



Published in final edited form as:

Int J Med Robot. 2008 June ; 4(2): 131–138. doi:10.1002/rcs.185.

Development of the VBLaST™: A Virtual Basic Laparoscopic Skill Trainer

Anderson Maciel, Ph.D.,

A postdoctoral research associate in Department of Applied Computation, Institute of Informatics, Federal University of Rio Grande do Sul, Porto Alegre, Brazil

Youquan Liu, Ph.D.,

A postdoctoral research associate in Department of Mechanical, Aerospace and Nuclear Engineering Rensselaer Polytechnic Institute, Troy, USA

Woojin Ahn, B.Eng.,

Ph.D. student in Department of Mechanical Engineering, Korea Advanced Institute of Science and Technology, Daejeon, Korea

T. Paul Singh, M.D. Ph.D.,

The director of the Minimally Invasive Surgery Center for Training and Education, Albany Medical College, Albany, NY, USA

Ward Dunnican, M.D., and

An Assistant Professor of Surgery in the Minimally Invasive Surgery Center for Training and Education, Albany Medical College, Albany, NY, USA

Suvranu De, Sc.D.

An associate professor in Department of Mechanical, Aerospace and Nuclear Engineering, Rensselaer Polytechnic Institute, Troy, USA

Abstract

Background—Minimally invasive surgery has become more and more important in modern hospitals. In this context, increasingly more surgeons need to be trained to master the necessary laparoscopic surgical skills. The Fundamentals of Laparoscopic Surgery (FLS) training tool box has now been adopted by the Society of Gastrointestinal Endoscopic Surgeons (SAGES) as an official training tool. While useful, there are major drawbacks of such physical tool boxes including the need to constantly replace training materials, inability to perform objective quantification of skill and the inability to easily adapt to training surgeons on surgical robots such as the da Vinci® which provides high resolution stereo visualization.

Methods—To overcome the limitations of the FLS training tool box, we have developed a Virtual Basic Laparoscopic Skill Trainer (VBLaST™) system, which will allow trainees to acquire basic laparoscopic skill training through the bimanual performance of four tasks including peg transfer, pattern cutting, ligating loop and suturing. A high update rate of about 1 kHz is necessary to ensure continuous haptic interactions and smooth transitions.

Results—The outcome of this work is the development of an integrated visio-haptic workstation environment including two Phantom® Omni™ force feedback devices and a 3D display interface from Planar Systems, Inc whereby trainees can practice on virtual versions of the FLS tasks in 2D

as well as in 3D, thereby allowing them to practice both for traditional laparoscopic surgery as well as that using the da Vinci® system. Realistic graphical rendering and high fidelity haptic interactions are achieved through several innovations in modeling complex interactions of tissues and deformable objects.

Conclusions—Surgical skill training is a long and tedious process of acquiring fine motor skills. Even when complex and realistic surgical trainers with realistic organ geometries and tissue properties, which are currently being developed by academic researchers as well as the industry, mature to the stage of being routinely used in surgical training, basic skill trainers such as VBLaST™ will not lose their relevance. It is expected that residents would start on trainers such as VBLaST™ and after reaching a certain level of competence would progress to the more complex trainers for training on specific surgical procedures. In this regard, the development of the VBLaST™ is highly significant and timely.

Keywords

Computer Graphics; Interaction techniques; Minimally Invasive Surgery; User Interfaces—Haptic I/O

1. Introduction

With its many advantages of minimal postoperative pain, few postoperative adhesions, minimized blood loss, low risk of surgical complications, short hospital stay, and early return to normal activities, minimally invasive surgery (MIS) has revolutionized general surgery in the past decade in treating both malignant and benign diseases. Surgical skill training is a long and tedious process of acquiring fine motor skills. The Society of American Gastrointestinal Endoscopic Surgeons (SAGES) has recently introduced the Fundamentals of Laparoscopic Surgery (FLS) training tool box (1), consisting of a box covered by an opaque membrane through which two 12 mm trocars are placed on either side of a 10 mm zero-degree laparoscope connected to a video monitor. Inside the box five premanufactured tasks including peg transfer, pattern cutting, ligating a loop and suturing can be performed. The mechanical tool box shown in FIGURE 1 is similar to the FLS system and is used in training the same set of tasks.

Potential drawbacks of such mechanical toolbox systems are: 1) the training material has to be constantly replaced after it is cut or sutured, 2) objective quantification of skill is difficult and 3) unless specialized and expensive stereo cameras are used, the images obtained are in 2D and therefore these systems are not suitable to train for robotic surgical systems such as the da Vinci®. To overcome these problems, we have developed, for the first time, a virtual basic laparoscopic skill trainer (VBLaST™) whereby tasks such as the ones available in the FLS system, may be performed on the computer. The development of the VBLaST™ is not just an exercise in adapting existing technology to a few procedures but involves several innovations in modeling complex interactions of tissues and deformable objects.

Most existing work on laparoscopic skill trainers focus on the metrics for the student performance measurement (2). They often present a set up of the real toolbox for the participants to perform a number of tasks. Several surgical simulators have already been validated for resident and surgeon training (3). Some are limited to 2D video-based training (4). The most notable of these is the McGill Inanimate System for Training and Evaluation of Laparoscopic Skills (MISTELS). This system involves performing laparoscopic simulated tasks on inanimate objects within a box trainer. Such tasks have been extensively tested and validated as an effective training tool. A reduced set of the MISTELS tasks has been adapted to the FLS training box. FIGURE 2. illustrates the four tasks that can be performed in the box.

In (5), the authors show that the use of force-feedback results in a significantly improved training transfer compared to training without force feedback. The use of stereo vision has also been shown to benefit performance of surgical tasks, while virtual environments have significant advantage over the mechanical toolbox systems (6), (7). The Minimally Invasive Surgery Trainer - Virtual Reality (MIST-VR) program is an example of these kinds of trainers. Potential drawbacks of these trainers include their high cost, lack of tactile feedback, and lack of stereoscopic visualization.

In the US and in Europe several other educational institutes and centers are involved in developing surgical simulation technology including the Stanford University Medical Media and Information Technologies (SUMMIT) and the National Bio-Computation Center; Center for Integration of Medicine and Innovative Technology (CIMIT) at Harvard Medical School, Human Interface Laboratory and Biorobotics Laboratory at University of Washington; National Area Medical Simulation Center at Uniformed Services University (USU); the Center for Human Simulation (University of Colorado); the Biomedical Interactive Technology Center (Georgia Institute of Technology); the Center for Robotics and Computer Assisted Surgery (Carnegie Mellon University); the KISMET group in Germany, the LASSO group at ETH and Bristol Medical Simulation Centre in the University of Karlsruhe, Germany. In addition to academic research and development, several companies such as Boston Dynamics, Immersion Medical (includes products developed by HT Medical), Melerit, Mentice, MedSim, MusculoGraphics, Novint, Reachin Technologies and Virtual Presence have tried or are currently trying to develop medical simulators. However, to the best of our knowledge, none of them are developing a virtual version of the FLS trainer. In that sense VBLaST™ is unique.

Surgical skill training is a long and tedious process of acquiring fine motor skills (8). Even when complex and realistic surgical trainers with realistic organ geometries and tissue properties, such as the ones being currently developed, are routinely used in surgical training, basic skill trainers such as VBLaST™ will not lose their relevance. It is expected that residents would start on trainers such as VBLaST and after reaching a certain level of competence would move on to the more complex trainers for training on specific surgical procedures.

In Section 2 we present the various computational algorithms developed to achieve a very high level of visual and haptic realism. In Section 3 we present computational results followed by concluding remarks in Section 4.

2. Computational Methods

Consistent with the FLS system we have developed the following tasks within VBLaST™ (see FIGURE 3):

Peg transfer (FIGURE 3a): Each of 6 virtual rings may be lifted from a virtual pegboard with the left hand, transferred to the right hand and placed on another pegboard.

Pattern cutting (FIGURE 3b): A 4 cm diameter premarked circular pattern may be cut out of a 10 cm × 10 cm piece of virtual gauze suspended between alligator clips.

Ligating loop (FIGURE 3c): A 3-dimensional tubular structure is presented in space. Using bimanual manipulation a virtual loop is securely fashioned about a pre-drawn line on the tubular structure.

Suturing (FIGURE 3d): A virtual suture is tied using either an intracorporeal or extracorporeal knot, using 3-dimensional bimanual manipulation with a curved needle.

