Macromolecules

pubs.acs.org/Macromolecules Article

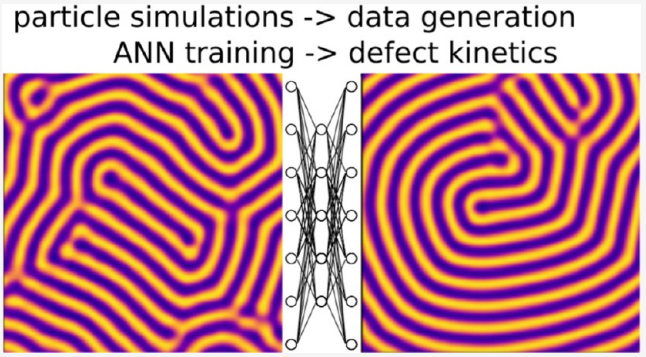
Combining Particle-Based Simulations and Machine Learning to Understand Defect Kinetics in Thin Films of Symmetric Diblock Copolymers

Ludwig Schneider* and Juan J. de Pablo*

Cite This: *Macromolecules* 2021, 54, 10074−10085 Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

ABSTRACT: The self-assembly of soft matter provides a practical and scalable route toward the production of nanostructured materials, with minimal need for direct intervention at nanoscopic length scales. Symmetric diblock copolymers, which can self-assemble into a lamellar phase, are a prototype for this class of materials. In this work, we introduce a machine learning model that is trained by intermediate time-scale simulations of a soft, coarse-grained model. The aim of the model is to simulate defect kinetics in the lamellar morphology, as the material relaxes toward equilibrium. To do so, we exploit the physical characteristics of overdamped dynamics and formulate the problem of time evolution as a Markov chain. The trained artificial neural network (ANN) predicts a time-independent transition probability from one time step to the next. As a result, we arrive at a method that can be repeatedly applied to generate long-time trajectories. Predicting defect kinetics in this manner provides hitherto unavailable insights into the late-time dynamics of block copolymer relaxation. The neural network is purposely designed to be independent of input size, which enables training on small systems, and enabling predictions over large scales. As a demonstration of these capabilities, in this work, we leverage the ANN to obtain information about the statistics of defect motion and lifetimes over a long-range ordering process.



INTRODUCTION

The self-assembly features of diblock copolymer materials provide access to nanometer structure manufacturing without a need for direct process control on such length scales. These materials and their blends with homopolymers have in fact been shown to meet the strict demands of the semiconductor industry for the fabrication of electronic devices at nanometer length scales.[1−10] A key challenge toward the implementation of any self-assembly strategy is understanding the underlying kinetics, including predicting the final morphology of a materials system from a given initial condition.

Over the last decades, a variety of methods for the simulation of diblock copolymers have been proposed. This includes theoretical descriptions,[11] all-atom and coarse-grained simulations,[12] soft coarse-grained models,[13−15] continuum models,[16−19] and field-theoretic models.[20−23] Each of these methods serves an important role in understanding block polymers and their self-assembly. Recent publications summarize and address these efforts.[24−26] However, even with the use of modern supercomputers, they all struggle to predict the full kinetics of the self-assembly on laboratory time scales of hours or days. For diblock copolymers, the equilibrium phase diagram has been established with regular morphologies as a function of the block repulsion χN and volume fraction f.[20] For symmetric diblock copolymers, the equilibrium morphology is lamellar. The challenge is that after spinodal decomposition, the system often occupies a local minimum in the underlying free-energy landscape, which consists of basins separated by high energy barriers. Thermally activated annealing is therefore protracted. While the equilibrium crystal structures are similar to those of atomistic hard condensed matter systems, the kinetics of ordering or self-assembly is governed by a fundamentally different mechanism in polymeric systems.

Advanced sampling methods, including the string algorithm,[27,28] have been applied successfully to study the kinetics of prototypical defects.[2,4,29−31] However, the computational effort associated with these methods is considerable, and the corresponding calculations often require knowledge of both the initial and final configurations of the system. Such knowledge is available for specific prototypical situations, but, more generally, it is lacking if the goal is to predict the long-time defect kinetics

Received: July 27, 2021
Revised: September 24, 2021
Published: October 28, 2021

https://doi.org/10.1021/acs.macromol.1c01583
Macromolecules 2021, 54, 10074−10085

for a given initial condition that exhibits many different defects. The interplay between these defects and their annihilation is what ultimately determines the final configuration.

An alternative approach toward the simulation of a material over long time scales is to combine traditional physics-based approaches with machine learning (ML) techniques to speed up long-time predictions. As ML techniques become more accessible, their application becomes more accessible.[32,33] This combination is suitable under the assumption that most of the molecular physics is important during the early stages of morphology formation[17,34] and that the long-time dynamics need not rely on the full molecular details to predict defect kinetics.[29] More specifically, the early stages are simulated with a full particle-based physics model until an initial configuration is attained throughout several relaxation times of the polymer T_R. It is possible to use the same model to directly simulate the first steps of defect kinetics, but this incurs insignificant computational costs. With this true kinetics, we train an ML model to learn the inherent dynamics to transform one configuration over the time interval $t \rightarrow t + \Delta t$, where Δt spans several molecular relaxation times T_R and covers the defect kinetics time scale.

The generation of multiple replicas of the early time kinetics is inherently parallel—optimally suited of large supercomputers, while the ML model coarse grains all molecular details out and can be "integrated" with large time steps Δt. This latter feature makes long time-scale accessible.

■ ASSEMBLING AN ML MODEL FOR THE PREDICTION OF DEFECT KINETICS

Three aspects must be considered in the construction of the ML model: data generation, the architecture of the ML model, and the training procedure that combines the architecture with the model.

Data Generation with a Particle-Based Model. We use a soft, coarse-grained model to simulate the self-assembly and defect kinetics. Polymers are represented as Gaussian chains, with interaction centers connected via harmonic springs. The diblock copolymer molecules are modeled with a discretization of $N = 32$. We choose symmetric lamellae-forming diblocks with $N_A = N_B = 16$ beads per block. The spring constant connecting the beads is selected such that the end-to-end distance is $R_{e,0} = b_0\sqrt{N-1} = 1$ the natural unit of length, where the average bond length is denoted by b_0. The coarse graining leads to soft nonbonded interactions[13] that are, in this case, evaluated on a grid as a function of the local densities $\rho_\alpha(r)$, with $\alpha \in \{A, B\}$ indicating the particle type. The resulting nonbonded interactions can be described via an excess free-energy functional

$$\frac{\mathcal{H}_{nb}[\{\hat{\phi}_\alpha\}]}{k_B T} = \frac{\rho_0 \Delta L^3}{N} \sum_{c\in\{cells\}} (\mathcal{K}_{fluc.}[\{\hat{\phi}_\alpha(c)\}] + \mathcal{K}^{\alpha}_{inter.}[\{\hat{\phi}_\alpha(c)\}]) \quad (1)$$

where densities are allocated on a discrete cubic grid with a grid size $\Delta L = R_{e0}/6$ and ρ_0 describes the particle density of a homogeneous system. Here, $\hat{\phi}_\alpha$ describes normalized particle densities given by a particle configuration $\{r_i\}$.

The first part of the functional controls density fluctuations via

$$\mathcal{K}_{fluc.}[\{\hat{\phi}_\alpha\}] = \frac{\kappa_0 N}{2}(\hat{\phi}_A(c) + \hat{\phi}_B(c) - 1)^2 \quad (2)$$

and the model parameter, $\kappa_0 N$, enforces near incompressibilty.[14] For this work, we choose an inverse compressibility modulus of $\kappa_0 N = 42$, which is high enough to constrain density fluctuations but low enough to avoid an artificial slow down of the simulations.

The second term of the excess free-energy functional controls the incompatibility of the two polymer blocks A and B

$$\mathcal{K}^0_{inter.}[\{\hat{\phi}_i\}] = -\frac{\chi_0 N}{4}(\hat{\phi}_A(c) - \hat{\phi}_B(c))^2 \quad (3)$$

The Flory–Huggins model parameter χN_0 controls this repulsion; for this study, we select $\chi N_0 = 17$ for all systems. Note that $\chi N_0 = 17$ reliably results in weak segregation and lamellar microstructures.

As a demonstration study, we choose to perform predictions of defect kinetics in a two-dimensional system. Hence, we choose a box geometry with a pseudo-thin film with periodic boundary conditions in all directions. The box dimensions are $L_x \times L_y \times L_z = 21.33R_{e,0} \times 21.33R_{e,0} \times 1R_{e,0}$. The z-dimension is smaller than the natural lamellar spacing $L_z < L_0 \approx 1.53\ R_{e,0}$, which forces lamellae to stand up. As a result, we can average out the z-direction and obtain densities on a collocation grid with dimensions $N_x \times N_y = 128 \times 128$. The densities are normalized by ρ_0 and can thus be similarly interpreted as one-channel images: optimal for ML algorithms.

