Necessity for More Realistic Contact Simulation

Mabel M. Zhang
Open Robotics (OSRF)
mabel@openrobotics.org

***Abstract*—Given the recent progress in large-scale data-driven methods, sim-to-real is quickly becoming an enabler for access to machine learning research, alleviating hardware costs and physical constraints. Simulation's role in validating new methods is unlikely to diminish. While success in simulation is far from a guarantee of success in the real world, it serves as a valuable baseline and sanity check during development. In addition, mission-critical tasks with high costs and strict time constraints, such as deep sea and space missions, rely on simulation to determine the minimum plausibility of the system.**

As visual simulation matures and visual perception's limits are acknowledged in the manipulation community, tactile perception is gaining steady interest. Unlike visual simulation, which relies less on physics than on rendering, contact simulation depends largely on the fidelity of physics modeling, which is crucial for interaction-heavy tasks like manipulation and locomotion.

At the present, contact simulation is primitive even for rigid bodies, not to mention soft and deformable bodies. This presents a severe lack of minimum baseline and accessibility, especially given the limited availability and flakiness of cutting-edge contact sensors. Any improved contact simulation is a prerequisite, before sim-to-real transfer can even come into relevance in contact perception. This position paper supports this view by summarizing the state of the art, challenges, and the latest progress as cursors forward.

I. State of the Art

The need for more realistic physics modeling marks the difference between physics-dependent algorithms, such as interactions with the world in manipulation and locomotion, and dominantly visual algorithms, such as decisions in autonomous driving. While the latter may be trained in graphics-intensive game engines, which are highly optimized for realistic rendering, the former poses minimum fidelity requirements that are not yet met by robotics-oriented physics engines.

In simulation tools commonly used for robotics, contacts are based on single points. This does not reflect realistic physics, which is analog, spatially distributed, and never at an exact infinitesimal point. Other than contact modeling itself, other properties such as friction and material are difficult to model. Soft and deformable objects pose even more challenges. Adding to the fact that the nature of deformable bodies are more complex, mainstream simulation software are predominantly designed for rigid body dynamics. Yet another dimension to the sim-to-real gap is real-time execution. All of the problems stated above are difficult even with no requirements on speed.

A number of physics engines and simulation software exist. Examples of physics engines commonly used in robotics that are freely available include Open Dynamics Engine (ODE) [1], [2], Dynamic Animation and Robotics Toolkit (DART) [3], and Bullet [4]. Most simulation software are designed primarily for rigid body dynamics. Examples include open source ones such as Gazebo [5], NVIDIA PhysX [6], Drake [7], Klampt [8], and Pinocchio [9]–[11], and proprietary ones such as MuJoCo [12]. Some of the above handle soft and deformable body dynamics [3], [4], [6], [12]. Fewer are specifically designed for soft bodies, such as VoxCAD [13], and biomedical-oriented simulation like SimBody [14] and Simulation Open Framework Architecture (SOFA) [15]. At the present, the capability of soft body simulation is limited, and it is not widely used in robotics.

Other than contact simulation, contact sensor simulation is lacking, as sensor surfaces are often deformable. Moreover, the sensor data is directly affected by force distribution over a contact area, which is not well modeled, since contacts are modeled by discrete points. Volume-based contact simulation has been studied [16] but is not widely available for robotics.

II. Challenges and Progress

A. *Game engines vs. physics engines*

Simulation is by nature a central topic in graphics and animation, where the goal is to synthesize environments, with some that may resemble the real world. In contrary, the role of simulation in robotics is, understandably, an aside. It is not traditionally viewed as a core robotics problem. As a result, even though increasingly many roboticists rely on simulation, it does not receive an adequate amount of academic attention to move significantly forward, especially in the area of physics simulation.

In practice, this translates to that robotics researchers are limited to off-the-shelf simulation software that do not have realistic physics. This leads to early abandonment of simulation for the real world, in applications that can afford it. However, in applications that are mission-critical (such as space and deep sea), require high repetitions, demand close inspection of dynamics in slow time steps, or in general are expensive to run or require reproducible results, a working system in simulation is a prerequisite that cannot be forgone.