In Section 2.1 we present the hardware system that we have developed. In sections 2.2 and we present innovations in collision detection and physical modeling that have contributed to the development of the VBLaST™ system.

2.1. Interface

A stereoscopic visio-haptic workstation has been developed including two Phantom® Omni™ force feedback devices and a 3D display interface from Planar Systems, Inc. There is provision to turn the stereoscopic display on (VBLaST2D) and off (VBLaST3D) at will for the same task. FIGURE 4 shows the whole system. During the simulation, the participants use the Phantoms to control the virtual surgical instruments to interact with the virtual training materials. On the virtual side, the instruments represented in the system are tissue graspers, suture graspers, scissors and the ligating loop. On the hardware side, we have developed a plug-and-play interface that allows us to instrument real surgical hand tools and attach them to the Phantom stylus gimbal using 6.3 mm headphone jacks. For this, we take advantage of the 6.3 mm jacks available when the Omni stylus cover is removed. Such interface converts the opening and closing of the tool handles into measurable electric potentials using potentiometers. The software interface, in turn, converts the voltage read through a USB interface into angles that define the orientation of instruments movable parts.

2.2. Collision detection and Physics-based modeling

For any interactive 3D application, collision detection and dynamic response computation between the tools and models in the environment are essential. That is necessary so that one can pick, move or even cut objects at the same time that effort responses are generated.

Collision handling is the key to the whole interactive simulation system, and due to the high update rate required for the haptics device (around 1000 Hz), the algorithm must be very efficient. An efficient dynamic point algorithm is used for line-based collision detection and response, readers can refer to (9) for more details. The basic idea of the algorithm is that only one point is used for collision detection and response, but this point is constrained to lie on a line at a location which is instantaneously closest to the mesh. The line is represented by its end points *A* and *B* and a dynamic point *P*, which is chosen to be the closest point on the line to any potentially colliding triangle *t* (FIGURE 5). The dynamic point position on the line is updated at haptic frequencies and hence to the user, due to inherent latencies of the order of 1 ms in the human haptic system, it is virtually indistinguishable from a line.

At the same time, various physics-based algorithms are used for interaction with the various virtual objects throughout the different tasks. The methods used in each task are detailed below.

Peg-transfer—The peg-transfer task relies on implementing correct rigid body dynamics (FIGURE 3.a). Momentum and torques are computed depending on interaction with the tools and pegs. Collisions produce impulses, which change the trajectory and spinning when the rings are released and provide force-feedback when they are held by a tool. For the other three tasks, however, deformations are present. We use mass-spring-damper models to simulate deformations and penalty forces to handle contact with deformable objects. Sponge, gauze, suture, ligation are all modeled as deformable objects.

Mesh-mesh collision detection is necessary only in the peg-transfer task. As only rigid bodies are present, we use a boundary volume hierarchy to check triangles intersections only with the cylinder in the vicinity of the tool.

Pattern-cutting—The gauze model for the pattern cutting task (FIGURE 3b) is a textured mass-spring square membrane attached at the corners. It undergoes deformation as it collides

with the scissor shaft and blades. Whenever the same triangle is in contact with the two blades and the blades are closing, the springs around the triangle are removed, causing the gauze to open up at that spot. Moreover, as with the real pattern cutting task, bimanual cutting is also simulated. The user can use a grasper on the left hand to grab the gauze and ease the cutting with the right hand by placing the gauze in the appropriate angle.

As graspers and scissors are composed of basically three long parts (two blades or claws and one shaft), three dynamic points are used for each tool to detect collisions with the gauze model during the pattern-cutting task: one for the instrument shaft; two for the grasper claws or scissor blades.

As for collision response, penalty methods are used to deform the model and calculate force-feedback. Moreover, collision information, e.g. contacting triangles, and blades/claws angular variations are used to decide about when to cut and when to grab the mesh. Typically, closing blades that contact the same triangle causes the triangle to be cut. For the grasper, the mesh is grabbed at the triangle that intersects the two claws when they are completely closed.

Suturing—In the suturing task (FIGURE 3d), collision detection for the needle-tissue interaction is performed using a needle discretized into line segments (more details below). This works straightforward while the needle is completely outside the tissue. When the tip of the needle is in contact with the tissue, the contact force is computed. If this force exceeds a threshold value, the tissue is penetrated. After the first penetration, and while a portion of the needle is inside the tissue, physics-based constraints are applied. The skeleton generated by the discretization is used to define local coordinate systems at each node. The coordinate systems are oriented according to the tangent at nodal positions, as in FIGURE 6a. The displacement of each node is then allowed freely (only with friction) in the direction of its respective z-axis. Any other displacement is penalized. The penalty force is implemented as a spring attracting the node to its projection on the z-axis (FIGURE 6b).

The suture-thread, in turn, is modeled as a 1D mass-spring structure. The thread is discretized into a set of mass points. To allow the thread to withstand axial tension and smooth bending with a finite curvature much like the real suture, each mass point is connected with three front and three rear mass points by springs as shown in FIGURE 7. Furthermore, we observed that smooth bending curvature is not achieved if the number of mass points is large which leads to a sharp edge at the picked region.

Thus, mass points are also down-sampled from the original discretization. Deformation of the suture is then computed using the original mass points, and the rest of the nodes are interpolated by a spline curve to update new positions for graphics rendering.

Whenever a thread is involved, self-collisions may occur and must be detected. Self-collision of the thread is checked using the line segments connecting the mass-points (a_i and b_i in FIGURE 8). A self-collision occurs if the shortest distance \vec{ab} between two line segments $\vec{a_1a_2}$ and $\vec{b_1b_2}$ is lesser than twice the radius of the thread (10). This approach leads to the non-trivial problem that the colliding line segments can pass through each other without detection as the radius of the thread is small. Thus, we use a small time step synchronizing with haptic refresh rate not to miss any intersections, which in practice gives accurate results with the dynamics of the application. Self-collision response, in turn, is based on impulse and repulsion force (11). Impulse alters relative velocity of a pair of collided line segments to zero by perfect inelastic collision. Repulsion force allows the collided line segments to move in opposite directions to avoid the collision.

The direction of the repulsion force is given by \vec{ab} . Self-collision of the thread allows to tie a knot (FIGURE 9). When the knot is tight enough to stay in place it is just fixed at that position.

Ligating loop—Interestingly, the above strategy (suture modeling) does not work for the ligating loop problem (FIGURE 3c) since we have to deal with an elastic loop whose shape is hard to maintain without using stiff springs. Highly stiff springs result in instability when the loop collides with the sponge or is tightened. We have developed a new strategy combining a rigid loop structure with a simple mass spring, anchoring the mass points to positions on the rigid loop with springs as shown in FIGURE 10. The yellow nodes represent the loop's original position and the green ones the deformed position. Between each yellow-green node pair there is a spring, represented in red in the figure. The yellow nodes move with the tool. The green ones, in turn, are attracted to the new positions of the yellow ones. When the loop collides with another object, as in FIGURE 10b, the red springs allow the deformation required to avoid penetration, while attracting the green nodes so that the loop shape is restored when the collision ceases. As only the loop with the green nodes is displayed, the ligating-loop looks fully deformable, but keeps track of its original shape through the use of the yellow nodes and red springs. This model provides natural looking behavior with very high efficiency.

As the loop geometry is represented as a chain of line segments, one dynamic point is applied to each segment for collision detection with the sponge model. This is analogous to the method used for graspers and scissors for both collision detection and response, except that the loop is deformable. The loop deformation is however independent of the collision method.

3. Results

We have implemented an interactive PC-based surgical simulation framework and tested it on an Intel(R) Core™ 2 Quad 2.66GHz machine with a GeForce 8800 GTX graphics card. Customized vertex and pixel programs are used for textured shading, which provides state of the art interactive graphics realism. This simulator utilizes two force feedback devices to provide bimanual surgical tasks, and a dual polarized monitor based stereo vision system by Planar Systems, Inc.

TABLE 1 presents performance information for the 4 tasks. The system runs at haptic frequencies and displays graphics at 60 Hz, being 30 Hz for each monitor/eye. Such frequencies provide smooth graphical displays with no flickering and vibration-free haptics rendering. The haptic frequencies are essentially dependent on the performance of the collision detection.

