MRS MATERIALS RESEARCH SOCIETY®

Advancing materials. Improving the quality of life.

OPEN DATA CHALLENGE

IOWA STATE UNIVERSITY
OF SCIENCE AND TECHNOLOGY

Deep Generative Models for Creating Synthetic Microstructures

Viraj Shah, Balaji Sesha Sarath Pokuri, Rahul Singh,
Chinmay Hegde, Baskar Ganapathysubramanian, Soumik Sarkar

Contact email: viraj@iastate.edu

Objective

- A key problem in computational material science deals with understanding the effect of material distribution (i.e., **microstructure**) on material performance.
- The challenge we consider here is to **synthesize microstructures with desired physical and chemical properties**, given a finite number of microstructure images, evaluated based on the **physical invariances that the microstructure exhibits**.
- Conventional approaches are based on stochastic optimization and are **computationally intensive**.
- We introduce Machine learning based generative models for the **fast synthesis of binary microstructure** images.
- Our model is a **Wasserstein Generative Adversarial Network** that uses a finite number of training images to **synthesize new microstructures that satisfy the physical invariances** respected by the original data.
- For the training of our model, we curated a dataset of **Binary 2D microstructural images of polymer phase separation.** We made our dataset available publically.

- In this presentation, we discuss:
  1. Details of **dataset generation**
  2. **Physical invariances** under consideration: volume fraction and two-point correlation
  3. Introduction to deep generative adversarial networks (**GANs**) and our method.
  4. **Microstructure images generated** by our WGAN model, and their analysis
  5. **Conclusion** on usefulness and effectiveness of our model

1.1 Dataset

We curated a dataset of **Binary 2D microstructural images of polymer phase separation (CH - dataset).**

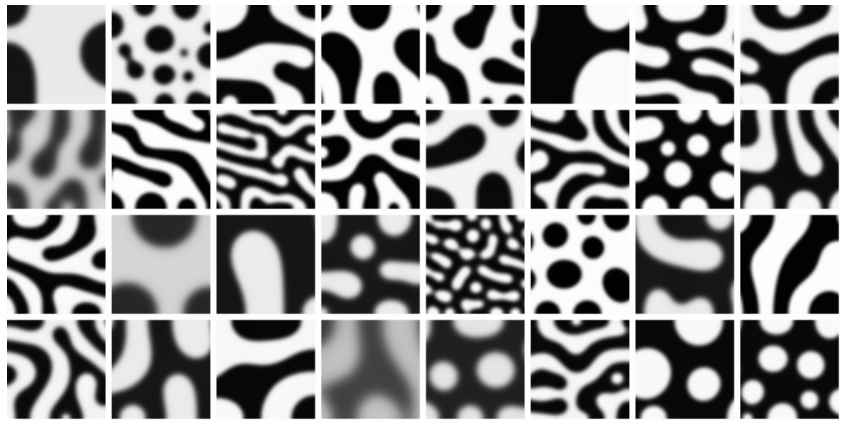
- This dataset was generated through the simulation of a time evolving **Cahn -Hilliard equation [1]**, describing **phase separation in binary polymer blends**. Thus, we call it **CH-dataset**.
- Several realizations of the equation were done through **different values of volume fractions** and **binary interaction parameters**.
- Morphologies were outputted at **constant** time intervals.

Details of the dataset

- Collection of **34672 grayscale images**,
- resolution of each image is **101x101**.
- Each pixel takes a floating point value **between 0 and 1.**
- **Dataset and supporting code is made public at:**

  https://zenodo.org/record/2580293

Samples from the dataset



1.2 Dataset Generation

dataset was generated through the simulation of a time evolving **Cahn -Hilliard equation [1]:**

$$\frac{\partial \phi}{\partial t} = \nabla\left(M\nabla\left(\frac{\partial f}{\partial \phi} - \epsilon^2\nabla^2\phi\right)\right)$$

$$f(\phi) = \phi\ln(\phi) + (1-\phi)\ln(1-\phi) + \chi_{12}\phi(1-\phi)$$

- Simulations were performed on a **101×101** (pixels) square domain, for **10 values** of $\chi_{12} \in [2.2, 3.4]$;

