DIFF-PIC: REVOLUTIONIZING PARTICLE-IN-CELL NUCLEAR FUSION SIMULATION WITH DIFFUSION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

The rapid development of AI highlights the pressing need for sustainable energy, a critical global challenge for decades. Nuclear fusion, generally seen as an ultimate solution, has been the focus of intensive research for nearly a century, with investments reaching hundreds of billions of dollars. Recent advancements in Inertial Confinement Fusion have drawn significant attention to fusion research, in which Laser-Plasma Interaction (LPI) is critical for ensuring fusion stability and efficiency. However, the complexity of LPI upon fusion ignition makes analytical approaches impractical, leaving researchers depending on extremely computation-demanding Particle-in-Cell (PIC) simulations to generate data, presenting a significant bottleneck to advancing fusion research. In response, this work introduces Diff-PIC, a novel framework that leverages conditional diffusion models as a computationally efficient alternative to PIC simulations for generating high-fidelity scientific LPI data. In this work, physical patterns captured by PIC simulations are distilled into diffusion models associated with two tailored enhancements: (1) To effectively capture the complex relationships between physical parameters and corresponding outcomes, the parameters are encoded in a physically-informed manner. (2) To further enhance efficiency while maintaining high fidelity and physical validity, the rectified flow technique is employed to transform our model into a one-step conditional diffusion model. Experimental results show that Diff-PIC achieves 16,200$\times$ speedup compared to traditional PIC on a 100 picosecond simulation, with an average reduction in MAE / RMSE / FID of 59.21% / 57.15% / 39.46% with respect to two other SOTA data generation approaches.

1 INTRODUCTION

Sustainable energy stands as one of the paramount challenges of our era, particularly with the rapid advancement of AI. The recent successful demonstration of fusion ignition (Abu-Shawareb et al., 2024) underscores the transformative potential of fusion as a sustainable energy source. In 2023 and 2024, the National Ignition Facility (NIF) achieved groundbreaking milestones, generating 3.4 MJ and 5.2 MJ of fusion energy, respectively, from 2.2 MJ input energy. Given that the estimated output could reach $\sim$120 MJ (Suter et al., 2004), there is a growing demand for a deeper understanding of the fundamental science behind ignition efficiency, especially the physical mechanisms governing the interaction between the laser and the plasma emitted when the laser bombards the fuel pellet. However, the Laser-Plasma Interaction (LPI) is a complex multi-body problem, which traditionally relies on time-stepping method, particularly, Particle-in-Cell (PIC) simulations (Tskhakaya et al., 2007; Langdon, 2014; Sulsky et al., 1995; Arber et al., 2015; Liewer & Decyk, 1989). Despite being the preeminent standard for modeling the physics of LPI, PIC simulations are exceedingly intensive in computation, often requiring tens of millions of CPU hours, consuming millions of dollars in order to obtain meaningful outputs (Germaschewski et al., 2016; Derouillat et al., 2018; Bastrakov et al., 2012). The computational overhead of PIC simulations has become a daunting bottleneck in fusion research, raising the pressing need for innovative methodologies capable of generating high-quality scientific data with substantially reduced computational burden.

Over the years, numerous CPU-GPU implementations have been developed for PIC simulations (Fonseca et al., 2002; Bowers et al., 2008; Sgattoni et al., 2015). While invaluable, these efforts remain within the scope of the time-stepping approach that iteratively executes over infinitesimal time intervals, falling short in addressing the inherent computational overhead of long-term simulations. Recent

advancements in generative AI – diffusion models, however, present a novel approach to bypass the constraint. Diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song & Ermon, 2019) have demonstrated exceptional capabilities in Computer Vision (CV) (Song et al., 2023; Ramesh et al., 2022; Ho et al., 2022), synthesizing highly complex data distributions that match real data with high fidelity. From the perspective of energy-based models (LeCun et al., 2006; Grathwohl et al., 2019), a diffusion model effectively constructs a highly complex energy field that governs the evolution of variables, analogous to the motion of particles in cells. In fact, diffusion models are rooted in the diffusion concept in physics, where particles move according to the energy field. This has sparked significant interest in their potential for generating scientific data, as recent applications of diffusion models in molecular dynamics simulations have demonstrated their promise in this domain (Wu & Li, 2023; Petersen et al., 2023).

Although diffusion models exhibit outstanding compatibility for generating PIC simulation data, two critical research gaps must be addressed. ❶ *Physical soundness* must be ensured. In contrast to traditional PIC simulations that directly take continuous physical parameters as constraints, it remains unclear how diffusion models can effectively capture and distill complex physical patterns. ❷ *Substantial efficiency improvement* must be achieved. The step-by-step denoising process in diffusion models is computationally demanding. Although the requirement for infinitesimal time intervals has been relaxed, the process still presents significant challenges similar to those faced by time-stepping methods, limiting the practicality of diffusion models as advanced alternatives for PIC simulations.

In light of these challenges, we propose a distillation framework for physical patterns, titled **Diff-PIC**, that leverages diffusion models to efficiently generate a snapshot of *arbitrary* time, under *arbitrary* simulation parameters within certain ranges. To specifically address the two challenges above: ❶ We develop a conditional diffusion model with a Physically-Informed Parameter Encoder. This encoder allows the model to capture the relationship between continuous physical parameters and PIC simulation data, distilling physical phenomena into Diff-PIC. ❷ We employ the rectified flow technique to eliminate the requirement for the multi-step denoising, further optimizing the runtime efficiency of Diff-PIC. As highlighted in Fig. 1, orders-of-magnitude speedup is achieved compared to PIC simulations and other generative approaches, including Generative Adversarial Networks (GAN) (Karras et al., 2020) and Normalizing Flow (NF) (Zhang & Chen, 2021), dubbed "GAN-PIC" and "NF-PIC".

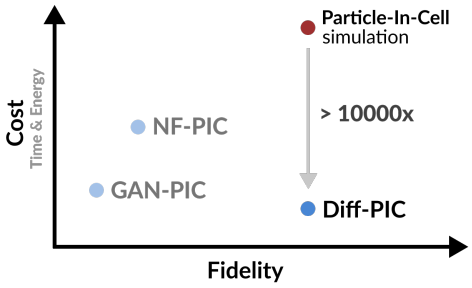

Figure 1: Overview of the proposed method.

In summary, our work represents the first known effort to tackle the imperative challenges associated with generating high-quality PIC simulation data for LPI using diffusion models renowned in the field of generative AI. The core contributions of our work include:

- We propose Diff-PIC, a pioneering study that utilizes diffusion models as a computationally efficient alternative to PIC simulations (see §3.1). By making all resources publicly available, we aim to establish Diff-PIC as a robust baseline and a valuable benchmark for the research community, thereby accelerating the advancement in scientific data generation for nuclear fusion research.

- We develop a physically-informed conditional diffusion model (see §3.2) that seamlessly integrates physical simulation parameters into the diffusion model. The designed condition encoder facilitates the generalization of effective simulation data within and beyond existing simulation parameters, endowing the model with robust generalization capabilities and adaptable transferability.

- We implement the rectified flow technique to transform our model into a one-step diffusion model (see §3.3), thereby enhancing its efficiency in generating high-fidelity fusion data.

- Experimental results in §4.2 demonstrate that our method achieves a remarkable speedup of 16,200 times compared to traditional PIC simulations while preserving high fidelity and physical validity of the generated data – 0.014, 0.019, 1.03 in terms of MAE, RMSE, and FID, respectively, with an average reduction of 59.21% / 57.15% / 39.46% compared to two other SOTA generative models.