There are a few motivational and practical differences between graphics and robotics that justify more attention to be devoted to robotics-oriented simulation research. First, in balancing the trade-off between physics and rendering, two computationally intensive components in simulation, graphics applications naturally prioritize rendering, while robotics has firm practical requirements on high-fidelity physics. Second, tools developed for the entertainment industry have a large computational budget, with high-performing GPUs for real-time computation. Offline rendering is allowed for non-real-time needs. In robotics, computational power is often limited by payload, power consumption, and heat dissipation. At the same time, real-time computation is still required and is often associated with critical safety risks.

Third, even though game engines have physics components, they do not have sensing components that translate directly to sensor readings in the real world. In simulation, raw physical quantities such as force at specific points may be accessible, but the real world does not have such oracles and rely on sensors placed on robots. Simulation of sensors other than camera is generally unavailable or unrealistic, even in robotics-oriented software. This means in order to make use of simulated physical quantities in the real world, the corresponding quantities, such as force, must first be inferred from real sensor data. Such inference is still active research and does not represent true physics. The lack of sensor simulation leaves a gap in data bewteen simulation and the real world.

Recently, traditionally graphics communities have initiated robotics-oriented simulations [6], [17], [18], targeting large-scale and computationally expensive learning.

B. Contact sensor simulation

For stationary or mobile robots on the ground, the limits of physics simulation often cited are friction, surface material, contact forces, real-time computation, to name a few. For aerial, surface, and underwater robots, the array of limitations expand to even less explored areas like fluid dynamics. This extended abstract focuses on contact simulation, a small slice of the array with large demand and room for improvement.

Other than simulating contact physics, the challenges of which are outlined in Section I, the simulation of contact sensors can provide complementary information from rendering. Specifically, optical tactile sensors use camera images to infer physical properties such as force normals, shear, torque, and other quantities.

The past year has seen a surge of new optical tactile sensors, with DIGIT [19] and OmniTact [20] based on the concept of GelSight [21], Li et al. [22] and GelTip [23] combining the concepts of GelSight and TacTip [24] to measure both tactile and force information, and NeuroTac [25] based on TacTip.

Up to now, the manufacturers of tactile sensors have often been the only users, with a few commercial exceptions like BioTac [26] and TakkTile [27]. However, with the recent release of GelSight manufacturing method [28] and the low-cost 3D-printable DIGIT, the use of optical tactile sensors is expected to increase. Simulation of these sensors will not only widen access but also provide a platform for large-scale learning, which GelSight and BioTac have already been used in the past, only in the real world.

Toward that end, in the past year, Gomes et al. [29] simulated the GelSight sensor [28] using a depth camera in Gazebo. Narang et al. [30] developed finite element modeling using ANSYS to simulate the deformable surface of the BioTac sensor. Ding et al. [31] designed a soft body simulation model for TacTip using Unity [18], by applying operations on mesh vertices. The model learned from simulation is then transferred to the real world for edge prediction with no real-world data.

The unique advantage brought forth by optical contact sensors is that physics and rendering, opposite ends of the performance trade-off, can now be complementary. Difficult problems in contact physics simulation may be mitigated by optical tactile data.

C. Physics vs. rendering

Physics and rendering are two core components of simulation. Often, one is traded off for the other, because both are computationally expensive. Alternatively, it may be possible to supplement one with the other.

In the case of contact simulation, optical tactile sensor simulation can be used to add rich physical data otherwise difficult and slow to compute. GelTip [23] and [22] are examples of real sensors that combine the camera data of the tactile imprint on the elastomer sensor surface with the physical data from force pins embedded in the sensor.

In the case of soft bodies, one possible way to improve simulation with learning is to optimize simulation parameters using real-world visual data. For example, interaction with deformable objects can be recorded in the real world. Physical quantities such as surface normals can then be obtained from depth cameras and used to correct the simulated normals. Liang et al. [32] used real-world object poses to optimize parameters in a high-fidelity simulation that supports deformable body dynamics [33].

III. Conclusion

Contact simulation of higher fidelity than the current primitive state is a minimum requirement for accessibility and reproducibility of contact-based research, which is gaining traction in locomotion and manipulation. Otherwise, the data gap between simulation and reality remains wide, and the data are in different formats, fundamental physical quantities and digital sensor input, respectively. Only when the gap is narrowed would sim-to-real methods have meaningful simulation data to train on, especially given the quantity and dimensionality required for data-driven approaches, even in sim-to-sim. Such developments will not only propagate manipulation and locomotion research beyond the pre-contact visual stage, but also enable wider and more diverse community participation.