  10 values of total volume fraction $\phi_0 = \int_\Omega \phi d\Omega \in [0.3, 0.5]$. Also, $\epsilon = 1e-2$, $l_x, l_y = 1$, $M = 1$, $dt = 1e-3$.
- Data was saved after every prescribed number of **time steps (20 time steps).**
- Data was then **augmented** by reflecting and flipping the phases(black/white)

- Because data was generated using an **energy dissipating equation**, data generated using one simulation has:
  - The same (volume) fraction for all times ($\int_\Omega \phi d\Omega = constant\ \forall\ t$),
  - With increasing time, the average domain size increases as $r_{avg} \propto t^{\frac{1}{3}}$. *i.e.*, r_{avg}increases with time.

2. Physical Invariances

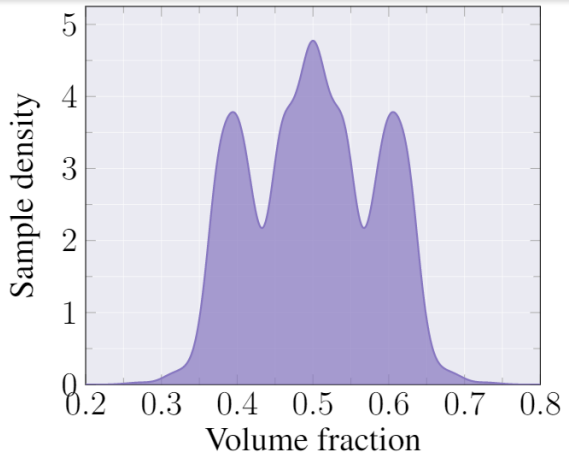
We consider the underlying material to be a **two-phase homogeneous, isotropic** material.
Our setup for statistical characterization of microstructure follows with [2].

- A phase function $\phi^{(1)}(.)$ is used to characterize this two-phase system, defined as: $\phi^{(1)}(\boldsymbol{r}) = \begin{cases} 1, \boldsymbol{r} \in V_1, \\ 0, \boldsymbol{r} \in V_2, \end{cases}$

  where V_1 is the region occupied by phase 1 and V_2 is the region occupied by phase 2.
- The 1-point correlation function, $p_1^{(1)}$, commonly known as **volume fraction**, is defined as: $p_1^{(1)} = \mathbb{E}_{\boldsymbol{r}} \phi^{(1)}(\boldsymbol{r})$
- The **two-point correlation function**, $p_2^{(1)}$, is defined as: $p_2^{(1)}(r_{12}) = \mathbb{E}_{\boldsymbol{r_1}, \boldsymbol{r_2}}[\phi^{(1)}(\boldsymbol{r_1})\, \phi^{(1)}(\boldsymbol{r_2})]$

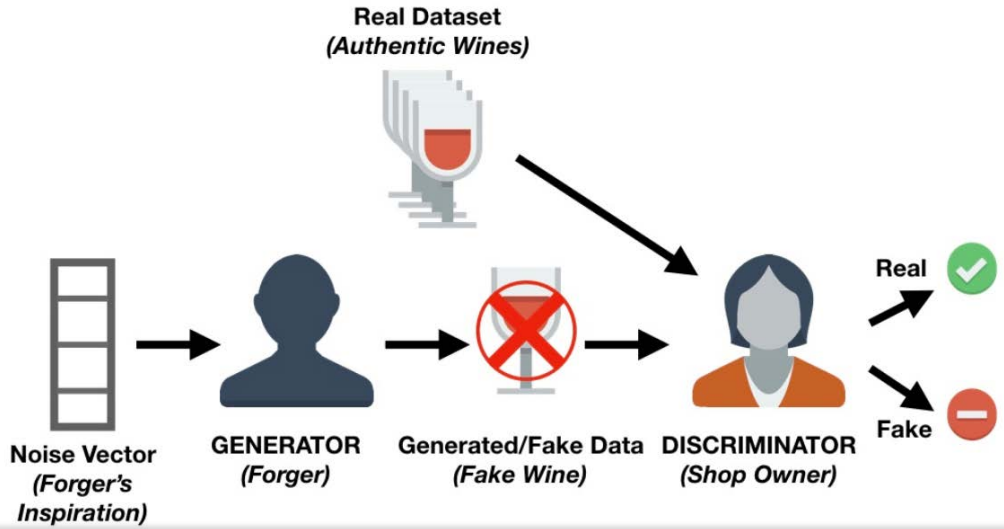
- Our aim is to synthesize microstructures **that satisfy certain target statistical properties** of the material distribution.
- We choose these statistical properties to be **total volume fraction of a material (p_1)** and **2-point correlation (p_2)**.

