Extending Gazebo simulator for surgical robotics: tissue and suture modeling

Artur Sagitov[1], Hongbing Li[2], Natalia Schiefermeier-Mach[3], Evgeni Magid[1]
[1]Laboratory of Intelligent Robotic Systems, Intelligent Robotics Department, Kazan Federal University, Kazan, Russia
[2]Department of Instrument Science and Engineering, Shanghai Jiao Tong University, Shanghai, China
[3]Health University of Applied Sciences Tyrol, Innsbruck, Austria
E-mail: sagitov@it.kfu.ru, natalia.schiefermeier-mach@fhg-tirol.ac.at, magid@it.kfu.ru
http://kpfu.ru/robolab.html

Abstract

Active use of a simulator as a training tool has proven to be advantageous to a human surgeon, but there is no open source and convenient universal surgery simulation of a robot surgeon. This paper presents an extension of Gazebo simulator for surgical robots using Robot Operating System. We present software architecture that allows modeling robot interaction with different types of tissue and suture.

Keywords: robotic surgery, tissue modelling, suture modelling, Gazebo, ROS.

1. Introduction

Robotic surgery field experiences exponential growth and emerging new techniques present new challenges for training of surgeons. We note the increase in quality of robotic surgery simulations and incorporation of those into modern surgical training curriculum. Simulators have proven to be advantageous in a practice, but they are designed for a human training and it is not possible to use those for developing autonomous procedures.

Surgery simulation is an interactive physical simulation that involves rigid and deformable objects. Simulation development engages various applied mathematics branches including numerical analysis, geometric modeling, computational mechanics, collision detection, and rendering, while their computational efficiency is achieved with compound algorithms and multi-threading that take full advantages of modern hardware capabilities. Therefore, developing surgical simulations is a challenging task that requires an expertise in all of the aforementioned areas. One of the ways to address this is combining available separate domain frameworks into a single modular package.

We used Gazebo simulation as a base for our surgery robot simulation because of its support for robot operating system (ROS)[1], but a number of changes were required as Gazebo is not capable of deformable objects simulation. Gazebo consists of two main components: a server and a client. The server evaluates physics of the environment: entire state of each object being involved in the simulation, forces and velocities of each object and external controllers' input. The client uses server output to displays the environment in a graphical user interface in order to provide visualization and control. The server can use different physics' engines each being optimized for different use cases. We have selected Bullet physics engine for existing soft body routines and had to implement a new external plugin for Gazebo in order to add soft body modeling to Gazebo.

Modern vector graphics editors and computer games use a special data structure that represents all entities - a scene graph, which contains different object representations in the scene. In our simulation we organize objects, their relations and algorithms in a hierarchical data structure that is similar to scene graphs. Each object is decomposed into a set of properties: state vectors, mass, forces, constraints, topology, integration scheme, and solving process. Simulation algorithms can be customized for each component individually as well as for physical models themselves. A physical object is represented as a combination of four models: internal model, inertial model, collision model and visual model. We propagate forces and movements between the models concurrently during the simulation.

2. Related Work

In a field of rigid body simulations for robotics there is a number of different products, including an open-source software Gazebo[2], proprietary V-REP, WorkspaceLt, RoboticSimulation, Webots and others. Those simulators rely on different physics engines (e.g., Open Dynamics Engine (ODE), Bullets, DART or NVidia PhysX) and are fast and efficient in rigid-body simulations and easy to use. The main disadvantage from our perspective is that very few of them are capable of modeling realistic soft bodies. Usually soft object modelling involves a use of structural multi-physics tools such as ComSol. As those tools use continuous mechanics algorithms the disadvantage of this method is a slow computational speed and the requirement of deep understanding of the physical processes behind.

