

Physics-Informed Spatiotemporal Deep Learning for Emulating Coupled Dynamical Systems

Anishi Mehta,^{1,3} Cory Scott,^{2,3} Diane Oyen,³ Nishant Panda,³ Gowri Srinivasan³

¹Georgia Institute of Technology, ²University of California-Irvine, ³Los Alamos National Laboratory

Abstract

Accurately predicting the propagation of fractures, or cracks, in brittle materials is an important problem in evaluating the reliability of objects such as airplane wings and concrete structures. Efficient crack propagation emulators that can run in a fraction of the time of high-fidelity physics simulations are needed. A primary challenge of modeling fracture networks and the stress propagation in materials is that the cracks themselves introduce discontinuities, making existing partial differential equation (PDE) discovery models unusable. Furthermore, existing physics-informed neural networks are limited to learning PDEs with either constant initial conditions or changes that do not depend on the PDE outputs at the previous time. In fracture propagation, at each time-step, there is a damage field and a stress field; where the stress causes further damage in the material. The stress field at the next time step is affected by the discontinuities introduced by the propagated damage. Thus, both stress and damage fields are heavily dependent on each other; which makes modeling the system difficult. Spatiotemporal LSTMs have shown promise in the area of real-world video prediction. Building on this success, we approach this physics emulation problem as a video generation problem: training the model on simulation data to learn the underlying dynamic behavior. Our novel deep learning model is a *Physics-Informed Spatiotemporal LSTM*, that uses modified loss functions and partial derivatives from the stress field to build a data-driven coupled dynamics emulator. Our approach outperforms other neural net architectures at predicting subsequent frames of a simulation, enabling fast and accurate emulation of fracture propagation.

Introduction and Motivation

Brittle materials fail suddenly with little warning due to the growth of micro-fractures that quickly propagate and coalesce. Prediction of fracture propagation in brittle materials is a multi-scale modeling problem whose time dynamics are well understood at the micro-scale but do not scale well to the macro-scale necessary for practical evaluation of materials under strain (White 2006; Hyman et al. 2016; Kim et al. 2014). Fracture formation in brittle materials

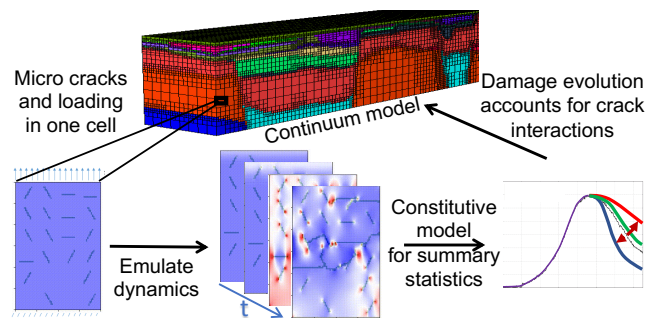


Figure 1: Machine learning emulates dynamics of micro cracks to inform the continuum model.

is typically simulated using parallel implementations of finite discrete element methods (FDEM). Industrial software packages applying these methods have been developed, many of which are capable of representing the high-fidelity dynamics and are extremely parallelized (Hyman et al. 2015; Rougier et al. 2014). Yet, these codes are unable to simulate samples large enough to have real-world scientific applications, due to the large computational requirements of simulating the behavior at the spatial and temporal resolutions necessary. Upscaled continuum representations are used as an approximation because they discard topological features of the simulated material and are therefore faster; however, precisely because they omit these features, they fail to match experimental observations (Vaughn et al. 2019). Thus, we develop a spatio-temporal machine learning model to emulate the micro-scale physics model and estimate the necessary quantities of interest needed to ensure accuracy of the continuum-scale model, as in Figure 1.

The goal is to predict summary statistics, or quantities of interest, for both the damage field and the stress field in a simulated 2-dimensional material from initial conditions until the point of failure (when a single fracture spans the width of the material). The dynamics of the stress field cannot be modeled without the damage and vice versa. When damage is static, the evolution of stress over the material mimics properties of fluid flow. However, the damage caused in the material changes the behavior of stress to no longer be

governed by a single PDE, e.g. stress accumulates at crack tips and causes cracks (each of which is a discontinuity in the stress field) to spread further. Thus, instead of using solid state dynamics equations to predict this stress field, we must extend approaches successfully demonstrated in machine learning to couple the dynamics of the damage field and stress field.

