Research Statement

Huamin Wang

The fast development of virtual reality and digital manufacturing industries is quickly redefining what computer graphics is and what computer graphics research should be. In this rapid transition, it is imperative to have fast and reliable computational tools for real-time simulation and modeling of deformable solids, shells, and fluids. Toward this goal, I focus my research in four directions. Below are the research summaries in these directions.

1 Real-Time Simulation of Deformable Bodies

Real-time simulation of deformable bodies is a highly demanded feature in entertainment, biomedical, and scientific applications, such as virtual plastic surgery and animation of buildings in an earthquake. However, deformable body simulation is often known to be computationally expensive, due to the common use of small time steps for addressing the numerical instability issue. Recent research on strain limiting [18, 21] and position-based dynamics [13, 14] suggests to avoid numerical instability by using geometric constraints to replace elastic forces. These techniques are simple and compatible with GPU acceleration, but they lack the physical meanings of their underlying elastic material properties. The more recent projective dynamics technique [1, 10] describes geometric constraints as specific elastic energies in quadratic forms. This allows the simulation to be iteratively solved by a local constraint step and a global linear system step, the latter of which runs fast on the CPU, but not on the GPU. In general, fast and accurate simulation of deformable bodies is still a challenging problem in computer graphics. I believe that my following research will lead to the ultimate solution to this problem.

GPU-based simulation My research on GPU-based simulation is based on the fact that deformable body simulation can be formulated into a nonlinear optimization problem and many dynamics solvers can be interpreted as nonlinear optimization algorithms. For example, projective dynamics is essentially a gradient descent method with a constant matrix preconditioner. Therefore, the method converges fast in the first few iterations, but not so much later as the preconditioner gets less ineffective. This new knowledge motivates me to rethink the design of a real-time deformable body simulator, especially on the GPU.

My very first discovery [20] is that the gradient descent method with a Jacobi preconditioner can be highly effective, after being accelerated by the Chebyshev semi-iterative method. This can be explained by two unique characteristics of deformable body simulation: 1) most elastic models allow the diagonals of their Hessian matrices to be evaluated as the Jacobi preconditioners; 2) the Chebyshev method is effective, as long as the optimization problem is mildly nonlinear within a time step. My experiment showed that the resulting open-source simulator can simulate a deformable body with 60K elements using a h=1/30s time step at 60FPS or more. I believe that this work opens the door to more efficient and reliable simulation of deformable bodies, using other optimization algorithms and/or multi-GPU architectures.

Spectral and spatial acceleration *Model reduction* and *domain decomposition* are two popular techniques used for accelerating deformable body simulation spectrally and spatially. However, both of them have limitations: model reduction cannot handle deformations outside of a predefined subspace, and domain decomposition cannot easily solve the discontinuity issue on the domain boundary. To address these issues, my recent work [22] formulated subspace simulation of all of the domains into a unified simulation system. Based on this system, our research team invented new cubature approximation schemes and matrix reordering approaches to further speed up the simulation performance. This system can efficiently simulate multi-domain deformable bodies under the same material model with limited discontinuity artifact, for the first time as far as we know.

The next research step I will take is to treat these techniques as preconditioners in full-space simulation. Since they do not work as solvers, their limitations should be less problematic. The challenge is how to improve the convergence rate of an iterative solver, in which these techniques are used. My research team will also investigate the combination of the techniques with multi-grid methods, for more speedups in a multi-resolution way.

Adaptive remeshing for fracture simulation A solid object can fracture when the stress exceeds its yield strength. Simulating such a phenomenon requires the object to be remeshed, which is known to be complex and difficult [7] in the past. Fortunately, remeshing can be simplified, if we consider surface meshes only. Our research team first studied the adaptive remeshing scheme for multi-layered thin plates [3]. Based on constrained Delaunay triangulation, this scheme adaptively refines the mesh around fracture fronts to create fine fracture details. Our team later extended the adaptive remeshing idea to fracture simulation of solids [6]. Instead of refining volumetric meshes during dynamic simulation, our method refines the fractured surface as post-processing. This method allows highly detailed fracture surfaces to be generated within minutes for the first time. Without the method, physics-based simulators need hours or even days to generate similar details at this level.

2 Fast and Accurate Clothing Simulation

The trillion-dollar fashion industry deeply influences our lives everyday. Being able to efficiently and accurately simulate clothing will not only broaden graphics applications, but also have a great potential to revolutionize the whole fashion industry, from design to retailing. My research team is one of the leading teams in this field worldwide. My work on cloth dynamics solvers has already been covered in Section 1. In this section, I will summarize my additional work focused on cloth collisions and contacts.

GPU-based continuous collision detection Compared with self-collisions of deformable solids, self-collisions of cloth are much more complex. Existing cloth simulators often use the continuous collision detection (CCD) technique, which is difficult to be developed with efficiency and accuracy. Since false negatives are more critical than false positive, researchers [2] suggested the use of error bounds to avoid false negatives caused by numerical errors. But then how to determine error bound values becomes a dilemma: smaller values may not sufficiently eliminate false negatives, while bigger values will cause too many false positives. To address this issue, I systematically analyzed the errors occurred in CCD and derived an error bound estimation scheme [19] to strictly ensure the safety of CCD for the first time, without using expensive exact arithmetics.

