Analysis of a Generative Adversarial Network and a Denoising Diffusion Probabilistic Model for the Design of Nuclear Reactor Loading Patterns

Gabriel Pereira Coelho[1,2⋆], Lenilson Moreira Araujo[1], and Anderson Alvarenga de Moura Meneses[1,2]

[1]Laboratory of Computational Intelligence, Institute of Engineering and Geosciences, Federal University of Western Pará, Brazil

[2]Postgraduate Program in Computer Science, Institute of Engineering and Geosciences, Federal University of Western Pará, Brazil

gabriel.pc@discente.ufopa.edu.br, lenilson.araujo@ufopa.edu.br

anderson.meneses@ufopa.edu.br

Abstract. *At the end of each operation cycle in a nuclear power plant, a portion of the spent nuclear fuel must be replaced, while the remaining fuel is rearranged within the core, forming a Loading Pattern (LP), an NP-hard combinatorial problem subject to safety constraints. Although LP design is still predominantly supported by optimization metaheuristics, the application of deep generative models to this problem lacks comparative evidence. This work compares StyleGAN2-ADA and a Denoising Diffusion Probabilistic Model (DDPM) for LP generation, representing LP configurations as images. DDPM required less hyperparameter tuning and trained substantially faster than StyleGAN2-ADA. Statistically significant differences were found between the models, with DDPM achieving better structural compliance and generating markedly more configurations outside the training set, suggesting broader exploration of the feasible LP space. Overall, the findings support DDPM as a favorable option for LP generation under the conditions tested.*

Keywords: *Generative Artificial Intelligence, Diffusion Models, Generative Adversarial Networks, Nuclear Energy, Galerkin Finite Element Method.*

1. Introduction

During the energy production cycle in a Nuclear Power Plant (NPP), spent fuel must be replaced, while the remaining assemblies are rearranged within the reactor core, forming a configuration of Fuel Assemblies (FAs) known as a Loading Pattern (LP). This rearrangement of FAs is essential for maintaining the nominal power of the plant and ensuring the structural integrity of the system. Figure 1 shows the reactor core in top view together with its symmetry axes: because the core layout repeats identically across these axes, only a reduced region of the core (e.g., one quarter or one eighth) needs to be specified to fully define the LP, as detailed for the benchmark used in this work in Figure 2. The task of determining the position of the FAs within the core is a combinatorial problem known

⋆Corresponding author.

as Loading Pattern Optimization or the Nuclear Reactor Reload Problem (NRRP), which is an NP-hard problem that seeks to balance economic efficiency, operational safety, and physical constraints [Levine 1986]. Although the LP design is still predominantly supported by optimization metaheuristics, the application of deep generative artificial intelligence to such problem has not yet been thoroughly assessed, with limited comparative evidence between models.

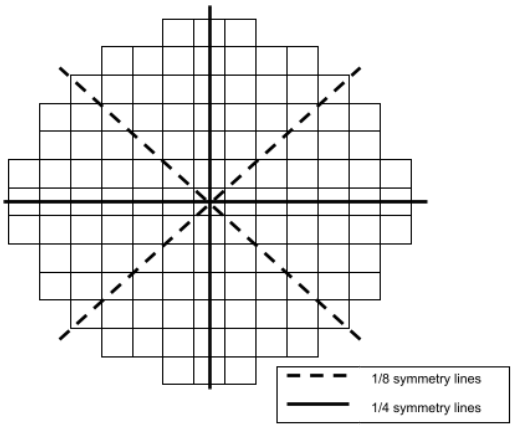


Figure 1. Schematic 2D representation of a nuclear reactor core (top view) and its symmetry lines, which allow the Loading Pattern to be defined over a reduced region of the core.