We integrate the simulations in time via the single-chain-in-mean-field (SCMF) algorithm, which approximates the slow moving and soft, nonbonded interactions in an instantaneous field but describes strong bonded interactions explicitly for smart Monte-Carlo (SMC) moves.[14,35] This ensures correct Rouse-like polymer dynamics, while the SOft coarse-grained MC Acceleration (SOMA) GPU implementation is sufficiently efficient to facilitate the generation of data.[36]

For each sample, we start with a random homogeneous state and quench the block repulsion instantaneously to $\chi N_0 = 17$. What follows is a spinodal decomposition into a lamellar fingerprint pattern with many defects. After a period of $\Delta t = 5\ T_R$, where $T_R = R^2_{e,0}/D$ is the molecule relaxation time,[a] spinodal decomposition transitions into kinetics that is dominated by defect motion. We record nine further density snapshots with the same time interval $\Delta t = 5T_R$ in between. Hence, we obtain nine pairs of densities $\{\rho(t, r), \rho(t + \Delta t, r)\}$ that can be used for supervised learning techniques. The idea is to learn the characteristic defect annihilation kinetics for this fixed period. Although we successfully trained models for all of these pairs, the majority of our presentation here focuses on the last pair, for the time step between $t = 45T_R$ (input) and $t = 50T_R$ (output) after the quench.

We prepared 2096 independent samples in this manner. The initial particle positions were randomly created with different seeds. Each simulation uses a different initial seed for the Monte-Carlo (MC) algorithm, ensuring independent time evolution. From this pool, 48 samples were randomly selected for the test set. The remaining 2048 samples constitute the training set for the supervised ML techniques. Each sample could be simulated within about one and a half hours on a single Nvidia A100 GPU, with a total of 3 144 GPU hours invested into the sample preparation. Each of the samples is an independent simulation, and hence parallel preparation with up to 2096 GPUs is possible.[b]

In addition, we prepared 84 larger samples with spatial dimensions of $L_x \times L_y \times L_z = 170.664R_{e,0} \times 170.664R_{e,0} \times 1R_{e,0}$, which results in a grid resolution of 1024 × 1024 after z-

https://doi.org/10.1021/acs.macromol.1c01583
Macromolecules 2021, 54, 10074−10085

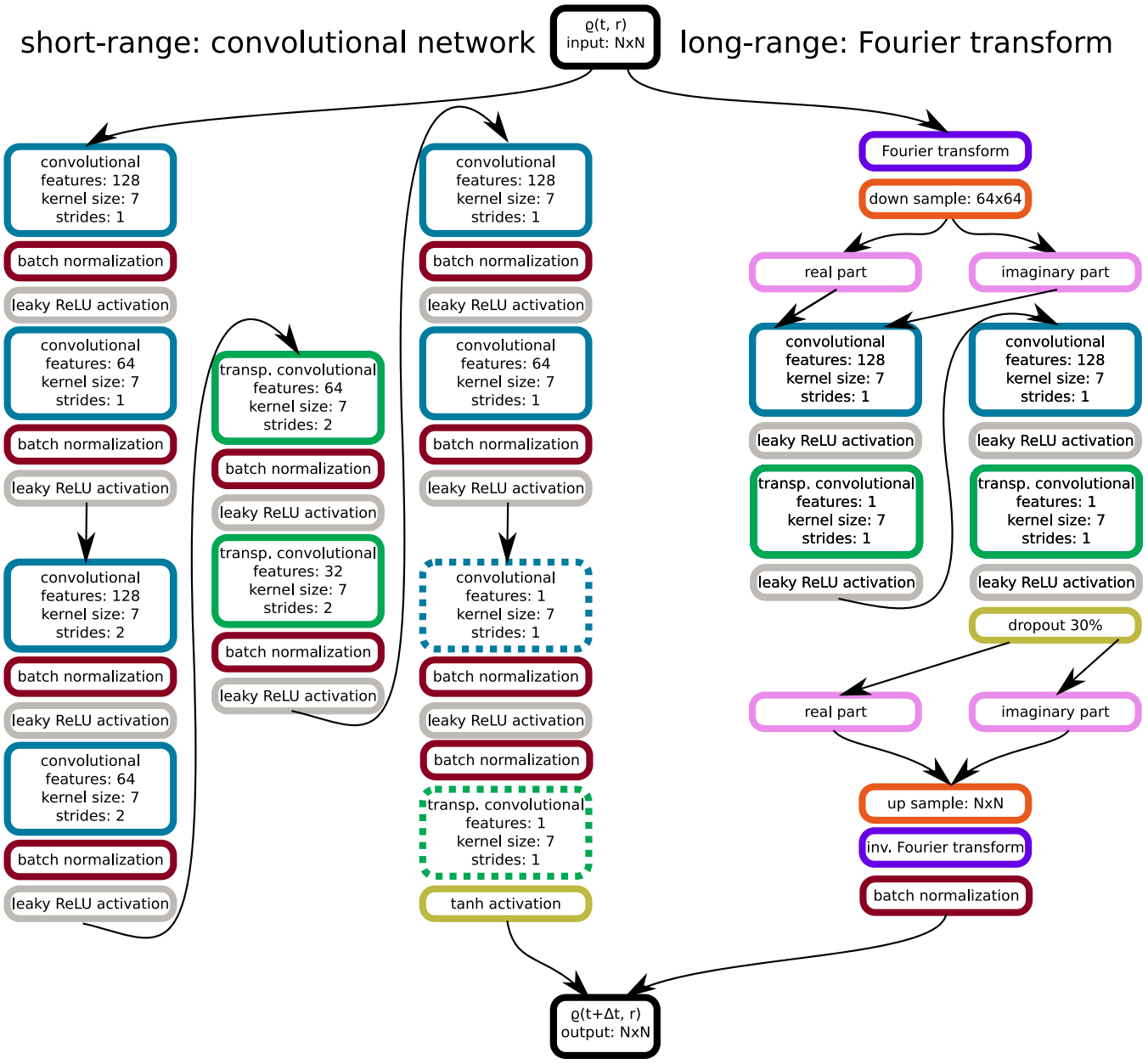


Figure 1. Overview of the ANN as implemented and trained with the TensorFlow2/Keras framework.[40] The left side depicts the deep-convolutional network with several layers and varying features and strides. The transposed convolutional layers, in combination with a final layer with feature size 1, result in a fully convolutional network that operates independently of the spatial input size of ρ. The right side shows the long-range part of the ANN: a Fourier transform is downsampled to a fixed resolution and presented to a trainable convolutional network. After upsampling and an inverse Fourier transform, the manipulated densities ρ on both sides of the network are summed with trainable weights to obtain the final prediction.

direction marginalization. These samples are not used during training, but we use them to verify (a) that our predictions are physically reasonable and (b) that our prediction network is independent of input and output size.

Architecture of the ML Model. We use an artificial neural network (ANN) to predict how a given conformation of lamellar defects $\rho(t)$ evolves during the $\Delta t = 5T_R$ time period $\tilde{\rho}(t + \Delta t)$.[37] We assume that this transition can be expressed as $\tilde{\rho}(t + \Delta t) = f[\rho(t)]$. The ANN introduced here now fulfills the function of a map f: $N \times N \rightarrow N \times N$, where each pixel is an element of $\mathbb{R}_{\geq 0}$ and the domain is $[0, \approx 1]$ for the normalized densities.[c] For a given initial condition, repeated application of this map generates a trajectory for the defect kinetics. This simple approach of using only $\rho(t)$ as input and neglecting the further history of the trajectory is possible because the motion is expected to be overdamped. Even the simulation of the underlying particle model simulates the motion as overdamped. Hence, this Markov-chain assumption is justified and obviates the need for recurrent neural network architectures.[38] Without the Markovian assumption, similar approaches have succeeded in the field of active-nematic hydrodynamics,[39] but the overdamped kinetics of diblock copolymers allows for a simpler approach to predict trajectories.

In contrast to the true trajectory, which is driven by thermally activated defect annihilation and includes an inherent random component, the map is entirely deterministic. The goal is for the ANN to predict the most likely trajectory of a real system.

https://doi.org/10.1021/acs.macromol.1c01583
Macromolecules 2021, 54, 10074−10085

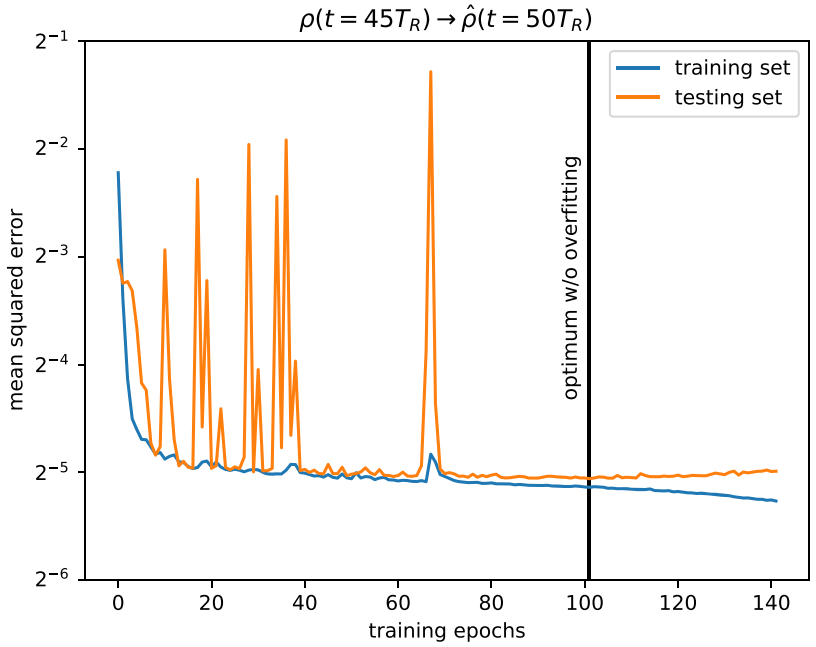


Figure 2. Evolution of the loss of the training and test set with presenting all samples of the training set to the ANN for the prediction of $\rho(t = 45T_R) \rightarrow \hat{\rho}(t = 50T_R)$. The mean squared error reduces dramatically at first and settles to a minimum plateau after a few local minima are overcome. After about 100 epochs, the loss of the training set decreases without a similar evolution in the test set loss, indicating overfitting. Hence, we rewind to the lowest test set loss for the final model. The final mean squared error is on the order of 2^{-5} with a data range of the density of $[0, \approx 1]$.

