Towards Fast Biomechanical Modeling of Soft Tissue Using Neural Networks

Felix Meister *
Siemens Healthineers
Digital Technology & Innovation
meister.felix.ext@siemens.com

Tiziano Passerini
Siemens Healthineers
Digital Technology & Innovation
tiziano.passerini@siemens.com

Viorel Mihalef
Siemens Healthineers
Digital Technology & Innovation
viorel.mihalef@siemens.com

Ahmet Tuysuzoglu
Siemens Healthineers
Digital Technology & Innovation
ahmet.tuysuzoglu@siemens.com

Andreas Maier
Pattern Recognition Lab
University of Erlangen-Nuremberg
andreas.maier@fau.de

Tommaso Mansi
Siemens Healthineers
Digital Technology & Innovation
tommaso.mansi@siemens.com

Abstract

To date, the simulation of organ deformations for applications like therapy planning or image-guided interventions is calculated by solving the elastodynamics equations. While efficient solvers have been proposed for fast simulations, methods that are both real-time and accurate are still an open challenge. An ideal, interactive solver would be able to provide physically and numerically accurate results at high frame rate, which requires efficient force computation and time integration. Towards this goal, we explore in this paper for the first time the use of neural networks to directly learn the underlying biomechanics. Given a 3D mesh of a soft tissue segmented from medical images, we train a neural network to predict vertex-wise accelerations for a large time step based on the current state of the system. The model is trained using the deformation of a bar under torsion, and evaluated on different motions, geometries, and hyperelastic material models. For predictions of ten times the original time step we observed a mean error of 0.017 mm $\pm$ 0.014 (0.032) at a mesh size of 50 mm x 50 mm x 100 mm. Predictions at 20dt yield an error of 2.10 mm $\pm$ 1.73 (4.37) and by further increasing the prediction time step the maximum error rises to 38.3 mm due to an artificial stiffening. In all experiments our proposed method stayed stable, while the reference solver fails to converge. Our experiments suggest that it is possible to directly learn the mechanical simulation and open further investigations for the direct application of machine learning to speed-up biophysics solvers.

1 Introduction

Computational modeling of organ biomechanics has been investigated for disease understanding, therapy planning and image-guided interventions [1, 3, 6, 11]. For instance, an image-guided surgery system for the resection of hepatocarcinoma may use the physics-based computation of the

*Pattern Recognition Lab, Friedrich-Alexander-University Erlangen-Nürnberg, felix.meister@fau.de

32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, Canada.

deformation field from a personalized model of a patient's liver and a sparse set of tracked points to robustly register acquisitions from pre- and intraoperative imaging modalities [10]. Simulations rely on numerical solvers that use the time and space discretization of the equations and constitutive laws to compute the deformation. Accurately calculating the motion in real-time is still an open challenge, but necessary for efficient personalization of models which commonly requires iterative approaches. While simple methods like mass-spring systems proved to be fast, the result may not be accurate enough to capture the nonlinear behaviour of organ tissue under large deformations. In contrast, finite element methods (FEM) may yield precise results, but at the price of not being real-time [8]. Common approaches in the literature to improve the performance comprise the usage of reduced order models or warping techniques [2, 12]. Ideally, an interactive solver would be able to provide physically and numerically accurate results at large time-steps, with efficient force computations.

Towards this goal, we explore in this work the application of deep learning to speed-up biomechanical simulations. We hypothesize that a neural network, as a universal function approximator, is able to learn the underlying biomechanics. We evaluate our method by training a neural network on a synthetic dataset with fixed material model and apply it on various geometries, motions, and material models.