References

[1] R. L. Smith, "Open Dynamics Engine," 2001. [Online]. Available: https://www.ode.org/

[2] E. Drumwright, J. Hsu, N. Koenig, and D. Shell, "Extending Open Dynamics Engine for robotics simulation," in *International Conference on Simulation, Modeling, and Programming for Autonomous Robots*, vol. 6472. Berlin, Heidelberg: Springer, 2010, pp. 38–50.

[3] J. Lee, M. X. Grey, S. Ha, T. Kunz, S. Jain, Y. Ye, S. S. Srinivasa, M. Stilman, and C. K. Liu, "DART: Dynamic Animation and Robotics Toolkit," *Journal of Open Source Software*, vol. 3, no. 22, p. 500, Feb. 2018. [Online]. Available: https://dartsim.github.io/

[4] E. Coumans, "Bullet physics simulation," in *SIGGRAPH*, July 2015. [Online]. Available: https://pybullet.org/

[5] N. Koenig and A. Howard, "Design and use paradigms for Gazebo, an open-source multi-robot simulator," in *IEEE/RSJ International Conference On Intelligent Robots and Systems (IROS)*, Sendai, Japan, 28 September–2 October 2004. [Online]. Available: http://gazebosim.org/

[6] NVIDIA, "PhysX." [Online]. Available: https://developer.nvidia.com/physx-sdk

[7] R. Tedrake and the Drake Development Team, "Drake: Model-based design and verification for robotics," 2019. [Online]. Available: https://drake.mit.edu

[8] K. Hauser, "Robust contact generation for robot simulation with unstructured meshes," in *International Symposium on Robotics Research (ISRR)*, vol. 114, Apr. 2013, pp. 357–373. [Online]. Available: https://github.com/krishauser/Klampt

[9] J. Carpentier and N. Mansard, "Analytical Derivatives of Rigid Body Dynamics Algorithms," in *Robotics: Science and Systems (RSS)*, Pittsburgh, Pennsylvania, June 2018.

[10] J. Carpentier, G. Saurel, G. Buondonno, J. Mirabel, F. Lamiraux, O. Stasse, and N. Mansard, "The Pinocchio C++ library – a fast and flexible implementation of rigid body dynamics algorithms and their analytical derivatives," in *IEEE International Symposium on System Integrations (SII)*, 2019.

[11] J. Carpentier, F. Valenza, N. Mansard, *et al.*, "Pinocchio: fast forward and inverse dynamics for poly-articulated systems," 2015–2019. [Online]. Available: https://stack-of-tasks.github.io/pinocchio

[12] E. Todorov, T. Erez, and Y. Tassa, "MuJoCo: A physics engine for model-based control," in *IEEE/RSJ International Conference On Intelligent Robots and Systems (IROS)*, Vilamoura, Portugal, 7–12 October 2012. [Online]. Available: http://www.mujoco.org/

[13] J. Hiller and H. Lipson, "Dynamic simulation of soft multimaterial 3d-printed objects," *Soft Robotics (SoRo)*, vol. 1, no. 1, Feb. 2014. [Online]. Available: https://www.creativemachineslab.com/voxcad.html

[14] M. A. Sherman, A. Seth, and S. L. Delp, "Simbody: multibody dynamics for biomedical research," in *Symposium on Human Body Dynamics*, ser. Procedia IUTAM, vol. 2, 2011, pp. 241–261. [Online]. Available: https://simtk.org/projects/simbody

[15] F. Faure, C. Duriez, H. Delingette, J. Allard, B. Gilles, S. Marchesseau, H. Talbot, H. Courtecuisse, G. Bousquet, I. Peterlik, and S. Cotin, "SOFA: A Multi-Model Framework for Interactive Physical Simulation," in *Soft Tissue Biomechanical Modeling for Computer Assisted Surgery*, ser. Studies in Mechanobiology, Tissue Engineering and Biomaterials, Y. Payan, Ed. Springer, June 2012, vol. 11, pp. 283–321. [Online]. Available: https://www.sofa-framework.org/

[16] F. Faure, F. Falipou, J. Allard, C. Duriez, P. G. Kry, H. Courtecuisse, B. Wang, D. K. Pai, and C. Barbier, "Volume-based Contact," in *Computational Contact Mechanics: Advances and Frontiers in Modeling Contact*, Feb. 2014.