Distribution of volume fraction for our dataset



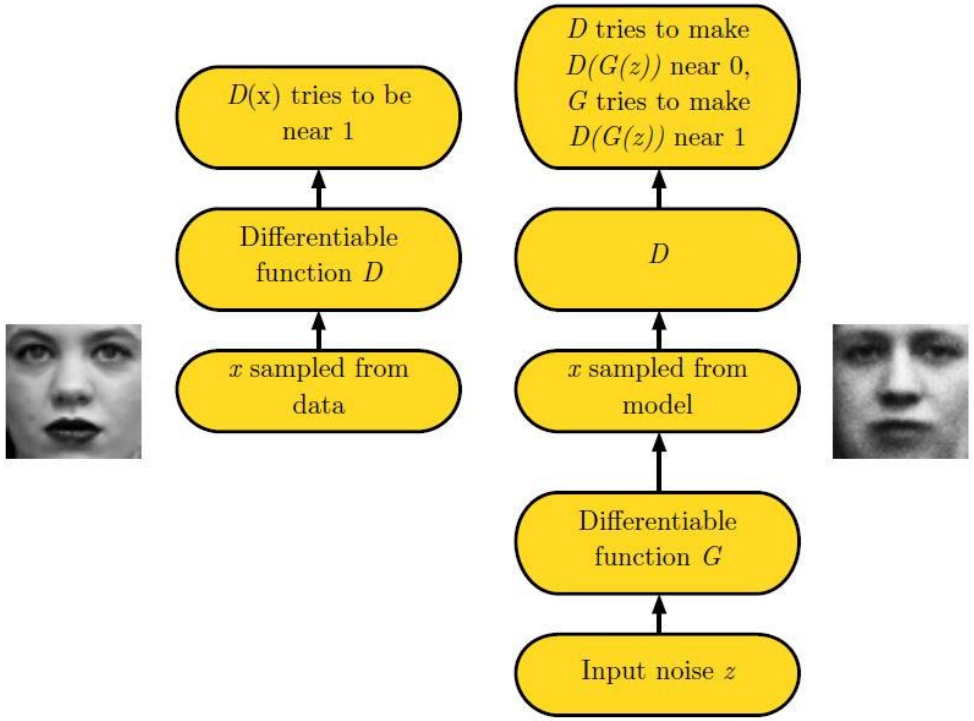
3.1 Introduction to Generative Adversarial Networks [4]

The basic idea of GANs is to set up a minimax game between two players: **generator** and **discriminator** [4].



- We can think of the **generator** as being like a forger, trying to make fake wine, and the **discriminator** as being like wine-shop owner, trying to allow legitimate wines and catch fake wines.
- To succeed in this game, the **forger must learn to make wine that is indistinguishable from genuine wine** implying that the generator network must learn to create samples that are drawn from the same distribution as the training data.