2 BACKGROUND

Inertial Confinement Fusion (ICF) is a method of achieving controlled nuclear fusion by using intense energy pulses to compress and heat small fuel pellets (Keefe, 1982; Betti & Hurricane, 2016), typically containing isotopes of hydrogen such as deuterium and tritium. This process unfolds the nuclear fusion reaction as delineated below:

$$^{2}\mathrm{H} + {}^{3}\mathrm{H} \rightarrow {}^{4}\mathrm{He} + {}^{1}\mathrm{neutron} + \textbf{Energy}. \tag{1}$$

Given the ubiquity of these hydrogen isotopes in the ocean, nuclear fusion holds immense potential to provide "near-infinite" energy by achieving the necessary temperature and pressure conditions to initiate fusion reactions, attaining a positive net energy gain (*i.e.*, output energy surpassing the input). To optimize the sophisticated initiation of fusion, an advanced understanding of the underlying LPI mechanism is essential. For this purpose, PIC is considered a crucial tool to provide theoretical insights into LPI, due to its capability of predicting and interpreting physical phenomena.

Particle-in-Cell Simulations. The PIC method is a computational technique widely used in the study of plasma physics and fusion energy research (Tskhakaya et al., 2007; Jiang et al., 2015; Derouillat et al., 2018). Developed in the mid-$20th$ century, the PIC method has become a cornerstone in the simulation of complex plasma behaviors, enabling researchers to delve into the intricate dynamics of particles and electromagnetic fields (Lange, 1978; Lewis et al., 1972). To highlight, PIC is especially useful in LPI studies (Arber et al., 2015; Strozzi et al., 2012; Klimo et al., 2010), which involves complex dynamics of electrons and ions. PIC simulations track the trajectories and interactions of these charged particles under the influence of electromagnetic fields, providing insights into both shock wave formation and heating mechanisms that are essential for ICF.

In essence, PIC is an iterative time-stepping method applied to atomic particles such as electrons and ions. Within each iteration, particles are systematically arranged into discrete cells according to their spatial distribution, with their positions and velocities being updated over infinitesimally small time steps, typically on the scale of femtoseconds (10^{-15} seconds). Unlike molecular dynamics widely applied in biology (Geng et al., 2019; Das et al., 2018), PIC simulations in LPI are characterized by intensive electromagnetic fields, which exert a significant influence on particle trajectories as follows:

$$\frac{d\mathbf{v}_i}{dt} = \frac{q_i}{m_i}\left(\mathbf{E}(\mathbf{r}_i, t) + \mathbf{v}_i \times \mathbf{B}(\mathbf{r}_i, t)\right). \tag{2}$$

For a particle at position $\mathbf{r}_i$ with charge q_i and mass m_i, the equation describes how the velocity $\mathbf{v}_i$ evolves according to an energy field that consists of electric field $\mathbf{E}$ and magnetic field $\mathbf{B}$. To ensure accuracy, simulating LPI at the scale of mere hundreds of picoseconds (*i.e.*, 10^{-10} seconds) requires hundreds of thousands of sophisticated PIC iterations. This imposes significant demands on computational storage and processing capabilities. As a result, the PIC methodology has emerged as a stringent bottleneck in fusion research, significantly constraining progress in this domain.

Diffusion models have emerged as a prominent class of generative models within the realm of artificial intelligence, offering an innovative methodology for the synthesis of high-fidelity data (Ho et al., 2020; 2022; Ramesh et al., 2022). Named after the physical concept, diffusion models use the idea of diffusion, which in physics refers to the random movement of particles from regions of high concentration to regions of lower concentration, often driven by thermal energy.

In machine learning, diffusion models are generative models that progressively add noise to data in a forward process, then gradually remove it in the reverse process to generate new data. From the perspective of energy-based models, the reverse process can be seen as moving variables through an energy landscape, where the model transitions variables from high-energy, noisy states (where data are unstructured) to low-energy, clean states (where data are structured). In this sense, diffusion models generate data by allowing variables to evolve in a manner akin to particles diffusing within an energy field, granting the opportunity to effectively apply physical methods to evaluate the diffusion, specifically, Langevin dynamics:

$$d\mathbf{x} = -\nabla_{\mathbf{x}} E(\mathbf{x})dt + \sqrt{2\beta}d\mathbf{W_t} \tag{3}$$

The evolution of variables $\mathbf{x}$ is influenced by both the energy field $E(\mathbf{x})$ and the stochastic motion term $\mathbf{W_t}$ regulated by the diffusion coefficient β. Through the dynamics, variables move towards low

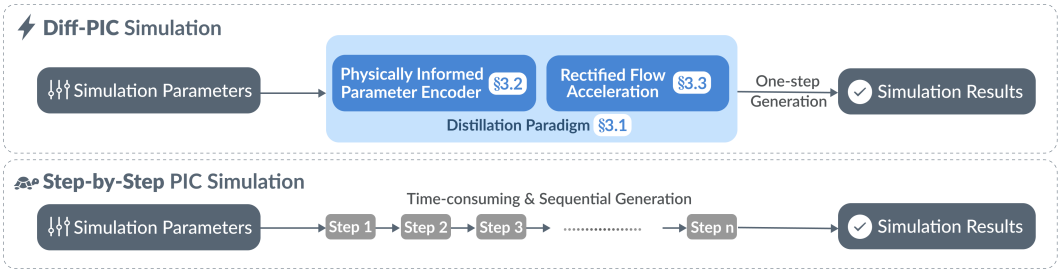

Figure 2: Workflow of the proposed Diff-PIC compared to traditional step-by-step PIC simulation.

energy regions, conceptually mirroring the behavior of particles in PIC simulations. This similarity encourages an exploration of diffusion models as potential alternatives to traditional PIC simulations.

3 DIFF-PIC

In this section, we introduce Diff-PIC, a physically-informed conditional diffusion model tailored for generating high-fidelity synthetic data for LPI in nuclear fusion. As illustrated in Fig. 2, accepting physical parameters as inputs, the parameters are encoded and integrated with the model main body through the physically informed parameter encoder. Additionally, the rectified flow mechanism is adopted to break free from the time-stepping paradigm in both PIC simulation and traditional diffusion models, unlocking the model's capability to generate scientific data in one single step.

3.1 THE OVERALL DISTILLATION FRAMEWORK

Since LPI is governed by the behavior of electromagnetic field that dominates plasma dynamics, the tasks specifically assigned to Diff-PIC are to generate high-fidelity electric fields as representative cases under various physical parameters. Due to the polarization of the input laser, the resulting electric field only oscillates in a 2D plane, making 2D field descriptions sufficient to capture primary features of LPI. This approach is also a standard practice in PIC simulations.

In this work, we focus on the following critical physical parameters in LPI: "Electron Temperature (T_e) in keV," "Ion Temperature (T_i) in keV," and "Laser Intensity (I) in W/m^2." Additionally, to enable direct and versatile generation of the electric field at a specific simulation time (a snapshot), we include a user-defined parameter t_{as} that represents the simulation time for an arbitrary snapshot. Through a standard learning process in diffusion models (Ho et al., 2020; Song et al., 2021), the relationships between the four parameters and the resulting electric field are captured, as the model learns to progressively transform Gaussian noise into realistic electric fields. Once trained, the model takes the four parameters as inputs and produce electric field snapshots $\mathbf{E}(t_{as}, \theta)$ corresponding to the specified parameters $\theta = \{T_e, T_i, I\}$. These generated snapshots can be used for multiple purposes, including data augmentation, parameter exploration, and studying the effects of different physical parameters. Furthermore, the diffusion model aids in the interpretation and visualization of complex plasma phenomena, providing valuable insights for researchers in the field of nuclear fusion. Essentially, the proposed techniques offer two advantages:

- *Data Dependency Relaxation.* We treat snapshots from various parameter combinations as distinct distributions. This approach decouples data dependency in model training, enabling the model to efficiently learn from individual snapshots while generalizing across a wide array of scenarios.
- *Systemic efficiency.* Unlike traditional PIC simulations that generate data sequentially over time, the proposed diffusion model can directly produce data for any target snapshot (see Fig. 4). This non-sequential behavior allows for substantially more efficient data generation and analysis (see Table 3), enabling researchers to focus on specific times of interest without needing to simulate the entire LPI process from the beginning.

