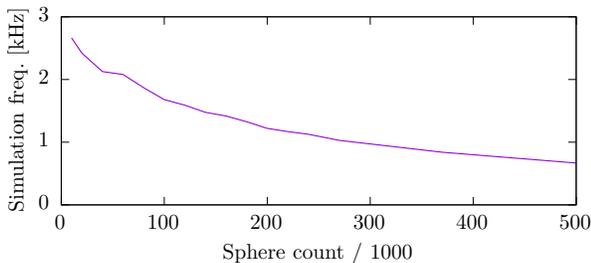


Fig. 11: Performance graph for a recording of realistic cutting interaction with varying degrees of packing accuracy. An average simulation frequency of 1 kHz can be maintained for sphere packings with less than 300k spheres.



Fig. 12: Our method supports arbitrary tools for material cutting. *Left*: A spherical burr removing caries in a tooth. *Right*: Branding iron used on a bunny.

## V. RESULTS

We implemented our algorithms in a library in C++ and CUDA using the CHAI 3D library [6] to communicate with regular haptic devices and an FRI-based library [15] to communicate with the KUKA robot. We decided to access our simulation library via a plugin for popular game engines. This facilitated implementing the algorithm in other game engines in the future. So far, we have successfully utilized the library in two major game engines, Unreal and Unity. The plugin runs asynchronously to the game engines to avoid the frame limit of the main rendering thread (see Fig. 10).

To test the performance of our algorithm, including the collision detection, force- and torque computation and the material removal, we recorded a typical cutting interaction from the dental simulator using a Phantom Omni device (Fig. 4 shows the result).

We ran the simulation on a single PC with a i7-4770K CPU, 16 GB DDR3 RAM, and two GTX 1080 GPUs. Obviously, the accuracy depends on the density of the sphere packing. Consequently, we tested several different sphere packings. Fig. 11 shows the preliminary results with a not yet finally optimized implementation. Our experiment shows that we can achieve haptic rates for nearly 300k spheres. Actually, a preliminary user test suggests that about 10k spheres are sufficient to accurately model a human tooth. The produced forces are very stable (see Fig. 13) and we have had no issues when applying our algorithm to powerful haptic devices such as a 140N capable industrial KUKA robot.

We applied our methods to a variety of applications, such as hip replacement, root canal access opening, as well as dental caries removal. In these cases, we worked

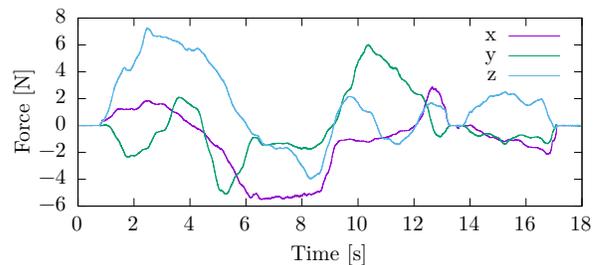


Fig. 13: Force recording while milling a tooth.

together closely with medical experts and could adapt parameters to the experts' requests. However, all the test cases use only simple instruments for material removal, but our method supports the cutting of arbitrary tools. Fig. 12 shows our method works with arbitrary tools, like the branding with a complex 3D model.

## VI. CONCLUSION

We have presented a novel collision detection & force feedback algorithm for haptics-enabled medical simulation with support for material removal. Our material removal model is, to the best of our knowledge, the first that supports continuous feedback for arbitrary tool models. Our method can easily be parallelized and runs completely on the GPU. Our results show that we achieve haptic rates even for very detailed models.

In the future, we plan to further optimize our GPU implementation and to further investigate the accuracy of the algorithm through extensive user tests in different medical scenarios. Furthermore, we are planning to develop more sophisticated optimization methods to determine the density distribution in the acetabulum based on our experimental data. Another avenue for future work is the support for deformable tools and materials. We also would like to evaluate the usefulness of our application in training users through rigorous user studies with experienced medical experts.

## ACKNOWLEDGEMENTS

This work is supported in part by the German Federal Federal Ministry of Education and Research (BMBF) under the grant 16SV8355.