The algorithm we use is based on line collision and depends only on the numbers of lines we are checking for collision. FIGURE 11 shows an example of cutting where three lines are used (the long shaft of the instrument, the upper and lower blades). As the task is bimanual, we eventually have six lines to check. In the example of FIGURE 12, twenty-eight lines are used, which causes the frequency to drop considerably but still provides smooth contact feedback of the deformable instrument with a deformable object. The physics-based simulation frequencies depend on the model complexity and integration methods and vary considerably, but the material parameters are tuned so that it is high enough to deliver very dynamic and responsive deformations and contacts.

4. Concluding remarks

We have developed, for the first time, a virtual basic laparoscopic skill training (VBLaST™) simulator which allows training for both traditional minimally invasive surgical procedures as well as procedures on the da Vinci® robotic surgery system. The system includes realistic stereo graphics interface and bimanual haptics interface to provide an immersive training

environment. In comparison with existent simulators, the VBLaST™ system brings better graphics realism through extensive use of customized GPU pipelines for texturing and lighting, high performance force-feedback through optimized collision detection and response algorithms, and 30 Hz per eye stereo vision even with complex deformable geometries.

Obvious advantages of the VBLaST™ over real mechanical training tool boxes include unlimited training material, easy portability, customized training regimens and objective, automatic and immediate assessment of surgical skills. Moreover, previous works have already studied and confirmed the benefits of haptics and stereo vision realism for an effective training tool (5), (6), (7).

We are now in the process of developing assessment metrics and performing validation studies of the VBLaST™ system. Isolated groups of trainees will practice on the physical box and on VBLaST™ for comparison. Also, a glove-based tracking system is being developed to allow us to capture the hand and finger motions of the trainees for the purpose of objective quantification of skill training.

Acknowledgments

Grant

Partial support of this work through the R01 EB005807-01 grant from NIH/NIBIB and the PDJ program of CNPq are gratefully acknowledged.

References

1. Peters JH, Fried GM, Swanstrom LL, Soper NJ, Sillin LF, Schirmer B, Hoffman K. SAGES FLS Committee. Development and validation of a comprehensive program of education and assessment of the basic fundamentals of laparoscopic surgery. *Surgery* 2004;135(1):21–27. [PubMed: 14694297]
2. Cotin, S.; Stylopoulos, N.; Ottensmeyer, MP.; Neumann, PF.; Rattner, D.; Dawson, S. Metrics for laparoscopic skills trainers: The weakest link!. Proceedings of the 5th International Conference on Medical Image Computing and Computer-Assisted Intervention-Part I (MICCAI '02); Springer-Verlag: London, UK. 2002. p. 35-43.
3. Satava, RM. *Surgical Endoscopy*. Vol. 15. Springer: New York; 2004. Accomplishments and challenges of surgical simulation; p. 232-241.
4. Simulab Corporation [home-page on the Internet]. Surgical simulators and medical trainers: Trauma, laparoscopic, orthopedic, and general; c2004. Available from <http://www.simulab.com>.
5. Kim, HK.; Rattner, DW.; Srinivasan, MA. Lecture Notes in Computer Science: Medical Image Computing and Computer-Assisted Intervention - MICCAI 2003. Vol. 2878. Springer Berlin: Heidelberg; The Role of Simulation Fidelity in Laparoscopic Surgical Training; p. 1-8.
6. Downes, M.; Cavusoglu, M.; Gantert, W.; Way, L.; Tendick, F. Medicine Meets Virtual Reality. San Diego, CA: 1998. Virtual Environments for training critical skills in laparoscopic surgery; p. 316-322.
7. Tendick F, Downes M, Goktekin T, Cavusoglu MC, Feygin D, Wu X, Eyal R, Hegarty M, Way LW. A virtual environment testbed for training laparoscopic surgical skills. Presence: Teleoper. Virtual Environ 2000;9(3):236–255.
8. Kneebone R. Simulation in surgical training: educational issues and practical implications. *Med Educ* 2003;37(3):267–277. [PubMed: 12603766]
9. Maciel, A.; De, S. Medicine Meets Virtual Reality. Long Beach, CA: 2008. An efficient dynamic point algorithm for line-based collision detection in real time virtual environments involving haptics. (to appear)
10. Eberly, DH. 3D game engine design: a practical approach to real-time computer graphics. San Francisco: Morgan Kaufmann Publishers Inc.; 2000.
11. Bridson R, Fedkiw R, Anderson J. Robust treatment of collisions, contact and friction for cloth animation. *ACM Trans. Graph* 2002;21:594–603.

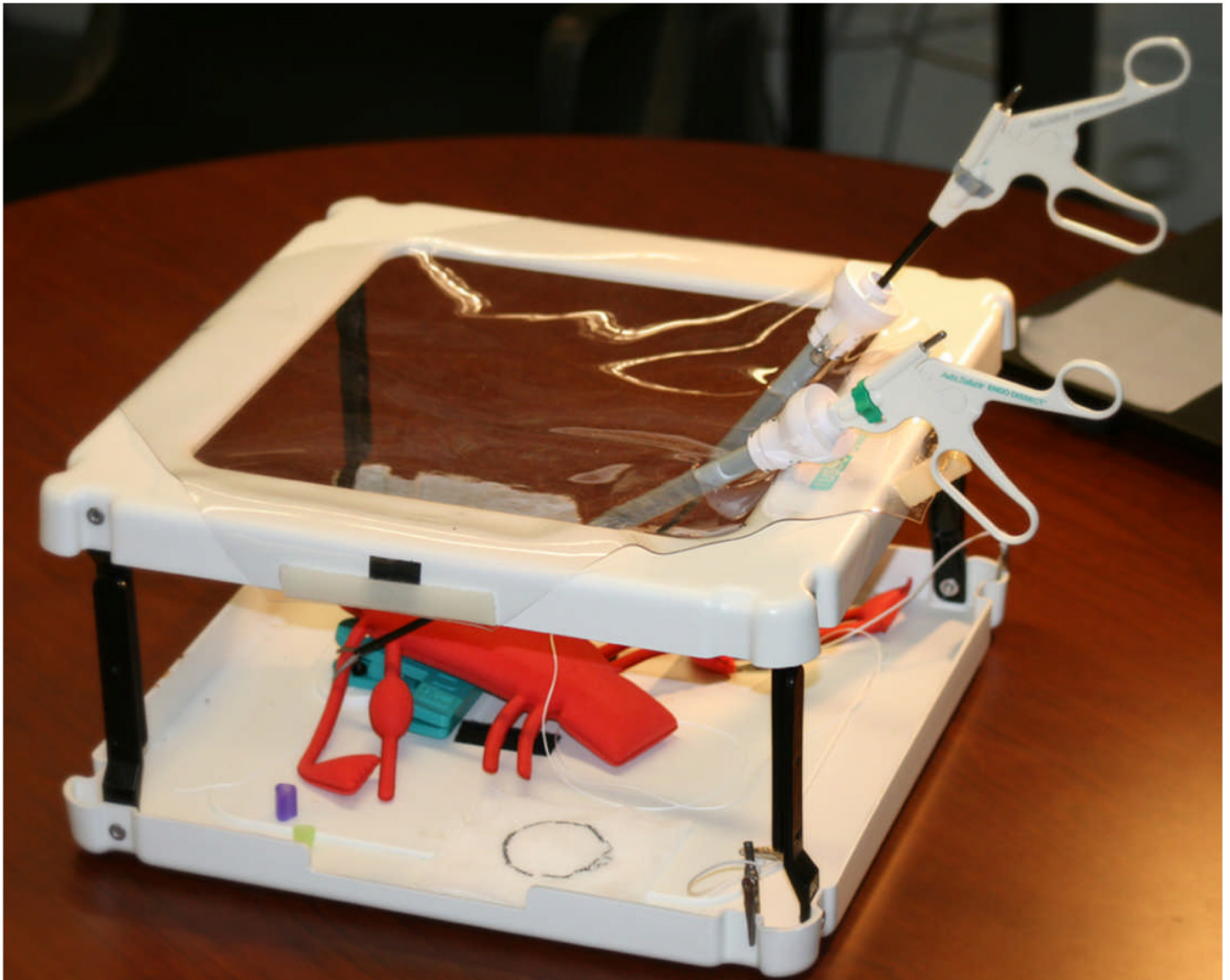


FIGURE 1.
Mechanical training toolbox.