It is tempting to treat damage and the stress tensor at each location simply as different channels in the same time series and apply methods from the extensive prior work on video prediction (Wang et al. 2018). However this approach is ineffective because although the damage and stress fields are highly coupled, they have dramatically different dynamics in time. Therefore, one model cannot easily predict both quantities simultaneously. The damage data is binary-valued and sparse: most of the finite elements remain undamaged for the entire simulation, as shown in Figure 2a. The stress data is real-valued, where values as small as 10^{-6} are significant yet magnitudes also range up to 10^8 (see Figure 2); and the stress field has spatial discontinuities wherever damage has occurred. Furthermore, unlike video prediction which is concerned with precise pixel-by-pixel accuracy, we need to emulate the most important features of the simulation over a long time horizon (hundreds of frames in the future) with high enough accuracy to predict several quantities of interest needed by the continuum model.

In order to capture the long-term frame dependencies, recurrent neural networks (RNNs) (Williams and Zipser 1995) have been recently applied to video predictive learning. Former state-of-the-art models applied complex nonlinear transition functions from one frame to the next, constructing a dual memory structure (Wang et al. 2018) upon Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber 1997b). To emulate the spatio-temporal model, we propose a *Physics-Informed Spatiotemporal LSTM* model. First, linear interpolation is used to coarsen the damage data to retain the important fracture features while discarding the uninformative undamaged regions. Next, a modified recurrent neural network learns temporal evolution in the latent space representation (Wang et al. 2018). Finally, the predictions from the recurrent neural network are passed to the decoder sub-network of the convolutional autoencoder, and decoded into time-advanced simulation states. As input to the convolutional autoencoder network, we include point estimates of partial derivatives of stress values. This allows us to predict *Coupled Dynamical PDEs* unlike existing PDE discovery models.

Results show that this approach makes accurate predictions of fracture propagation. Our method outperforms other neural net architectures at predicting subsequent frames of a simulation, and reproduces physical quantities of interest with higher fidelity.

Related Work

Machine learning-based prediction of behavior in physical systems in general, and partial differential equations specifically, is an area of active research. Broadly, machine learning approaches to PDE emulation fall into one of two categories. In the first category are approaches that acceler-

ate methods to solve a PDE whose form is known using data; for example, (Han, Jentzen, and E 2018). (Long, She, and Mukhopadhyay 2018) uses convolutions in the LSTM cells in a fully convolutional network to train a PDE solver with varying input perturbations. Our work falls under the second category of approaches; which emulate the behavior of a system governed by PDEs; such as fluid dynamics (Kim et al. 2019; Wiewel, Becher, and Thuerey 2019; White, Ushizima, and Farhat 2019; Guo, Li, and Iorio 2016). Unlike these fluid dynamics emulators where the boundary conditions and topology are constant, in our case the evolution of the damage field changes both the boundary conditions and topology. Furthermore, our problem has a bi-directional relationship between stress (which is governed by a PDE, when damage is constant) and damage (which is not) and hence we cannot fit a simple PDE and simply unroll forward in time.

Physics-Informed Spatiotemporal Model

Formally, the problem to solve is: given initial conditions, predict a time series of damage field and stress field evolution. The initial conditions given to this generative model are some number of simulated frames, from which the rest of the time-series is predicted.

Architecture of the deep learning model

Our data-driven approach to predict physical behavior in a complex system leverages advances in deep neural networks (Hinton et al. 2012). We use a Convolutional Neural Network (CNN) (Krizhevsky, Sutskever, and Hinton 2012) to learn a nonlinear mapping from the stress and damage values in local neighborhoods at time t to the stress and damage fields at the next time-step. CNNs are designed for problems with high spatial correlation and translation invariance, making them an ideal choice for physical problems.

In prior work, we found that using a CNN alone to make predictions at the next time step, tends to make biased predictions of lower stress and damage values than the truth. As we unroll the predictions over time, these errors compound resulting in highly inaccurate predictions of stress values after 10 or so frames in time; and virtually no predictions of damage occurring. We incorporate an explicit modeling of the time component using a recurrent neural network (RNN) which shares weights over subsequent time-steps of the input (Pearlmutter 1989). The hidden state of the RNN after consuming an entire time series thus is a fixed-length encoding of that (varying-length) time series. Specifically, we use a Long Short-Term Memory network (LSTM) that allows the network to separately “remember” both long-term *global* context, as well as short-term *recent* context (Hochreiter and Schmidhuber 1997a).

