A Transformer-Based GAN Architecture for Simulating Carbon Nanotube Structures in the Frequency Domain

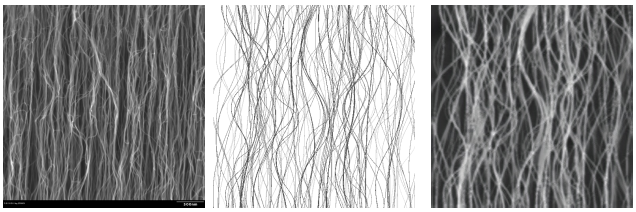
Trevor Bohl[1], Minasadat Attari[1], Matthew R. Maschmann[2,3], Filiz Bunyak[1,3*]
[1]Department of Electrical Engineering and Computer Science
[2]Department of Mechanical and Aerospace Engineering
[3]MU Materials Science and Engineering Institute
University of Missouri-Columbia, MO, USA
{tdbkcx, ma8pz, maschmannm, bunyak}@missouri.edu

Abstract

The design and discovery of new materials with specific properties is a costly and time-consuming process because of the vast search space and the high cost of experimental testing. Simulation and machine learning offer promising solutions to address these challenges. Carbon nanotubes (CNTs) are highly promising nanoscale materials renowned for their exceptional mechanical, electrical, and thermal properties, and have a wide range of potential applications. The properties of CNTs are related to their structural characteristics, which are determined by the growth parameters. In this paper, we present CNT-Former, a generative adversarial network (GAN) using a transformer-based architecture and frequency domain encoding to simulate CNT structures based on given growth parameters. Our ultimate objective is to identify the optimal growth parameters for the desired structures and properties. This approach provides a more scalable alternative to traditional finite element simulators while maintaining high accuracy in simulating CNT structures. Additionally, it allows for the integration of real, multi-modal experimental data, grounding the simulations in actual experimental results. Experimental results demonstrate promising structural fidelity and strong class discrimination capabilities for CNT-Former.

1. Introduction

Carbon nanotubes (CNTs) [7] are cylindrical nanostructures made of carbon atoms arranged in a hexagonal lattice. CNTs offer unique mechanical, electrical, and thermal properties that are useful across various industries, from consumer electronics and energy devices to biomedical and healthcare [9]. CNTs are typically grown through chemical vapor deposition (CVD) or other similar methods [10]. The growth process involves the deposition of carbon atoms onto a substrate, where they form CNTs under specific conditions. The growth conditions determine the structure and properties of the resulting CNTs. Development of new CNT materials with desired properties is an expensive and time-consuming process, since many experiments need to be conducted to grow and test the new CNTs. Simulation and machine learning offer new opportunities in the design and discovery of new materials. Previous work developed a physics-based finite element simulation to model carbon nanotube (CNT) forests [10, 14]. This model considers attraction between neighboring CNTs and allows the growing and deforming CNTs to interact and react based on a balance of forces [14] (Figure 1). This physics-based simulation effectively captures the morphology and overall properties of CNT forests, allowing researchers to analyze and predict their mechanical properties. By combining this with a custom style transfer [11], current methods also enable generation of high-fidelity, photo-realistic images of CNT forests. These images are critical for training and evaluation of image analysis methods, which are essen-



(a) (b) (c)

Figure 1. Real and simulated CNT forest images. (a) Real CNT forest imaged using scanning electron microscope (SEM); (b) CNT forest simulated using physics-guided finite element simulations [14]; (c) SEM-like photo-realistic image of the simulated CNT forest using neural style transfer [11]

.

*Corresponding author.

tial for segmentation and characterization [6, 11], for non-destructive material property prediction, and for exploring structure-property relationships. Although highly valuable, this physics-based finite element simulation has two main limitations: its high computational cost and its inability to integrate real experimental data. The finite element simulation involves inverting large matrices, which makes it slow and prone to numerical instability. Furthermore, this approach lacks a mechanism for incorporating experimental data, such as the structure of CNTs grown under specific environmental conditions to improve the simulation.