We have two constraints for the construction of the ANN: (i) it should incorporate long and short length-scale effects and (ii) the ANN should work independently of the input and output spatial dimensions, $N \times N$, of the density ρ. The first consideration is met by combining a deep-convolutional network for short length scales[37] and a new approach using a Fourier transform of the density to capture effects on long length scales. Figure 1 depicts all details of the ANN architecture for this study.

Short-range interactions are implemented via a deep-convolutional network. We use a kernel size of 7 for convolutional kernels, which represents a length scale of more than one $R_{e,0}$ with a grid spacing of $\Delta L/6$. With a natural lamellar spacing of about $L_0 \approx 1.53R_{e,0}$, the kernel always contains at least one A/B interface. This acts as a short-range recognition in the system and has similarities to continuum model formulations such as that of Swift and Hohenberg.[18] Implementing the model with discrete Laplace operators is essentially identical to a convolutional layer with fixed weights. Using convolutional layers with trainable weights and multiple features (teal in Figure 1) greatly expands the capabilities of the network compared to a simple Laplace operator. Additionally, we are using a convolutional layer with strides 2 > 1 to reduce in succession to compact the information. Expanding these layers via transposed convolutional layers (green in Figure 1)[41] essentially implements the popular encoder/decoder principle for fully convolutional neural networks.[42] All but the final layers are batch normalized (maroon in Figure 1) and activated with the leaky ReLU nonlinearity (gray in Figure 1).[43]

Additionally, we are using a convolutional and a transposed convolutional layer with a feature size of unity at the end, which transforms the output size into the original input size (dashed borders in Figure 1). Using this technique, instead of customary dense layers, yields a fully convolutional network[44] and has the advantage of resulting in a network that is independent of the input and output size. Any input size, with minor restrictions discussed later, is transformed into an output of the same size. As a result, it is possible to train the network with densities over a domain of 128×128 and infer results for densities much larger than this. Physically, this assumes that all relevant physics can be observed in the smaller system.

A convolutional network has a finite length scale—any given point can only obtain information about its apparent neighborhood. The interaction between defects is believed to be influenced over long distances. To enable such interactions, we also integrate a Fourier branch into the network (right side of Figure 1). For this branch, the density is Fourier transformed first $\rho(r) \rightarrow \rho^*(q)$ (purple in Figure 1). Representing the density in Fourier space for the physical description has an analogue in the Ohta−Kawasaki model[16] for continuum descriptions as well. In a second step, we downsample $\rho^*(q)$ to a fixed matrix size of 64×64. The downsampling conserves the q-vectors present in the image independent of input size (orange in Figure 1). The real and imaginary parts of the Fourier transform are separated (pink in Figure 1) and fed through the same shallow stack of convolutional and transposed convolutional networks and share the same weights. We finally upsample the processed Fourier transform to its original $N \times N$ size before applying an inverse Fourier transform, thereby obtaining the long-range prediction to the predicted density $\hat{\rho}(t + \Delta t)$. Combining the result with the short-range result from the deep-convolutional side yields the predicted density $\hat{\rho}(t + \Delta t)$.

Restrictions. The network architecture, as introduced above, has limitations. For technical reasons, the boundary condition of the convolutional layer is of the mirror type. This is in contrast to the periodic boundary condition of the simulations. Hence, it is expected that predictions close to the boundaries exhibit differences from simulations. This limitation can be circumvented by tiling the input into a larger image before it is fed into the network and cutting into the area of interest afterward. Since the inference of predictions is computationally inexpensive this is an acceptable solution.

https://doi.org/10.1021/acs.macromol.1c01583
Macromolecules 2021, 54, 10074−10085

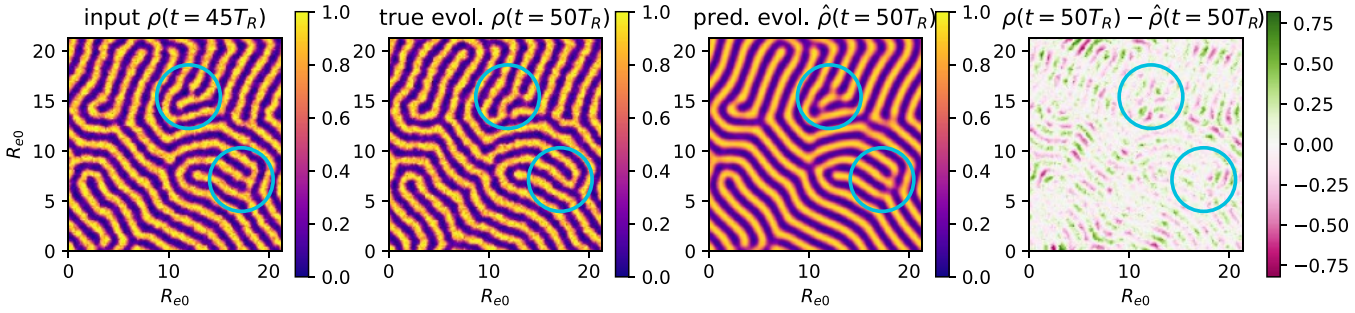


Figure 3. Comparison of the true, simulated evolution versus the predicted kinetics with the trained model. From left to right: input density $\rho(t = 45T_R)$, simulated evolution $\rho(t = 50T_R)$, via ANN predicted evolution $\hat{\rho}(t = 50T_R)$, and pixel difference between true evolution and prediction. Blue circles highlight regions where defect kinetics occurs.

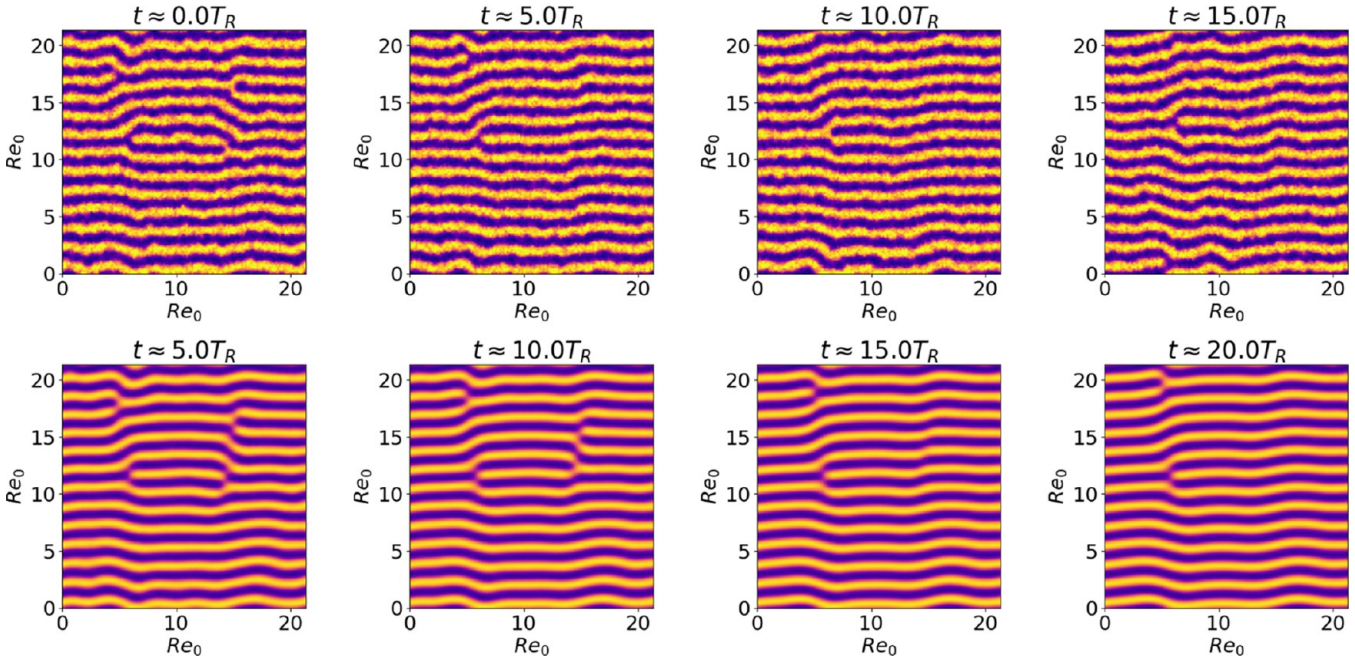


Figure 4. Top row: Simulated trajectory of a prepared defective configuration at low defect density. On the right, annihilation occurs first by joining the two opposing defects. In a second step, the defect on the left is repelled from the central, additional lamellae and moves up. Bottom row: Same evolution but predicted by the ANN instead. The characteristics of how and in which order the defects are annihilated are identical to the true results, but the timing is slightly different. The network evolution has been predicted with an image 4 times the actual presented size and is only cut for publication. This ensures that no boundary condition artifacts influence the results.