The spatial and temporal elements can be combined with a *Convolutional LSTM* or ConvLSTM which maintains the spatial structure of the input as it processes time series. We find that the best model is a Spatiotemporal LSTM (ST-LSTM) (Lu, Hirsch, and Scholkopf 2017). The main reason for the improved predictive power is the inclusion of the *Spatiotemporal Memory* in each LSTM block in addition to

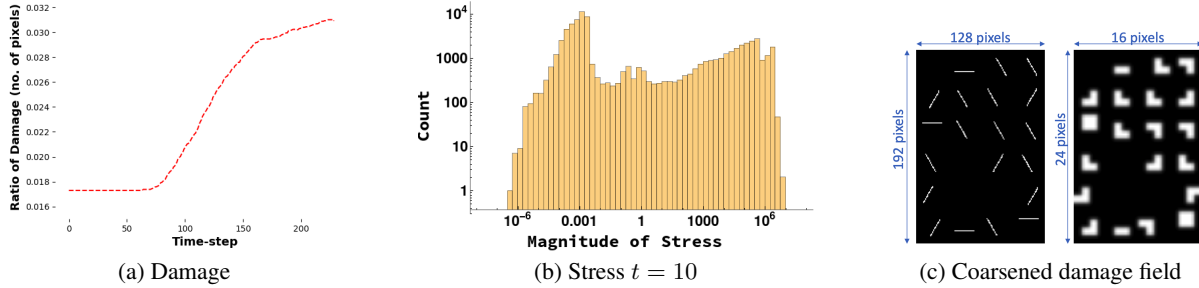


Figure 2: (a) Ratio of pixels that are damaged versus time. (b) Histogram of absolute value of stress values at one time frame, for an example simulation run. Note that values range from very small ($\approx 10^{-8}$) to very large ($\approx 10^8$). (c) Example of coarsened damage field at the initial time frame for an example simulation. Field is coarsened using the Lanczos method by a factor of 8.

the *Temporal Memory*. While temporal states are only shared horizontally between time-steps, the spatiotemporal state is shared between the stacked ST-LSTM blocks. This enables efficient flow of spatial information. We make the memory representations of the ST-LSTM cells common between all input fields. This feature allows us to model the highly co-dependent nature of the stress and damage fields. Figure 3 shows our novel physics-informed architecture. We introduce various aspects inspired by the physical properties of the damage propagation problem allowing for a closer fitting PDE.

Coarsening of input damage images

The damage field is very sparse with damaged pixels forming less than 2% of the entire spatial domain. The increase in damaged pixels from the initial seed damage at $t = 0$, to the damage at the final step when the sample has failed, is less than 0.2% of the total pixels, as shown in Figure 2a. This makes it difficult for an ML model to capture and predict this information, since (formulated as a binary classification task) the two prediction classes are extremely imbalanced. Furthermore the distance between cracks is quite large relative to the size of a crack which complicates the use of convolutional filters. Hence, we coarsen the damage data with a linear Lanczos method (Lanczos 1950) with a filter of 3×3 (see Figure 2c). We then convert the damage field to a binary 0-1 field by applying a threshold of 0.11 which is a standard threshold in this domain beyond which damage cannot be repaired, i.e. any pixels with values higher than this will be considered as a damaged pixel and all others are non-damaged. In this manner, we effectively coarsen the fields by a factor of 8. Empirically, we find that this coarsening method preserves the important features to accurately predict physical quantities of interest.

Informing the model with partial derivatives

The FDEM model that we are emulating is a Markovian process: the damage and stress fields of the next time-step are completely determined by the current state. Unlike the FDEM model, the machine learning model does not have the actual PDE to define the dynamics, and so we predict each time-step from up to k previous time-steps. We use $k = 3$ so

that dynamic information can be observed.