Recent years have seen rapid advancements in generative AI, particularly in generative adversarial networks (GANs). GANS frame the learning process, the mapping from latent space to generated data points, as a zero-sum adversarial game between two players [5]. These models have been well studied and widely used in the field of generative AI, having seen the quickest adaptation in computer vision for image generation [1, 16], and have since rapidly expanded into other areas and data modes. Our study proposes an adversarially trained deep generative neural network framework for generating individual CNT samples, in interest of further enhancing the quality of data we are able to generate for designing automated image analysis methods for CNT forests.

These advances have led to the proposal of several adversarially trained networks for signal synthesis, many of which employ autoregressive methods in both the generator and discriminator [2, 15, 21]. While these methods are powerful, they are hindered by high computational complexity during both training and inference. In contrast, transformers [19] offer a model architecture that can be trained for sequence prediction and modeling without the need for autoregression at training time. As such, the transformers, particularly the transformer encoders, lend themselves nicely as architectures for learning mappings between the latent space and the data distribution that we aim to model. While the progression of thought for signal GANs is promising, transformer encoders were initially used as a backbone in computer vision tasks. The Vision Transformer (ViT) [4] demonstrated that embedding image patches as a sequence enabled transformer encoders to excel in image classification tasks. Building on this, TransGAN [8] applied the same principle to GANs for image generation, showing that a fully transformer encoder-based backbone could effectively drive image synthesis. TTS-CGAN [13] demonstrated that a GAN architecture with a transformer backbone could be effectively applied to signal generation by treating signals as one-dimensional images and adapting the ViT's patch division to this representation.

The Fourier transform plays a central role in signal analysis by converting a signal from the time domain to the frequency domain. This transformation provides valuable insights into the frequency components of a signal, which are often more useful for analysis. The Fourier transform represents periodic signals as a sum of sinusoidal functions with varying frequencies, where these components are encoded using complex numbers. Many standard operations that deep networks rely on are not defined on the outputs of a Fourier transform directly, one has to take a few liberties, or compute other abstractions on the components. For example, in the MelGAN architecture [12], the outputs of a short-time Fourier transform are multiplied with a Mel basis in order to form a Mel spectogram as input to the network. Fourier encodings of this sort are done to allow networks to learn very high frequency data more easily, and stably [17].

In this paper, we present a novel CNT simulator based on generative adversarial networks (GANs) with a transformer-based architecture and frequency domain encoding. This approach not only addresses the high computational cost of physics-based finite element simulations but also facilitates the integration of real multi-modal experimental data, grounding the simulation in actual data and enhancing its representational accuracy.

2. Methods

2.1. Physics-guided CNT simulation

The synthetic dataset used in our study was generated through physics-based finite element simulations, as described in [10, 14]. This model accounts for van der Waals (vdW) attraction between neighboring CNTs and enables the growing and deforming CNTs to interact and respond based on a balance of forces [14]. Given a set of input parameters, the physics-based simulation produces a CNT forest model $\mathcal{M} = \{M_1, M_2, \ldots, M_m\}$ where each M_i represents an individual CNT defined by its coordinates $M_i = \{(x_1, y_1), (x_2, y_2), \ldots, (x_k, y_k)\}$. The input parameters such as CNT growth rate and density affect the morphology, particularly curvature, of the generated CNTs.

2.2. Network architecture

We developed a generative adversarial network (GAN), *CNT-Former*, with a transformer-based architecture and frequency-domain encoding. This network is illustrated in Figure 2 and is described in the following.

Transformer structure: The structure of the main encoder blocks used as feature extractors within both networks uses a pre-norm structure [20], and learned positional encodings were used in both networks.

Generator: The generator of the proposed CNT-Former network is made up of three stacked transformer encoders, broken up by linear reprojection layers and succeeded by a single linear output head for the final reprojection into $\frac{N}{2} \times 2$ Fourier coefficients, which are then passed to the discrimi-