GAN model

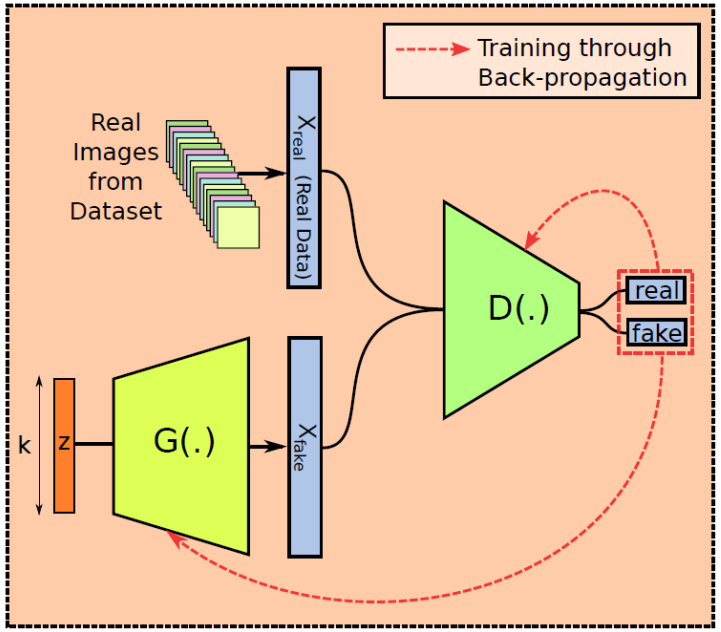


$$\min_G \max_D V(D,G) = E_{x\sim p_{data}(x)}[logD(x)] + E_{z\sim p_z(z)}[\log(1-D(G(z)))]$$

Image credits: (left) datacamp blog, (right) GAN tutorial by Ian Goodfellow

3.2 Our methodology

We use our CH dataset as the training data for a **improved version** of GAN – called **Wasserstein GAN with Gradient Penalty [5].**



WGAN-GP-CH	WGAN-GP-CHp1	WGAN-GP-CHp2
•Trained over CH dataset that contains all 34672 images covering entire range of p1 and p2.	•Trained over CHp1 dataset with ~6000 images of CH dataset having a volume fraction (p_1) value between 0.35 to 0.45.	•Trained over CHp2 dataset with~3000 images of CH dataset having a (p_2) value equal to 0.0625.

- Using the **Cahn-Hilliard (CH)** dataset, we prepare two smaller datasets referred as **CHp1** and **CHp2** by segregating the images based on their p_1 and p_2 values respectively.
- The first subset **CHp1** is a collection of all images of CH dataset having a volume fraction **(p_1) value between 0.35 to 0.45**.
- The second subset is segregated on the basis of p_2 values of images, and contain all the images from CH dataset having 2 – point correlation **(p_2) value equal to 0.0625.**
- We train **3** WGAN-GP using **CH, CHp1 and CHp2** as the training data respectively.
- As these segregated datasets typically contain images with similar statistical properties (either p_1 or p_2), we can testify the ability of our model to preserve such properties by observing the p1 or p2 values of the images generated by these 3 networks.

4.1 Results

Our trained WGAN-GP models are able to generate microstructures closely resembling the real microstructures. Moreover, these generated microstructures respect the statistical constraints (p_1 and p_2)