The stress field, without damage, follows a 2nd-order PDE. To build a deep learning model that fits to such PDEs, we include the 1st and 2nd order partial spatial and temporal derivatives as input. Using $k = 3$ time-steps as input allows us to capture temporal derivative information accurately. We use the gradient and Hessian calculating functions of Tensorflow to calculate the gradients (Abadi and others 2015). At each step, we append the derivatives to the input fields and predict them as part of the next-step prediction. This enables the spatiotemporal memory blocks of the network to carry information about spatial and temporal derivatives of stress. Through this, we overcome the issue of vanishing gradients discussed in (Wang et al. 2018), as well as capture the monotonically increasing nature of the damage field. We ensure the mean squared error of the predicted derivatives and derivatives calculated from the stress fields lies within a pre-decided threshold σ as a self-check. These modeling choices were imbibed from the physical principles governing the damage propagation process creating a novel "physics-informed" deep learning model. Empirical results show that this physics-informed approach of training the model significantly improves accuracy.

Experiment

We focus on three variants of LSTM models: Stacked LSTM, convLSTM, and ST-LSTM. All three models take in the first $k = 3$ time-steps of coarsened data as input frames. Next, to encourage the model to fit to a PDE, we calculate the 1st and 2nd derivatives of the stress field w.r.t. time and append that information to the inputs. The LSTM block calculates predicted values for the next frame corresponding to each pixel in the input frames. Each model with derivatives as inputs is called Physics-Informed. The whole model is unrolled to predict the entire simulation.

The dataset consists of 61 simulations each of which has 260 time-steps. We split our dataset into 41 simulations used for training, 10 for validation, and 10 as test cases. We train our models until saturation which we reach between 350-400 epochs. To prevent the model from overfitting, we perform one round of cross-validation at the end of each epoch.

Our models are designed to allow us to plugin different

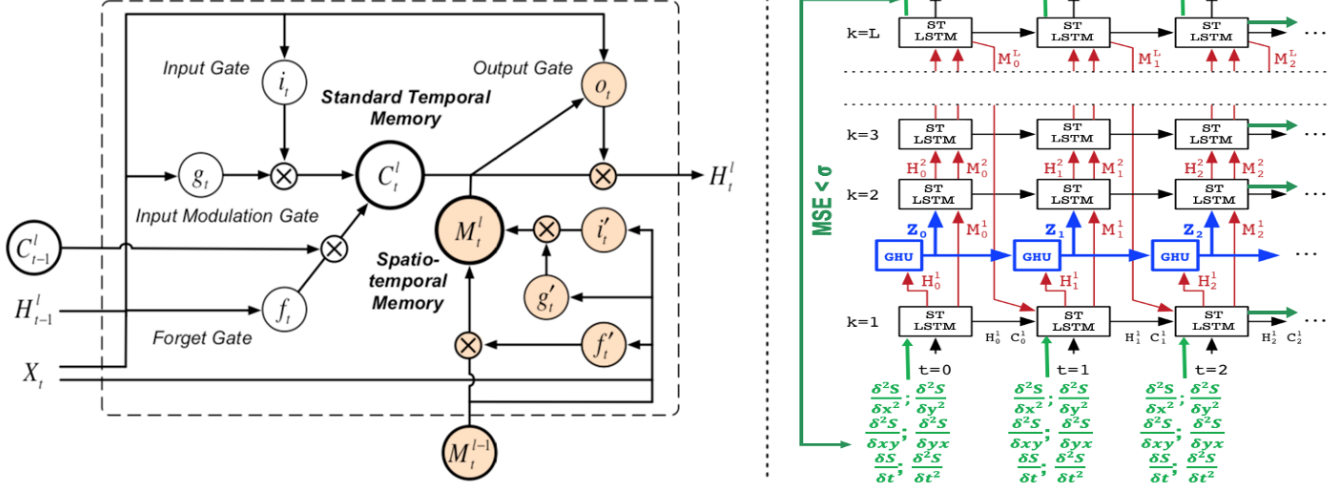


Figure 3: Left: a basic SpatioTemporal LSTM (ST-LSTM) cell. Right: The novel Physics-Informed LSTM generative model. The spatial and temporal memories work in parallel: the red lines denote the deep transition paths of the spatial memory, while horizontal black arrows indicate the update directions of the temporal memories. The green lines indicates direction flow of the stress field derivatives. Gradient Highway Units (GHU) help overcome the problem of vanishing gradients.