Traditionally, solutions for the NRRP have been supported by Optimization Metaheuristics, with studies employing techniques such as Particle Swarm Optimization (PSO) [Meneses et al. 2009], Genetic Algorithms [Chapot et al. 1999], and Cuckoo Search [Meneses et al. 2020], among other approaches. Recently, studies utilized Wasserstein Generative Adversarial Networks (WGAN) to generate LPs, showing that generative models can learn the statistical distribution of configurations treated as images [Meneses and Araujo 2026]. Denoising Diffusion Probabilistic Models (DDPMs) have since achieved strong results in image generation [Ho et al. 2020, Dhariwal and Nichol 2021], motivating their application to the NRRP.

The main contribution of this work is a comparative analysis of StyleGAN2-ADA [Karras et al. 2020b] and DDPM [Ho et al. 2020] for LP generation in Pressurized Water Reactor (PWR) cores, assessing generative quality, reactor physics compliance, novelty of configurations, and training efficiency as computational tools for nuclear engineering.

2. Theoretical Background

The NRRP, or LP design, consists of determining the arrangement of FAs in a PWR core under coupled neutronic, thermal hydraulic, operational, and safety constraints. Local changes in assembly position produce nonlinear effects on power distribution and cycle behavior, making exhaustive search impractical [Levine 1986]. Metaheuristics remain central in NRRP practice but their performance is sensitive to initialization and hyperparameter tuning [Meneses et al. 2009]. This motivates generative modeling as an alternative: the model learns regularities in feasible LP spaces and samples candidates directly [Goodfellow et al. 2016].

In this work, LPs are represented as matrix structured images, preserving spatial relationships among FAs for use with convolutional architectures. Figure 2 illustrates the IAEA-2D [Argonne National Laboratory 1977] quarter-core geometry used to define the LP representation: exploiting the $1/4$ symmetry shown in Figure 1, only this 9×9 region needs to be specified, and the full-core LP is recovered by reflecting it across the two symmetry axes. Each cell of the grid corresponds to one FA position, and its number identifies the FA type placed at that position (classes 1-4 in the figure, encoded as classes 0-3 in the data), with each class representing a distinct FA composition: classes 1 and 3 are fuel assemblies of different compositions, class 2 denotes positions occupied by a control rod (an assembly that absorbs neutrons to regulate reactivity and power), and class 4 denotes reflector positions. It is this labeled 9×9 grid that is subsequently converted into the grayscale/RGB image used to train and evaluate the generative models, as described in Section 3.1; an LP is thus treated in the generative pipeline as an image whose classes are pixel intensities, rather than as a vector representation as in the case of metaheuristics for the NRRP.

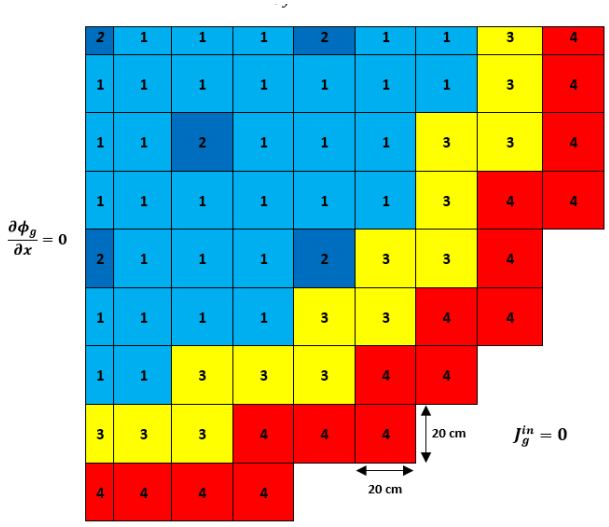


Figure 2. IAEA-2D quarter-core geometry with $1/4$ symmetry.

Generative Adversarial Networks (GANs) synthesize samples through adversarial optimization between generator and discriminator. StyleGAN variants improved image quality and control of generated structure, from the original StyleGAN formulation [Karras et al. 2019] to StyleGAN2-ADA [Karras et al. 2020b, Karras et al. 2020a]. Still, adversarial training may exhibit instability and mode collapse, reducing diversity in multimodal data regimes.