3.2 PHYSICALLY-INFORMED PARAMETER ENCODER

PIC simulations employ continuous physical parameters as inputs, which necessitate a seamless and continuous transition in the resulting synthesized data as input parameters are adjusted. Consequently, an encoder is considered exceptionally beneficial in this scenario, responsible for transforming

domain-specific inputs into embeddings comprehensible by the model, meanwhile preserving the physical continuity of output snapshots. In particular, these inputs comprise the simulation parameters θ and the target simulation snapshot t_{as} that the conditional diffusion model aims to generate.

Given the extensive range of physically feasible parameters and the limited data available during training, an optimal encoder must excel in both *interpolation* and *extrapolation* — critical measures of the model's ability of generalization. Interpolation capability refers to the encoder's proficiency in generating suitable embeddings for new parameters that, although not encountered during training, lie between observed parameters. Extrapolation capability, conversely, pertains to generating embeddings for parameters that fall outside the range of those observed during training. Notably, both capabilities are indispensable for addressing the LPI problem, in order to cover a large enough, and fine-grained enough parameter space to provide sufficient insight into further LPI evaluations.

To meet the two essentials, we introduce a Physically-Informed Parameter Encoder (PIPE) as shown in Fig. 3. To encode the simulation parameters θ, we incorporate two distinct types of encoders. For interpolation, we employ Positional Encoding (Vaswani et al., 2017) (denoted "$\sim$" in a circle) to learn a continuous representation of input parameters, enabling smooth transitions between observed parameters. To augment extrapolation capability, we enhance the encoding process with a polynomial encoder ("P" in a circle), utilizing transformation functions constructed as a linear combination of polynomial basis functions $f_i(\theta)$ of varying degrees:

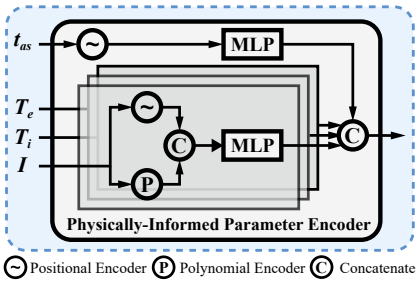

Figure 3: The proposed PIPE.

$$\mathcal{P}(\theta) = \sum_{i=0}^{n} f_i(\theta), \tag{4}$$

where n denotes the maximum order of polynomial terms, and the polynomial $\mathcal{P}(\theta)$ can be chosen as Chebyshev polynomials and Legendre polynomials, based on the characteristics of the parameter space. This polynomial enhancement allows the encoder to generate plausible embeddings for parameters well beyond those encountered during training, ensuring robust performance across a broader spectrum of simulation scenarios. Subsequently, we concatenate ("C" in a circle) the embeddings from these two encoders and apply a Multi-Layer Perceptron (MLP) to further refine the embeddings. The MLP, with its trainable parameters, learns to combine and transform the concatenated embeddings, resulting in a more informative representation of the input parameters. For encoding the simulation time step t_{as}, we utilize Positional Encoding (Vaswani et al., 2017) followed by an MLP layer. This approach is specifically chosen to learn continuous representations that facilitate smooth transitions between consecutive snapshots, thereby enhancing the model's temporal coherence. In summary, this design offers the following advantages:

- *Algorithmic generalization.* PIPE improves the generalizability of the conditional diffusion model (see Table 2). The dual-encoding strategy captures non-linear relationships by incorporating both positional and polynomial encodings, empowering the model to adeptly manage a diverse array of simulation parameters, ranging from scenarios encountered during the training phase to parameters that lie beyond the spectrum of the training data.
- *Adaptive transferability.* By fine-tuning the pre-trained model with a new dataset, this methodology facilitates adaptation to other fields where precise and efficient simulations are imperative for deciphering intricate physical phenomena.

3.3 Rectified Flow-Based Acceleration

To enable rapid generation of high-fidelity synthetic data, we employ the Rectified Flow Acceleration (RFA) technique in model optimization. Based on the principles of rectified flow (Liu et al., 2022; Esser et al., 2024; Liu et al., 2024), RFA converts the original complex denoising trajectory from initial noise ϵ to the target electric field snapshot $\mathbf{E}(t_{as})$ into a direct and straight path – the shortest route between two distributions. During training, RFA minimizes the following objective function with straight ordinary differential equations:

$$\arg\min_{\zeta} \quad \mathbb{E}\left[\int_0^1 \left\|(\mathbf{E}(t_{as},\theta) - \epsilon) - \zeta(\mathbf{E}_t, t \mid t_{as}, \theta)\right\|^2 dt\right], \tag{5}$$

where $\mathbf{E}_t = t\mathbf{E}(t_{as}, \theta) + (1-t)\epsilon$ denotes the linear interpolation between $\mathbf{E}(t_{as}, \theta)$ and ϵ across the diffusion timeline, with t ranging from 0 to 1. The score-based model ζ, approximated using the U-Net, defines the learned trajectory. By minimizing the expectation of the squared deviations between the straight path $\mathbf{E}(t_{as}, \theta) - \epsilon$ and the learned trajectory $\zeta(\mathbf{E}_t, t \mid t_{as}, \theta)$, RFA promotes the adoption of the shortest and most direct path in the denoising process, thus significantly reducing the denoising time. Once this time-dependent score-based model ζ is trained, we further straighten the learned trajectories through an interactive reflow procedure (Liu et al., 2024).

In summary, the RFA module provides additional benefits for our paradigm:

- *Streamlined Denoising Process.* RFA significantly accelerates the denoising process (see Table 3) by converting the complex data trajectory from initial noise to the target snapshot into a direct denoising step. By distilling the typically winding diffusion path into the shortest route between two distributions, RFA greatly reduces the time required for generating high-fidelity synthetic data.
- *Robust Optimization.* RFA leverages the principles of rectified flow to minimize deviations between the winding data trajectory and the optimal, shortest path. This direct path approach reduces the possibility of error accumulation that can occur with more winding, iterative methods.

4 Evaluation

4.1 Experimental Setup

Datasets. We provide a new dataset comprising 6,615 simulations across varied physical simulation parameters, each containing 80 snapshots of electric fields along two orthogonal directions denoted E1 and E2. The data were generated by OSIRIS (Fonseca et al., 2002), a well-established PIC simulation software suite. The dataset covers diverse parameters, including T_e, T_i, and I, all of which are critical parameters influencing the resultant electric fields. To foster further advancements in fusion and scientific data generation research, we will release the dataset publicly.

Metrics. To validate physical soundness, Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) are used to evaluate the electric field difference and the energy difference between the Diff-PIC generated and the ground truth produced by PIC simulations. To better demonstrate the relative error, the dataset is normalized to [0,1]. To further evaluate the difference in the generated and the ground truth data distributions, the Fréchet Inception Distance (FID) metric is also employed, reflecting the fidelity of the electric fields produced by DiffPIC.

Baselines. We compare Diff-PIC with two other SOTA generative models, Generative Adversarial Networks (Karras et al., 2020) and Normalizing Flow (Zhang & Chen, 2021). The baseline models are implemented based on the setups provided in the original papers. Since neither of them originally supports learning meaningful embeddings for the physical parameters, for fair comparison, we equip them with the proposed PIPE to establish two baselines: GAN-PIC and NF-PIC. More implementation details are provided in Appendix A.2.

Diff-PIC configurations. The architectural foundation of our model is predicated on the U-Net framework (Ronneberger et al., 2015), incorporating a series of three down-sampling blocks followed by three up-sampling blocks. The training regimen of our model encompasses an extensive 600 epochs, employing a batch size of 64, a configuration empirically validated to secure model convergence. Furthermore, the training protocol adheres to a fixed learning rate of 5×10^{-4}, optimized via the Adam optimizer (Kingma & Ba, 2015). The optimization objective is delineated in Eq. 5. In addition, in terms of the encoders, the positional encoding mechanism yields a 16-dimensional embedding from a single input, while the polynomial encoder operates with a highest polynomial order of 4. More implementation details are provided in Appendix A.1.