ML architectures, choose whether to include partial derivative information, and test various loss functions. This modular approach makes it easy to train the necessary components as needed. Our experiments show that we achieve the best performance by using 6 ST-LSTM blocks stacked on top of each other, each of size 128. We use tanh (Hochreiter and Schmidhuber 1997a) as the activation function for the LSTM and Leaky-ReLu (Maas, Hannun, and Ng 2013) for the CNN. For our experiments, we test several combinations of loss functions such as L1 loss, L2 loss, L2 loss weighted by pixel values, cross-entropy loss, etc. We use the same loss functions for all our models to directly compare performance. For the best results, we treat the damage fields as a binary classification problem i.e. deducing whether a given pixel i, j is damaged or not, and use a cross-entropy loss (Goodfellow, Bengio, and Courville 2016). The cross-entropy loss L_D for the damage field is:

$$L_D = -0.5 \sum_{i,j} \sum_{c=\{1,2\}} y_{D_{ij},c} \log(p_{D_{ij},c}), \quad (1)$$

where y is a binary indicator (0 or 1) if class label c is the correct prediction for damage field observation D_{ij} and p is the probability of the model predicting class c for damage field observation D_{ij} .

For the stress fields, we use L1 and L2 losses and a gradient difference loss (GDL) which sharpens the image prediction (Mathieu, Couprie, and LeCun 2016). The stress field

loss function L_S is given by:

$$L_S = \alpha_3 L_{\text{GDL}} + \sum_{i,j} \left(\alpha_1 (S_{ij} - \hat{S}_{ij})^2 + \alpha_2 |S_{ij} - \hat{S}_{ij}| \right),$$

$$L_{\text{GDL}} = \sum_{i,j} \left(\left| |S_{i,j} - S_{i-1,j}| - |\hat{S}_{i,j} - \hat{S}_{i-1,j}| \right| + \left| |S_{i,j-1} - S_{i,j}| - |\hat{S}_{i,j-1} - \hat{S}_{i,j}| \right| \right), \quad (2)$$

where i, j ranges over the pixels, \hat{S}_{ij} are the predicted stress values, S_{ij} are the true values and α_1 , α_2 , and α_3 are hyperparameters that weight the relative importance of each term of the loss function. We use $\alpha_1 = 0.3$, $\alpha_2 = 0.1$, and $\alpha_3 = 0.1$.

Prediction of quantities of interest

The continuum-scale model requires as input several quantities that describe a material behavior under given conditions. These quantities of interest (QoI) are: (a) number of cracks as a function of time; (b) distribution of crack lengths as a function of time; and (c) maximum stress over the field as a function of time. To predict these quantities of interest, we collect stress and damage predictions from our physics-informed spatiotemporal generative model; then, we calculate the QoI.

Evaluation Metrics

We evaluate model performance with two standard video similarity metrics and by quantifying the prediction of quantities of interest. **MSE**: Mean Squared Error compares the squared difference between prediction and truth, averaged

Table 1: Evaluation of models. Low is good for **MSE** and **QoI**. High is good for **SSIM**. Best score is **bold**.

Model	MSE	SSIM	QoI
Stacked LSTM	8.14	0.61	0.35
Physics-Informed Stacked LSTM	6.81	0.72	0.31
ConvLSTM	3.73	0.86	0.28
Physics-Informed ConvLSTM	2.10	0.87	0.21
ST-LSTM	1.55	0.94	0.12
Physics-Informed ST-LSTM	1.23	0.92	0.09

over all pixels. **SSIM**: The Structural Similarity Index Metric considers perception-based similarity between two images (Wang et al. 2004). Note that higher is better for SSIM. **QoI**: We weigh the quantities of interest (QoI) defined above equally and measure the mean absolute error which indicates how well the continuum model will perform with this model as an emulator.

Results

Our Physics-Informed ST-LSTM outperforms other models particularly on MSE and on predicting the QoI, as shown in Table 1. ST-LSTM does perform slightly better according to the SSIM metric, but the difference is small and SSIM measures visual similarity which is not our main goal. Qualitatively, we see in Figures 5 and 6 our Physics-Informed ST-LSTM model can faithfully emulate both the stress and damage field propagation. The Stacked LSTM model in particular, tends to predict overly smooth stress fields and no change in damage, even with the Physics Informed model.