The second issue with the network regards its input size. While the overall size of the input and output is only limited by the available computational memory, the explicit size has to be a power of 2. The inversion of convolutional layers with stride 2 with transposed convolutional layers is only exact if the input size has a sufficient number of 2s in its prime decomposition. Additionally, the down- and upsampling procedures for the Fourier branch of the network are only implemented for networks that are powers of 2 larger than 64. This limitation can be circumvented in the same way as the first. Each image of interest can be embedded into a larger image with a power-two-sized dimension.

Training of the ML Model. The data set and ANN architecture are combined into a predictive model via supervised learning. The density $\rho(t)$ is considered the input for the ANN that produces a predicted output $\tilde{\rho}(t + \Delta t)$. This output can be compared with the true simulated output $\rho(t + \Delta t)$. We compute the pixel-wise mean squared error between $\tilde{\rho}$ and ρ as a loss function. We use an Adam optimizer[45] with an initial learning rate of 10^{-3} to vary the trainable weights of the ANN and minimize the loss. The complexity of the ANN makes it prone to overfitting, and we compensate by monitoring the loss of the test set during training. If the loss of this test set does not decrease over 40 epochs of training, we restore the optimal weights of the lowest test loss and stop the training early. For the different initial times $t/T_R \in \{5, 10, 15, 20, 25, 30, 35, 40, 45\}$, we find that the optimal number of epochs before overfitting sets is between 45 and 101 epochs of training. For the remainder of this work, we use the model trained with the initial time of $t = 45T_R$, which triggered the early stopping event after epoch 101. The evolution of the observed losses during training is visualized in Figure 2.

RESULTS AND DISCUSSION

The first step to evaluate the performance of the model we have trained is to examine its predictions when applied to a sample of the test set. Figure 3 visualizes the application of the trained ANN to an input of the test set and compares it to the true

https://doi.org/10.1021/acs.macromol.1c01583
Macromolecules 2021, 54, 10074−10085

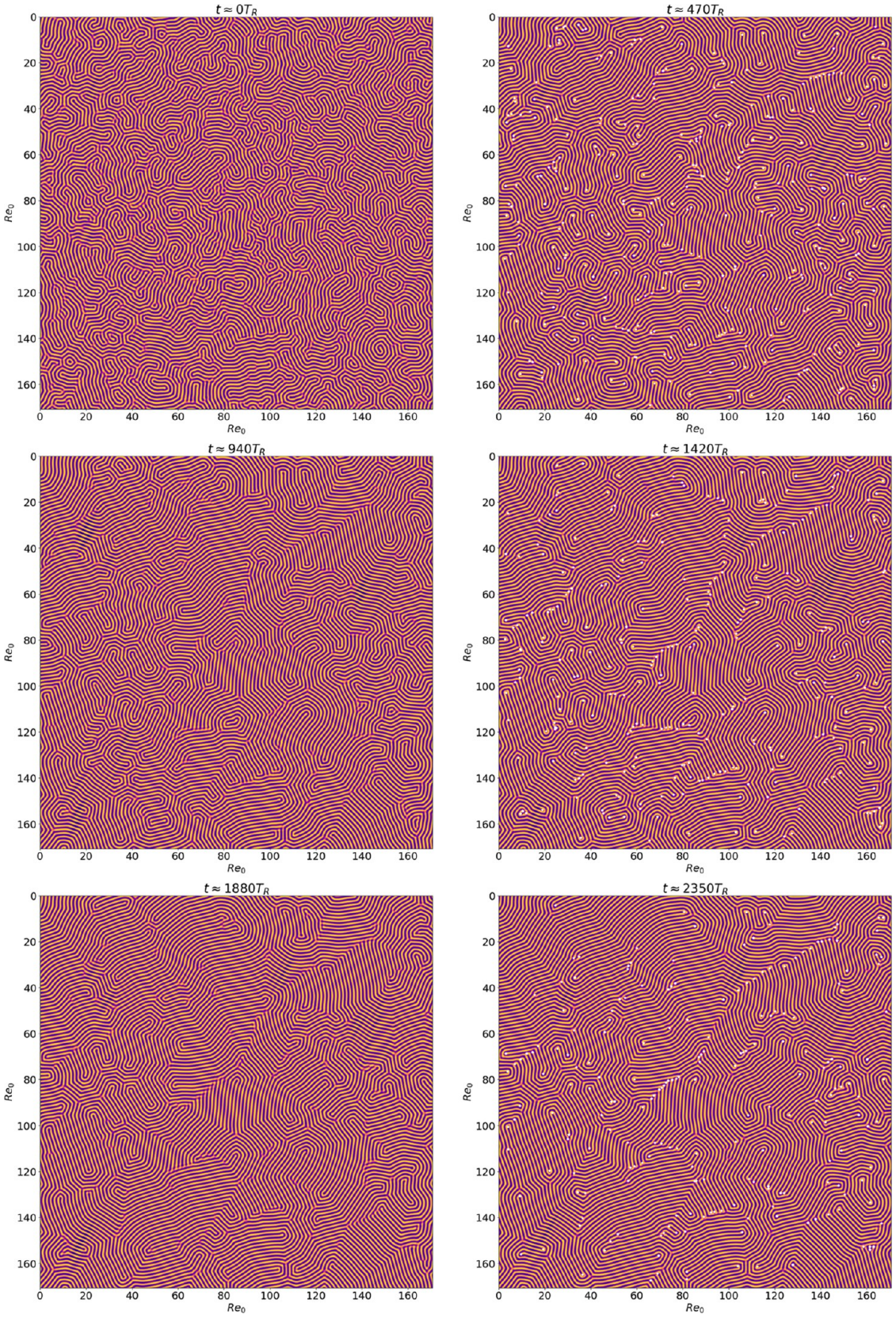


Figure 5. Long-time and large-scale trajectory as predicted with the ANN model. The transition from a fingerprint-like pattern toward a grain structure of lamellae with differing orientations is visible. The visualized trajectory is one of the 84 samples used to determine the correlation length of Figure 7. The frames on the right demonstrate defect detection with white stars. The Supporting Information contains animated versions of this figure.

simulated evolution. At first sight, it is apparent that the ANN prediction is significantly smoother than true simulation results, which include thermal noise in density fields. Further inspection of the pixel difference between the two reveals that the shape of

https://doi.org/10.1021/acs.macromol.1c01583
Macromolecules 2021, 54, 10074−10085

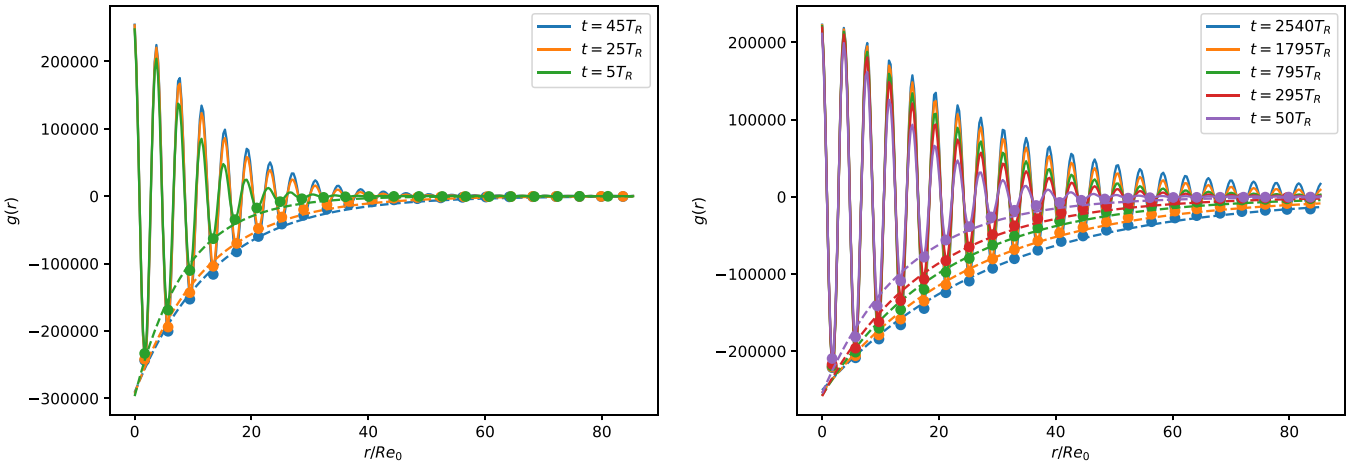


Figure 6. Radial autocorrelation $g(r)$ for short time scales and a simulated trajectory (left) and longer times via the predictions of the ANN model. The data from 84 independent samples have been averaged for each time step. The correlation length can be accessed by fitting an exponential decay as an envelope around the oscillating $g(r)$. Dashed lines demonstrate these fits for the minima.

the A/B interface of the lamellae is not faithfully reconstructed by the prediction. If we were interested in building a prediction that reconstructs the full physics, this would be disappointing. However, there is a multitude of other methods and models that have been specialized for this, *e.g.*, self-consistent field theory (SCFT).[20]

However, we are interested in the kinetics of defects. We highlight areas of defect motion with blue circles in Figure 3. A comparison between the true and the predicted evolution shows a slide in these defects that is similar for both. There are minor differences in how defects are transformed, but this is not unexpected since the simulation has a random, thermally activated component which the prediction is lacking. However, the fact that the prediction changes the same defects and does not alter the other defects is encouraging. We conclude that the network can identify and move defects for a configuration similar to that presented during the training set.