4.2 Main Results

In this section, we evaluate Diff-PIC on three key aspects for comprehensive comparisons. (1) The interpolation ability and extrapolation ability. (2) The physical validity of the Diff-PIC generated data. (3) The speedup and power efficiency compared to traditional PIC simulations.

Interpolation and Extrapolation. To evaluate the interpolation capability of Diff-PIC, we sample a specified range for each simulation parameters (T_e, T_i, and I), totally 500 simulations and $500 \times 80 = 40,000$ snapshots. Then, we randomly split these 500 simulations into training and testing set with the

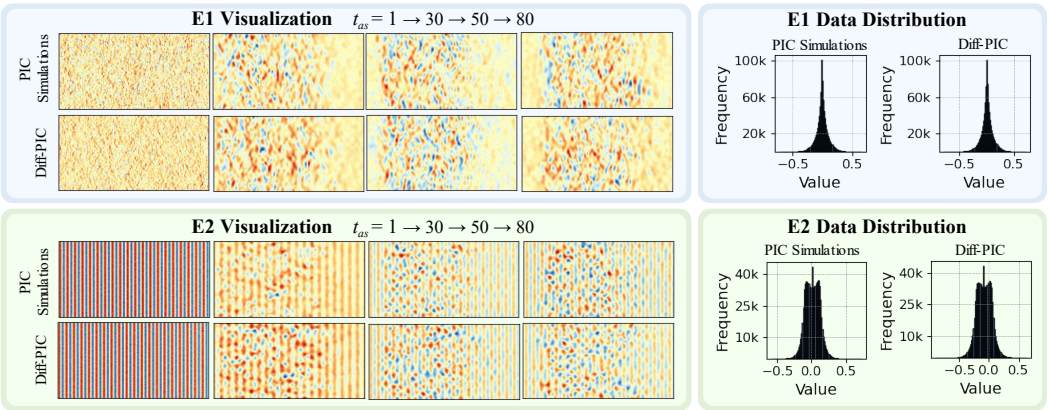

Figure 4: **Visualization and Comparison** of PIC simulations and Diff-PIC.

Table 1: **Quantitative results** for interpolation evaluation.

Method	Training Set for E1			Testing Set for E1			Training Set for E2			Testing Set for E2		
	MAE↓	RMSE↓	FID↓	MAE↓	RMSE↓	FID↓	MAE↓	RMSE↓	FID↓	MAE↓	RMSE↓	FID↓
GAN-PIC	4.59e-2	5.31e-2	2.32	4.73e-2	5.84e-2	2.51	1.82e-2	2.07e-2	0.973	1.97e-2	2.18e-2	1.03
NF-PIC	4.46e-2	5.12e-2	2.06	4.61e-2	5.35e-2	2.42	1.70e-2	2.03e-2	0.914	1.86e-2	2.45e-2	0.986
Diff-PIC	1.56e-2	2.67e-2	1.21	1.68e-2	2.29e-2	1.62	7.95e-3	9.32e-3	0.328	8.26e-3	1.03e-2	0.341

ratio of 80% and 20%. We train Diff-PIC and baselines on the training set, and report the performance of Diff-PIC on the training and testing set in Table 1. The reasonably low MAE, RMSE, and FID scores indicate that the proposed Diff-PIC is able to synthesize high-quality scientific data similar to what PIC generates, meanwhile significantly outperforming baselines on all three metrics. On average, Diff-PIC achieves 59.25% reduced MAE, 57.77% reduced RMSE, and 49.21% reduced FID for both testing sets, compared to the baselines.

In addition, Fig. 4 compares the results for one randomly selected simulation produced by Diff-PIC and PIC respectively in the testing set. Throughout the snapshots ($t_{as} = 1 \rightarrow 30 \rightarrow 50 \rightarrow 80$ for example), the distributions of the synthetic data closely follow the ground truths, indicating that physical continuity is maintained over time. Additionaly, the distributions on the right demonstrate that our proposed model successfully captures the data distributions in the ground truths.

In terms of extrapolation capability evaluation, we progressively extend the range of physical parameters from the range selected for training. In particular, the ranges are extended by 10% and 20% respectively as case studies. The results on Table 2 demonstrate that Diff-PIC achieves approximately only 2% relative absolute error in extrapolation tasks, significantly outperforming the other generative counterparts. On average, Diff-PIC achieves 59.16% reduced MAE, 56.53% reduced RMSE, and 29.70% reduced FID for all test cases, compared to the baselines.

Physical Validity. In addition to the high quality of the synthetic data, we test the physical validity of the generated electric fields at each snapshot with energy, a fundamental concept used to describe the general property of a physical system. In particular, we randomly pick a simulation in testing set, the energy results of its electric fields are illustrated in Fig. 5. We observe that the pattern of the ground truth is captured by Diff-PIC, with fairly low MAE and RMSE incurred. Specifically, compared to the PIC simulations, Diff-PIC achieves 0.035 / 0.046 MAE / RMSE on E1 training set, 0.041 / 0.055 MAE / RMSE on E1 testing set, 0.020 / 0.025 MAE / RMSE on E2 training set, and 0.036 / 0.045 MAE / RMSE on E2 testing set. Especially for electric field E2, the oscillation in energy is properly preserved in the synthetic data produced by Diff-PIC. Furthermore, this evaluation highlights that Diff-PIC is capable of generating sequentially continuous data, demonstrating the effectiveness of the proposed distillation paradigm and the Physically-Informed Parameter Encoder.

Speedup and Power Efficiency. In addition to the traditional PIC approach, baselines are also included in the comparison shown in Table 3. The PIC simulations are run on the Perlmutter supercomputer in the National Energy Research Scientific Computing (NERSC) facility, with AMD EPYC 7763 CPUs. As essential fusion phenomena typically appear at approximately 100 ps, the

Table 2: **Quantitative results** for extrapolation evaluation.

Method	E1 10%			E1 20%			E2 10%			E2 20%		
	MAE↓	RMSE↓	FID↓	MAE↓	RMSE↓	FID↓	MAE↓	RMSE↓	FID↓	MAE↓	RMSE↓	FID↓
GAN-PIC	5.24e-2	6.32e-2	1.97	5.73e-2	6.98e-2	2.15	2.18e-2	3.25e-2	1.15	2.93e-2	3.89e-2	1.29
NF-PIC	4.74e-2	5.31e-2	1.85	5.41e-2	6.46e-2	2.08	1.90e-2	3.16e-2	1.04	2.63e-2	3.42e-2	1.17
Diff-PIC	1.83e-2	2.40e-2	1.74	2.18e-2	2.62e-2	1.82	9.47e-3	1.36e-2	0.536	1.13e-2	1.85e-2	0.673

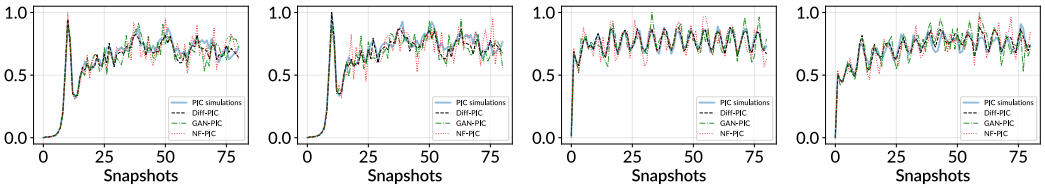

Figure 5: **Energy evaluation** of electric fields for training and test sets with 80 snapshots. From left to right: E1 Training Set, E1 Testing Set, E2 Training Set, and E2 Testing Set.

costs of PIC simulation at 100 ps are selected as the reference across the comparisons. The GPU results for Diff-PIC and baselines are obtained on an Nvidia RTX 4090 GPU to demonstrate the availability of this approach to general users. The CPU results for these approaches are acquired on an Intel 13th Gen i9-13900KF CPU. The results highlight that Diff-PIC-GPU achieves over $10^4\times$ speedup versus traditional PIC simulation, as well as $10^4\times$ in terms of reduced energy cost.