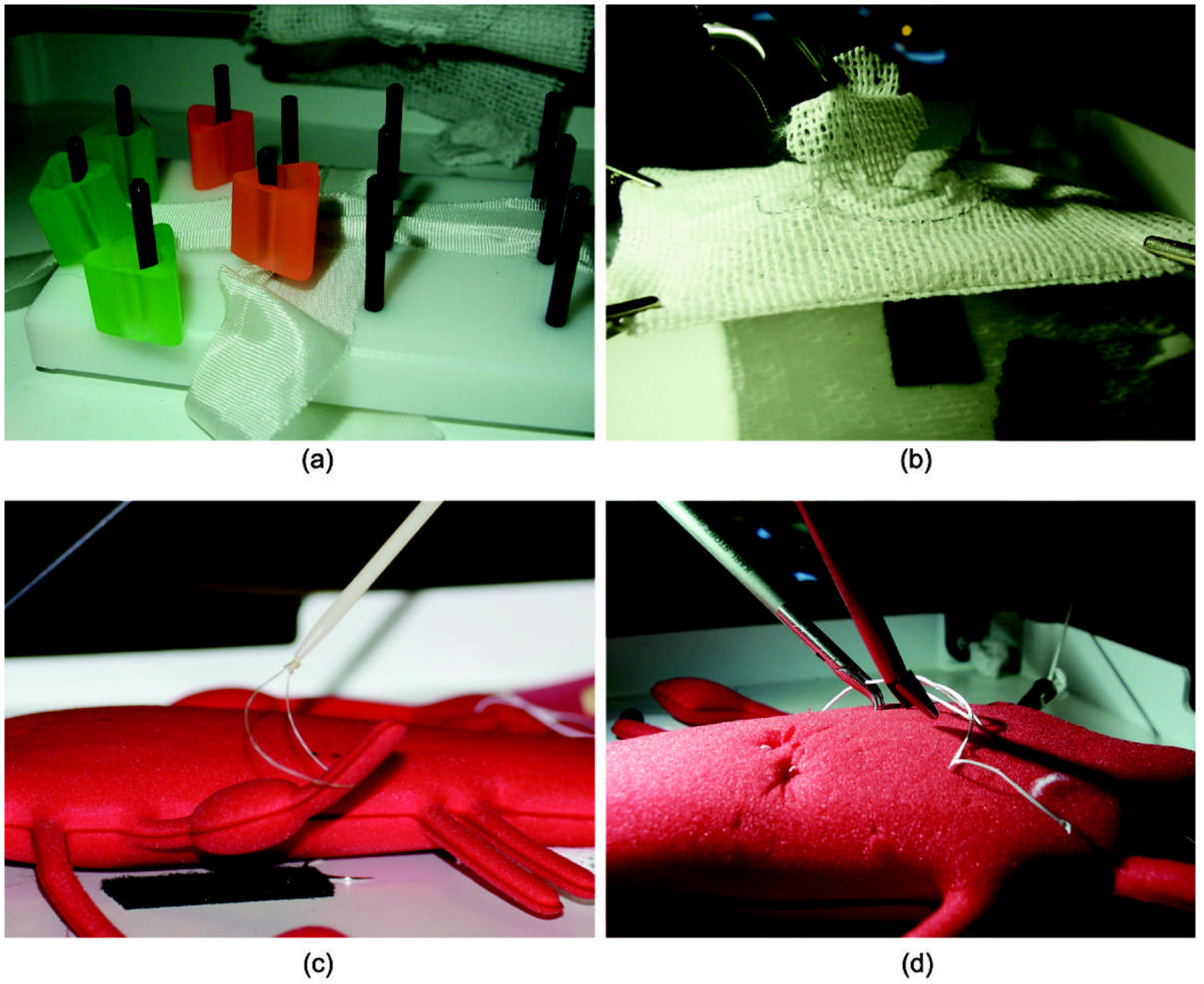


FIGURE 2.
The 4 MISTELS tasks used in the FLS toolbox: (a) peg transfer; (b) circular pattern cutting;
(c) ligating loop; (d) suturing.

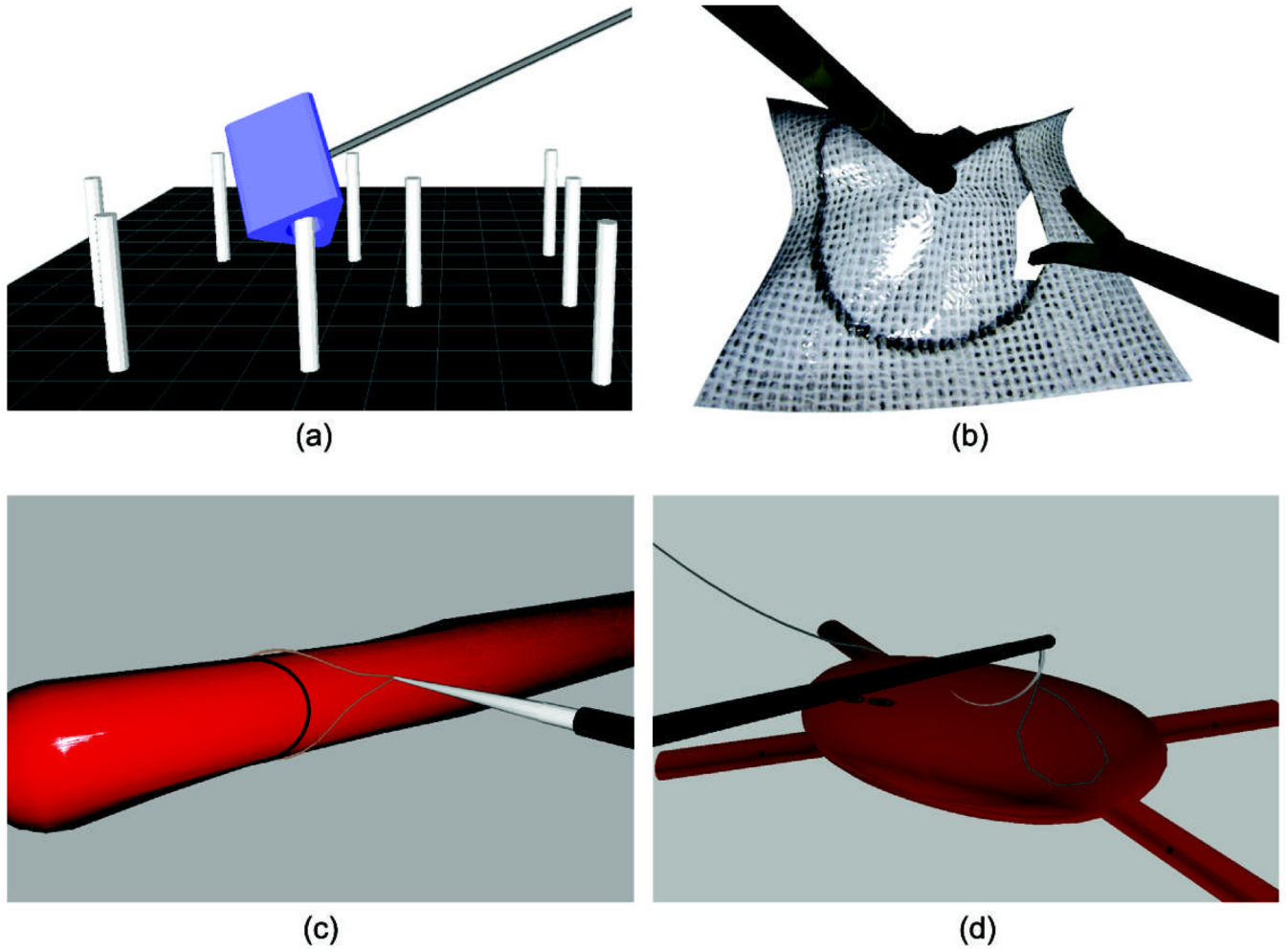


FIGURE 3. The 4 VBLaSTTM tasks: (a) peg transfer; (b) circular pattern cutting; (c) ligating loop; (d) suturing.