Our model learns an approximation to the physical equations governing the evolution of stress and damage fields allowing it to make predictions on previously unseen conditions. The quantities of interest are then extracted from these predictions. As an example, Figures 4 and 7 show the results for these quantities of interest for a held-out test simulation. From this example, we can see generally that our model predicts cracks coalescing with neighbor cracks slightly earlier than when it actually occurs; causing (a) the total damage to be overestimated, (b) the number of cracks are underestimated, and (c) the length of individual cracks are overestimated during the most dynamic parts of the simulation. This is likely due to the coarsening of the damage field and is not a major concern. We predict the entire stress field for all three directions (or channels) of stress and then extract the maximum value from our prediction to compare against the maximum value in the ground truth stress field. Figure 7 shows that our model routinely under-estimates the maximum stress value, yet generally gets the trend and peaks of the time series. This is a typical result from machine learning prediction, which tends not to predict extreme values. We could improve our prediction of this quantity by optimizing specifically for the prediction of the maximum stress rather than predicting the entire stress field, but leave this for future work.

Run-time and speed-up

High-fidelity simulators for material failure are computationally expensive, taking on the order of 1500 CPU-hours to run one simulation of a 2-dimensional material for 260 time-steps, such as in the dataset we use (Rougier et al. 2014). Physics-Informed ST-LSTM accelerates the entire workflow by generating approximate QoI in a fraction of the time. We train each model to saturation in 10-12 hours on four GeForce GTX 1080 Ti 2.20GHz GPUs, after which emulation of the physical behavior is on the order of milliseconds, rather than minutes, per timestep. This is a speedup on the order of 50,000 times faster. Furthermore, once trained, the model can generate QoI for any number of simulations drawn from the same initial conditions.

Discussion

The complexity of our model architecture and loss functions are necessary for accurately emulating a complex spatiotemporal process over a long time horizon. The LSTM learns the monotonically increasing nature of the damage field without any constraints being imposed. This physically-plausible learned model is an important result that favored the use of LSTMs that can capture time-dependent evolution better than conventional neural network architectures. Explicitly calculating the partial derivatives and including them as input improves prediction. This is because the model now fits to a PDE which is a closer approximation to the original physical problem. The dual memory representation of spatial and temporal information in our ST-LSTM cell improves performance of our model on this problem significantly. The failure of Stacked-LSTM (Hermans and Schrauwen 2013) is also evidence of this. Furthermore, both local and global spatio-temporal information is important to reduce compounding errors to make predictions at any given time.

We see that the maximum stress is consistently under-predicted, even after weighting the losses by actual stress values. We believe this is due to the inherent nature of ML finding an average representation from training data and the inherently difficult inference problem of estimating a maximum statistic. However, an important point to take note of is that our model is able to follow the peaks and trends of the maximum stress quite accurately. Future work in uncertainty quantification could learn the correction in our maximum stress estimate.

The damage model tends to predict crack coalescence early. We coarsen the simulation data before giving it as input to our model, which proportionately reduces the non-damaged regions between cracks. Due to this, our model tends to predict crack coalescence a few steps earlier than ground truth. However, the model is able to converge to the correct number of cracks towards the end of the simulations (see Figure 4). In future work, learning a coarse representation, such as with a convolutional autoencoder (Masci et al. 2011), could learn to correct this bias.

Conclusion

Emulation of complex physical systems has long been a goal of artificial intelligence because although we can write down

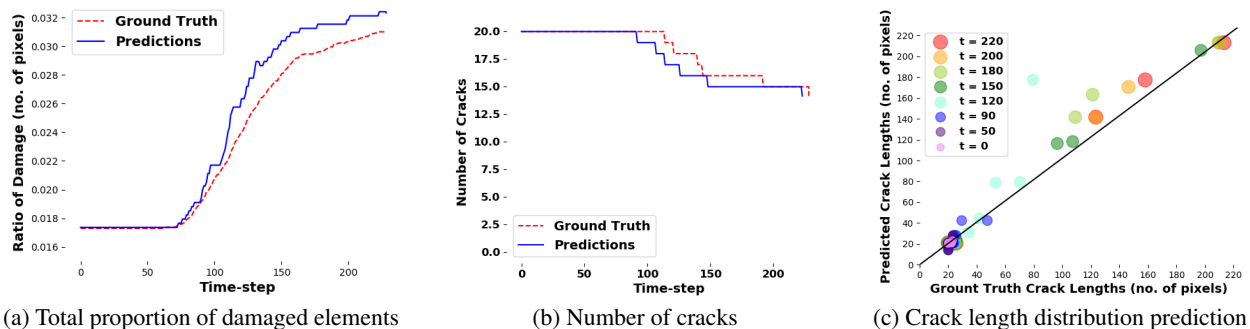


Figure 4: Example predicted damage quantities of interest for one held-out simulation.