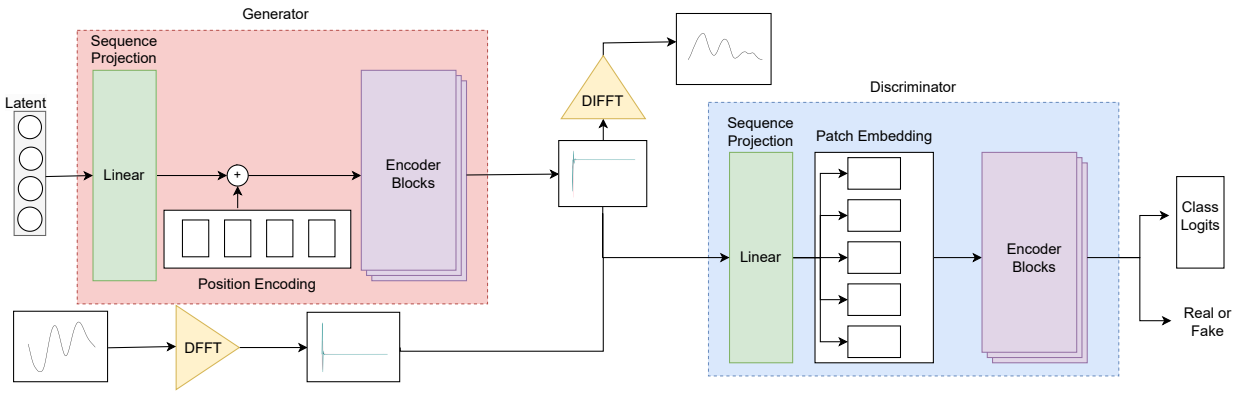


Figure 2. CNT-Former network architecture. The generator (pink block) outputs Fourier components, which are input to the Discriminator (blue block) directly without transforming back to the temporal domain. The generator outputs both adversarial logits and classification logits, which helps the model to adhere to the class distributions of the data.

nator. The generator is also provided with discretized input parameter labels, which are mapped through a learned embedding and concatenated onto the latent vector before the whole vector is fed into the generator. The adversarial loss per sample for the generator is defined as

$$\mathcal{L}_{\mathcal{G}_{adv}} = -\mathcal{P}_{\mathcal{F}} \tag{1}$$

where $\mathcal{P}_{\mathcal{F}}$ is the adversarial logit of the discriminator for the fake sample. The classification loss for the generator is defined as the cross-entropy between the discriminator predicted class logits for the fake sample, and the class indices that the generator received as input in generating that sample,

$$\mathcal{L}_{\mathcal{G}_{cls}} = -\sum_{i=1}^{N_C} y_f(i) \log(\mathcal{P}_{FC}(i)) \tag{2}$$

where $\mathcal{P}_{FC}$ denotes the discriminator's class predictions on the fake sample. The combined loss is defined as,

$$\mathcal{L}_{\mathcal{G}} = \alpha \mathcal{L}_{\mathcal{G}_{adv}} + \beta \mathcal{L}_{\mathcal{G}_{cls}} \tag{3}$$

where α, β are hyperparameters set by the user. In our experiments, we found $\alpha = 1.2$ and $\beta = 0.8$ to work best heuristically. This combined loss encourages the generator to generate samples which match the class distribution in addition to winning the adversarial game against the discriminator [3, 13].

Discriminator: The discriminator is made up of a single transformer encoder layer and two linear classification heads, one for adversarial logits and one for class logits, with the input reshaped into patches and a class (cls) token appended to each patch [4, 13]. Since the data are relatively low dimensional, the data in the discriminator is also linearly reprojected into a sequence with higher feature dimensionality and higher sequence length in order to allow for the patch divisions. We found in practice that these learned reprojections were more effective than simple padding or up-sampling. The same was true in the generator. The adversarial loss per sample for the discriminator is defined as the hinge loss for the adversarial predictions:

$$\mathcal{L}_{\mathcal{D}_{adv}} = \max(0, 1 + P_R) + \max(0, 1 - P_F) \tag{4}$$

where P_R and P_F denote the discriminator predictions for the real and fake samples, respectively. The per batch loss is simply the mean of these per sample scores. The classification loss is simply the cross-entropy between the discriminator's class logit outputs, and the class indices for the real signal.

$$\mathcal{L}_{\mathcal{D}_{cls}} = -\sum_{i=1}^{N_C} y_r(i) \log(\mathcal{P}_{RC}(i)) \tag{5}$$

where $\mathcal{P}_{RC}$ denotes the discriminator's class predictions on the real samples. The combined loss is a weighted sum of these two losses,

$$\mathcal{L}_{\mathcal{D}} = \alpha \mathcal{L}_{\mathcal{D}_{adv}} + \beta \mathcal{L}_{\mathcal{D}_{cls}} \tag{6}$$

2.3. Frequency-domain processing

Both generator and discriminator were designed to operate in frequency domain through the use of Fourier coefficients. The discriminator was designed to accept a sequence of size $(\frac{N}{2}, 2)$, representing the real and imaginary components of the Fourier decomposition of a CNT signal. This reformulates the generator's task, shifting it from learning a signal based on 1D spatial information to learning it through 2D frequency components. The generator was trained to generate $(\frac{N}{2}, 2)$ frequency components of a signal, customized with respect to the input class indices.