DDPMs learn generation by reversing a progressive noise corruption process through iterative denoising [Ho et al. 2020]. Compared with adversarial optimization, diffusion training is often more stable and tends to improve mode coverage, and diffusion models have achieved competitive image synthesis quality against GAN baselines [Dhariwal and Nichol 2021].

Let $\mathbf{x}_0$ be a clean LP image sample. In the forward process, Gaussian noise is progressively added for T steps according to a variance schedule $\{\beta_t\}_{t=1}^{T}$, yielding noisy variables $\mathbf{x}_t$. The reverse process is parameterized by a U-Net that predicts and removes

noise iteratively to reconstruct structured samples [Ho et al. 2020]. The training objective is noise prediction; sampling starts from random noise and applies the learned reverse transitions.

For LP generation, the comparison between GAN and DDPM is based on multiple criteria: training stability, diversity, fidelity, generation cost, and reactor domain adherence. GAN models provide fast sampling after training, whereas DDPM models usually offer more stable optimization and broader coverage at higher sampling cost due to iterative denoising [Ho et al. 2020, Nichol and Dhariwal 2021].

Because LP design is an engineering task, visual quality alone is insufficient. Assessment must combine image quality control, such as the Fréchet Inception Distance (FID) [Heusel et al. 2017] used internally to monitor StyleGAN2-ADA training, with reactor physics indicators such as the Effective Multiplication Factor (k_{eff}) and the Maximum Assembly Relative Power (P_{rm}) calculated with the Galerkin Finite Element Method (GFEM) [Meneses et al. 2024], which are the metrics used to compare the two models, following protocols adopted in prior PWR studies [Meneses et al. 2009, Meneses and Araujo 2026].

3. Methodology

Two generative models were compared for LP generation: StyleGAN2-ADA and DDPM. The methodology was structured in sequential steps, as shown in Figure 3.

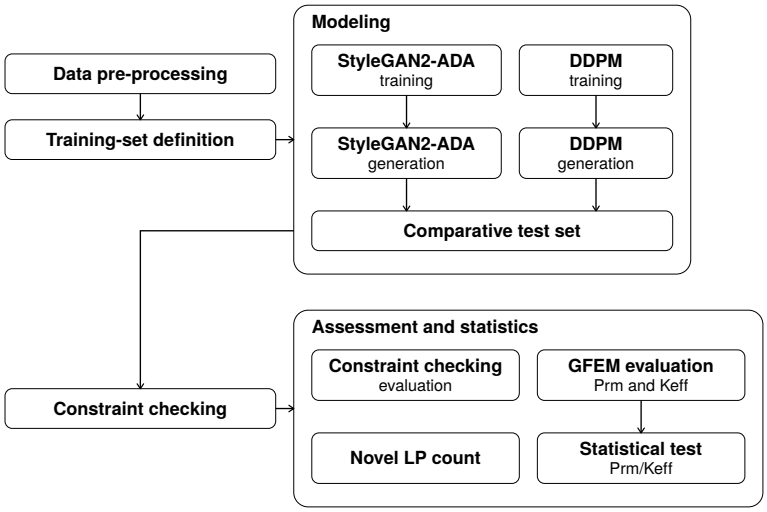


Figure 3. Workflow of the present work.

3.1. Data Preprocessing

The dataset consists of 30,546 feasible LPs from the IAEA benchmark problem [Argonne National Laboratory 1977], generated via PSO following the 1/4 symmetry representation in [Meneses et al. 2009, Meneses and Araujo 2026]. Each LP is encoded as a 9×9 integer matrix (classes 0-3). The official StyleGAN2-ADA implementation requires square, power-of-two input resolutions [Karras et al. 2020a], a constraint the native 9×9 matrix does not satisfy; the LP was therefore resized to 64×64 (rather than

the minimal 32×32 option) to provide more pixels per FA cell and reduce class-recovery ambiguity, via nearest-neighbor interpolation, mapping classes to grayscale values {0, 85, 170, 255} and converting to RGB by channel replication, to produce a common training format used by both models. Representative samples are shown in Figure 4.