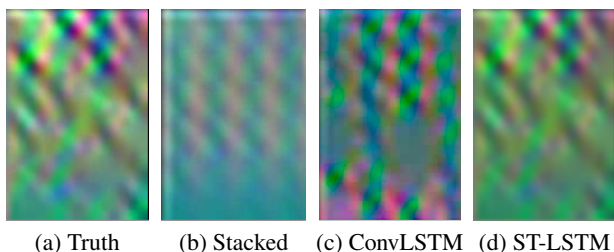


Figure 5: Example predicted stress field of a test frame. Results shown are all from the Physics-Informed version of the model. The three directions of the stress field are visualized with false-color by mapping each channel of stress direction (S_{xx} , S_{xy} , S_{yy}) to an image color channel (red, green, blue).

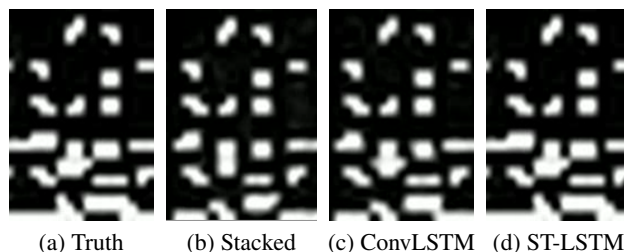


Figure 6: Example predicted damage field of a test frame. Results shown are all from the Physics-Informed version of the model. White means damaged and black means not damaged.

the micro-scale physics equations of such a system, it is computationally intractable to simulate the physics model to obtain meaningful predictions on a large scale; yet the macro-scale patterns of these dynamic systems can be quite intuitive to humans (Lerer, Gross, and Fergus 2016). We present Physics-Informed ST-LSTM, an extension and application of Spatiotemporal LSTM (ST-LSTM) neural network models to emulate the time dynamics of a physical simulation of stress and damage in a material. Unlike PDE emulators that assume a PDE form, our entirely data driven framework, can be used equally well on high dimensional experimental studies where binary variables can arise. We demonstrate that ST-LSTMs outperform two other machine learning models at predicting these time dynamics and physical quantities of interest, and furthermore that all three models increase in performance when they are physics-informed, that is they have access to the underlying physics of the simulation. Physics information comes both in the form of spatiotemporal derivatives, and in a loss function which takes into account the QoI. We furthermore demonstrate that a reduced-order model can gainfully capture the time dynamics of these physical QoI without needing pixel-perfect accuracy, an important step towards using machine learning to massively accelerate prediction of complex physics.

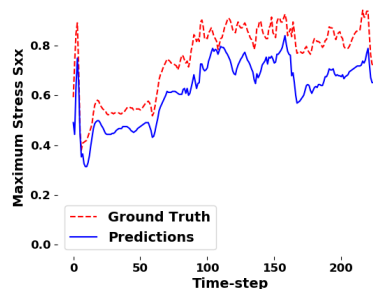


Figure 7: Maximum stress for a heldout simulation.

Acknowledgments

Research supported by the Laboratory Directed Research and Development program of Los Alamos National Laboratory (LANL) under project number 20170103DR. AM supported by the LANL Applied Machine Learning Summer Research Fellowship. CS supported by the LANL Center for Non-Linear Studies.

References

Abadi, M., et al. 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.
 Goodfellow, I.; Bengio, Y.; and Courville, A. 2016. *Deep Learning*. MIT Press.