Modeling soft objects is an active field of research, and a number of methods exist for soft tissue deformation in a surgical simulation. Mass-spring models (MSM) approximate a soft tissue as a set of point masses being interconnected with massless elastic springs. Advantages of MSM are their easy implementation and efficient evaluation. Disadvantages include difficulty in distribution of material properties and dependency of deformation upon a selected discretization topology[3]. When deformation accuracy is a concern, a finite element method (FEM) is considered. FEM represents a soft tissue model as a continuous object and constructs built-in equations to model complex mechanical behavior of soft tissues. Equations approximating geometric and material properties being involved in deformation are numerous, which makes FEM computationally expensive; moreover, it can become numerically unstable. To address the issue of the computational efficiency of FEM model reductions[4], tensor pre-computation[5], multi-grid solvers[6] and domain decomposition[7] could be applied. FEM that is used in real time surgical simulations often uses simplified linearized equations for material law and strain computation. These simplifications are applicable for small deformations and cause significant errors for large deformations[8]. To correct for large scale deformations a geometric nonlinear tensor-mass model has been applied[9] but it imposes complicated computations using quadratic formulations of strain. To address geometric nonlinearity involving rotational deformations FEM could be supplemented with a co-rotational formulation[10]. In meshless method[11] a deformation could be evaluated without explicit node connections with a penalty of additional computations of node adjacency metric at each iteration. Lack of explicit topology is also an issue[12]. Formulation of Lagrangian dynamics for FEM taking into account geometric and material nonlinearities method[13] depends upon constant topology of an object as it uses precomputations, and has shown numerical stability with only small time steps. To address deformation nonlinearity method of hyperelastic mass links a new approach[14] was proposed that provides fast computation, but as of now is applicable only for homogenous objects[15]. Boundary element method[16] was unable to address objects anisotropic and heterogeneous properties. Using specialized hardware, such as Graphics Processing Unit (GPU), could be used to handle computational load[17]. Simulation framework SOFA[18] was developed with a focus on deformable objects and complex interactions, and can simulate a large variety of models, including rigid-bodies, MSM and FEM with hyperelastic masses.
Our implementation for Gazebo simulator is a somehow intermediate approach between simple game physics engines and a complicated FEM analysis.

3. Tissue and suture models

Both tissue and suture are represented as soft deformable bodies. Assuming suture simple geometry, we directly use a default 2D discretization for solving constitutive equations. The tissues are a 3D object and to construct a mechanical model we had to discretize the tissue domain. We use CGAL (an efficient geometric C++ library) to generate a hexahedral polygonal mesh from 3D model. We adjusted discretization parameters to have the best deformation accuracy given selected limit of volume elements to address the performance. Discretization example is presented in Fig.1.

Deformable objects' behavior is a subject to the laws of continuum mechanics for material modeling, Lagrangian multipliers for constraints solving, and Signorini problem for contact resolution. Internal forces of a simulated object are dependent on a current state and a sum of all known external forces. Amount of deformation depends only on external loads. We use Hooke's law for this dependency because of computation time constraints. Deformation is treated as purely elastic behavior. Each tissue type specifies Young modulus and Poisson's ratio as parameters. When a potential contact with a robot end-effector has been detected, we measure the distance between the robot and an obstacle at a contact point. Using contact mechanics, we find an area of the contact patch and find a displacement vector by resolving Signorini problem.