To further reduce the collision cost, my colleagues and I recently studied the development of an end-to-end simulation system that exploits GPU parallelism for time integration, collision detection, and collision handling. This system has already demonstrated its high performance on the GPU and it is expected to serve as the foundation for future research in this direction.

Cloth contact mechanics The second topic I am interested in is how to handle collisions and contacts, once they get detected. My research team studied the data-driven approach [5] for modeling complex cloth friction effects, and invented a fast algorithm for animating air trapped within multiple cloth layers. For virtual fitting room applications, we investigated fast detection and handling of collisions between virtual clothing and human bodies [12], and automatic human body posing from point cloud data [17]. Although the clothing simulation research still has a long way to go, our team is making solid research progress and our contributions will play an important role in the success of fashion-related applications in the future.

3 Physics-based Modeling of Printable Shapes

As the 3D printing technology starts to revolutionize how things are made in everyday life, it still suffers from many limitations, such as *long printing time*, *expensive printing materials*, and *limited printing volume*. To address these limitations, my research team investigates physics-based modeling of printable shapes, by leveraging our expertise in physics-based simulation.

A level-set-based partitioning framework A natural way to reduce the printing time and overcome the printing volume limit is to partition a large 3D model into pieces, each of which can be quickly built in parallel by small printers. The main question is how to define and optimize the partitioning, so that it can meet various quality criteria. Inspired by the wide use of level set methods in image and mesh segmentation, we proposed a level-set-based framework for the partitioning of printable shapes [23]. Under this framework, we presented a series of level set methods to improve the intrinsic qualities of the partitioning, such as boundary smoothness, gap invisibility, and strength against separation. Meanwhile, we developed a physics-based packing algorithm, which can work as a metric for level set methods to reduce the space occupied by the pieces within a printer or container. This framework has already demonstrated its effectiveness and flexibility in the design of high-quality partitioning models, which cannot be easily produced by previous methods.

Future research Under the proposed level-set-based framework, I plan to further investigate better algorithms to address multiple printability issues. I am also interested in level set methods for optimizing support structures of partitioned pieces within a printer or container. Finally, I would like to improve the strength of an assembled model by struts and/or connectors, based on our ongoing research on real-time physics-based simulation.

4 Particle-based Simulation of Fluids and Solids

I became more interested in particle-based simulation in recent years, because of its simplicity and flexibility in animating many real-world fluid and solid behaviors. For fluids, I am specifically interested in simulating thin features, such as bubbles, streamlets, and capillary waves. For example, my research team defined a foam bubble as a particle and modeled its irregular shape by a Voronoi diagram [4]. Using this representation, our simulator can efficiently animate foams with a large number of bubbles, which are too computationally expensive to handle otherwise. Later my colleagues and I studied tensile instability in smoothed particle hydrodynamics. Based on this study, we developed a new surface tension scheme for preserving thin features in free-surface flows [8]. Recently, we presented a novel technique [15] to animate capillary waves on top of a particle-based simulator, so that the animation result can be enriched by wave effects that are lost in particle-based simulation.

Brush painting system To explore the use of fluid simulation in practical applications, our research team collaborates with Adobe Research on the development of a real-time physics-based brush painting system. This system uses a hybrid fluid representation: the paint near a brush is represented by particles, while the rest of the paint is represented by a signed distance field defined on an Eulerian grid. To implement this system on the GPU, we presented a series of new methods to handle the challenge of coupling paint, brush, and canvas together. The prototype of this system has been presented at SIGGRAPH Asia 2016 and GPU Technology Conference 2016.

Many of our users, including artists and reviewers, acknowledged that this is the best brush painting system they have ever seen!

A unified meshless framework Given the recent research progress in smoothed particle hydrodynamics, peridynamics, and position-based dynamics, I believe that it is now the right time to formulate a meshless framework for unified simulation of both solids and fluids. Unlike previous frameworks [11, 16], my framework will use absolutely no mesh, even for elastic bodies. This framework is attractive only when it is efficient and compatible with GPU acceleration. To this end, my team will explore the transfer of real-time simulation techniques from a mesh-based representation to a meshless representation, which has not been comprehensively studied before.

Other fluid simulation methods Besides particle-based methods, I have also investigated other ways to simulate fluids in recent years. This includes a deformable surface model for water drop simulation [24] and an Eulerian convection scheme for avoiding numerical dissipation and simulating fluid mixing effects [9].

References

[1] Bouaziz, S., Martin, S., Liu, T., Kavan, L., and Pauly, M. Projective dynamics: Fusing constraint projections for fast simulation. *ACM Trans. Graph. (SIGGRAPH) 33*, 4 (July 2014), 154:1–154:11.

[2] Bridson, R., Marino, S., and Fedkiw, R. Simulation of clothing with folds and wrinkles. In *Proceedings of SCA* (2003), pp. 28–36.