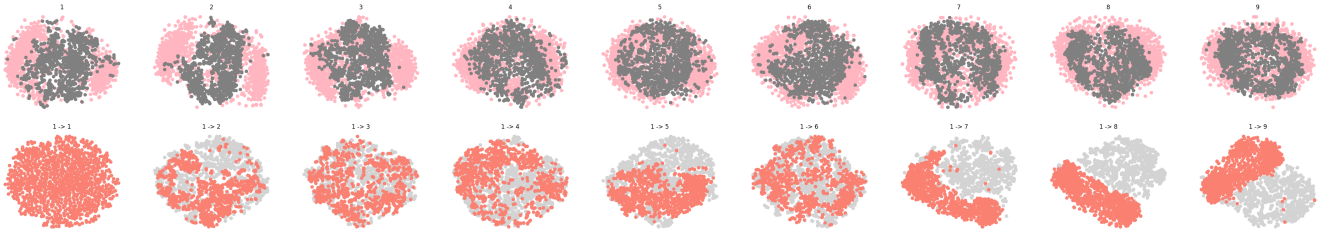
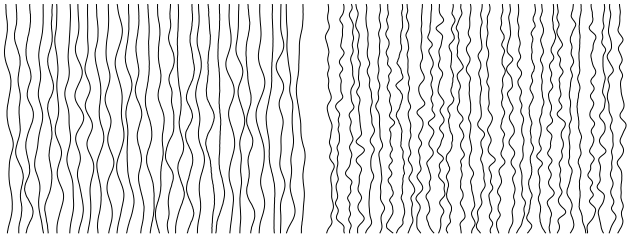


Figure 3. t-SNE (t-Distributed Stochastic Neighbor Embedding) analysis [18] between physics-based and CNT-Former generated CNTs for each of the nine classes (top row) and between class 1 and classes 1-9 of CNT-Former generated CNTs (bottom row).



(a) class=1 (b) class=9

Figure 4. CNT-Former generated CNT samples from the lowest and highest frequency classes, plotted in the temporal domain after being reconstructed from the output Fourier components.

3. Experiments

The proposed network was trained for 50 epochs over the data. We found that the data distribution generally required a smaller batch size, and so training was carried out in batches of size 16, for a total of 51207 training updates. The generator and critic were both set to have hidden dimensions of 256, 16 attention heads, and a model depth of 12, although we don't doubt that a larger model might improve results. Training took roughly 45 hours on one NVIDIA A100 GPU.

3.1. Experimental results and evaluation

Figure 4 displays sample CNTs generated from two input parameter classes using the proposed CNT-Former network. These results not only highlight the structural fidelity to real CNTs but also demonstrate high within-class and low inter-class structural similarities.

3.1.1. t-distributed Stochastic Neighbor Embedding

To further evaluate the quality of the generated CNTs, we employed the t-distributed Stochastic Neighbor Embedding (t-SNE) [18] dimensionality reduction technique to visualize the latent structure of the CNTs.

The top row of Figure 3 illustrates pairwise embeddings between physics-based and CNT-Former generated CNTs across all nine classes. The close clustering of corresponding data points indicates strong structural similarity, underscoring the generative capacity of CNT-Former in replicating the distribution of physically simulated CNTs.

The bottom row of Figure 3 presents pairwise t-SNE embeddings between CNT-Former generated CNTs from class 1 and those from classes 1 through 9. The clear separation between these clusters demonstrates the model's ability to capture class-specific variations, highlighting its discriminative power.

It is important to note that for certain class pairs, particularly between low-frequency and middle-frequency CNTs, the structural differences were inherently subtle, even in the physics-based simulated data that serve as our ground truth. This intrinsic similarity limits the degree of inter-class separation observable in the t-SNE plots and partially explains why CNT-Former also exhibits less distinction between these specific classes. These overlaps reflect the challenge posed by ambiguous class boundaries in the input data itself, rather than a limitation of CNT-Former.