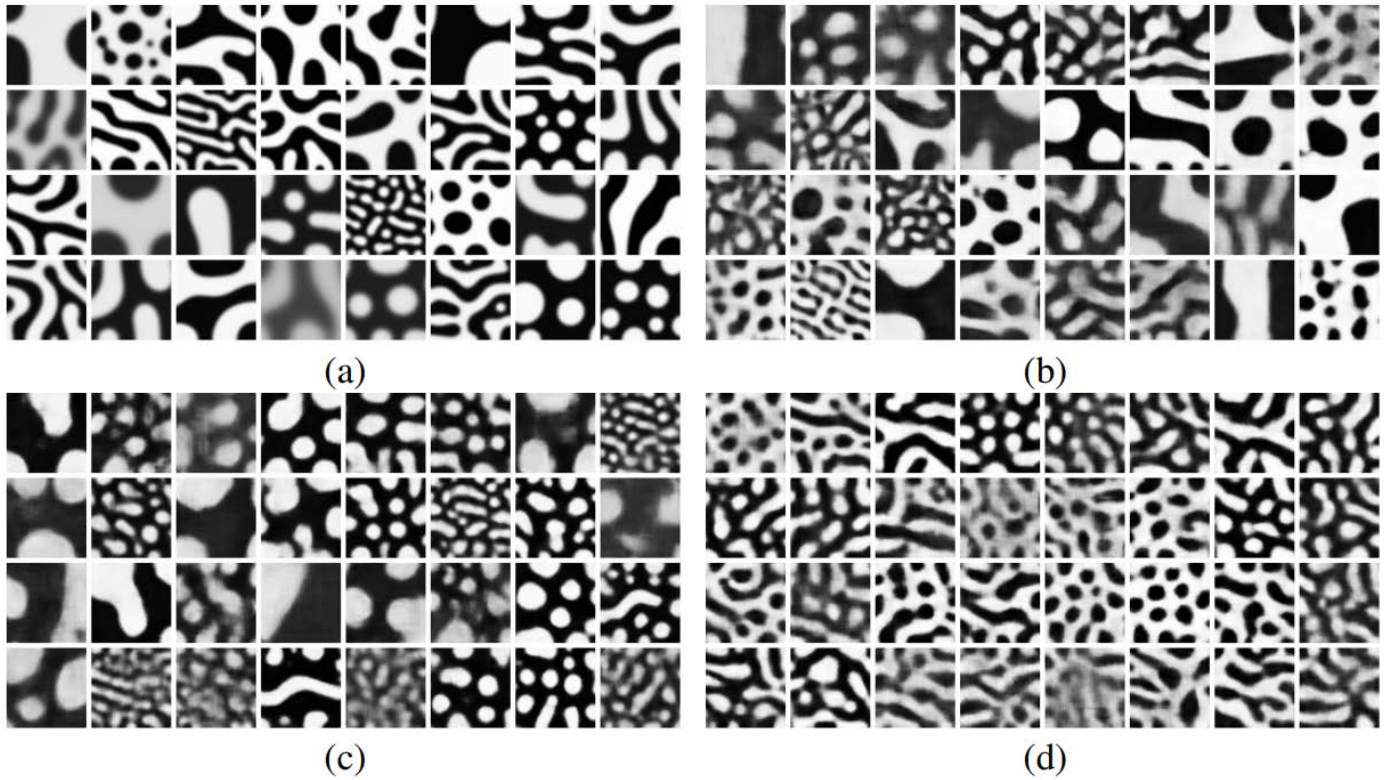


Figure 2: (a) Sample images from Cahn-Hilliard dataset; (b) samples generated by WGAN-GP trained on CH-dataset; (c) Samples generated by WGAN-GP trained over the morphologies from CH_{p1} dataset (only includes the images with volume fraction between 0.35 to 0.45); (d) Samples generated by WGAN-GP trained over the morphologies from CH_{p2} dataset (only includes the images with $2-$point correlation equal to 0.0625).

4.2 Analysis of the Results

We analyze (i) the **statistical properties** of the generated miscrostructures; (ii) **interpolation behavior** of our trained WGAN models; (iii) **Free energy of generated microstructures** to present interesting findings.