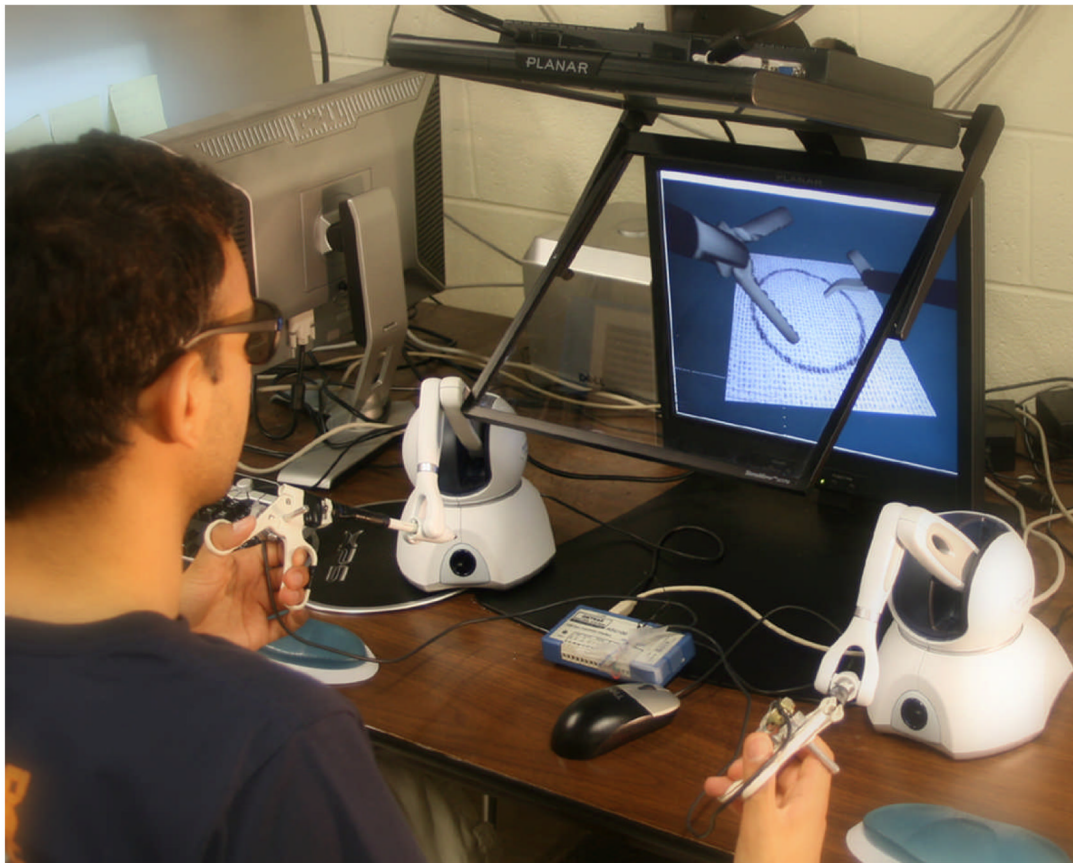


FIGURE 4.
The VBLaST™ workstation.

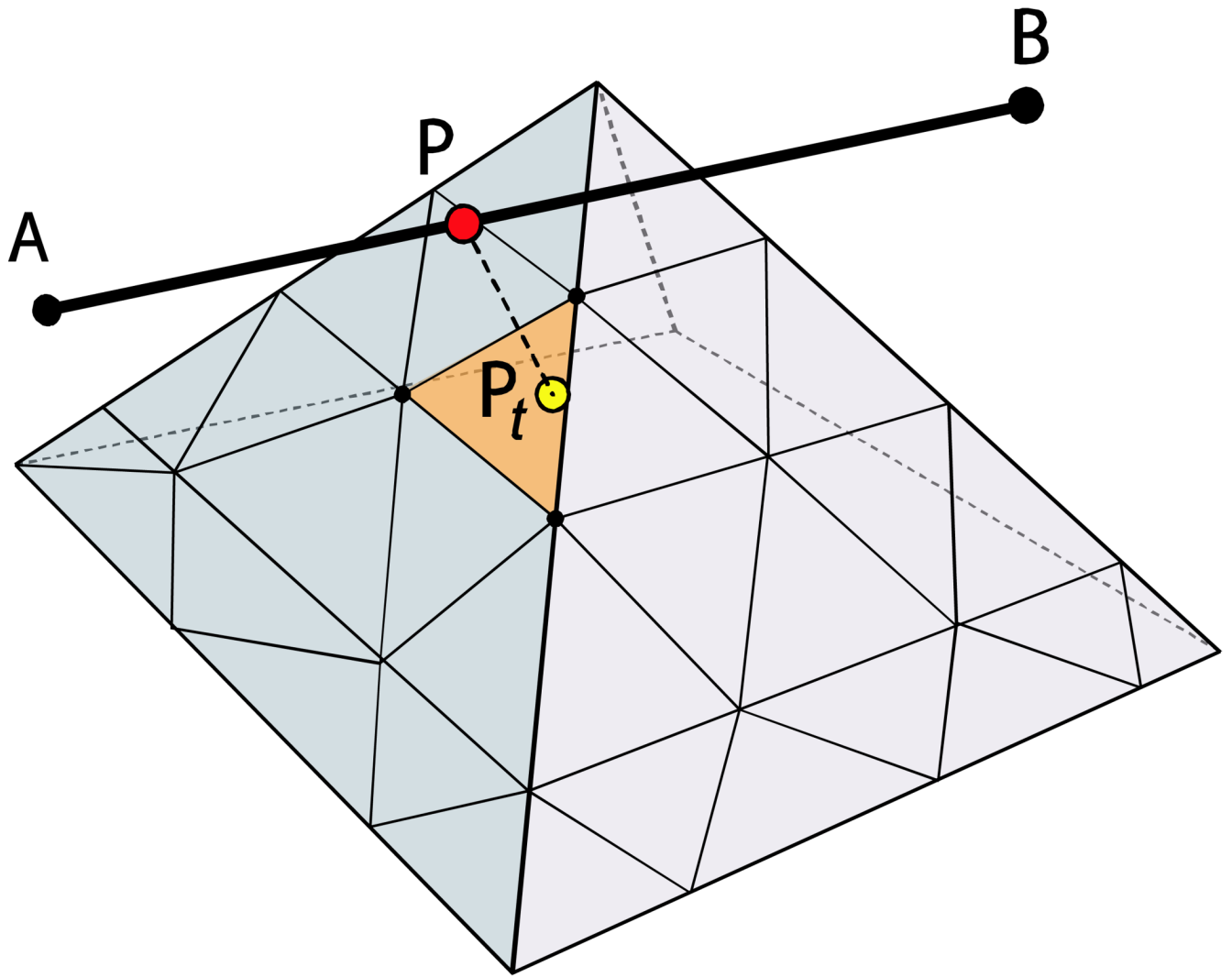


FIGURE 5. The dynamic point algorithm for a line interacting with a 3D mesh. Current dynamic point P and the closest point in the mesh P_t .

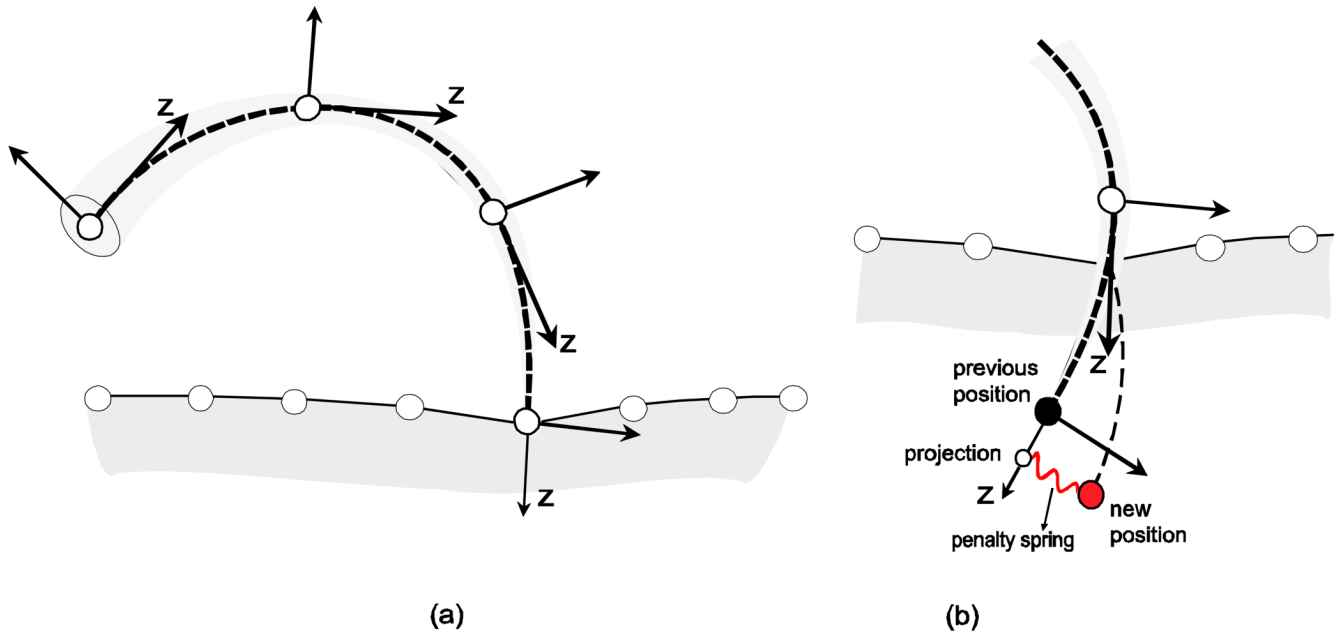


FIGURE 6. Needle-tissue interaction. (a) the needle is discretized into segments for collision detection; (b) the local frame of each segment is used to calculate resistance inside the tissue.