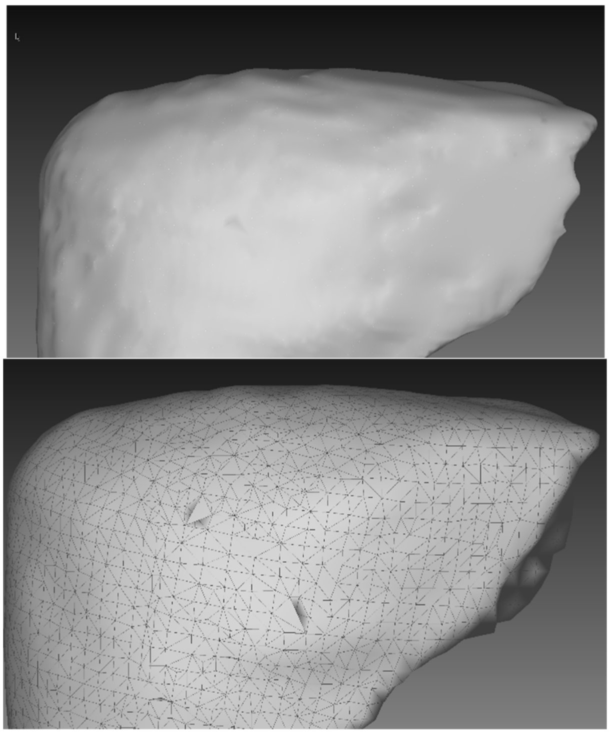

Fig. 1. Model discretization and triangulation

4. Gazebo integration

A scene contains the robot and its environment and is described within XML file. Soft tissues are defined using an extension of URDF format with components for discretization options, contact configuration and options for numerical integration and rendering. Using a synchronized tissue model, we collect force vectors and contact points from different representations to obtain an entire system. Applying our Gazebo plugin, we connect the Gazebo world and the soft tissue to allow interaction with other objects. Plugin synchronizes locations of each of soft body nodes between different representations. Rendering module of the plugin utilizes the state of the physical simulation to render a visualization because Gazebo does not have deformable meshes rendering capability. We render our meshes using custom OGRE renderer as a plugin to Gazebo that is implemented through *rendering::visual* class.

In our current configuration of the plugin we allow interaction of a rigid body object (manipulator) with a soft tissue in a realistic manner. Behavior of the simulated soft tissue is sensitive to parameter selection, mechanical parameters and tissue topology which should be specified manually.

5. Examples of abdomen simulation

We are using abdomen models available from OpenHELP phantom, which are designed with realistic anatomy and material characteristics, for our autonomous suturing system prototype testing[19]. Using plugin for each of the tissue type, abdomen models were imported to Gazebo with proper mechanical properties. Model of the interaction is represented on Fig. 2.

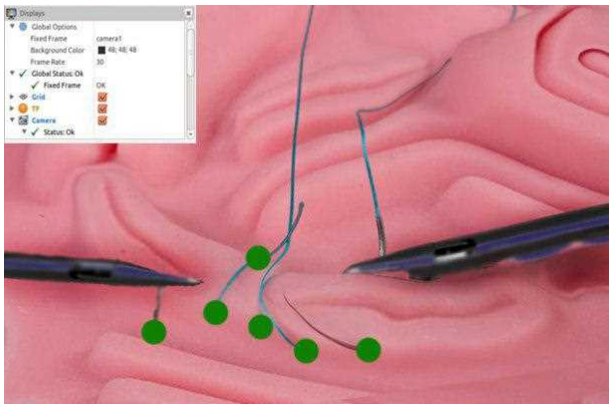

Fig. 2. Integrated model in Gazebo

We used a custom manipulator to interact with the soft tissue to observe the interaction results. The interaction simulation took around 5 seconds for 1 real time second (time step of 5ms, 10000 mesh nodes) using Haswell-E i7-5820K. The complete soft tissue model therefore was successfully implemented for

Gazebo, although the performance should be improved further.

6. Conclusions

We developed and implemented a new soft tissue Gazebo plugin with an emphasis on performance and have achieved an acceptable level of usability. The plugin demonstrates realistic interaction with soft tissue. As a future work, we consider to improve performance with a GPU-based approach and to allow for complex elastic interactions' simulation. We intend to turn this plugin into a basis for long-term developments in robot surgery automation.

Acknowledgements

This work was supported by the Russian Foundation for Basic Research and the Government of the Republic of Tatarstan, project ID 18-48-160037. Part of the work was performed according to the Russian Government Program of Competitive Growth of Kazan Federal University.