[17] NVIDIA, "Isaac SDK," 2018. [Online]. Available: https://developer.nvidia.com/isaac-sdk

[18] Unity Technologies, "Unity Simulation." [Online]. Available: https://unity.com/products/simulation

[19] M. Lambeta, P.-W. Chou, S. Tian, B. Yang, B. Maloon, V. R. Most, D. Stroud, R. Santos, A. Byagowi, G. Kammerer, D. Jayaraman, and R. Calandra, "DIGIT: A Novel Design for a Low-Cost Compact High-Resolution Tactile Sensor with Application to In-Hand Manipulation," *IEEE Robotics and Automation Letters (RA-L)*, vol. 5, no. 3, pp. 3838–3845, Feb. 2020.

[20] A. Padmanabha, F. Ebert, S. Tian, R. Calandra, C. Finn, and S. Levine, "OmniTact: A Multi-Directional High Resolution Touch Sensor," in *IEEE International Conference on Robotics and Automation (ICRA)*, Paris, France, June 2020.

[21] W. Yuan, R. Li, M. A. Srinivasan, and E. H. Adelson, "Measurement of Shear and Slip with a GelSight Tactile Sensor," in *IEEE International Conference on Robotics and Automation (ICRA)*, Seattle, WA, USA, May 2015.

[22] W. Li, J. Konstantinova, Y. Noh, Z. Ma, A. Alomainy, and K. Althoefer, "An Elastomer-based Flexible Optical Force and Tactile Sensor," Seoul, South Korea, Apr. 2019.

[23] D. F. Gomes and S. Luo, "Exploiting touch sensing around fingers," in *ViTac Workshop, IEEE International Conference on Robotics and Automation (ICRA)*, Paris, France, June 2020.

[24] N. F. Lepora and B. Ward-Cherrier, "Superresolution with an optical tactile sensor," in *IEEE/RSJ International Conference On Intelligent Robots and Systems (IROS)*, Hamburg, Germany, Sept. 2015.

[25] B. Ward-Cherrier, N. Pestel, and N. Lepora, "Neurotac: A neuromorphic optical tactile sensor applied to texture recognition," in *IEEE International Conference on Robotics and Automation (ICRA)*, Paris, France, June 2020.

[26] J. A. Fishel, V. J. Santos, and G. E. Loeb, "A robust micro-vibration sensor for biomimetic fingertips," in *IEEE International Conference on Biomedical Robotics and Biomechatronics*, Scottsdale, AZ, USA, Oct. 2008, pp. 659–663.

[27] Y. Tenzer, L. P. Jentoft, and R. D. Howe, "The Feel of MEMS Barometers: Inexpensive and Easily Customized Tactile Array Sensors," *IEEE Robotics & Automation Magazine*, vol. 21, no. 3, pp. 89–95, Sept. 2014.

[28] W. Yuan, S. Dong, and E. H. Adelson, "GelSight: High-Resolution Robot Tactile Sensors for Estimating Geometry and Force," *IEEE Sensors Journal*, vol. 17, no. 12, Nov. 2017.

[29] D. F. Gomes, A. Wilson, and S. Luo, "GelSight simulation for sim2real learning," in *ViTac Workshop, IEEE International Conference on Robotics and Automation (ICRA)*, 2019.

[30] Y. Narang, K. V. Wyk, A. Mousavian, and D. Fox, "Interpreting and Predicting Tactile Signals via a Physics-Based and Data-Driven Framework," in *Robotics: Science and Systems (RSS)*, Corvalis, Oregon, USA, July 2020.

[31] Z. Ding, N. F. Lepora, and E. Johns, "Sim-to-Real Transfer for Optical Tactile Sensing," in *IEEE International Conference on Robotics and Automation (ICRA)*, Paris, France, June 2020.

[32] J. Liang, A. Handa, K. V. Wyk, V. Makoviychuk, O. Kroemer, and D. Fox, "In-Hand Object Pose Tracking via Contact Feedback and GPU-Accelerated Robotic Simulation," in *IEEE International Conference on Robotics and Automation (ICRA)*, Paris, France, June 2020.

[33] M. Macklin, K. Erleben, M. Müller, N. Chentanez, S. Jeschke, and V. Makoviychuk, "Non-smooth Newton Methods for Deformable Multi-body Dynamics," *ACM Transactions on Graphics*, no. 140, Oct. 2019.