The next step is to determine if the ANN can be applied repeatedly to obtain a trajectory of defect motion. This has not been explicitly trained into the network, so we have to verify it separately. We choose an example system of the same size but with a specifically prepared morphology with a low defect density. Such a morphology is expected to appear much later in the time evolution. So if we are successful, we would demonstrate both that (a) a trajectory can be obtained via a repeated application and (b) the defect kinetics can be predicted at later times as well.

Figure 4 compares the trajectories. The top row visualizes the defect kinetics as obtained by simulations, and the bottom row can be contrasted against the simulation trajectory. Even in this situation, the ANN correctly identifies defects and changes them. The trajectory demonstrates that the ANN can annihilate defects in the same way as the simulation does. The prepared morphology contains more than a single defect and the ANN changes defects in the same temporal order as the simulation, even as the timing of the motion and annihilation are slightly different between simulation and prediction. So, overall, these results serve to confirm that it is possible to predict trajectories with the proposed method; the kinetics of individual defects are correctly ordered and annihilated.

We now turn our attention to long trajectories and verify that the observed differences in time do not propagate through the trajectory. With longer trajectories, we also require larger length scales, as the defect annihilation increases the correlation length, ξ. To avoid finite-size effects, the system size has to be larger than the correlation length at all times $L \gg \xi(t)$.

We choose a system 4 times larger in the linear dimension than the test and training sets. The first relaxation times T_R are simulated and, starting from $t = 45T_R$, we use the trained ANN to predict the kinetics. One example out of 84 total considered trajectories is visualized in Figure 5. We predict the evolution until a maximum time that corresponds to $t = 2540T_R$. The size of the system and the length of the trajectory are longer than we can comfortably simulate with the particle-based model. Additionally, the larger system allows us to demonstrate that our model is indeed size independent and can reliably predict defect kinetics for systems much larger than the samples of the training set.

Inspecting the trajectory demonstrates how the fingerprint pattern slowly evolves into a system that can be best described as grains of different lamellar orientations. Close inspection reveals how defects move through the system and eventually annihilate with opposing defects to yield more regular structures. The length of the trajectory even allows inspecting how the domain boundaries move and how entire grains get absorbed into larger ones. These dynamics are well known from experimental studies but have rarely been observed in simulations. Overall, the trajectory is consistent with our expectations.

A visual inspection, however, is insufficient to clarify whether the observed dynamics are correctly predicted with our trained ANN model or not. We therefore compare the evolution of the correlation length ξ as a function of time, which can be accessed from a spatial autocorrelation of the density $g(r) = (\rho^*\rho)(r)$. A lamellar morphology is expected to adopt the form $g(r) = \cos(2\pi r/L_0)\cdot e^{-r/\xi}$ and can be conveniently calculated from the collective structure factor $S(q)$.[29] Figure 6 visualizes the autocorrelation $g(r)$ for simulated and predicted trajectories. The expected form of the correlation is observed, and the correlation length can be obtained by fitting an enveloping exponential decay.

Various studies have investigated this correlation length as a function of time and found a power-law relation $\xi(t) \propto t^{\eta}$ with an exponent $\eta \in [0.066, 0.3]$.[29,46−50] For early times, power laws with steeper slopes have been observed, and a transition into the late-time dynamics has been observed. For this work, we focus on time scales beyond these initial times since they can be easily

https://doi.org/10.1021/acs.macromol.1c01583
Macromolecules 2021, 54, 10074−10085

accessed via conventional simulation techniques. By examining the correlation length, we can study the relation both during first time steps $t < 45T_R$ from the simulation trajectory as well as the correlation as predicted via the ANN model $t > 45T_R$.

Figure 7 presents the averaged correlation lengths for short simulated trajectories as well as extended trajectories predicted

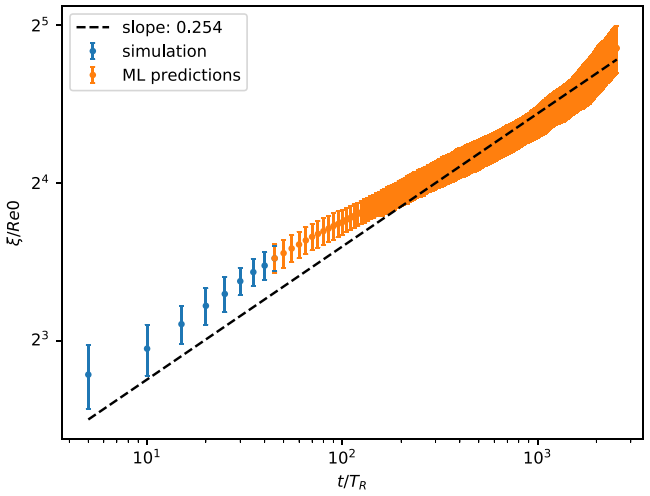


Figure 7. Correlation length ξ as a function of time. The expected power-law behavior is observed, and the prediction seamlessly extends the simulation trajectory far beyond the originally simulated spatial and temporal scales.

via ML. Both follow a power law with an exponent of $\eta \approx 0.254$, well within the literature values. The relatively high exponent is a consequence of the model parameters, which include a low $\sqrt{N}$ and low $\chi N = 17$. The predicted correlation lengths seamlessly extend the correlation lengths of explicitly simulated trajectories. These findings serve to confirm that our ANN model is capable of predicting the defect kinetics, including the correct time evolution. We emphasize that the model was not trained with data simulated beyond $t > 50T_R$, and yet its predictions appear to be correct after thousands of relaxation times. We conclude that basic relations that determine the defect kinetics can be learned from short time scales, as long as they are well beyond the spinodal decomposition regime.

Insights into Defect Dynamics. Beyond providing a useful machine learning tool, the proposed method also enables access to a large number of trajectories of defects. Such an amount of data has not been available from simulations or experiments. We now use this data set to gain insights into the statistics of defects. To automatically identify defects in the morphology, we skeletonize lamellae (A and B) domains and identify line ends via 9×9 masks on the density grid.[S1,S2] All line ends are defects, but this approach does not identify all defects. Notably, junctions are not detected. However, it captures dislocations and disclinations with high precision, see Figure 5 for identified defects and Figure 9 for the skeleton. This raw defect detection approach is sanitized by removing defects when they are closer than $1R_{e,0}$, thereby avoiding double detection. Using subsequent frames, we can connect defects to trajectories by associating defects that are closer than $1R_{e,0}$ from one frame to the next. These trajectories enable us to track the distance that a defect covers from the position it first appeared at to the point in space where it gets annihilated. This also permits the estimation of the lifetime of each defect.

The first observation we can extract from this analysis is the average density of defects. A geometric argument indicates that the defect density scales as $\rho_{defect} \propto \xi^{-2}$, a power law that has also been validated in experiments.[48] Figure 8 presents the results. At

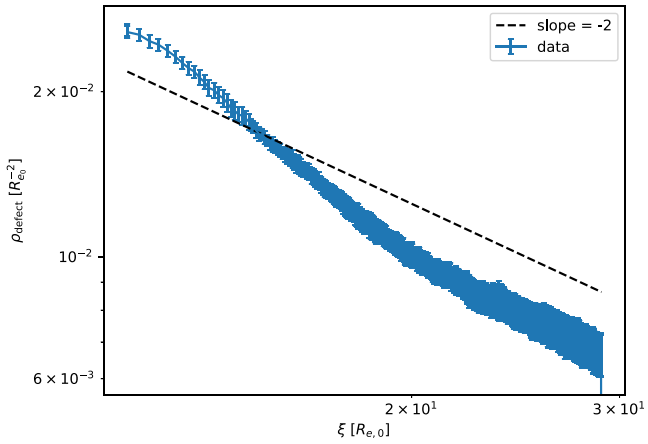


Figure 8. Defect density averaged over 15 morphology trajectories as a function of the corresponding correlation length ξ. At early times, we observe deviations from the predicted power law, but, at later times, the power law is accurately reproduced.

early times, we find deviations from the predicted power law. A possible explanation includes that the defects we are not detecting with our methodology are skewing the results. We also note that the cited experimental measurements do not feature sufficient data points to rule out similar deviations. For later times, however, we can confirm our prediction with the experiments.