- Guo, X.; Li, W.; and Iorio, F. 2016. Convolutional neural networks for steady flow approximation. In *ACM SIGKDD Int. Conference on Knowledge Discovery and Data Mining*.
- Han, J.; Jentzen, A.; and E, W. 2018. Solving high-dimensional partial differential equations using deep learning. *Proceedings of the National Academy of Sciences*.
- Hermans, M., and Schrauwen, B. 2013. Training and analysing deep recurrent neural networks. In *Neural Information Processing Systems*.
- Hinton, G.; Deng, L.; Yu, D.; Dahl, G.; Mohamed, A.-r.; Jaitly, N.; Senior, A.; Vanhoucke, V.; Nguyen, P.; Kingsbury, B.; et al. 2012. Deep neural networks for acoustic modeling in speech recognition. *IEEE Signal Processing Magazine*.
- Hochreiter, S., and Schmidhuber, J. 1997a. Long short-term memory. *Neural Computation*.
- Hochreiter, S., and Schmidhuber, J. 1997b. Long short-term memory. *Neural Computation* 9(8):1735–1780.
- Hyman, J. D.; Karra, S.; Makedonska, N.; Gable, C. W.; Painter, S. L.; and Viswanathan, H. S. 2015. dfnworks: A discrete fracture network framework for modeling subsurface flow and transport. *Computers and Geosciences*.
- Hyman, J.; Jiménez-Martínez, J.; Viswanathan, H. S.; Carey, J. W.; Porter, M. L.; Rougier, E.; Karra, S.; Kang, Q.; Frash, L.; Chen, L.; et al. 2016. Understanding hydraulic fracturing: A multi-scale problem. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*.
- Kim, J.; Um, E. S.; Moridis, G. J.; et al. 2014. Fracture propagation, fluid flow, and geomechanics of water-based hydraulic fracturing in shale gas systems and electromagnetic geophysical monitoring of fluid migration. In *SPE Hydraulic Fracturing Technology Conference*.
- Kim, B.; Azevedo, V. C.; Thuerey, N.; Kim, T.; Gross, M.; and Solenthaler, B. 2019. Deep fluids: A generative network for parameterized fluid simulations. In *Computer Graphics Forum*.
- Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*.
- Lanczos, C. 1950. *An Iteration method for the solution of the eigenvalue problem of linear differential and integral operators*. United States Governm. Press Office.
- Lerer, A.; Gross, S.; and Fergus, R. 2016. Learning physical intuition of block towers by example. In *International Conference on Machine Learning*.
- Long, Y.; She, X.; and Mukhopadhyay, S. 2018. Hybridnet: Integrating model-based and data-driven learning to predict evolution of dynamical systems. In *Conference on Robot Learning*.
- Lu, C.; Hirsch, M.; and Scholkopf, B. 2017. Flexible spatio-temporal networks for video prediction. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Maas, A. L.; Hannun, A. Y.; and Ng, A. Y. 2013. Rectifier nonlinearities improve neural network acoustic models. In *Proceedings of International Conference on Machine Learning*.
- Masci, J.; Meier, U.; Cireşan, D.; and Schmidhuber, J. 2011. Stacked convolutional auto-encoders for hierarchical feature extraction. In *Artificial Neural Networks and Machine Learning*.
- Mathieu, M.; Couprie, C.; and LeCun, Y. 2016. Deep multi-scale video prediction beyond mean square error. In *International Conference on Learning Representations*.
- Pearlmutter, B. A. 1989. Learning state space trajectories in recurrent neural networks. *Neural Computation*.
- Rougier, E.; Knight, E. E.; Broome, S. T.; Sussman, A. J.; and Munjiza, A. 2014. Validation of a three-dimensional finite-discrete element method using experimental results of the split hopkinson pressure bar test. *International Journal of Rock Mechanics and Mining Sciences*.
- Vaughn, N.; Kononov, A.; Moore, B.; Rougier, E.; Viswanathan, H.; and Hunter, A. 2019. Statistically informed upscaling of damage evolution in brittle materials. *Theoretical and Applied Fracture Mechanics*.
- Wang, Z.; Bovik, A. C.; Sheikh, H. R.; and Simoncelli, E. P. 2004. Image quality assessment: from error visibility to structural similarity. In *IEEE Transactions on Image Processing*.
- Wang, Y.; Gao, Z.; Long, M.; Wang, J.; and Yu, P. S. 2018. Pre-dnn++: Towards a resolution of the deep-in-time dilemma in spatiotemporal predictive learning. In *International Conference on Machine Learning*.
- White, C.; Ushizima, D.; and Farhat, C. 2019. Neural networks predict fluid dynamics solutions from tiny datasets. *arXiv preprint arXiv:1902.00091*.
- White, P. 2006. Review of methods and approaches for the structural risk assessment of aircraft. Technical report, Australian Government Department of Defence, Defence Science and Technology Organisation, DSTO-TR-1916.
- Wiewel, S.; Becher, M.; and Thuerey, N. 2019. Latent space physics: Towards learning the temporal evolution of fluid flow. In *Computer Graphics Forum*.
- Williams, R. J., and Zipser, D. 1995. Gradient-based learning algorithms for recurrent networks and their computational complexity.