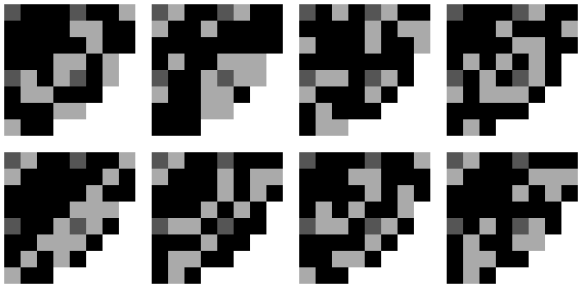


Figure 4. Examples of 8 LPs used for training.

3.1.1. StyleGAN2-ADA.

The official PyTorch implementation [Karras et al. 2020a] was used without data augmentation, as geometric transforms would alter the physical meaning of LP configurations and photometric ones would distort the discrete class encoding.

3.1.2. DDPM.

For DDPM, the Hugging Face Diffusers library was utilized, specifically following the official unconditional image generation example provided in the library's repository [von Platen et al. 2022] using a 2D U-Net [Ho et al. 2020].

3.2. Model Training and Settings

Training and hardware settings for both models are summarized in Table 1. Both models were trained on the SDumont cluster at the Laboratório Nacional de Computação Científica (LNCC) [LNCC SERED 2025].

StyleGAN2-ADA used Python 3.7 with PyTorch 1.7.1 and CUDA 11.0; DDPM used Python 3.10 with PyTorch 2.5.1, CUDA 12.1, and Hugging Face Diffusers [von Platen et al. 2022]. Training was monitored via TensorBoard. For analysis, we used the StyleGAN2-ADA checkpoint with best native FID50k_full (19,152 kimg); since the official DDPM pipeline does not compute FID, and an external implementation would not be comparable to StyleGAN2-ADA's native score [Heusel et al. 2017], the final DDPM checkpoint was instead selected by training convergence (Section 4.2). Training took approximately 27 hours for StyleGAN2-ADA and 3.5 hours for DDPM.

3.3. GFEM Evaluation of Generated LPs

For evaluation, synthetic LP sets at 64×64 resolution were generated by both StyleGAN2-ADA and DDPM. Following the groupwise evaluation style used in a previous LP study based on WGAN [Meneses and Araujo 2026], we used 1000 LPs from StyleGAN2-ADA

Table 1. Training and hardware settings for StyleGAN2-ADA and DDPM.

Shared settings	
Training set	30,546 LPs
Input format	64×64 RGB images
Batch size	64
Platform	SDumont cluster (LNCC)
Hardware	2× Xeon Skylake 6252 + 4× V100 GPUs
StyleGAN2-ADA	
Training budget	25,000 kimg
R_1 regularization	2
Snapshot interval	250 training ticks
Quality monitoring	FID50k_full
DDPM	
Training budget	200 epochs (≈24,000 steps)
Optimizer	AdamW
Learning rate	10^{-4}
LR schedule	Cosine with 500 warmup steps
Diffusion process	1,000-step linear β schedule

and DDPM. To compute k_{eff} and P_{rm} with GFEM [Meneses et al. 2024], generated images were converted to the LP representation required by the reactor physics pipeline. As in the previous WGAN methodology [Meneses and Araujo 2026], a constraint checking step after generation was applied before reactor physics evaluation, enforcing the benchmark LP structural requirements; when a generated LP did not satisfy those requirements,class rounding was applied in a way equivalent to output rounding in classification tasks. RGB outputs were then converted to grayscale, mapped to discrete LP classes, and finally transformed into the 9×9 matrix format used by GFEM. Because visually valid LP images do not necessarily correspond to neutronic configurations close to criticality or to acceptable power distributions, GFEM evaluated k_{eff} and P_{rm}. The later was used as a surrogate safety indicator associated with power peaking. The maximum assembly relative power is defined as