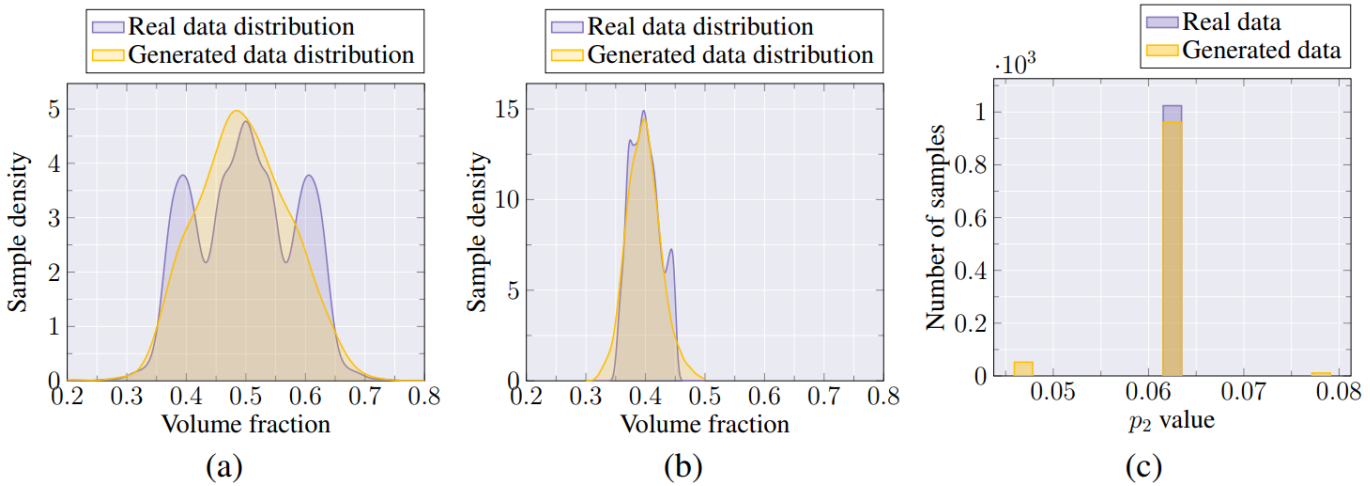


Figure 3: (a) Comparisons of the distributions of volume fractions of training dataset and that of the samples generated by WGAN-GP trained over entire CH dataset; (b) Comparisons of the distributions of volume fractions of training dataset and that of the samples generated by WGAN-GP trained over the CH_{p1} dataset; (c) Histograms of p_2 correlation values for samples from CH_{p2} dataset and samples generated by WGAN-GP trained over CH_{p2} dataset.

- Fig. 3 provides the density plots/histograms of p_1 and p_2 values of the images for both training data and generated data.
- The striking similarities in the spread of both the density plots/histograms suggest that our **network successfully reproduces the statistical properties** of the real (training) images in the simulated images.
- In Fig. 3 (a,b), densities of p_1 value is compared between the real data and generated data for network trained on CH dataset and CHp1 dataset.
- We provide the histogram for p2 values for models trained using CHp2 in Fig. 3 (c). **Both the histograms closely match.**

4.3 Analysis of the Results

We display interesting behavior of the learned image manifold **through results of interpolation over latent vectors z**

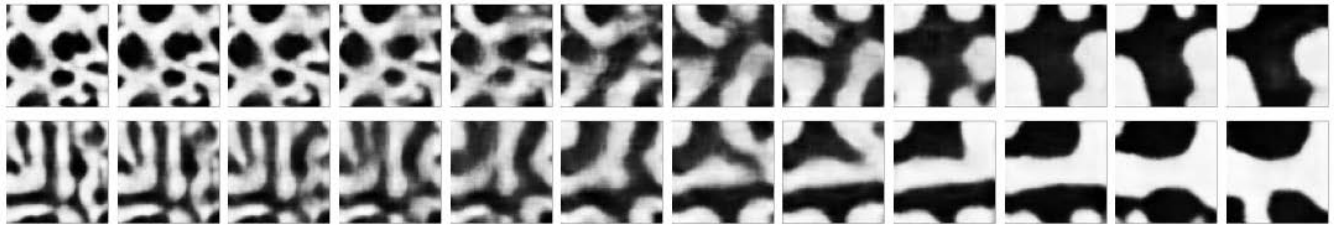
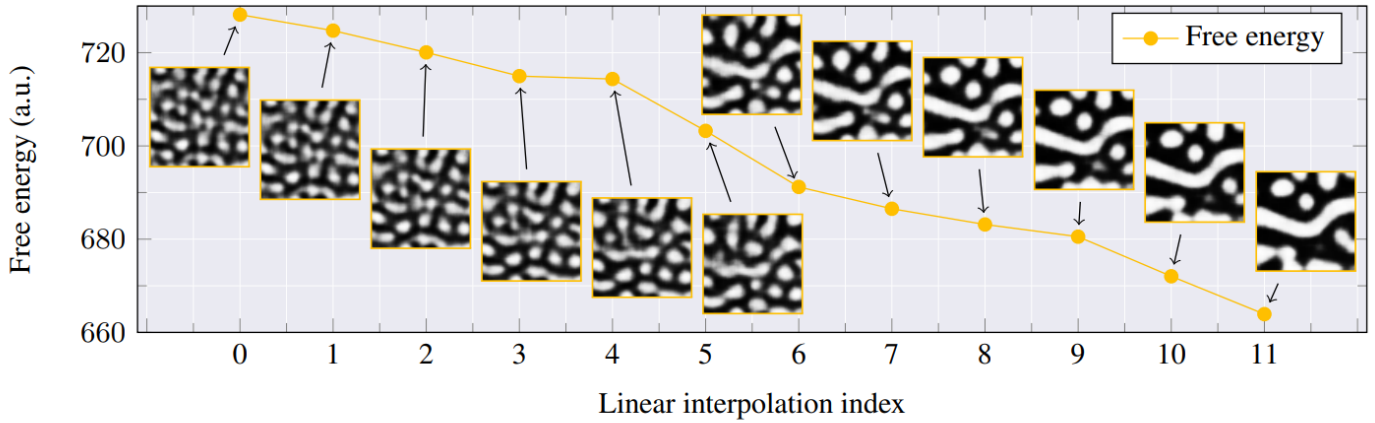