[3] Busaryev, O., Dey, T. K., and Wang, H. Adaptive fracture simulation of multi-layered thin plates. *ACM Trans. Graph. (SIGGRAPH) 32*, 4 (July 2013), 52:1–52:6.

[4] Busaryev, O., Dey, T. K., Wang, H., and Ren, Z. Animating bubble interactions in a liquid foam. *ACM Trans. Graph. (SIGGRAPH) 31*, 4 (July 2012), 63:1–63:8.

[5] Chen, Z., Feng, R., and Wang, H. Modeling friction and air effects between cloth and deformable bodies. *ACM Trans. Graph. (SIGGRAPH) 32*, 4 (July 2013), 88:1–88:8.

[6] Chen, Z., Yao, M., Feng, R., and Wang, H. Physics-inspired adaptive fracture refinement. *ACM Trans. Graph. (SIGGRAPH) 33*, 4 (July 2014), 113:1–113:7.

[7] Clausen, P., Wicke, M., Shewchuk, J. R., and O'Brien, J. F. Simulating liquids and solid-liquid interactions with Lagrangian meshes. *ACM Trans. Graph. 32*, 2 (Apr. 2013), 17:1–15.

[8] He, X., Wang, H., Zhang, F., Wang, H., Wang, G., and Zhou, K. Robust simulation of sparsely sampled thin features in SPH-based free surface flows. *ACM Trans. Graph. 34*, 1 (Dec. 2014), 7:1–7:9.

[9] He, X., Wang, H., Zhang, F., Wang, H., Wang, G., Zhou, K., and Wu, E. Simulation of fluid mixing with interface control. In *Proceedings of SCA* (2015), pp. 129–135.

[10] Liu, T., Bargteil, A. W., O'Brien, J. F., and Kavan, L. Fast simulation of mass-spring systems. *ACM Trans. Graph. (SIGGRAPH Asia) 32*, 6 (Nov. 2013), 214:1–214:7.

[11] Macklin, M., Müller, M., Chentanez, N., and Kim, T.-Y. Unified particle physics for real-time applications. *ACM Trans. Graph. (SIGGRAPH) 33*, 4 (July 2014), 153:1–153:12.

[12] Mao Ye, Huamin Wang, N. D. X. Y., and Yang, R. Real-time human pose and shape estimation for virtual try-on using a single commodity depth camera. *IEEE Transactions on Visualization and Computer Graphics (IEEE Virtual Reality) 20*, 4 (2014), 550–559.

[13] Müller, M. Hierarchical position based dynamics. In *Proceedings of VRIPHYS* (2008), pp. 1–10.

[14] Müller, M., Chentanez, N., Kim, T., and Macklin, M. Strain based dynamics. In *Proceedings of SCA* (2014), pp. 21–23.

[15] Sheng Yang, Xiaowei He, H. W. S. L. G. W. E. W., and Zhou, K. Enriching SPH simulation by approximate capillary waves. In *Proceedings of SCA* (2016).

[16] Stam, J. Nucleus: Towards a unified dynamics solver for computer graphics. In *Computer-Aided Design and Computer Graphics* (2009), pp. 1–11.

[17] Tamal K. Dey, Bo Fu, H. W., and Wang, L. Automatic posing of a meshed human model using point clouds. *Computer and Graphics (Shape Modeling International) 16*, C (2015), 14–24.

[18] Thomaszewski, B., Pabst, S., and Strasser, W. Continuum-based strain limiting. *Computer Graphics Forum (Eurographics) 28*, 2 (2009), 569–576.

[19] Wang, H. Defending continuous collision detection against errors. *ACM Trans. Graph. (SIGGRAPH) 33*, 4 (July 2014), 122:1–122:10.

[20] Wang, H. A Chebyshev semi-iterative approach for accelerating projective and position-based dynamics. *ACM Trans. Graph. (SIGGRAPH Asia) 34*, 6 (Oct. 2015), 246:1–246:9.

[21] Wang, H., O'Brien, J., and Ramamoorthi, R. Multi-resolution isotropic strain limiting. *ACM Trans. Graph. (SIGGRAPH Asia) 29*, 6 (Dec. 2010), 156:1–156:10.

[22] Wu, X., Mukherjee, R., and Wang, H. A unified approach for subspace simulation of deformable bodies in multiple domains. *ACM Trans. Graph. (SIGGRAPH Asia) 34*, 6 (Oct. 2015), 241:1–241:9.

[23] Yao, M., Chen, Z., Luo, L., Wang, R., and Wang, H. Level-set-based partitioning and packing optimization of a printable model. *ACM Trans. Graph. (SIGGRAPH Asia) 34*, 6 (Oct. 2015), 214:1–214:11.

[24] Zhang, Y., Wang, H., Wang, S., Tong, Y., and Zhou, K. A deformable surface model for real-time water drop animation. *IEEE Transactions on Visualization and Computer Graphics 18*, 8 (Aug. 2012), 1281–1289.