Ablation studies on the proposed PIPE. To further demonstrate the effectiveness of PIPE, we conduct ablation studies by replacing the PIPE component in Diff-PIC with other commonly used encoders: MLP and Transformer decoder. Detailed implementations and comparison results are presented in Appendix A.3. The results show that PIPE outperforms both MLP and Transformer decoder in both interpolation and extrapolation tasks. MLP struggles with both interpolation and extrapolation. The Transformer decoder, while powerful for sequence modeling and capturing relationships in discrete token sequences, is not inherently designed to effectively process continuous physical parameters, showing inferior performance.

In addition, to provide more insights into the effectiveness of PIPE, we conduct another ablation study to evaluate the contributions of the positional encoder and polynomial encoder separately. Detailed implementations and comparison results are shown in Appendix A.3, highlighting the individual importance of these encoders. Specifically, combining both positional and polynomial encoders consistently outperforms using either encoder alone in both interpolation and extrapolation tasks. In interpolation tasks, the positional encoder demonstrates better performance than the polynomial encoder, suggesting its important role in interpolation scenarios. Conversely, in extrapolation tasks, particularly at larger ranges (20%), the polynomial encoder shows relatively better performance, indicating its importance for extrapolation capabilities. These results validate our design choice of combining both encoders in PIPE to leverage their complementary strengths.

4.3 Discussion

To provide further insights and highlight the value of this work, it is worth noting that our approach exhibits outstanding scalability compared to traditional PIC simulations. In PIC, for N particles, the computing complexity can generally reach a formidable $TN\log N$ with T time steps. In contrast, Diff-PIC is not sensitive to the number of particles in space nor the number of time steps, since it focuses on generating the macroscopic data (*e.g.*, electric field, which is usually considered more informative than individual particle status) directly for a specific time. For larger particle spaces,

Table 3: **Speedup** and **Energy Consumption Reduction**.

Method	PIC-100 ps	**Diff-PIC**-GPU	**Diff-PIC**-CPU	GAN-PIC-GPU	GAN-PIC-CPU	NF-PIC-GPU	NF-PIC-CPU
Speedup	1.00×	1.62e4×	519×	1.56e4×	523×	9.21e2×	24×
Energy Reduction	1.00×	1.01e4×	1.05e3×	1.13e4×	1.47e3×	8.14e2×	53×

the speedup achieved by Diff-PIC can be easily improved by extra orders of magnitude, further accelerating the research of fusion, or other research areas involving large-scale PIC simulations.

5 Related Work

Particle-in-Cell Simulations have long been fundamental to modeling physical processes in fusion research (Taccogna & Minelli, 2017; Garrigues et al., 2016). However, the computational intensity of PIC simulations presents significant challenges (Verboncoeur, 2005). To mitigate these computational constraints, various methods have explored GPU and hybrid CPU-GPU acceleration technologies. Studies such as (Abreu et al., 2010; Burau et al., 2010; Decyk & Singh, 2011; Kong et al., 2011; Suzuki et al., 2011) have utilized parallel computing, high memory bandwidth, and multiple processors to expedite simulations. Architecturally, the simulator optimized for the Kepler GPU architecture, as discussed in (Shah et al., 2017), underscores the potential of specific GPU architectures to enhance simulation efficiency. For more intricate simulations, research efforts like (Xu et al., 2012; Chen et al., 2012) have developed hybrid CPU-GPU implementations, Wang & Zhu (2021) introduced a hybrid approach for multi-core and multi-GPU systems, highlighting the continuous integration and evolution of these technologies in advancing PIC simulations. Despite these advancements, these approaches remain reliant on the fundamental PIC framework, which may not completely address the computational burden due to the inherent algorithmic complexity of the PIC method. In recent years, rapid advancements in deep learning have opened new pathways for accelerating scientific simulations. Machine learning-based approaches have emerged, such as predicting a vector space that approximates the PIC system solution (Kube et al., 2021), and learning the probability of interactions between potential collision pairs (Bilbao et al., 2022). However, the approach by (Kube et al., 2021) depends on a pre-computed vector space and may not generalize well to novel scenarios, while (Bilbao et al., 2022)'s method focuses on binary interactions, overlooking the complex many-body interactions in PIC simulations.

Contrastingly, our proposed method overcomes these limitations by employing a conditional diffusion model to distill the complex patterns captured by PIC simulations from a limited training dataset. Utilizing a time-dependent score-based model, our approach can efficiently generate high-fidelity synthetic data (see Fig. 4 and Table 1) without the computational expense of traditional PIC algorithms (see Table 3). This results in a significant reduction in computational cost while maintaining high simulation accuracy. Moreover, our method is highly adaptable, as it can be fine-tuned with minimal additional data to suit various physical parameters. This flexibility renders our approach suitable for a wide range of applications in fusion research and beyond, where efficient and accurate simulations are crucial for understanding complex physical phenomena.

Diffusion Models in Scientific Research have emerged as a formidable tool across a myriad of scientific domains. These models, which employ a stochastic process to incrementally convert a pristine data sample into a noise-distributed version and subsequently reverse this process. For instance, in materials science and chemistry, (Wu & Li, 2023) introduced a diffusion model for molecular dynamics simulations, demonstrating the generalizability in generating molecular trajectories. (Arts et al., 2023) presented an approach that integrates diffusion models with coarse-grained molecular dynamics to develop a new force field for simulating protein dynamics. By leveraging score-based generative models, they trained a model on coarse-grained structures to produce a force field that enhances the performance and realism of protein simulations without requiring force inputs during training. (Duan et al., 2023) introduced an object-aware SE(3)-equivariant diffusion model for rapidly generating accurate 3D transition state structures, significantly reducing the computational burden typically associated with quantum chemistry calculations. In astrophysics, diffusion models have been utilized to generate synthetic observations and simulate complex astrophysical phenomena. (Smith et al., 2022) proposed a diffusion model for generating realistic galaxy images, aiding in the analysis of large-scale sky surveys. Diffusion models have also found applications in climate science and Earth system modeling. For instance, (Oyama et al., 2023) employed a deep generative model to super-resolve spatially correlated multiregional climate data, enhancing the spatial resolution of global climate simulations, which is crucial for long-term climate projections and infrastructure development planning. (Li et al., 2024) explored the generative emulation of weather forecast ensembles with diffusion models, illustrating their effectiveness as scalable and cost-efficient alternatives to traditional ensemble forecasts, thus improving the reliability and accuracy of predictions for extreme weather events. In particle physics, (Imani et al., 2024) introduced a diffusion model for generating

high-quality Liquid Argon Time Projection Chamber (LArTPC) images, showcasing the model's ability to handle the challenges of sparse but locally dense particles.

While these studies highlight the expanding interest and application of diffusion models across various scientific domains, their potential within fusion research, specifically as an alternative to PIC simulations, remains underexplored. Our work aims to bridge this gap by proposing Diff-PIC, a conditional diffusion model that integrates the capability of PIC for generating high-fidelity synthetic data in fusion research. Diff-PIC leverages the inherent advantages of diffusion models to provide a computationally efficient alternative to traditional PIC simulations (see Table 3).

6 Conclusions

This paper presents Diff-PIC, a pioneering approach that leverages the capabilities of diffusion models to generate high-fidelity synthetic data for LPI, offering a computationally efficient alternative to conventional PIC simulations in nuclear fusion research. By integrating a Physically-Informed Parameter Encoder and applying the Rectified Flow Acceleration, Diff-PIC significantly augments the diffusion model's capacity to manage diverse experimental parameters, thereby expediting the generation of high-fidelity synthetic data. These advancements facilitate rapid, resource-efficient exploration of the design space, markedly diminishing the computational demands associated with PIC simulations. Our research not only catalyzes accelerated scientific discoveries within the realm of fusion research but also sets a novel precedent for the application of generative AI models in scientific simulations. Future investigations may focus on optimizing the distillation paradigm, harmonizing the simulation time t_{as} with the diffusion time t, and refining the condition encoder to encompass a broader spectrum of physical parameters.