$$P_{\text{rm}} = \max_{A} \left(P_A^N \right), \qquad P_A^N = \frac{\bar{P}_A}{C}, \tag{1}$$

where P_A^N is the normalized power of FA A, $\bar{P}_A$ is the average power of assembly A, and C is the core average normalization constant, given by

$$\bar{P}_A = \frac{1}{V_A} \int_{V_A} \sum_{g=1}^{G} \nu\Sigma_{f,g}\phi_g \, dV, \qquad C = \frac{1}{V_C} \int_{V_C} \sum_{g=1}^{G} \nu\Sigma_{f,g}\phi_g \, dV. \tag{2}$$

Thus, k_{eff} verifies the criticality behavior of each generated LP, whereas P_{rm} identifies local power concentration over the core.

3.4. Constraint Checking Evaluation

In the IAEA-2D benchmark, structural constraints define the exact composition and placement of FAs per symmetry group. For quartet positions (those lying on symmetry axes and

therefore occupying 4 equivalent core slots each), the valid configuration requires exactly 8 black-type, 2 light gray-type, and 3 dark gray-type FAs, with the 3 dark gray positions being spatially fixed. For octet positions (off-axis assemblies occupying 8 equivalent core slots each), the required configuration is exactly 13 black-type and 3 light gray-type FAs. Reflector assemblies are excluded from these counts. The constraint check was applied to all generated LPs before reactor physics evaluation. When a generated LP did not satisfy the requirements, corrections were applied by adjusting the number or position of FAs of each type. The Euclidean distance between the original generated LP and the corrected LP was used to quantify the magnitude of the correction needed.

3.5. Assessment of New LPs and Statistical Analysis

The assessment of new LPs not in the training datasets was quantified as the number of new LPs generated by each model relative to the training dataset used. Statistical analysis was then performed on P_{rm} and k_{eff} values by first applying the Shapiro–Wilk normality test and then the Mann–Whitney U test [Sheskin 2011] to compare StyleGAN2-ADA and DDPM for each metric. The difference between the samples was considered statistically significant when the probability of the result occurring by chance was lower than 5% ($p < 0.05$) [Sheskin 2011].

4. Results and Discussion

4.1. StyleGAN2-ADA

As shown in Figure 5, StyleGAN2-ADA exhibits a rapid initial reduction of discriminator loss, followed by stabilization near zero, while generator loss remains at higher levels. The R_1 penalty decreases after the initial phase but maintains oscillatory behavior throughout training. The model achieved its best FID score at 19,152 kimg of training, after which no further meaningful improvement was observed.

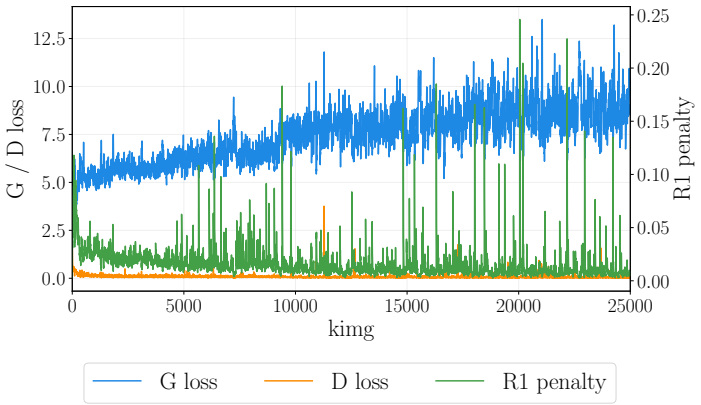


Figure 5. StyleGAN2-ADA training dynamics.