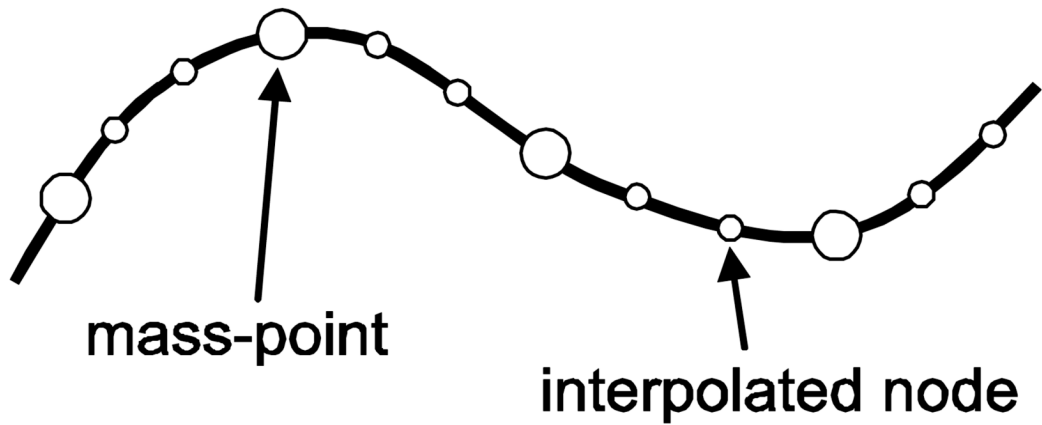
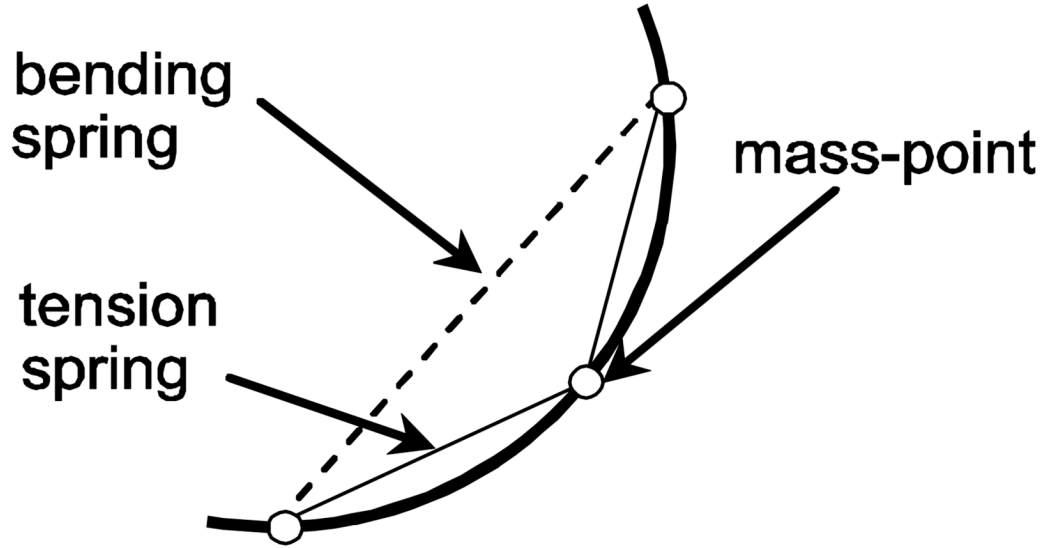
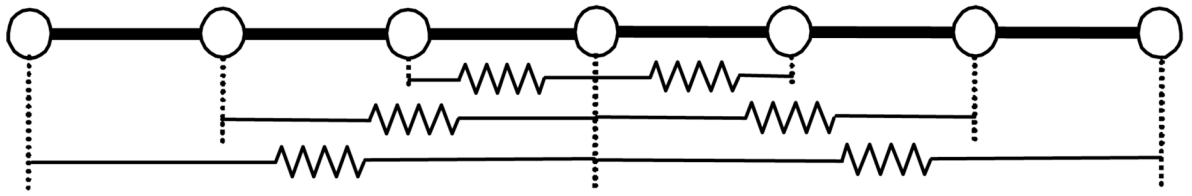


FIGURE 7.
The suture model using a 1D mass-spring.

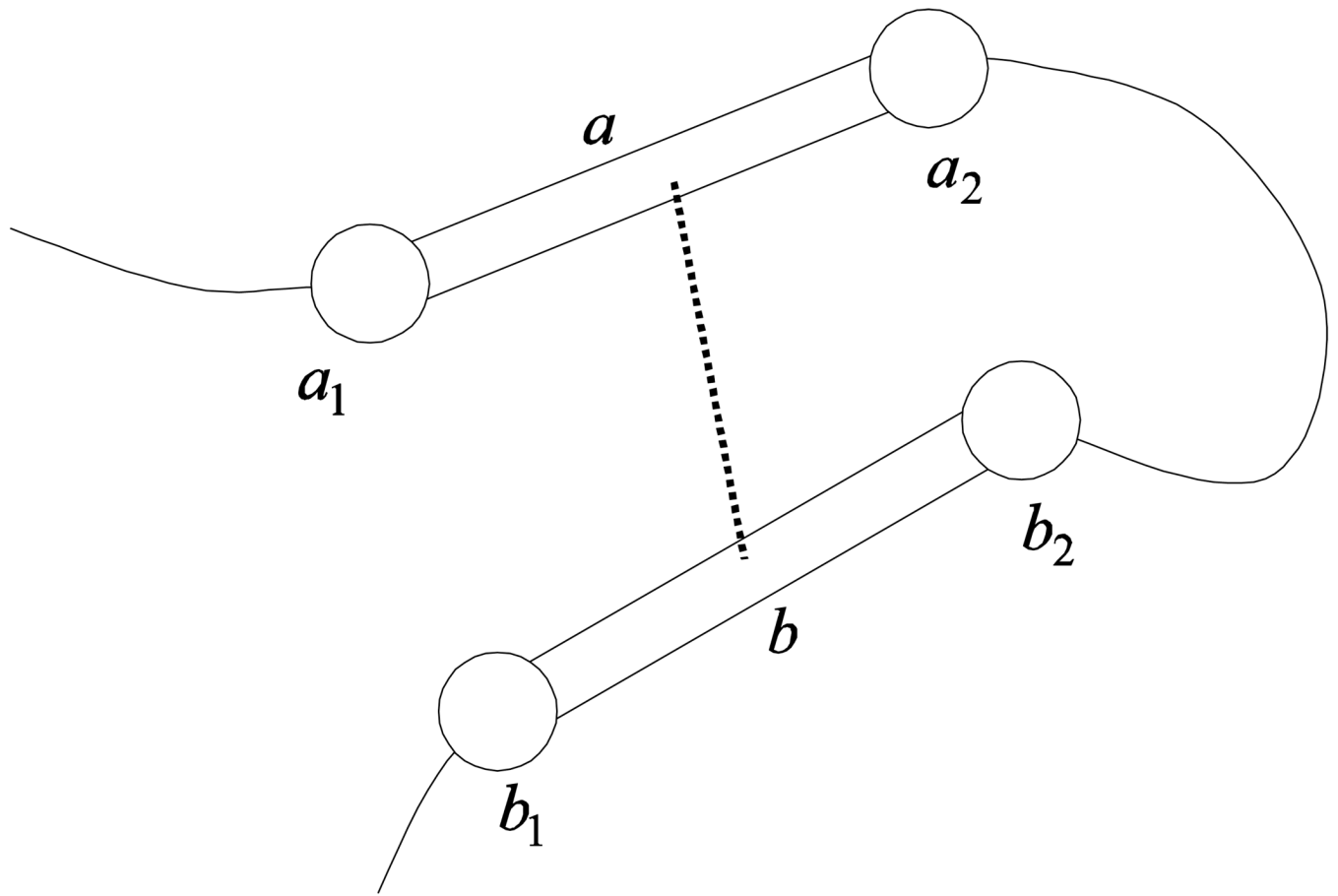


FIGURE 8.
Self-collision and response for the thread model.