2 Methods

Computing soft tissue deformation involves solving the underlying dynamics equation $\mathrm{M}\ddot{\mathbf{u}} + \mathrm{D}\dot{\mathbf{u}} + \mathrm{K}\mathbf{u} = \mathbf{f}_e$, where M is the lumped mass matrix, $\mathrm{D} = \mu\mathrm{M}$ is the Rayleigh damping, K is the stiffness matrix, and $\mathbf{f_e}$ are external forces. $\ddot{\mathbf{u}}$, $\dot{\mathbf{u}}$, and $\mathbf{u}$ are the vertex acceleration, velocity, and displacement, respectively. In this work, the material response is modeled by a standard exponential law derived from [5], although any other elastic model can be used. More precisely, the stress-strain energy function writes $\Psi = a/(2b)\,\exp(b(I_1 - 3)) + d(J - 1)$, where a, b, and d are free parameters, $I_1 = \mathrm{tr}(\mathbf{C})$ is the first invariant of the deformation tensor $\mathbf{C}$, and J the Jacobian determinant of the deformation. Finite element methods yield the solution using either explicit or implicit time integration. Implicit time integration is unconditionally stable, but one must solve a system of linear equations, which is computationally expensive and potentially inaccurate for large time steps. Explicit time integration requires small time steps to be stable and accurate, which results in far more operations to simulate the same interval, but can be easily parallelized and benefit from GPU-based acceleration. Due to these reasons, we consider in this work explicit FEM as the reference and explore the possibility of going beyond their stability limit.

In this work, the goal is to learn a model that predicts point-wise accelerations $\ddot{\mathbf{u}}_{\Delta t}$ for a large time step Δt given the current state of the system $\mathbf{\Phi}(t)$. A speed-up is thus achieved if enough computational steps are skipped in between such that it outweighs the network's execution time. At time t we define $\mathbf{\Phi}(t)$ as the displacement $\mathbf{u}(t)$, the velocity $\dot{\mathbf{u}}_{\Delta t}(t)$, and the total force $\mathbf{f}_T(t) = \mathbf{f}_e - \mathrm{K}\mathbf{u}$, where $\mathrm{K}\mathbf{u}$ denotes the internal force. The system is still parameterizable by the set of tissue parameters and boundary conditions by incorporating the total force. The regression task $\mathcal{R}$ is thus formulated as $\mathcal{R}(\mathbf{\Phi}(t)) = \mathcal{R}(\{\mathbf{f}_T(t), \dot{\mathbf{u}}_{\Delta t}(t), \mathbf{u}(t)\}) = \ddot{\mathbf{u}}_{\Delta t}(t)$. Finally, the displacement is updated as $\mathbf{u}(t + \Delta t) = \ddot{\mathbf{u}}_{\Delta t}(t)\Delta t^2 + 2\mathbf{u}(t) - \mathbf{u}(t - \Delta t)$.

To reduce the number of training samples required for successful learning we further introduce rotation-invariance by transforming the feature vector of each vertex $\mathbf{v}_i$ at time t into a local coordinate system that is defined by the tangent $\mathbf{t}_i(t)$, normal $\mathbf{n}_{i1}(t)$, and bi-normal $\mathbf{n}_{i2}(t) = \mathbf{t}_i(t) \times \mathbf{n}_{i1}(t)$ of its trajectory. For consistency over time we apply the parallel transport algorithm proposed by Hanson and Ma (see [4]), which repeatedly rotates an initial coordinate frame at $t = 0\,\mathrm{s}$ by the angle between two subsequent tangent vectors. By defining the tangent vector as the velocity, we can compress $\dot{\mathbf{u}}_{\Delta t}(t)$ to its magnitude, leading to a more robust feature vector consisting of seven values, which comprise the 3-D displacement vector, the velocity magnitude, and the 3-D total force vector.

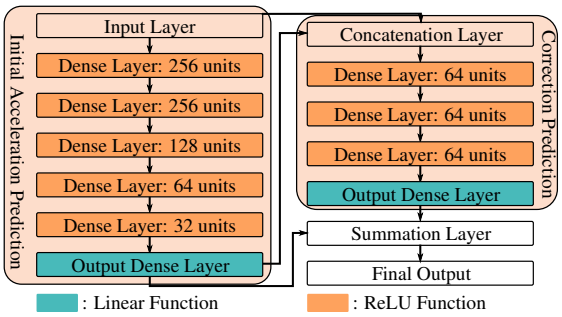

Figure 1: Architecture details.

We feed these features point-wise to a network of two functional parts (Figure 1). The first one uses five fully-connected layers with a decreasing number of leaky rectified linear units and a linear output layer to make an initial prediction of $\ddot{\mathbf{u}}_{\Delta t}$. To mitigate the error propagation over time, we apply a second network that uses the input and output of the first part as well as three non-linear and a single linear layer to compute the correction. Finally, we sum up both predictions to form the final output, which is clipped to the observed value range to discard outliers. Two separate mean-squared-error loss functions with respect to the ground truth acceleration are applied to both the acceleration prediction and the final output. The network is trained for 200 epochs using Adam optimizer with a learning rate of 0.001 and exponential decay parameters set to $\beta_1 = 0.9$ and $\beta_2 = 0.999$ [7].