Figure 9 shows the final configuration with skeletonized A lamellae. Additionally, we can identify the longer-lived trajectories of defects that are visible in the frame. It becomes

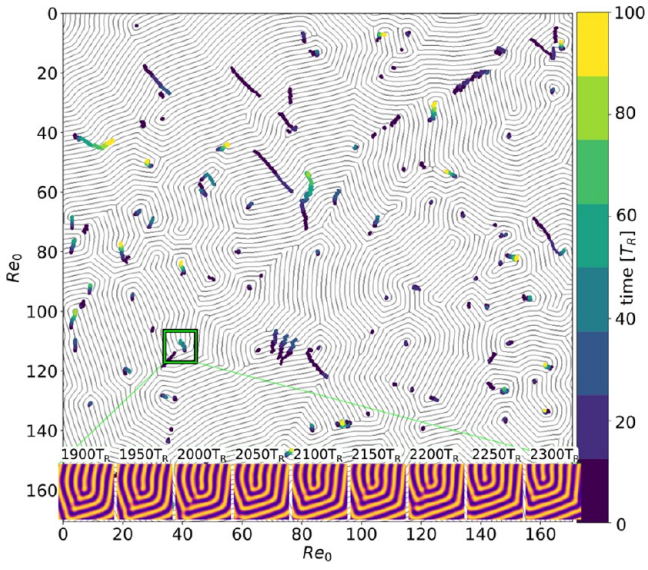


Figure 9. Skeletonized A lamellae of a final snapshot $t = 2540T_R$. Detected defects of this snapshot and their past trajectory are included. Visual inspection shows that most defects do not move far, but far-moving defects travel quickly, perpendicular to the lamellae (sliding). The inset shows one example of dislocation that does not undergo climb evaporation but rather only moves as dislocations slide by.

https://doi.org/10.1021/acs.macromol.1c01583
Macromolecules 2021, 54, 10074−10085

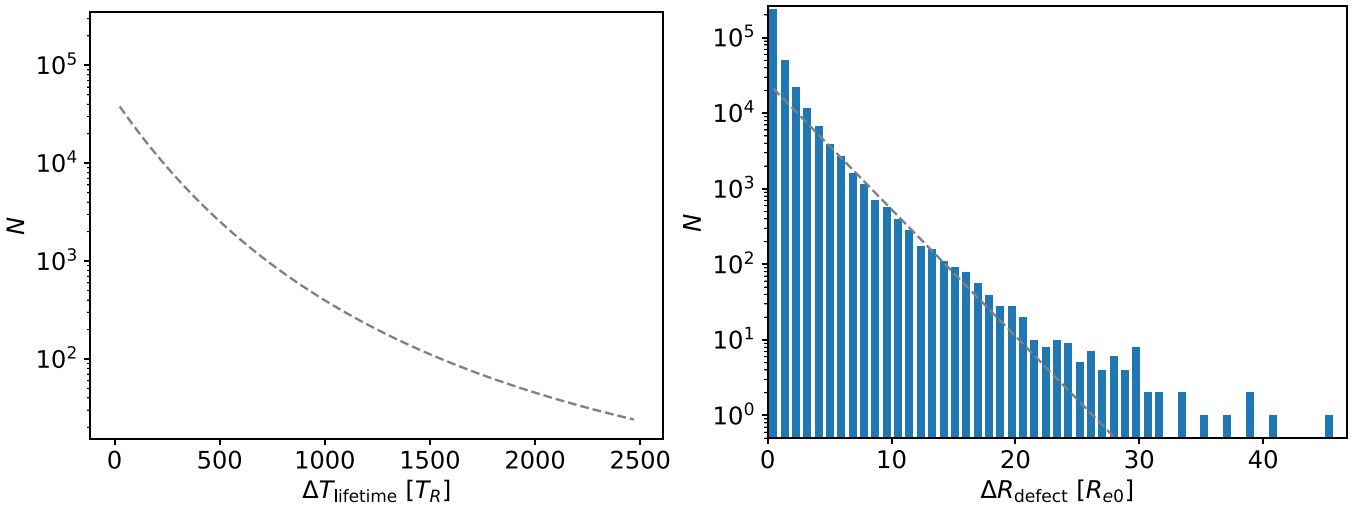


Figure 10. Histogram of 42 morphology evolution and the defect trajectories detected within. The left shows the lifetime of defects, the distance covered by defects. Since the maximum evolution covers $t = 2500T_R$, the data point at this time should be interpreted as lifetime longer than $2500T_R$. The dashed lines are to guide the eye: the lifetime with a nonexponential decay and the distance covered with an exponential decay.

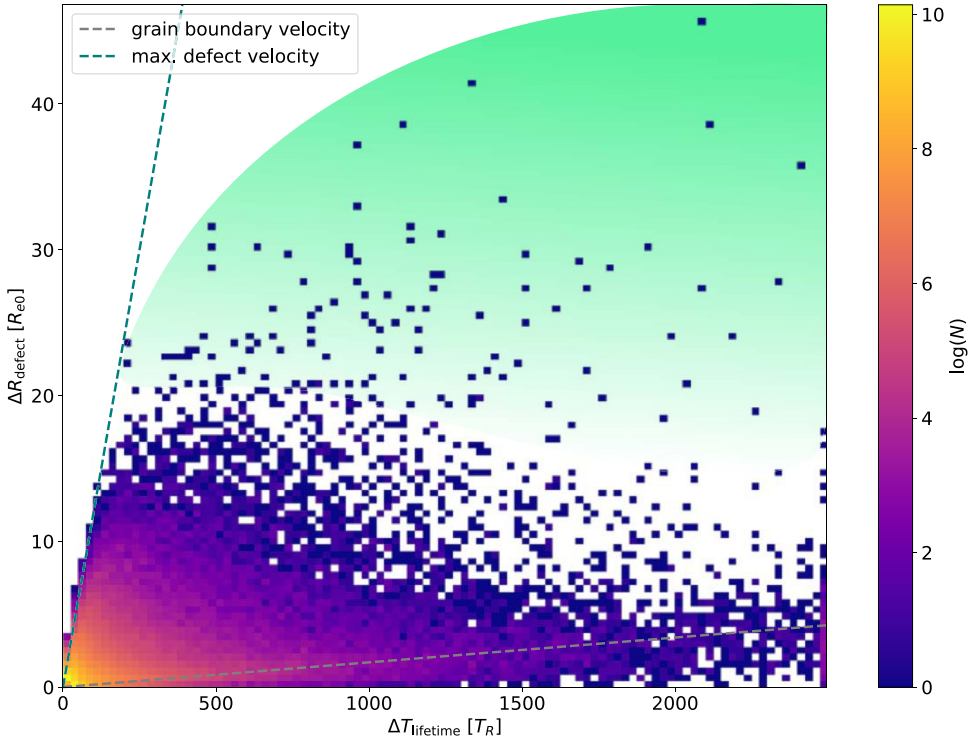


Figure 11. Correlation of the lifetime of defects with the distance they covered. Earlier observed especial defects are in the green region. Two velocities can be identified: one slow velocity covering the majority of the defects and the maximum possible velocity of defects bordering on the left.

clear that most defects are not mobile, especially as defects are localized in grain boundaries between two lamellar orientations. In contrast, there are a few defects with long trajectories that move quickly across multiple lamellae. We hypothesize that the majority of defects in any given configuration are located in the grain boundaries. However, there are a few, rare defects that travel in between grain boundaries and those cross lamellae perpendicularly. Such a defect motion is known as a slide. In the case of diblock copolymers, a slide motion requires only polymer chains adjacent to the defect to reorient. The polymers are already not in their preferred equilibrium configuration, and with the slide motion, the total energetic penalty is moderate in comparison with the alternative: an evaporation climb. The evaporation climb describes the motion of a defect that shrinks (or enlarges) the defective, ending lamella. Hence, the motion of the defect is parallel to the main orientation of the lamellae. The molecular mechanism for this kinetics requires polymer chains to switch from one lamella to the next. As a result, at least one of the blocks has to travel through a hostile region, which incurs a large energetic penalty. This effect has been experimentally studied before, and it has been shown that chain molecules can diffuse orders of magnitudes faster along the lamellae, compared to perpendicular lamellae, as required for a climbing mechanism.[48,53,54]

https://doi.org/10.1021/acs.macromol.1c01583
Macromolecules 2021, 54, 10074−10085

Consequently, evaporation climbs are rarely observed. The bottom inset of Figure 9 highlights a dislocation defect that does not show sliding. However, a potential climb motion of this defect is so protracted that it is barely visible. Instead, we observe the motion of that defect only as it merges with a neighboring defect, thereby changing the lamellae.

To quantify the difference between these boundary defects and free-moving defects, we analyze the lifetime of defects and the distance they cover in Figure 10. We can observe that the lifetime of defect trajectories decays rapidly. This is expected, as most defects annihilate quickly over the coarsening of the morphology. However, some defects have extraordinarily long lifetimes, well beyond the time evolution covered here.

The distance covered by defects is shown on the right-hand side of Figure 10. Again, most defects are relatively stationary, presumably inside a grain boundary. However, some defects do not follow the general trend of this histogram and indeed cover distances larger than $25R_{e,0}$. These defects are, however, rare, as they comprise less than 0.02% of the trajectories in the data set. We conclude that these are the defects that quickly move across lamellar interfaces, as observed earlier. Most of the research in the literature has focused on individual defects that can be associated with this type of defect, even though most defects follow the different, more complex dynamics of the grain boundary motion during the coarsening of the lamellar phase.