Figure 4: (a) Results of the linear interpolation over latent variable z for WGAN-GP trained over the entire CH dataset

- Randomly pick two different noise vectors z_1 and z_2, and linearly interpolate between them to obtain 10 more such noise vectors. All 12 images are plotted as in Fig. 4.



(b)

Figure 5: (a) Results of the linear interpolation over latent variable z for WGAN-GP trained over the images from CH_{p1} dataset (only includes the images with volume fraction between 0.35 to 0.45); volume fraction values are printed above each image; (b) Free energy of the interpolated morphologies (volume fraction of each image is $(0.41 \pm 5\%)$.

- Very interestingly, we observe that an unseen invariance of energy minimization is captured in the interpolation. As shown in Fig. 5 (b), the free energy of the morphologies decreases as the we move from one interpolation step to the next.
- In this process, the volume fraction (p_1) is also **preserved within reasonable limits**. This suggests that the WGAN framework is able to learn latent physical rules from the dataset.
- Such behavior can effectively be used to decide **series of manufacturing processes** for obtaining final morphology from initial morphology **without adding any new material.**

5. Conclusion

- We curated a **Binary 2D microstructural images of polymer phase separation (CH - dataset)** by solving CH equations and successfully used it with a machine learning model that **approaches computational results in microstructure synthesis tasks**.
- We also made our dataset **publically available**.
- We train three different WGAN models with full or subset of our CH dataset. We show that the generated images **respect the distribution of certain physical invariances** - specifically, can be used **for synthesizing a promising new material that meets a desired performance target.**
- We analyze our results to come up with **interesting properties of our trained WGAN models** such as interpolation and free energy reduction behavior and explore previously unknown correlations in **process–structure–property linkages**.

References

1. O. Wodo, J. Zola, B.S.S. Pokuri, P. Du, & B. Ganapathysubramanian, (2015). Automated, high throughput exploration of process–structure–property relationships using the MapReduce paradigm. Materials discovery, 1, 21-28.
2. Salvatore Torquato. Random heterogeneous materials: microstructure and macroscopic properties, volume 16. Springer Science & Business Media, 2013.
3. R. Singh, V. Shah, B. Pokuri, S. Sarkar, B. Ganapathysubramanian, and C. Hegde, Physics-aware Deep Generative Models for Creating Synthetic Microstructures, NeurIPS 2018 Workshop on Machine Learning for Molecules and Materials. (preprint arXiv:1811.09669)
4. I. Goodfellow, J. Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In NeurIPS, 2014.
5. I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville. Improved training of wasserstein gans. In NeurIPS, 2017.