3 Experiments & Discussion

We generated a synthetic training database by simulating the torsion of a regularly structured rod. The parameters of the exponential law are set to $a = 0.059$, $b = 8.023$, and $c = 60$. The bar consisted of 1458 points and 6528 tetrahedra, with a size of 50 mm x 50 mm x 100 mm. The motion was induced by fixing one side of the bar and applying an external force to the four corner points of the opposite side. The torsion was simulated for one second using the total lagrangian explicit dynamics (TLED) algorithm at a time step of dt $= 3e^{-5}$ s, the limit of stability in this configuration [9]. We discarded the first 0.3 s due to high-amplitude noise arising from the initial transient state. From the 0.7 s a total of 1,000 frames were sampled in equal intervals. For each frame we further rejected the points of the fixed plane and the four corner points due to the their discontinuous behaviour caused by the applied boundary conditions.

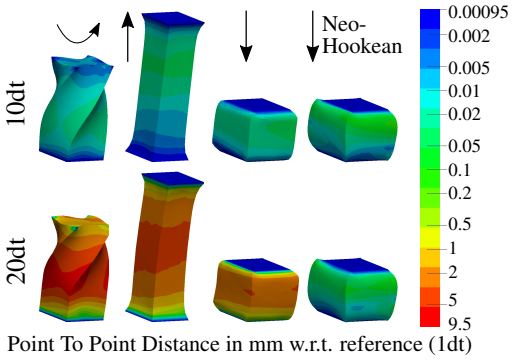

Figure 2: Error of the computed deformation w.r.t. the ground truth for various force induced motions (arrows) calculated with FEM solver at 1dt.

Our approach was first evaluated by training two networks to predict the acceleration for a time step of 10dt and 20dt, which was hence beyond TLED's stability limit, and applying them to four unseen deformations of the same bar: reversed torsion, stretch, compression, and compression under Neo-Hookean material response. The Neo-Hookean stress strain energy function was defined as $\Psi = c(I_1 - 3) + d(J - 1)^2$, whose governing parameters are set to $c = 0.5$ and $d = 15$. To be consistent with the training procedure, we applied a 'burn-in' phase by simulating the first 0.3 s with TLED at dt before using our neural network at 10dt & 20dt, respectively. We observed that for a prediction of 10dt the result is almost indistinguishable to the ground truth with a sub-millimeter mean error of 0.017 mm $\pm$ 0.014 (0.032) over the entire mesh at $t = 1.0$ s (see Figure 2). In contrast, the 20dt prediction exhibits a minor artificial stiffening resulting in a mean error of 2.10 mm $\pm$ 1.73 (4.37).

Finally, we trained two networks for a time step of 50dt and 100dt on the same rod training set, and applied them to simulate the bending of a cylinder and the oscillation of a liver lobe. The simulations were stable at these time steps, which are far beyond the stability limit of this setup. We discovered that in both cases the artifical stiffening is more profound compared to the 20dt simulation, thus leading to a larger maximum error of 38.3 mm (see Figure 3). This may be due to too few training samples and a too limited capacity of our network to learn the increased complexity.

4 Conclusion

In this work we explored the application of deep learning to speed-up the simulation of soft tissue deformation in medical imaging applications. In essence, a neural network is trained to predict point-wise accelerations for a larger time step than possible with traditional explicit finite element methods based on the total force, velocity, and displacement of the current state, which are transformed into a local coordinate system. Our network has been trained on one synthetic simulation and validated under different geometries, motions, and materials. We observed that for a prediction of ten and twenty times the original time step the resulting error is sufficiently small