References

1. A. Sagitov, T. Tsoy, H. Li, E. Magid. Automated open wound suturing: detection and planning algorithm. *J. of Robotics, Networking and Artificial Life*, Volume 5, Issue 2, September 2018, pp. 144 - 148
2. N. Koenig and A. Howard, Design and use paradigms for gazebo, an open-source multirobot simulator, in *Proc. of Conf. on Intelligent Robots and Systems*, vol. 3 (2004).
3. Y. Duan, W. Huang, H. Chang, W. Chen, J. Zhou, S.K. Teo, Y.Su, C.K. Chui, S. Chang, Volume preserved mass-spring model with novel constraints for soft tissue deformation, in *IEEE J. Biomed. Health Inform.*20 *(1),* (2016), pp. 268–280.
4. C. Yang, S. Li, Y. Lan, L. Wang, A. Hao, H. Qin, Coupling time-varying modal analysis and FEM for real-time cutting simulation of objects with multi-material sub-domains, in *Comput. Aided Geom.* Des.43, (2016), pp. 53–67.
5. S. Cotin, H. Delingette, N. Ayache, Real-time elastic deformations of soft tissues for surgery simulation, *IEEE Trans. Vis. Comput. Graphics,* 5 (1), (1999), pp. 62–73.
6. D. Weber, J. Mueller-Roemer, C. Altenhofen, A. Stork, D.W. Fellner, Deformation simulation using cubic finite elements and efficient p-multigrid methods, in *Comput. Graphics* UK 53, (2015), pp. 185–195.
7. J. Huang, X. Liu, H. Bao., B. Guo, H.Y. Shumb, An efficient large deformation method using domain decomposition, in *Comput. Graphics* UK30 (6), (2006), pp. 927–935.
8. Y. Payan, Soft Tissue Biomechanical Modeling for Computer Assisted Surgery, vol. 11. (Springer*,* Berlin, 2012).
9. S. Xu, P. X. Liu, H. Zhang, A nonlinear viscoelastic tensor-mass visual model for surgery simulation, in *IEEE Trans. Instrum.* Meas.60, (2011), pp. 14–20.
10. R. Plantefève, I. Peterlik, N. Haouchine, S. Cotin, Patient-specific biomechanical modeling for guidance during minimally-invasive hepatic surgery, in *Ann. Biomed.* Eng.44, (2016), pp. 139–153.
11. G.Y. Zhang, A. Wittek, G.R. Joldes, X. Jin, K. Miller, A three-dimensional nonlinear meshfree algorithm for simulating mechanical responses of soft tissue, in *Eng. Anal. Bound.* Elem.42, (2014), pp. 60–66.
12. V.P. Nguyen, T. Rabczuk, S. Bordas, M. Duflot, Meshless methods: a review and computer implementation aspects, in *Math. Comput. Simul.*79 (3), (2008), pp. 763–813.
13. K. Miller, G. Joldes, D. Lance and A. Wittek, Total Lagrangian explicit dynamics finite element algorithm for computing soft tissue deformation, in *Int. J. Numer. Methods Biomed.* Eng.23 (2), (2007), pp. 121–134.
14. F. Goulette, Z.-W. Chen, Fast computation of soft tissue deformations in real-time simulation with Hyper-Elastic Mass Links, in *Comput. Methods Appl. Mech.* Eng.295, (2015), pp. 18–38.
15. J. Zhang, Y. Zhong, and C. Gu, Ellipsoid bounding region-based ChainMail algorithm for soft tissue deformation in surgical simulation, in *International Journal on Interactive Design and Manufacturing*, vol. 12, no. 3, (2018), pp. 903-918.
16. P.Wang, A.A. Becker, I.A. Jones, A.T. Glover. S.D. Benford. C.M. Greenhalgh, M. Vloeberghs: Virtual reality simulation of surgery with haptic feedback based on the boundary element method, in *Comput. Struct*. 85 (7–8), (2007), pp. 331–339.
17. S.F. Johnsen, Z.A. Taylor, M.J. Clarkson, J. Hipwell, M. Modat, B. Eiben, L. Han, Y. Hu, T. Mertzanidou, D.J. Hawkes, S. Ourselin, NiftySim: a GPU-based nonlinear finite element package for simulation of soft tissue biomechanics, in *Int. J. Comput. Assist. Radiol. Surg.10* (7), (2015), pp. 1077–1095.
18. F. Faure, C. Duriez, H. Delingette, J. Allard, B. Gilles, S. Marchesseau, H. Talbot, H. Courtecuisse, G. Bousquet, I. Peterlik, S. Cotin, SOFA: A Multi-Model Framework for Interactive Physical Simulation, Soft Tissue Biomechanical Modeling for Computer Assisted Surgery, 11, Studies in *Mechanobiology, Tissue Engineering and Biomaterials,* (Springer, 2012), pp. 283-321.
19. A. Sagitov, H. Li, E. Magid. Toward automated open wound suturing using haptic feedback: detecting wounds and planning the suture. *Int. Conf. on Artificial Life and Robotics*, (2018), pp. 208-211.