## REFERENCES

- [1] E. Acosta and A. Liu. Real-time volumetric haptic and visual burrhole simulation. In *2007 IEEE Virtual Reality Conference*, pp. 247–250, March 2007. doi: 10.1109/VR.2007.352492
- [2] M. Arbabtafti, M. Moghaddam, A. Nahvi, M. Mahvash, B. Richardson, and B. Shirinzadeh. Physics-based haptic simulation of bone machining. *IEEE Transactions on Haptics*, 4(1):39–50, January 2011. doi: 10.1109/TOH.2010.5
- [3] O. R. Astley and V. Hayward. Design constraints for haptic surgery simulation. In *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*, vol. 3, pp. 2446–2451 vol.3, April 2000. doi: 10.1109/ROBOT.2000.846394
- [4] J. F. Blinn. A generalization of algebraic surface drawing. *ACM Trans. Graph.*, 1(3):235–256, July 1982. doi: 10.1145/357306.357310
- [5] V. B. Chial, S. Greenish, and A. M. Okamura. On the display of haptic recordings for cutting biological tissues. In *Proceedings 10th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems. HAPTICS 2002*, pp. 80–87, March 2002. doi: 10.1109/HAPTIC.2002.998944
- [6] F. Conti, F. Barbagli, D. Morris, and C. Sewell. Chai: An open-source library for the rapid development of haptic scenes. In *IEEE World Haptics, Pisa, Italy, March 2005*, March 2005.
- [7] C. A. Coulomb. Essai sur une application des regles de maximis et minimis a quelques problemes de statique relatifs a l'architecture (essay on maximums and minimums of rules to some static problems relating to architecture). 1973.
- [8] Statistisches Bundesamt. (Destatis). Fallpauschalenbezogene krankenhausstatisik (drg-statistik) operationen und prozeduren der vollstationären patientinnen und patienten in krankenhäusern (4-steller), October 2017.
- [9] I. Fazal and M. N. Karsiti. Needle insertion forces for haptic feedback device. In *2009 IEEE Symposium on Industrial Electronics Applications*, vol. 1, pp. 20–22, Oct 2009. doi: 10.1109/ISIEA.2009.5356495
- [10] B. Gonenc and H. Gurocak. Haptic interface with hybrid actuator for virtual needle insertion and tissue cutting. In *2012 IEEE Haptics Symposium (HAPTICS)*, pp. 451–455, March 2012. doi: 10.1109/HAPTIC.2012.6183830
- [11] Guanyang Liu, Yuru Zhang, D. Wang, J. Hao, P. Lu, and Y. Wang. Cutting force model of dental training system. In *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 925–929, Aug 2005. doi: 10.1109/IROS.2005.1545612
- [12] J. S. Henn, G. M. Lemole, M. A. T. Ferreira, L. F. Gonzalez, M. Schornak, M. C. Preul, and R. F. Spetzler. Interactive stereoscopic virtual reality: a new tool for neurosurgical education. *Journal of Neurosurgery*, 96(1):144 – 149, 2002.
- [13] H. Q. Huy Viet, T. Kamada, and H. T. Tanaka. An algorithm for cutting 3d surface meshes. In *18th International Conference on Pattern Recognition (ICPR'06)*, vol. 4, pp. 762–765, Aug 2006. doi: 10.1109/ICPR.2006.233
- [14] Kimin Kim, Y. Park, and J. Park. Volume-based haptic model for bone-drilling. In *2008 International Conference on Control, Automation and Systems*, pp. 255–259, Oct 2008. doi: 10.1109/ICCAS.2008.4694560
- [15] S. Knopp, M. Lorenz, L. Pelliccia, and P. Klimant. Using industrial robots as haptic devices for vr-training. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 607–608, March 2018. doi: 10.1109/VR.2018.8446614
- [16] Y. . Lim and S. De. On the use of meshfree methods and a geometry based surgical cutting algorithm in multimodal medical simulations. In *12th International Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2004. HAPTICS '04. Proceedings.*, pp. 295–301, March 2004. doi: 10.1109/HAPTIC.2004.1287212
- [17] M. Mahvash and A. M. Okamura. A fracture mechanics approach to haptic synthesis of tissue cutting with scissors. In *First Joint Eurohaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems. World Haptics Conference*, pp. 356–362, March 2005. doi: 10.1109/WHC.2005.8
- [18] W. A. McNeely, K. D. Puterbaugh, and J. J. Troy. Six degree-of-freedom haptic rendering using voxel sampling. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '99*, pp. 401–408. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 1999. doi: 10.1145/311535.311600
- [19] D. Wang, Yuru Zhang, Yuhui Wang, Y. . Lee, Peijun Lu, and Yong Wang. Cutting on triangle mesh: local model-based haptic display for dental preparation surgery simulation. *IEEE Transactions on Visualization and Computer Graphics*, 11(6):671–683, Nov 2005. doi: 10.1109/TVCG.2005.97
- [20] Weiyu Zheng, P. X. Liu, and Chengliang Li. Interactive simulation of progressive cutting model. In *2006 6th World Congress on Intelligent Control and Automation*, vol. 1, pp. 386–390, June 2006. doi: 10.1109/WCICA.2006.1712338
- [21] R. Weller and G. Zachmann. Inner sphere trees and their application to collision detection. *Virtual Realities: Dagstuhl Seminar 2008*, pp. 181–201, 01 2008. doi: 10.1007/978-3-211-99178-7\_10
- [22] Z. Yan, D. Wu, Y. Huang, and Z. Du. A simulation method of soft tissue cutting with haptics. In *2012 IEEE International Conference on Mechatronics and Automation*, pp. 2152–2157, Aug 2012. doi: 10.1109/ICMA.2012.6285676
- [23] Yan Zhou, Dongmei Zheng, and Dengming Hu. Research of virtual cutting technology based on haptic feedback. In *2009 IEEE 10th International Conference on Computer-Aided Industrial Design Conceptual Design*, pp. 2197–2201, Nov 2009. doi: 10.1109/CAIDCD.2009.5375139
- [24] Zhonghua Lu, G. Sankaranarayanan, D. Deo, Dingfang Chen, and S. De. Towards physics-based interactive simulation of

electrocautery procedures using physx. In *2010 IEEE Haptics Symposium*, pp. 515–518, March 2010. doi: 10.1109/HAPTIC.2010.5444609

- [25] C. B. Zilles and J. K. Salisbury. A constraint-based god-object method for haptic display. In *Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots*, vol. 3, pp. 146–151 vol.3, Aug 1995. doi: 10.1109/IROS.1995.525876