From these insights, we can further correlate the lifetime of defects with their total distance covered. Figure 11 depicts this correlation. The first observation is that not all defects that cover a large distance necessarily correlate with a shorter lifetime, which would identify their high velocity immediately. The explanation for this phenomenon is that defects may cover a section very quickly, but they can slow down after they hit a grain boundary and remain close to stationary afterward. An example trajectory of this kind can also be found in Figure 11. We can, however, identify a maximum velocity by assessing the left border on the correlation map. We assume that this velocity of $v_{max} \approx 0.12R_{e,0}/T_R$ approximates the speed of the sliding motion across an otherwise pristine region of lamellae. Additionally, we observe a concentration of points in another region of the correlation map. The relation of this correlation can also be associated with velocity. In this case, a much lower velocity of about $v_{av} \approx 1.7 \times 10^{-3}R_{e,0}/T_R$. We conclude that this is the velocity of longer-lived defects that reside primarily inside of grain boundaries and thus dominate the coarsening dynamics. Note that such a disparity in defect motion has been observed before.[50,53,54]

Computational Efficiency. The goal of this work is to develop a tool that is computationally more efficient than direct simulations, thereby providing access to longer time and length scales for the study of temporal evolution in ordered materials.

The upfront cost of the ML model is the combined time of the data generation: 3144 Nvidia A100 GPU h, plus about 6 h of training for the ML model on an Nvidia V100 GPU.

For the larger sample, with a linear dimension of about $L \approx 170R_{e0}$, 10 $\Delta t = 5T_R$ steps can be simulated in about 6.5 h with four Nvidia A100 GPUs. With the ANN model, the same time interval requires about 6 s on a desktop CPU. Hence, if we infer $(3144 + 6\ \text{h})/(6.5\ \text{h}/10) \approx 4846$ frames with the ANN model, we break even with the training cost. This neglects GPU resources usage during training while using four A100 GPUs for a single explicit simulation. Just the data presented in Figure 5 used 500 frames predicted with the ANN model—about one-tenth of the cost to break even; the data presented in Figure 7 more than breaks even with training cost.

CONCLUSIONS AND OUTLOOK

A method that combines traditional simulation techniques for particle-based models with modern ML techniques has been presented for the study of ordering dynamics in nanostructured materials. Particle-based simulations were used to generate the training data for an ANN model. The proposed approach was applied to investigate the defect kinetics of quasi 2D thin films of symmetric diblock copolymers. The computational efficiency of the ML model allows a break-even with the upfront cost of training data generation. As a proof of concept, we have restricted ourselves to two dimensions and only a single, easily accessible, set of parameters. However, expansion to full three-dimensional systems is straightforward, provided that the GPU hardware can accommodate the high-memory requirements of deep-convolutional networks. The proposed model is fully deterministic, which is at odds with the random nature of thermally activated processes. Moving forward, the expansion of the conventional ANN into a probabilistic model based on Bayesian theory could be beneficial to capture these processes more accurately.[55] It is also possible to explore more parameters and even more complex molecular architectures if the following basic requirements can be met: (a) the training set must contain a sufficient number of defect transitions, (b) the Markovian assumption must allow a trajectory prediction, and (c) all transitions must be well beyond the molecular relaxation time T_R. Note that the model presented here is not the only or best ANN architecture for the given task; the aim of this work has been to demonstrate that it is possible to construct such models.

Our results indicate that the combination of well-established simulation techniques with ML methods can provide insights into conditions that would otherwise be inaccessible with currently available high-performance computing (HPC) hardware. The machine learning data set generated here was used to gain insights into the statistics of defect kinetics. We found that the majority of defects are located inside the grain boundaries and move comparatively slowly. Climb motions are rare. A minority of defects, however, move quickly through otherwise pristine lamellar regions, primarily by the slide mechanism because of the lower energy penalty associated with such motion.

ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.macromol.1c01583.

Video animation of Figure 5 (AVI)
Video animation of Figure 5 (AVI)

AUTHOR INFORMATION

Corresponding Authors

Ludwig Schneider − *Pritzker School of Molecular Engineering, University of Chicago, 60637 Chicago, Illinois, United States;* orcid.org/0000-0002-3910-8217; Email: ludwigschneider@uchicago.edu.

Juan J. de Pablo − *Pritzker School of Molecular Engineering, University of Chicago, 60637 Chicago, Illinois, United States;* orcid.org/0000-0002-3526-516X; Email: depablo@uchicago.edu.

Complete contact information is available at:

https://doi.org/10.1021/acs.macromol.1c01583
Macromolecules 2021, 54, 10074−10085

https://pubs.acs.org/10.1021/acs.macromol.1c01583

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

This work is supported by NIST, through the Center for Hierarchical Materials Design (CHiMaD). The authors gratefully acknowledge the Gauss Centre for Supercomputing e.V. (www.gauss-centre.eu) for providing computing time through the John von Neumann Institute for Computing (NIC) on the GCS Supercomputer JUWELS at Jülich Supercomputing Centre (JSC). Additionally, the authors thank the JSC, Nvidia, Partec, and Atos for computing resources made available through the early access program of the JUWELS Booster system. The authors acknowledge the University of Chicago Research Computing Center for support of this work. Some of the simulations reported here were carried out on the GPU cluster supported by the NSF through grant DMR1828629.

■ ADDITIONAL NOTES

[a]The polymer relaxation time is determined from long times through the center-of-mass diffusion D

[b]For the preparation, we used a maximum of 400 GPUs in parallel.

[c]Thermal fluctuations and imperfectly enforced incompressibility can yield local fluctuations that exceed unity.

■ REFERENCES

(1) Nagpal, U.; Muller, M.; Nealey, P. F.; de Pablo, J. J. Free energy of defects in ordered assemblies of block copolymer domains. *ACS Macro Lett.* **2012**, *1*, 418−422.

(2) Hur, S.-M.; Khaira, G.; Ramírez-Hernández, A.; Müller, M.; Nealey, P.; de Pablo, J. J. Coarse-Grained Simulations of Defect Annihilation in Block Copolymer Thin Film via Solvent Annealing. *ACS Macro Lett.* **2015**, *4*, 11.

(3) Kambe, Y.; Arges, C. G.; Czaplewski, D. A.; Dolejsi, M.; Krishnan, S.; Stoykovich, M. P.; de Pablo, J. J.; Nealey, P. F. Role of Defects in Ion Transport in Block Copolymer Electrolytes. *Nano Lett.* **2019**, *19*, 4684−4691.

(4) Hur, S.-M.; Thapar, V.; Ramírez-Hernández, A.; Nealey, P. F.; de Pablo, J. J. Defect annihilation pathways in directed assembly of lamellar block copolymer thin films. *ACS Nano* **2018**, *12*, 9974−9981.

(5) Ruiz, R.; Kang, H.; Detcheverry, F. A.; Dobisz, E.; Kercher, D. S.; Albrecht, T. R.; de Pablo, J. J.; Nealey, P. F. Density multiplication and improved lithography by directed block copolymer assembly. *Science* **2008**, *321*, 936−939.

(6) Kim, S. O.; Kim, B. H.; Kim, K.; Koo, C. M.; Stoykovich, M. P.; Nealey, P. F.; Solak, H. H. Defect structure in thin films of a lamellar block copolymer self-assembled on neutral homogeneous and chemically nanopatterned surfaces. *Macromolecules* **2006**, *39*, 5466−5470.

(7) Edwards, E. W.; Müller, M.; Stoykovich, M. P.; Solak, H. H.; de Pablo, J. J.; Nealey, P. F. Dimensions and shapes of block copolymer domains assembled on lithographically defined chemically patterned substrates. *Macromolecules* **2007**, *40*, 90−96.

(8) Kim, S. O.; Solak, H. H.; Stoykovich, M. P.; Ferrier, N. J.; de Pablo, J. J.; Nealey, P. F. Epitaxial self-assembly of block copolymers on lithographically defined nanopatterned substrates. *Nature* **2003**, *424*, 411−414.

(9) Stoykovich, M. P.; Müller, M.; Kim, S. O.; Solak, H. H.; Edwards, E. W.; de Pablo, J. J.; Nealey, P. F. Directed assembly of block copolymer blends into nonregular device-oriented structures. *Science* **2005**, *308*, 1442−1446.

(10) Li, W. H.; Nealey, P. F.; de Pablo, J. J.; Müller, M. Defect removal in the course of directed self-assembly is facilitated in the vicinity of the order-disorder transition. *Phys. Rev. Lett.* **2014**, *113*, No. 168301.

(11) Leibler, L. Theory of Microphase Separation in Block Copolymers. *Macromolecules* **1980**, *13*, 1602−1617.

(12) Kremer, K.; Grest, G. S. Dynamics of entangled linear polymer melts: A molecular-dynamics simulation. *J. Chem. Phys.* **1990**, *92*, 5057−5086.

(13) Müller, M. Studying Amphiphilic Self-assembly with Soft Coarse-Grained Models. *J. Stat. Phys.* **2011**, *145*, 967−1016.