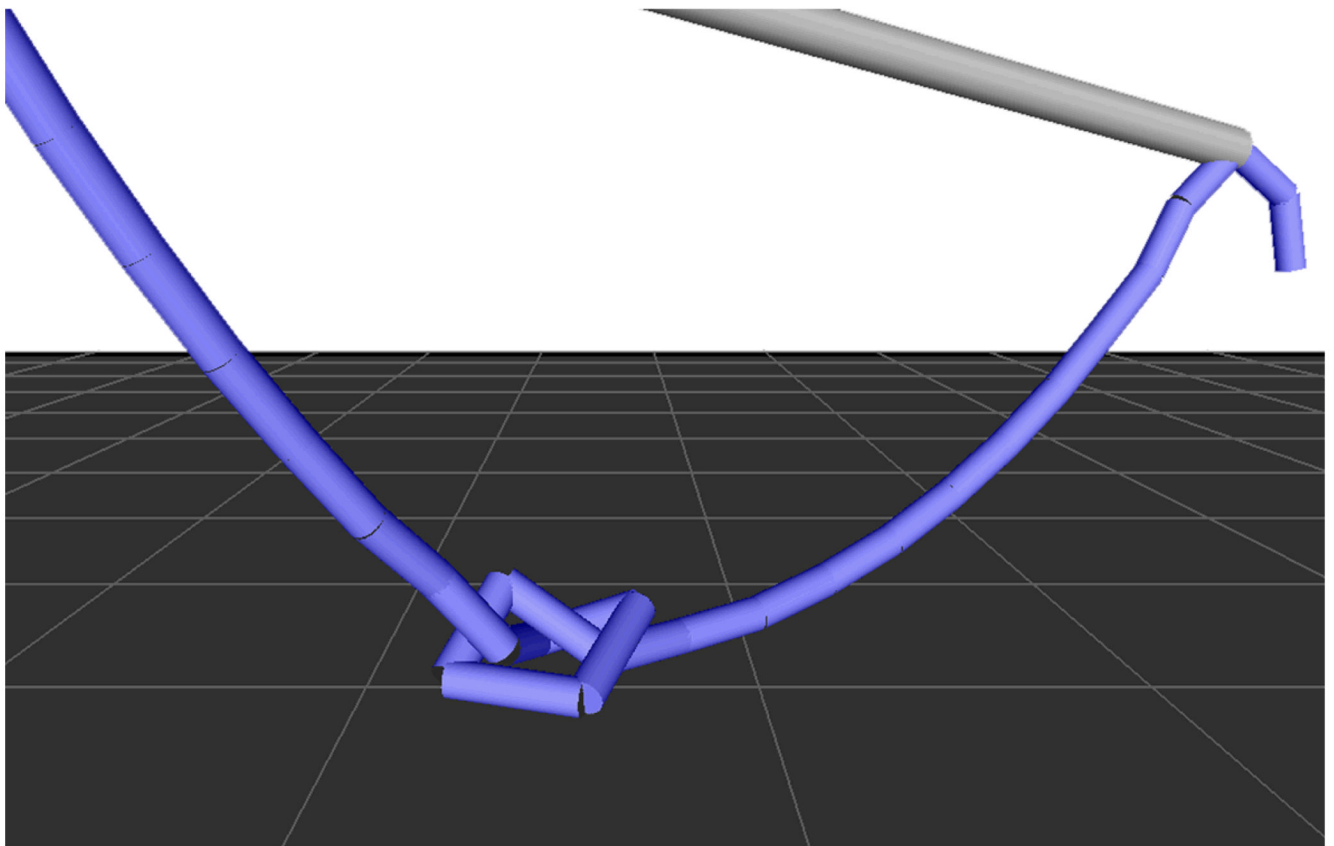


FIGURE 9.
Self-collision of the thread allows to tie a knot.

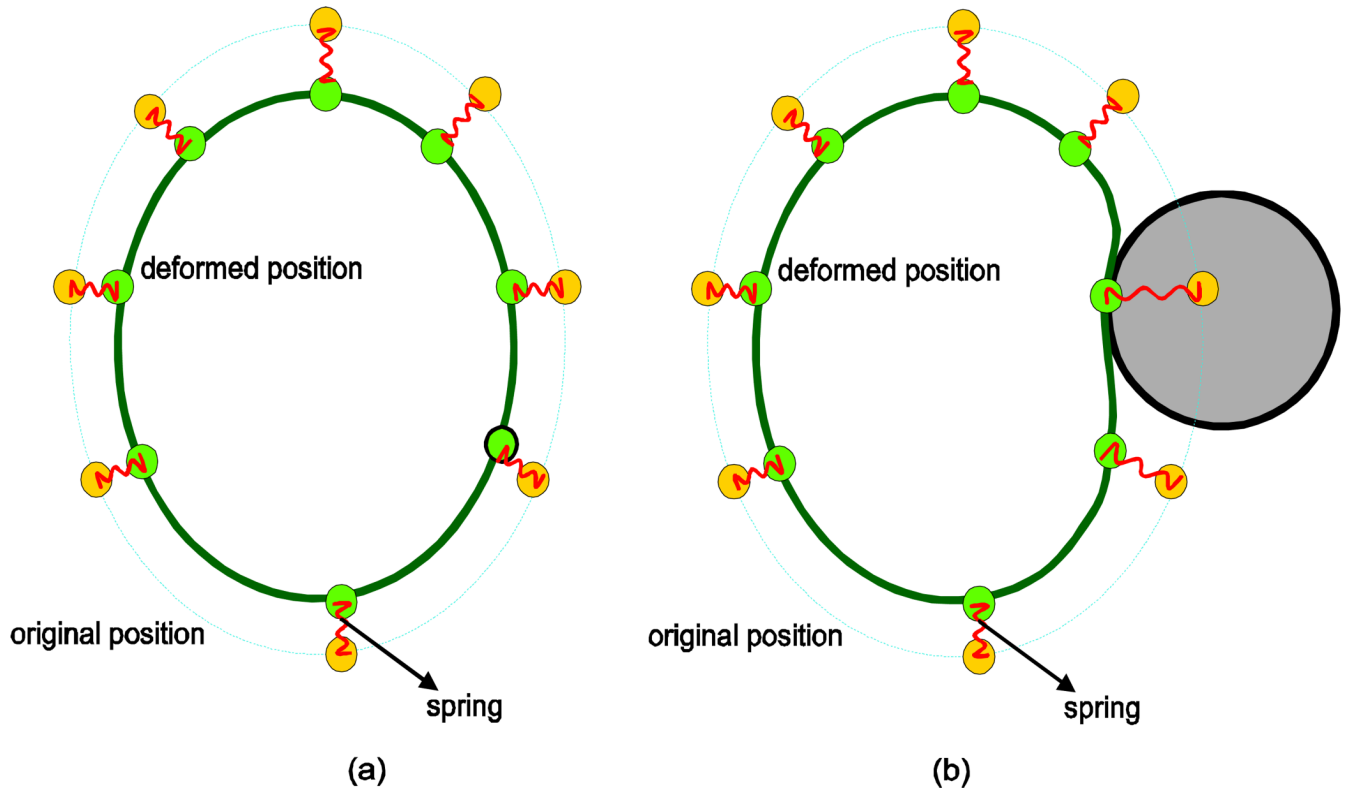


FIGURE 10. The ligating loop structure. (a) in rest position and (b) when in collision with another object.