4.2. DDPM

Figure 6 presents the training dynamics for the DDPM experiment. The loss decreases steadily throughout training, reaching near-zero values by the final epoch. The learning rate follows the expected warmup and decay profile, rising during the initial epochs and gradually decaying to zero at the end of training. The model converged well near the end

of the 200-epoch run; at the dataset and batch size used in this experiment, each epoch corresponds to approximately 120 training steps, so the full run totals approximately 24,000 steps. These loss trajectories support the adopted training strategy and provide context for the results discussed in Section 4.5.

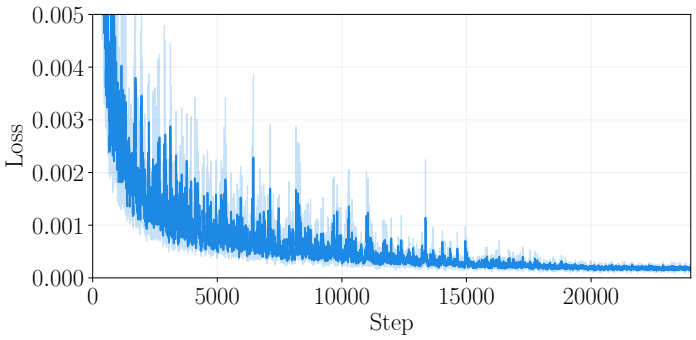
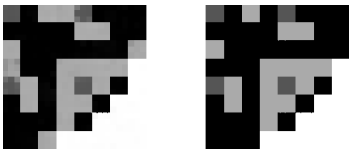


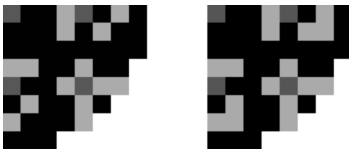
Figure 6. DDPM training curves (200-epoch run).

4.3. Constraint Checking Evaluation

For DDPM, 75.5% of the generated LPs satisfied the structural constraints without any correction, while for StyleGAN2-ADA all generated LPs required corrections, suggesting the DDPM model needs less postprocessing to produce structurally valid LPs. Visual inspection of the generated outputs reveals two categories of structural violations in StyleGAN2-ADA: visible color artifacts at assembly boundaries and incorrect counts of FA types, where the same FA class appears in the wrong quantity relative to the benchmark specification. In DDPM outputs, color errors are smaller and not perceptible by visual inspection alone; they become apparent only upon examining the raw pixel matrices, and in some cases FAs are placed in incorrect positions even when the color distribution appears plausible. Representative examples of generated LPs before and after constraint checking are shown in Figure 7, along with their corresponding Euclidean distances. Figure 7(b) is an outlier, since the vast majority of DDPM LPs required almost no correction. Descriptive statistics of the Euclidean distances for both models are summarized in Table 2.



(a) StyleGAN2-ADA: before (left) and after (right) constraint check. Euclidean distance: 989.04.



(b) DDPM: before (left) and after (right) constraint check. Euclidean distance: 716.95.

Figure 7. Representative generated LPs before and after constraint checking.

StyleGAN2-ADA required corrections in every generated LP, with a mean Euclidean distance of 405.23 and standard deviation of 251.94. DDPM achieved a mean

Table 2. Descriptive statistics of the Euclidean distance between generated images before and after the constraint check.

Statistic	StyleGAN2-ADA	DDPM
Count	1000	1000
Mean	405.23	2.98
Std. dev.	251.94	42.34
Min	36.00	0.00
25%	230.09	0.00
Median	279.12	0.00
75%	687.54	0.00
Max	989.04	717.18

distance of 2.98, with 75% of samples requiring no correction (median and lower quartiles at 0.00). The value of 717.18 for DDPM is the maximum found among the LPs that did require correction, but the distribution confirms that DDPM produces structurally compliant LPs more consistently than StyleGAN2-ADA.