Ethics

In our study, which introduces a novel dataset, we will implement comprehensive ethical safeguards to mitigate potential misuse and ensure responsible utilization, as detailed in the protocols included in the final release of models and datasets. These protocols encompass strict usage guidelines, access restrictions, the incorporation of safety filters, and monitoring mechanisms. We perform thorough risk assessments to identify potential misuse scenarios and develop tailored mitigation strategies, such as robust data governance frameworks. While not all research may necessitate stringent safeguards, we adhere to best practices, promoting ethical awareness and encouraging researchers to consider the broader impacts of their work. Additionally, we maintain detailed documentation for transparency and accountability for the data we release. These efforts underscore our commitment to upholding the highest standards of conduct in scientific inquiry, aiming to protect the interests of involved parties.

Reproducibility

Diff-PIC is implemented using PyTorch (Paszke et al., 2019). PIC simulations are executed on the Perlmutter supercomputer, located at the National Energy Research Scientific Computing (NERSC) facility, utilizing AMD EPYC 7763 CPUs. To ensure reproducibility, the complete implementation will be made publicly available.

References

Paulo Abreu, Ricardo A Fonseca, João M Pereira, and Luís O Silva. Pic codes in new processors: A full relativistic pic code in cuda-enabled hardware with direct visualization. *IEEE Transactions on Plasma Science*, 39(2):675–685, 2010.

H Abu-Shawareb, R Acree, P Adams, J Adams, B Addis, R Aden, P Adrian, BB Afeyan, M Aggleton, L Aghaian, et al. Achievement of target gain larger than unity in an inertial fusion experiment. *Physical Review Letters*, 132(6):065102, 2024.

TD Arber, Keith Bennett, CS Brady, A Lawrence-Douglas, MG Ramsay, Nathan John Sircombe, Paddy Gillies, RG Evans, Holger Schmitz, AR Bell, et al. Contemporary particle-in-cell approach to laser-plasma modelling. *Plasma Physics and Controlled Fusion*, 57(11):113001, 2015.

Marloes Arts, Victor Garcia Satorras, Chin-Wei Huang, Daniel Zuegner, Marco Federici, Cecilia Clementi, Frank Noé, Robert Pinsler, and Rianne van den Berg. Two for one: Diffusion models and force fields for coarse-grained molecular dynamics, 2023.

Sergey Bastrakov, R Donchenko, Arkady Gonoskov, Evgeny Efimenko, A Malyshev, Iosif Meyerov, and Igor Surmin. Particle-in-cell plasma simulation on heterogeneous cluster systems. *Journal of Computational Science*, 3(6):474–479, 2012.

R Betti and OA Hurricane. Inertial-confinement fusion with lasers. *Nature Physics*, 12(5):435–448, 2016.

PJ Bilbao, C Badiali, and F Cruz. Towards deep learning accelerated particle-in-cell simulations: application to compton scattering. *OSIRIS*, 10:3, 2022.

Kevin J Bowers, Brian James Albright, Lilan Yin, B Bergen, and Thomas JT Kwan. Ultrahigh performance three-dimensional electromagnetic relativistic kinetic plasma simulation. *Physics of Plasmas*, 15(5), 2008.

Heiko Burau, Renée Widera, Wolfgang Hönig, Guido Juckeland, Alexander Debus, Thomas Kluge, Ulrich Schramm, Tomas E Cowan, Roland Sauerbrey, and Michael Bussmann. Picongpu: a fully relativistic particle-in-cell code for a gpu cluster. *IEEE Transactions on Plasma Science*, 38(10): 2831–2839, 2010.

Guangye Chen, Luis Chacón, and Daniel C Barnes. An efficient mixed-precision, hybrid cpu–gpu implementation of a nonlinearly implicit one-dimensional particle-in-cell algorithm. *Journal of Computational Physics*, 231(16):5374–5388, 2012.

Sourav Das, Nikita Bora, Mostofa Ataur Rohman, Raju Sharma, Anupam Nath Jha, and Atanu Singha Roy. Molecular recognition of bio-active flavonoids quercetin and rutin by bovine hemoglobin: an overview of the binding mechanism, thermodynamics and structural aspects through multi-spectroscopic and molecular dynamics simulation studies. *Physical Chemistry Chemical Physics*, 20(33):21668–21684, 2018.

Viktor K Decyk and Tajendra V Singh. Adaptable particle-in-cell algorithms for graphical processing units. *Computer Physics Communications*, 182(3):641–648, 2011.

Julien Derouillat, Arnaud Beck, Frédéric Pérez, Tommaso Vinci, M Chiaramello, Anna Grassi, M Flé, Guillaume Bouchard, I Plotnikov, Nicolas Aunai, et al. Smilei: A collaborative, open-source, multi-purpose particle-in-cell code for plasma simulation. *Computer Physics Communications*, 222:351–373, 2018.

Chenru Duan, Yuanqi Du, Haojun Jia, and Heather J. Kulik. Accurate transition state generation with an object-aware equivariant elementary reaction diffusion model, 2023.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. *arXiv preprint arXiv:2403.03206*, 2024.

Ricardo A Fonseca, Luis O Silva, Frank S Tsung, Viktor K Decyk, Wei Lu, Chuang Ren, Warren B Mori, Shaogui Deng, Shiyoun Lee, T Katsouleas, et al. Osiris: A three-dimensional, fully relativistic particle in cell code for modeling plasma based accelerators. In *Computational Science—ICCS 2002: International Conference Amsterdam, The Netherlands, April 21–24, 2002 Proceedings, Part III 2*, pp. 342–351. Springer, 2002.

Laurent Garrigues, Gwénaël Fubiani, and Jean-Pierre Boeuf. Negative ion extraction via particle simulation for fusion: critical assessment of recent contributions. *Nuclear Fusion*, 57(1):014003, 2016.

Hao Geng, Fangfang Chen, Jing Ye, and Fan Jiang. Applications of molecular dynamics simulation in structure prediction of peptides and proteins. *Computational and structural biotechnology journal*, 17:1162–1170, 2019.

Kai Germaschewski, William Fox, Stephen Abbott, Narges Ahmadi, Kristofor Maynard, Liang Wang, Hartmut Ruhl, and Amitava Bhattacharjee. The plasma simulation code: A modern particle-in-cell code with patch-based load-balancing. *Journal of Computational Physics*, 318:305–326, 2016.

Will Grathwohl, Kuan-Chieh Wang, Jörn-Henrik Jacobsen, David Duvenaud, Mohammad Norouzi, and Kevin Swersky. Your classifier is secretly an energy based model and you should treat it like one. *arXiv preprint arXiv:1912.03263*, 2019.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.

Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *Journal of Machine Learning Research*, 23(47):1–33, 2022.

Zeviel Imani, Shuchin Aeron, and Taritree Wongjirad. Score-based diffusion models for generating liquid argon time projection chamber images, 2024.

Chenfanfu Jiang, Craig Schroeder, Andrew Selle, Joseph Teran, and Alexey Stomakhin. The affine particle-in-cell method. *ACM Transactions on Graphics (TOG)*, 34(4):1–10, 2015.

Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8110–8119, 2020.

Denis Keefe. Inertial confinement fusion. *Annual Review of Nuclear and Particle Science*, 32(1): 391–441, 1982.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

Ondrej Klimo, Shlomo Weber, VT Tikhonchuk, and J Limpouch. Particle-in-cell simulations of laser–plasma interaction for the shock ignition scenario. *Plasma Physics and Controlled Fusion*, 52(5):055013, 2010.

Xianglong Kong, Michael C Huang, Chuang Ren, and Viktor K Decyk. Particle-in-cell simulations with charge-conserving current deposition on graphic processing units. *Journal of Computational Physics*, 230(4):1676–1685, 2011.