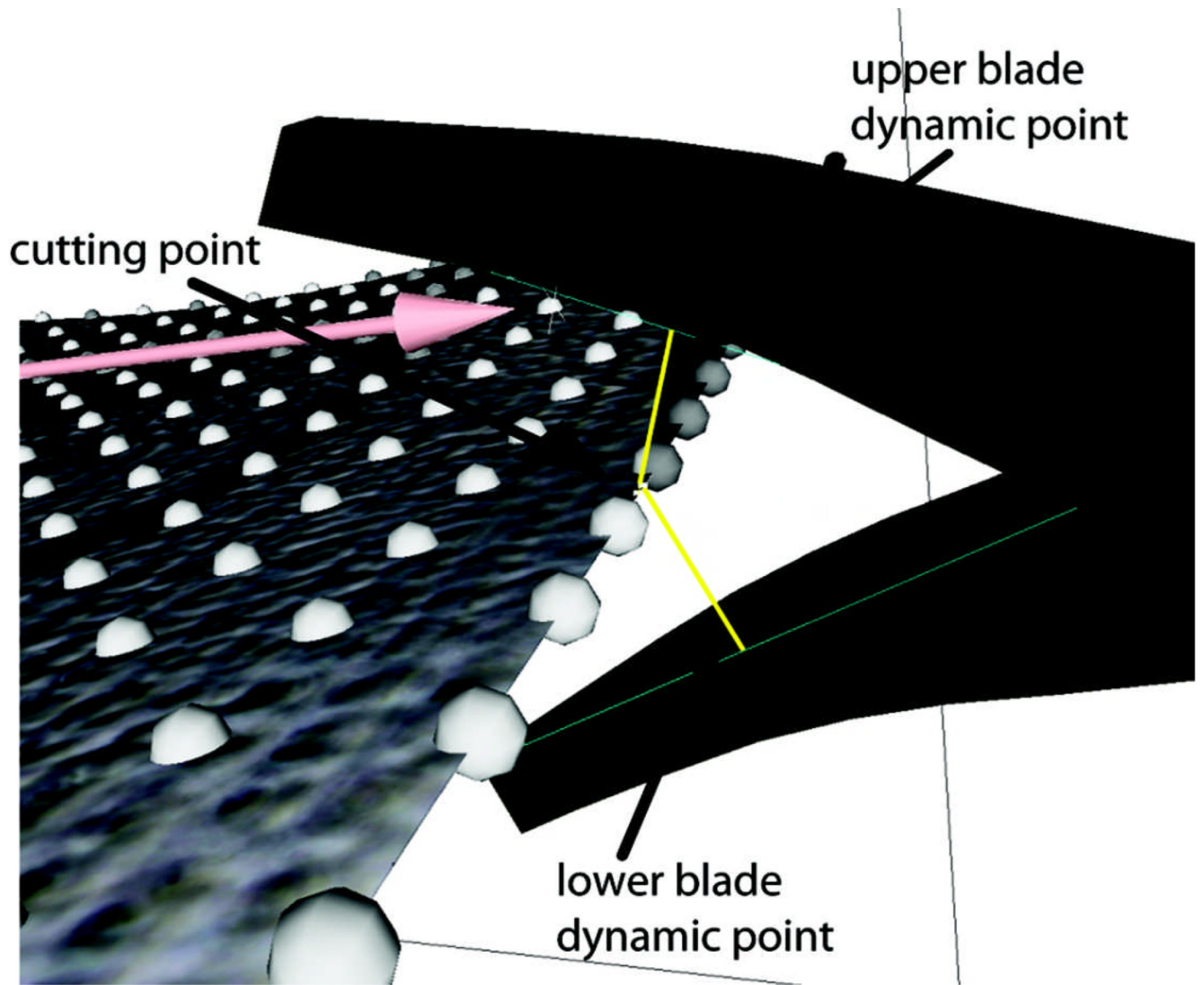


FIGURE 11. Collision detection in cutting. The instrument (scissors) is represented by 3 lines for collision detections and response. When the two lines representing the blades touch the closest point, a cutting point is established.

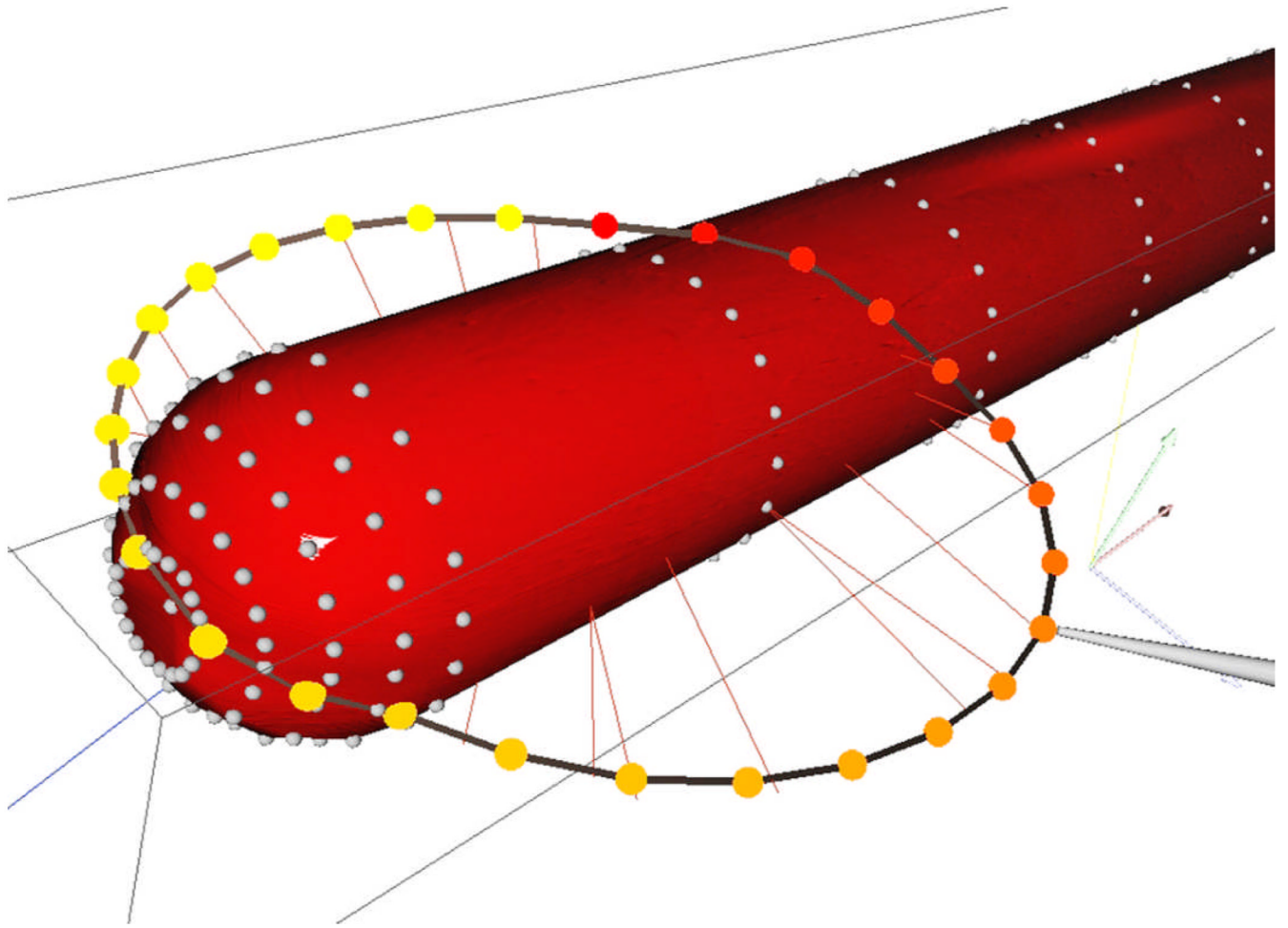


FIGURE 12.

Collision and deformation while manipulating the ligating loop. The loop is decomposed into twenty-eight line segments for the purpose of collision detection and response. The radial lines represent the relation of proximity

TABLE 1

Performance of the VBLaST™ for the four simulated tasks

	Pattern-cutting	Ligating-loop	Suturing	Peg-transfer
<i>Deforming faces</i>	512	720+28	1 878+50	0
Graphics display	60 Hz	60 Hz	60 Hz	60 Hz
Haptics rendering	3 500 Hz	600 Hz	4 000 Hz	8 000 Hz
Physics-based deformation	1 400 Hz	400 Hz	400 Hz	N/A

Values correspond to the number of frames per second of each execution thread. Notice that the execution threads are asynchronous to optimize multi-core efficiency.