(14) Daoulas, K. C.; Müller, M. Single chain in mean field simulations: Quasi-instantaneous field approximation and quantitative comparison with Monte Carlo simulations. *J. Chem. Phys.* **2006**, *125*, No. 184904.

(15) Detcheverry, F. A.; Pike, D. Q.; Nealey, P. F.; Müller, M.; de Pablo, J. J. Monte Carlo simulation of coarse grain polymeric systems. *Phys. Rev. Lett.* **2009**, *102*, No. 197801.

(16) Ohta, T.; Kawasaki, K. Equilibrium morphology of block copolymer melts. *Macromolecules* **1986**, *19*, 2621−2632.

(17) Rottler, J.; Müller, M. Kinetic Pathways of Block Copolymer Directed Self-Assembly: Insights from Efficient Continuum Modeling. *ACS Nano* **2020**, *14*, 13986−13994.

(18) Swift, J.; Hohenberg, P. C. Hydrodynamic fluctuations at the convective instability. *Phys. Rev. A* **1977**, *15*, 319.

(19) Müller, M.; Rey, J. C. O. Continuum models for directed self-assembly. *Mol. Syst. Des. Eng.* **2018**, *3*, 295−313.

(20) Matsen, M. W. The standard Gaussian model for block copolymer melts. *J. Phys.: Condens. Matter* **2001**, *14*, R21.

(21) Fredrickson, G. H. *The Equilibrium Theory of Inhomogeneous Polymers*; Oxford University Press, 2006; Vol. *134*.

(22) Ganesan, V.; Fredrickson, G. Field-theoretic polymer simulations. *Europhys. Lett.* **2001**, *55*, 814.

(23) Bates, F. S.; Fredrickson, G. H. Block copolymer thermodynamics: theory and experiment. *Annu. Rev. Phys. Chem.* **1990**, *41*, 525−557.

(24) Huang, C.; Zhu, Y.; Man, X. Block copolymer thin films. *Phys. Rep.* **2021**, *932*, 1.

(25) Hu, T.; Ren, Y.; Zhang, L.; Li, W. Impact of Architecture of Symmetric Block Copolymers on the Stability of a Dislocation Defect. *Macromolecules* **2021**, *54*, 773−782.

(26) Hu, T.; Ren, Y.; Li, W. Impact of Molecular Asymmetry of Block Copolymers on the Stability of Defects in Aligned Lamellae. *Macromolecules* **2021**, *54*, 8024.

(27) Weinan, E.; Ren, W.; Vanden-Eijnden, E. Simplified and improved string method for computing the minimum energy paths in barrier-crossing events. *J. Chem. Phys.* **2007**, *126*, No. 164103.

(28) Sun, D.-W.; Müller, M. Numerical algorithms for solving self-consistent field theory reversely for block copolymer systems. *J. Chem. Phys.* **2018**, *149*, No. 214104.

(29) Ren, Y.; Müller, M. Kinetics of pattern formation in symmetric diblock copolymer melts. *J. Chem. Phys.* **2018**, *148*, No. 204908.

(30) Ren, Y.; Müller, M. Impact of Molecular Architecture on Defect Removal in Lamella-Forming Triblock Copolymers. *Macromolecules* **2020**, *53*, 5337−5349.

(31) Hur, S. M.; Thapar, V.; Ramírez-Hernández, A.; Khaira, G. S.; Segal-Perez, T.; Ricon-Delgalilio, P. A.; Li, W.; Müller, M.; Nealey, P. F.; de Pablo, J. J. Molecular Pathways for Defect Annihilation in Directed Self-Assembly. *Proc. Natl. Acad. Sci. USA* **2015**, *112*, 14144.

(32) Mehta, P.; Bukov, M.; Wang, C.-H.; Day, A. G.; Richardson, C.; Fisher, C. K.; Schwab, D. J. A high-bias, low-variance introduction to machine learning for physicists. *Phys. Rep.* **2019**, *810*, 1−124.

(33) Carleo, G.; Cirac, I.; Cranmer, K.; Daudet, L.; Schuld, M.; Tishby, N.; Vogt-Maranto, L.; Zdeborová, L. Machine learning and the physical sciences. *Rev. Mod. Phys.* **2019**, *91*, No. 045002.

(34) Wang, G.; Ren, Y.; Müller, M. Collective Short-Time Dynamics in Multicomponent Polymer Melts. *Macromolecules* **2019**, *52*, 7704−7720.

(35) Rossky, P. J.; Doll, J.; Friedman, H. Brownian dynamics as smart Monte Carlo simulation. *J. Chem. Phys.* **1978**, *69*, 4628−4633.

(36) Schneider, L.; Müller, M. Multi-Architecture Monte-Carlo (MC) Simulation of Soft Coarse-Grained Polymeric Materials: SOft coarse grained Monte-carlo Acceleration (SOMA). *Comput. Phys. Commun.* **2019**, *235*, 463−476.

(37) LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436−444.
(38) Pathak, J.; Hunt, B.; Girvan, M.; Lu, Z.; Ott, E. Model-free prediction of large spatiotemporally chaotic systems from data: A reservoir computing approach. *Phys. Rev. Lett.* **2018**, *120*, No. 024102.
(39) Colen, J.; Han, M.; Zhang, R.; Redford, S. A.; Lemma, L. M.; Morgan, L.; Ruijgrok, P. V.; Adkins, R.; Bryant, Z.; Dogic, Z.; Gardel, M. L.; de Pablo, J. J.; Vitelli, V. Machine learning active-nematic hydrodynamics. *Proc. Natl. Acad. Sci. USA* **2021**, *118*, No. e2016708118.
(40) Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; Kudlur, M.; Levenberg, J.; Monga, R.; Moore, S.; Murray, D. G.; Steiner, B.; Tucker, P.; Vasudevan, V.; Warden, P.; Wicke, M.; Yu, Y.; Zheng, X. TensorFlow: A System for Large-Scale Machine Learning. In *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, Savannah, GA, 2016; pp 265−283.
(41) Zeiler, M. D.; Krishnan, D.; Taylor, G. W.; Fergus, R. Deconvolutional Networks. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010; pp 2528−2535.
(42) Wang, J.; He, H.; Prokhorov, D. V. A folded neural network autoencoder for dimensionality reduction. *Procedia Comput. Sci.* **2012**, *13*, 120−127.
(43) Maas, A. L.; Hannun, A. Y.; Ng, A. Y. Rectifier Nonlinearities Improve Neural Network Acoustic Models. In *Proceedings of the 30th International Conference on Machine Learning*, Atlanta, GA, 2013; pp 3.
(44) Long, J.; Shelhamer, E.; Darrell, T. Fully Convolutional Networks for Semantic Segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015; pp 3431−3440.
(45) Kingma, D. P.; Ba, J. Adam: A Method for Stochastic Optimization. arXiv Preprint arXiv:1412.6980, 2014.
(46) Garetz, B.; Balsara, N.; Dai, H.; Wang, Z.; Newstein, M.; Majumdar, B. Orientation correlations in lamellar block copolymers. *Macromolecules* **1996**, *29*, 4675−4679.
(47) Ruiz, R.; Sandstrom, R. L.; Black, C. T. Induced orientational order in symmetric diblock copolymer thin films. *Adv. Mater.* **2007**, *19*, 587−591.
(48) Ruiz, R.; Bosworth, J.; Black, C. Effect of structural anisotropy on the coarsening kinetics of diblock copolymer striped patterns. *Phys. Rev. B: Condens. Matter Mater. Phys.* **2008**, *77*, No. 054204.
(49) Xie, N.; Li, W.; Zhang, H.; Qiu, F.; Shi, A.-C. Kinetics of lamellar formation on sparsely stripped patterns. *J. Chem. Phys.* **2013**, *139*, No. 194903.
(50) Perego, M.; Lupi, F. F.; Ceresoli, M.; Giammaria, T. Jacopo.; Seguini, G.; Enrico, E.; Boarino, L.; Antonioli, D.; Gianotti, V.; Sparnacci, K.; Laus, M. Ordering dynamics in symmetric PS-b-PMMA diblock copolymer thin films during rapid thermal processing. *J. Mater. Chem. C* **2014**, *2*, 6655−6664.
(51) Zhang, T.; Suen, C. Y. A fast parallel algorithm for thinning digital patterns. *Commun. ACM* **1984**, *27*, 236−239.
(52) Lee, T.-C.; Kashyap, R. L.; Chu, C.-N. Building skeleton models via 3-D medial surface axis thinning algorithms. *CVGIP: Graph. Models Image Process.* **1994**, *56*, 462−478.
(53) Hahm, J.; Sibener, S. Time-resolved atomic force microscopy imaging studies of asymmetric PS-b-PMMA ultrathin films: Dislocation and disclination transformations, defect mobility, and evolution of nanoscale morphology. *J. Chem. Phys.* **2001**, *114*, 4730−4740.
(54) Tsarkova, L.; Knoll, A.; Magerle, R. Rapid transitions between defect configurations in a block copolymer melt. *Nano Lett.* **2006**, *6*, 1574−1577.
(55) Ghahramani, Z. Probabilistic machine learning and artificial intelligence. *Nature* **2015**, *521*, 452−459.

https://doi.org/10.1021/acs.macromol.1c01583
Macromolecules 2021, 54, 10074−10085