3.1.2. Inter-class scatter analysis

To further investigate the discriminative and generative properties of CNT-Former, we analyzed the inter-class scatter, defined as:

$$S_B = \sum_{k=1}^{C} n_k(\mu_k - \mu)(\mu_k - \mu)^T, \tag{7}$$

where C is the total number of classes, n_k is the number of samples in class k, μ_k is the mean vector of class k, μ is the overall mean vector across all classes, and $(\mu_k - \mu)(\mu_k - \mu)^T$ is the outer product that quantifies the deviation of class k's mean from the overall mean.

As illustrated in Figure 5, this inter-class scatter matrix S_B provides insight into how well the feature representations of different classes are separated, with higher values indicating stronger class-wise discrimination. The red bars, representing the inter-class scatter between physics-based and CNT-Former generated CNTs, remain consistently low, indicating that the generative model effectively replicates the real data structure and preserves the distributional characteristics of individual classes. The blue bars, representing the inter-class scatter between class 1 and classes 1-to-9 of the CNT-Former generated CNTS, are not only higher compared to red bars (similarities between physics-based and

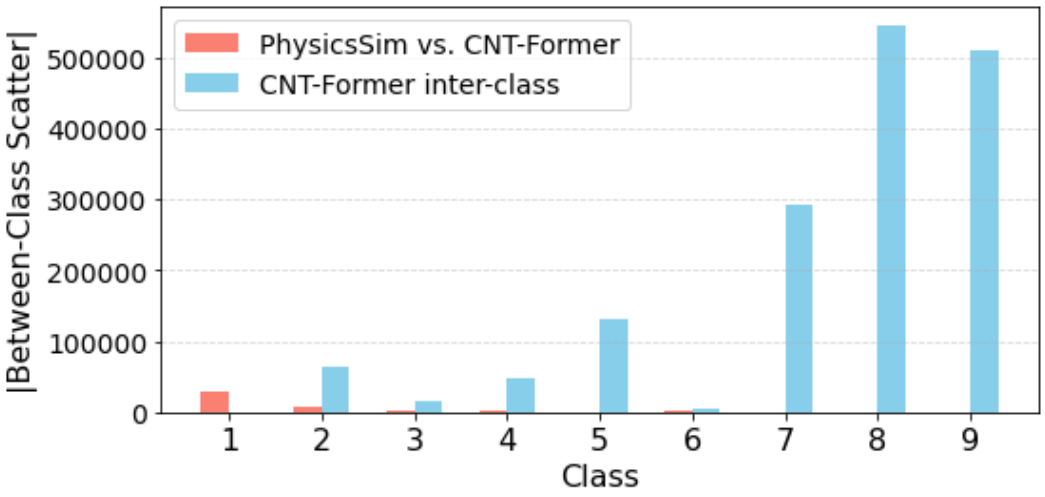


Figure 5. Inter-class scatter analysis between physics-based and CNT-Former generated CNTs (red) and between class 1 and classes 1-9 of CNT-Former generated CNTs (blue).

CNT-Former generated CNTs) but also grow significantly, particularly for the classes 7-to-9 (classes with highly different growth parameters compared to class 1). This pattern suggests that the generative model effectively learns to distinguish between different growth classes and associated structural properties. These results indicate that the model does not only replicate individual class distributions but also captures the relative differences between low- and high-frequency classes, reinforcing their distinguishability in the embedding space. We found that the higher frequency classes outperformed slightly in terms of their distribution quality when compared to the lower frequency classes. The model itself could also likely be pruned post training, as even the high frequency class CNTs have most of the Fourier components close to zero in both the real and the generated samples.

4. Conclusion and Future Works

In this paper, we introduced CNT-Former, a novel CNT structure simulator based on generative adversarial networks (GANs), using a transformer-based architecture and frequency-domain encoding. CNT-Former offers a scalable alternative to traditional simulators, reducing computational demands while maintaining the accuracy of simulated CNT structures. CNT-Former also enables integration of real, multi-modal experimental data to the model improving its accuracy. We demonstrated in our experiments that CNT-Former is able to generate diverse and realistic CNTs, while preserving signal feature differences between different growth parameter classes.