4.4. Comparison of Reactor Physics Metrics

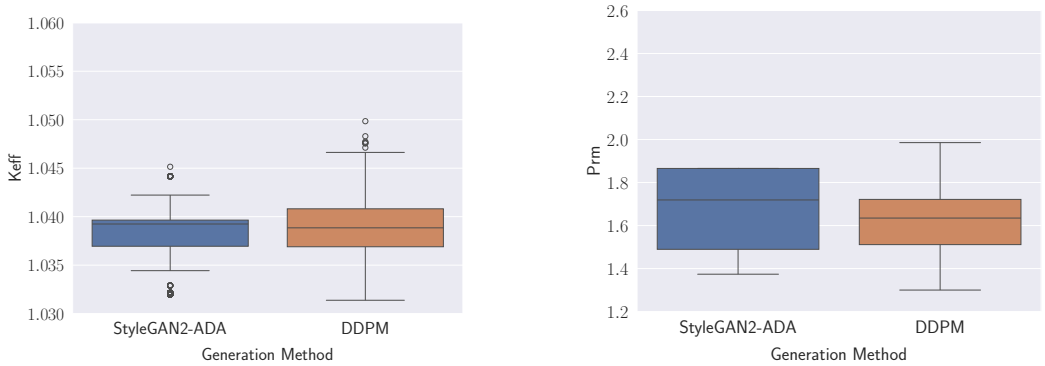
Descriptive statistics for k_{eff} and P_{rm} are organized in Table 3. Since the Shapiro–Wilk tests indicated non-Gaussian behavior in all groups ($p < 0.05$), the Mann–Whitney U test was used for pairwise comparison.

Figure 8 summarizes the distributions of k_{eff} and P_{rm} for StyleGAN2-ADA and DDPM. For k_{eff}, both models present similar central tendency and substantial interquartile overlap. Despite this visual proximity between boxplots, the Mann–Whitney U test revealed a statistically significant difference between the models ($U = 438411.0$, $p = 1.78 \times 10^{-6}$). For P_{rm}, the distributions are more separated, with StyleGAN2-ADA presenting a higher median (1.719 versus 1.635), and the difference is also statistically significant ($U = 621897.0$, $p = 3.00 \times 10^{-21}$).

These values are consistent with the safety thresholds adopted in a prior PWR loading pattern study on the same benchmark [Meneses and Araujo 2026] (k_{eff} close to criticality and P_{rm} around the 1.6-1.8 safety range), indicating that both models produced results within an acceptable engineering range.

Table 3. Descriptive statistics for k_{eff} and P_{rm}.

	k_{eff}		P_{rm}	
Statistic	StyleGAN2-ADA	DDPM	StyleGAN2-ADA	DDPM
Mean	1.03811	1.03899	1.67655	1.61936
Std. dev.	0.00275	0.00300	0.16156	0.13458
Median	1.03924	1.03885	1.71908	1.63483
Min	1.03197	1.03138	1.37392	1.30016
Max	1.04515	1.04985	1.86558	1.98558



(a) k_{eff} comparison (b) P_{rm} comparison

Figure 8. Statistical comparison between StyleGAN2-ADA and DDPM.

4.5. Assessment of New LPs

Both models were trained on the same D_{30k} dataset, and 1000 LPs were generated by each model for this assessment. Under this criterion, DDPM produced 527/1000 new LPs, whereas StyleGAN2-ADA produced only 12/1000 new LPs. Therefore, with the same training data, DDPM generated substantially more core configurations not present in the training set.

This suggests that DDPM explores a broader region of the feasible LP space, whereas StyleGAN2-ADA tends to reproduce configurations close to the training set, consistent with the reduced diversity of adversarial training.

The Shapiro–Wilk normality test indicated non-normal behavior in all groups for both P_{rm} and k_{eff}, so the comparison between models was performed with the Mann–Whitney U test. Table 4 consolidates the statistical test p-values for both metrics.

Table 4. Summary of normality and comparison tests between models.