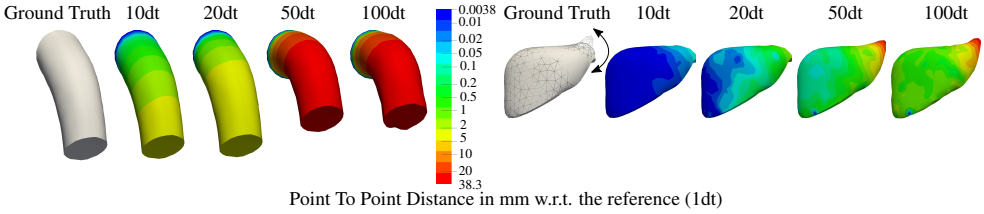

Figure 3: Illustration of the influence of the time step on the prediction of cylinder bending and liver lobe oscillation experiment. Excellent results up to 20dt, i.e. 20 times the inherent time step of the explicit solver, and artificial stiffening for further increased time steps.

w.r.t. the mesh dimension and the proposed method exceeds the stability of the reference FEM solver. While being beyond the stability limit, for larger time advances an artificial stiffening was detected, which is observed in implicit FEM solvers as well. These results suggest that the proposed AI-based solver has both beneficial properties of explicit and implicit FEM methods, namely a straight-forward parallelizibility and an extended stability. To the best of our knowledge, this is the first time a neural network was used to directly learn the biomechanical forces to speed-up the deformation simulations. Possible future research may use recent advances of recurrent neural networks to exploit the sequential nature of the problem for improved accuracy at very high time steps.

Disclaimer: This feature is based on research and is not commercially available. Due to regulatory reasons, its future availability cannot be guaranteed.

References

[1] A. E. Anderson, B. J. Ellis, and J. A. Weiss. Verification, validation and sensitivity studies in computational biomechanics. *Comput Methods Biomech Biomed Engin*, 10(3):171–184, Jun 2007. 17558646[pmid].

[2] M. G. Choi and H.-S. Ko. Modal warping: Real-time simulation of large rotational deformation and manipulation. *IEEE Transactions on Visualization and Computer Graphics*, 11(1):91–101, 2005.

[3] H. Delingette, X. Pennec, L. Soler, J. Marescaux, and N. Ayache. Computational models for image-guided robot-assisted and simulated medical interventions. *Proceedings of the IEEE*, 94(9):1678–1688, Sept 2006.

[4] A. Hanson and H. Ma. Parallel transport approach to curve framing. *Indiana University, Techreports-TR425*, 11:3–7, 1995.

[5] G. A. Holzapfel and R. W. Ogden. Constitutive modelling of passive myocardium: a structurally based framework for material characterization. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 367(1902):3445–3475, 2009.

[6] E. Kayvanpour, T. Mansi, F. Sedaghat-Hamedani, A. Amr, D. Neumann, B. Georgescu, P. Seegerer, A. Kamen, J. Haas, K. S. Frese, M. Irawati, E. Wirsz, V. King, S. Buss, D. Mereles, E. Zitron, A. Keller, H. A. Katus, D. Comaniciu, and B. Meder. Towards personalized cardiology: Multi-scale modeling of the failing heart. *PLOS ONE*, 10(7):1 – 18, 07 2015.

[7] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[8] U. Meier, O. López, C. Monserrat, M. Juan, and M. Alcaniz. Real-time deformable models for surgery simulation: a survey. *Computer methods and programs in biomedicine*, 77(3):183–197, 2005.

[9] K. Miller, G. Joldes, D. Lance, and A. Wittek. Total lagrangian explicit dynamics finite element algorithm for computing soft tissue deformation. *International Journal for Numerical Methods in Biomedical Engineering*, 23(2):121–134, 2007.

[10] D. C. Rucker, Y. Wu, L. W. Clements, J. E. Ondrake, T. S. Pheiffer, A. L. Simpson, W. R. Jarnagin, and M. I. Miga. A mechanics-based nonrigid registration method for liver surgery using sparse intraoperative data. *IEEE transactions on medical imaging*, 33(1):147–158, 2014.

[11] A. Wittek, N. M. Grosland, G. R. Joldes, V. Magnotta, and K. Miller. From finite element meshes to clouds of points: A review of methods for generation of computational biomechanics models for patient-specific applications. *Annals of Biomedical Engineering*, 44(1):3–15, Jan 2016.

[12] Y. Yang, D. Li, W. Xu, Y. Tian, and C. Zheng. Expediting precomputation for reduced deformable simulation. *ACM Transactions on graphics (TOG)*, 34(6), 2015.