The structure of individual CNTs is shaped by both input growth parameters and interactions with neighboring CNTs. While CNT-Former models CNT growth at the individual level and does not explicitly incorporate CNT-to-CNT interactions, its learned representations implicitly capture some of the effects of these interactions. Future future work will focus on extending CNT-Former to simulate the collective growth of entire CNT forests by explicitly modeling interactions among neighboring CNTs, and additionally adding measures to consider continuous parameters for CNT growth to enhance both modeling fidelity and interpretability. Additionally, we will continue to explore avenues for evaluation of generated samples, in order to evaluate future iterations against this one, and all of them against data collected from real physical experiments.

5. Acknowledgement

This material is based upon work supported in part by the U.S. Army Corps of Engineers, Engineering Research and Development Center—Information Technology Laboratory (ERDC-ITL) under Contract W912HZ24C0022. Computational resources for this research have been supported by the National Science Foundation (NSF) under Award Number: OAC-2322063 and the NSF National Research Platform, as part of GP-ENGINE Award Number: OAC-2322218. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the U.S. Government or agency thereof.

References

[1] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. *CoRR*, abs/1809.11096, 2018. 2

[2] Eoin Brophy, Zhengwei Wang, Qi She, and Tomas Ward.

Generative adversarial networks in time series: A survey and taxonomy, 2021. 2

[3] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation, 2018. 3

[4] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *CoRR*, abs/2010.11929, 2020. 2, 3

[5] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014. 2

[6] Taher Hajilounezhad, Rina Bao, Kannappan Palaniappan, Filiz Bunyak, Prasad Calyam, and Matthew R Maschmann. Predicting carbon nanotube forest attributes and mechanical properties using simulated images and deep learning. *npj Computational Materials*, 7(1):134, 2021. 2

[7] Sumio Iijima. Carbon nanotubes: past, present, and future. *Physica B: Condensed Matter*, 323(1-4):1–5, 2002. 1

[8] Yifan Jiang, Shiyu Chang, and Zhangyang Wang. Transgan: Two transformers can make one strong gan. *arXiv preprint arXiv:2102.07074*, 1(3), 2021. 2

[9] Ayesha Kausar, Irum Rafique, and Bakhtiar Muhammad. Review of applications of polymer/carbon nanotubes and epoxy/cnt composites. *Polymer-Plastics Technology and Engineering*, 55:1167 – 1191, 2016. 1

[10] Gordon Koerner, Ramakrishna Surya, Kannappan Palaniappan, Prasad Calyam, Filiz Bunyak, and Matthew R Maschmann. In-situ scanning electron microscope chemical vapor deposition as a platform for nanomanufacturing insights. In *ASME International Mechanical Engineering Congress and Exposition*, page V02BT02A052, 2021. 1, 2

[11] Prashanth Reddy Kotha, Minasadat Attari, Matthew Maschmann, and Filiz Bunyak. Deep style transfer for generation of photo-realistic synthetic images of cnt forests. In *2023 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*, pages 1–7. IEEE, 2023. 1, 2

[12] Kundan Kumar, Rithesh Kumar, Thibault de Boissiere, Lucas Gestin, Wei Zhen Teoh, Jose Sotelo, Alexandre de Brebisson, Yoshua Bengio, and Aaron Courville. Melgan: Generative adversarial networks for conditional waveform synthesis, 2019. 2

[13] Xiaomin Li, Anne Hee Hiong Ngu, and Vangelis Metsis. Ttscgan: A transformer time-series conditional gan for biosignal data augmentation, 2022. 2, 3

[14] Matthew R Maschmann. Integrated simulation of active carbon nanotube forest growth and mechanical compression. *Carbon*, 86:26–37, 2015. 1, 2

[15] Olof Mogren. C-rnn-gan: Continuous recurrent neural networks with adversarial training, 2016. 2

[16] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks, 2016. 2

[17] Matthew Tancik, Pratul P. Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan T. Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *CoRR*, abs/2006.10739, 2020. 2

[18] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9 (86):2579–2605, 2008. 4

[19] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023. 2

[20] Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tie-Yan Liu. On layer normalization in the transformer architecture, 2020. 2

[21] Jinsung Yoon, Daniel Jarrett, and Mihaela van der Schaar. Time-series generative adversarial networks. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2019. 2