R Kube, RM Churchill, and B Sturdevant. Machine learning accelerated particle-in-cell plasma simulations. *arXiv preprint arXiv:2110.12444*, 2021.

A Bruce Langdon. Evolution of particle-in-cell plasma simulation. *IEEE Transactions on Plasma Science*, 42(5):1317–1320, 2014.

Rolf Lange. Adpic—a three-dimensional particle-in-cell model for the dispersal of atmospheric pollutants and its comparison to regional tracer studies. *Journal of Applied Meteorology and Climatology*, 17(3):320–329, 1978.

Yann LeCun, Sumit Chopra, Raia Hadsell, M Ranzato, Fujie Huang, et al. A tutorial on energy-based learning. *Predicting structured data*, 1(0), 2006.

H Ralph Lewis, A Sykes, and JA Wesson. A comparison of some particle-in-cell plasma simulation methods. *Journal of Computational Physics*, 10(1):85–106, 1972.

Lizao Li, Robert Carver, Ignacio Lopez-Gomez, Fei Sha, and John Anderson. Generative emulation of weather forecast ensembles with diffusion models. *Science Advances*, 10(13):eadk4489, 2024.

Paulett C Liewer and Viktor K Decyk. A general concurrent algorithm for plasma particle-in-cell simulation codes. *Journal of Computational Physics*, 85(2):302–322, 1989.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow, 2022.

Xingchao Liu, Xiwen Zhang, Jianzhu Ma, Jian Peng, and Qiang Liu. Instaflow: One step is enough for high-quality diffusion-based text-to-image generation, 2024.

Norihiro Oyama, Noriko N Ishizaki, Satoshi Koide, and Hiroaki Yoshida. Deep generative model super-resolves spatially correlated multiregional climate data. *Scientific Reports*, 13(1):5992, 2023.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, 2019.

Magnus Petersen, Gemma Roig, and Roberto Covino. Dynamicsdiffusion: Generating and rare event sampling of molecular dynamic trajectories using diffusion models. In *NeurIPS 2023 AI for Science Workshop*, 2023.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015.

Andrea Sgattoni, Luca Fedeli, Stefano Sinigardi, Alberto Marocchino, Andrea Macchi, Volker Weinberg, and Anupam Karmakar. Optimising piccante-an open source particle-in-cell code for advanced simulations on tier-0 systems. *arXiv preprint arXiv:1503.02464*, 2015.

Harshil Shah, Siddharth Kamaria, Riddhesh Markandeya, Miral Shah, and Bhaskar Chaudhury. A novel implementation of 2d3v particle-in-cell (pic) algorithm for kepler gpu architecture. In *2017 IEEE 24th International Conference on High Performance Computing (HiPC)*, pp. 378–387. IEEE, 2017.

Michael J Smith, James E Geach, Ryan A Jackson, Nikhil Arora, Connor Stone, and Stéphane Courteau. Realistic galaxy image simulation via score-based generative models. *Monthly Notices of the Royal Astronomical Society*, 511(2):1808–1818, January 2022. ISSN 1365-2966.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, 2015.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *ICLR*, 2021.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *NeurIPS*, 2019.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *ICML*, 2023.

DJ Strozzi, M Tabak, DJ Larson, L Divol, AJ Kemp, C Bellei, MM Marinak, and MH Key. Fast-ignition transport studies: Realistic electron source, integrated particle-in-cell and hydrodynamic modeling, imposed magnetic fields. *Physics of Plasmas*, 19(7), 2012.

Deborah Sulsky, Shi-Jian Zhou, and Howard L Schreyer. Application of a particle-in-cell method to solid mechanics. *Computer physics communications*, 87(1-2):236–252, 1995.

LJ Suter, S Glenzer, S Haan, B Hammel, K Manes, N Meezan, J Moody, M Spaeth, L Divol, K Oades, et al. Prospects for high-gain, high yield national ignition facility targets driven by 2ω (green) light. *Physics of Plasmas*, 11(5):2738–2745, 2004.

Junya Suzuki, Hironori Shimazu, Keiichiro Fukazawa, and Mitsue Den. Acceleration of pic simulation with gpu. *Plasma and Fusion Research*, 6:2401075–2401075, 2011.

F Taccogna and P Minelli. Pic modeling of negative ion sources for fusion. *New Journal of Physics*, 19(1):015012, 2017.

David Tskhakaya, Konstantin Matyash, Ralf Schneider, and Francesco Taccogna. The particle-in-cell method. *Contributions to Plasma Physics*, 47(8-9):563–594, 2007.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.

John P Verboncoeur. Particle simulation of plasmas: review and advances. *Plasma Physics and Controlled Fusion*, 47(5A):A231, 2005.

Patrick von Platen, Suraj Patil, Anton Lozhkov, Pedro Cuenca, Nathan Lambert, Kashif Rasul, Mishig Davaadorj, Dhruv Nair, Sayak Paul, William Berman, Yiyi Xu, Steven Liu, and Thomas Wolf. Diffusers: State-of-the-art diffusion models. `https://github.com/huggingface/diffusers`, 2022.

Pan Wang and Xiangqin Zhu. Hybrid cpu-and gpu-based implementation for particle-in-cell simulation on multicore and multi-gpu systems. In *2021 Photonics & Electromagnetics Research Symposium (PIERS)*, pp. 155–161. IEEE, 2021.

Fang Wu and Stan Z Li. Diffmd: a geometric diffusion model for molecular dynamics simulations. In *AAAI*, 2023.

Ming Xu, Feiguo Chen, Xinhua Liu, Wei Ge, and Jinghai Li. Discrete particle simulation of gas–solid two-phase flows with multi-scale cpu–gpu hybrid computation. *Chemical engineering journal*, 207:746–757, 2012.

Qinsheng Zhang and Yongxin Chen. Diffusion normalizing flow. *Advances in neural information processing systems*, 34:16280–16291, 2021.

A Appendix

A.1 Implementation details of Diff-PIC

Physically-informed parameter encoder (PIPE). The PIPE module comprises two encoding strategies (i.e. positional encoders and polynomial encoders) followed by a single-layer MLP. The positional encoder (Vaswani et al., 2017), widely used in current Large Language Models, utilizes sinusoidal functions to encode the input parameters, facilitating smooth interpolation. For an input parameter x, the positional encoder of PIPE generates a 16-dimensional embedding using sine and cosine functions at varying frequencies:

$$Pos_i(x) = \begin{cases} \sin\left(\frac{x}{10000^{\frac{2i}{d}}}\right) & \text{if } i \text{ is even,} \\ \cos\left(\frac{x}{10000^{\frac{2i}{d}}}\right) & \text{if } i \text{ is odd,} \end{cases} \tag{6}$$

where $0 \leq i < 16$ and $d = 16$. Polynomial encoders include polynomial terms up to the fourth degree: (x, x^2, x^3, x^4). These polynomial terms are widely used to approximate a wide range of functions effectively, capturing nonlinear correlations and unbounded growth patterns in the data, which is crucial for extrapolation. For parameters T_e, T_i, I, each of them is processed separately through parallel positional and polynomial encoders, concatenated as a 20-dimensional embedding, and transformed by an MLP, resulting in a refined 20-dimensional embedding. The parameter t_{as} undergoes a positional encoder followed by MLP transformation, resulting in a refined 16-dimensional embedding. Finally, embeddings from these four parameters are concatenated to produce a final encoded representation, which is then combined with the simulation data as inputs to the Diff-PIC.

The architecture and configurations of the U-Net backbone. The visualization of the modified U-Net backbone is shown in Fig. 6, built using the HuggingFace Diffusers framework (von Platen et al., 2022). The U-Net backbone processes input images through an encoder-decoder architecture with skip connections. The encoding path comprises three downsampling blocks: a standard DownBlock2D followed by two AttnDownBlock2D modules, progressively reducing spatial dimensions by 4× (from H×W to H/4×W/4) while increasing the channel dimension from 64 to 256. Attention mechanisms are strategically integrated into both down- and up-sampling paths to capture long-range dependencies,

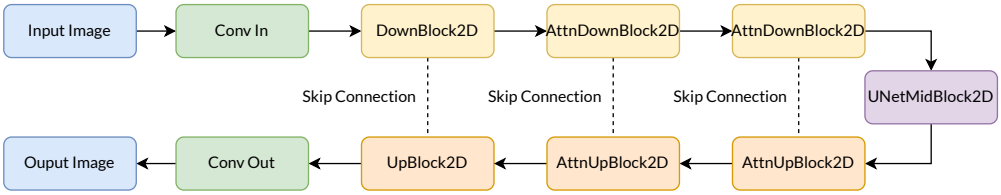

Figure 6: The U-Net backbone.

Table 4: Comparisons between MLP, Trans, and PIPE in interpolation tasks.

Method	Training Set for E1			Testing Set for E1			Training Set for E2			Testing Set for E2		
	MAE↓	RMSE↓	FID↓	MAE↓	RMSE↓	FID↓	MAE↓	RMSE↓	FID↓	MAE↓	RMSE↓	FID↓
MLP	3.72e-2	4.16e-2	2.06	3.90e-2	4.57e-2	2.15	1.71e-2	1.92e-2	0.926	1.85e-2	1.98e-2	0.932
Trans	1.61e-2	2.80e-2	1.35	1.72e-2	2.31e-2	1.70	8.61e-3	1.02e-2	0.375	9.41e-3	1.18e-2	0.409
PIPE	1.56e-2	2.67e-2	1.21	1.68e-2	2.29e-2	1.62	7.95e-3	9.32e-3	0.328	8.26e-3	1.03e-2	0.341

Table 5: Comparisons between MLP, Trans, and PIPE in extrapolation tasks.

Method	E1 10%			E1 20%			E2 10%			E2 20%		
	MAE↓	RMSE↓	FID↓	MAE↓	RMSE↓	FID↓	MAE↓	RMSE↓	FID↓	MAE↓	RMSE↓	FID↓
MLP	5.12e-2	6.01e-2	1.93	5.42e-2	5.64e-2	2.05	1.95e-2	2.62e-2	1.09	2.32e-2	2.91e-2	1.16
Trans	1.95e-2	2.61e-2	1.82	2.29e-2	2.73e-2	1.89	1.02e-2	1.55e-2	0.615	1.32e-2	2.04e-2	0.761
PIPE	1.83e-2	2.40e-2	1.74	2.18e-2	2.62e-2	1.82	9.47e-3	1.36e-2	0.536	1.13e-2	1.85e-2	0.673

with the attention operations performed on flattened spatial dimensions. The UNetMidBlock2D has attention and dual ResNet blocks. The decoder path mirrors the encoder with three upsampling blocks (two AttnUpBlock2D and one UpBlock2D), using skip connections to preserve fine spatial details. Each ResNet block employs GroupNorm normalization (with 32 groups) and SiLU activations, while convolution operations maintain padding to preserve spatial dimensions. The model culminates in a final convolution layer that projects the 64-channel features to a single-channel output while maintaining the original input resolution. The modified U-Net architecture combines local operations (convolutions) with global context (attention) and multiple conditions, making it well-suited for controlled diffusion-based generation tasks.

The loss function. The loss function of Diff-PIC is defined as Eq. 5.

A.2 Implementation details and comparisons of baselines

GAN-PIC is based on StyleGAN2 (Karras et al., 2020), a SOTA variant of Generative Adversarial Networks (GANs). It comprises two main components: a Generator and a Discriminator. The Generator in StyleGAN2 is enhanced with a style-based architecture that uses a mapping network to convert latent vectors into style vectors. These style vectors are then applied at various layers of the generator through adaptive instance normalization, allowing precise control over image attributes at different scales. The Discriminator's role is to differentiate between real and generated images, driving the adversarial training process that improves the realism and fidelity of the generated outputs. Additionally, StyleGAN2 incorporates architectural innovations such as progressive growing and skip connections, which contribute to training stability and high-quality image synthesis.

NF-PIC is implemented based on (Zhang & Chen, 2021) (setting the noise to zero), belong to normalizing flow based generative models. NF-PIC has a drift network and a score network, both modeled using U-Net architectures. But different from diffusion models, it relies on a sequence of invertible and differentiable transformations to construct complex distributions from simple ones, which are computationally intensive due to the need for invertibility and calculating the Jacobian determinant.

A.3 Ablation studies on the proposed PIPE

To further demonstrate the effectiveness of PIPE, we conduct ablation studies by replacing the PIPE component in Diff-PIC with other commonly used encoders: MLP (denoted as MLP) and Transformer decoder (denoted as Trans). Specifically, the two-layer MLP has 128 hidden units and ReLU activations. For the Transformer decoder, we first apply a positional encoder to generate a 16-dimensional embedding for each parameter (the same as the positional encoder in PIPE). This embedding is then processed by the Transformer decoder, which comprises multi-head self-attention and a two-layer MLP with ReLU activations. The comparison results are presented in Table 4 for interpolation tasks and Table 5 for extrapolation tasks. The results show that PIPE significantly outperforms both MLP and Transformer decoder in both interpolation and extrapolation tasks. MLP struggles with both interpolation and extrapolation. The Transformer decoder, while powerful for

Table 6: Ablation studies on different components of PIPE in interpolation tasks.

Method	Training Set for E1			Testing Set for E1			Training Set for E2			Testing Set for E2		
	MAE↓	RMSE↓	FID↓	MAE↓	RMSE↓	FID↓	MAE↓	RMSE↓	FID↓	MAE↓	RMSE↓	FID↓
only pos	1.64e-2	2.82e-2	1.37	1.76e-2	2.35e-2	1.72	8.63e-3	1.05e-2	0.379	9.42e-3	1.20e-2	0.412
only poly	1.76e-2	2.94e-2	1.52	1.92e-2	2.61e-2	1.85	1.14e-2	1.27e-2	0.517	1.13e-2	1.35e-2	0.546
PIPE	1.56e-2	2.67e-2	1.21	1.68e-2	2.29e-2	1.62	7.95e-3	9.33e-3	0.328	8.26e-3	1.03e-2	0.341

Table 7: Ablation studies on different components of PIPE in extrapolation tasks.

Method	E1 10%			E1 20%			E2 10%			E2 20%		
	MAE↓	RMSE↓	FID↓	MAE↓	RMSE↓	FID↓	MAE↓	RMSE↓	FID↓	MAE↓	RMSE↓	FID↓
only pos	1.97e-2	2.64e-2	1.83	2.31e-2	2.76e-2	1.93	1.03e-2	1.57e-2	0.624	1.35e-2	2.10e-2	0.812
only poly	1.91e-2	2.52e-2	1.78	2.25e-2	2.70e-2	1.89	9.75e-3	1.42e-2	0.581	1.26e-2	1.92e-2	0.724
PIPE	1.83e-2	2.40e-2	1.74	2.18e-2	2.62e-2	1.82	9.47e-3	1.36e-2	0.536	1.13e-2	1.85e-2	0.673

sequence modeling and capturing relationships in discrete token sequences, is not inherently designed to effectively process continuous physical parameters, showing inferior performance.

To provide more insights into the effectiveness of PIPE, we conduct another ablation study to evaluate the contributions of the positional encoder ("only pos"), polynomial encoder ("only poly") separately. The results are shown in Table 6 (interpolation tasks) and Table 7 (extrapolation tasks), highlighting the individual importance of these encoders. Specifically, combining both positional and polynomial encoders consistently outperforms using either encoder alone in both interpolation and extrapolation tasks. In interpolation tasks, the positional encoder demonstrates better performance than the polynomial encoder, suggesting its important role in interpolation scenarios. Conversely, in extrapolation tasks, particularly at larger ranges (20%), the polynomial encoder shows relatively better performance, indicating its importance for extrapolation capabilities. These results validate our design choice of combining both encoders in PIPE to leverage their complementary strengths.