Metric	Shapiro–Wilk p (StyleGAN2-ADA)	Shapiro–Wilk p (DDPM)	Mann–Whitney p
P_{rm}	1.84×10^{-28}	1.62×10^{-10}	3.00×10^{-21}
k_{eff}	1.89×10^{-24}	1.33×10^{-4}	1.78×10^{-6}

5. Conclusion

In the present work, DDPM required less manual hyperparameter selection and trained faster (about 3.5 hours versus 27 hours for StyleGAN2-ADA). The Mann–Whitney U test indicated statistically significant differences in both k_{eff} and P_{rm}, with StyleGAN2-ADA presenting higher median P_{rm}. Under the same D_{30k} training data, DDPM generated more configurations outside the training set, suggesting broader coverage of the feasible LP space. This pattern is consistent with the hypothesis that postprocessing compensates mainly local class assignment errors in StyleGAN2-ADA outputs, while DDPM tends to preserve class structure more precisely before correction. Together, these findings support a multiple criteria assessment (structural compliance, reactor physics metrics, assessment of new LPs, and training efficiency) when comparing generative methods for LP generation.

The results reinforce the potential of generative models as an alternative to metaheuristics for LP generation. Unlike PSO or genetic algorithms, which require iterative perturbations and hyperparameter tuning, DDPM generates candidate LPs by sampling from a learned distribution, with no search operators and fixed inference cost. Future work should extend the evaluation to larger and more diverse LP datasets and investigate generative architectures closer to the current state of the art, such as latent diffusion models, which enable higher resolution synthesis with reduced computational cost by operating in compressed latent spaces [Rombach et al. 2022]. Such approaches could be integrated with optimization or decision support workflows for practical fuel management of NPPs.

Acknowledgments

The authors acknowledge the National Laboratory for Scientific Computing (LNCC/MCTI, Brazil) for providing HPC resources from the SDumont supercomputer, which supported the computational results reported in this paper (`http://sdumont.lncc.br`). A.A.M.M. acknowledges the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) for financial support (Grant no. 304174/2025-4). G.P.C. acknowledges a scholarship from the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES). This study was financed in part by CAPES - Finance Code 001. The authors also thank the anonymous reviewers for their valuable comments and suggestions, which helped improve the quality of this manuscript.

Disclosure of interests. The authors have no competing interests to declare that are relevant to the content of this article.

Use of AI tools. During the preparation of this work, the authors used Claude (Anthropic) for language and readability review of the manuscript text. The authors reviewed and edited all content and take full responsibility for the content of the publication.

References

Argonne National Laboratory (1977). Benchmark problem book (supplement 2). Technical Report ANL-7416, Argonne National Laboratory.

Chapot, J. L. C., da Silva, F. C., and Schirru, R. (1999). A new approach to the use of genetic algorithms to solve the pressurized water reactor's fuel management optimization problem. *Annals of Nuclear Energy*, 26:641–655.

Dhariwal, P. and Nichol, A. (2021). Diffusion models beat GANs on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press, Cambridge, MA.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. (2017). GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Advances in Neural Information Processing Systems*, volume 30.

Ho, J., Jain, A., and Abbeel, P. (2020). Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851.

Karras, T., Aittala, M., Hellsten, J., Laine, S., Lehtinen, J., and Aila, T. (2020a). Training generative adversarial networks with limited data. *Advances in Neural Information Processing Systems*, 33:12104–12114.

Karras, T., Laine, S., and Aila, T. (2019). A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4401–4410.

Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., and Aila, T. (2020b). Analyzing and improving the image quality of StyleGAN. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8110–8119.

Levine, S. (1986). In-core fuel management of four reactor types. In Ronen, Y., editor, *Handbook of Nuclear Reactors Calculations*, volume II. CRC Press.

LNCC SERED (2025). Manual SDumont: 02 - caracteristicas. GitHub Wiki. Accessed: 2026-04-07.

Meneses, A. A. M. and Araujo, L. M. (2026). Design of nuclear fuel loading patterns for a PWR with Wasserstein generative adversarial networks. *Nuclear Engineering and Design*, 449:114765.

Meneses, A. A. M., Araujo, L. M., and Schirru, R. (2024). A GPU-accelerated linear system solution for the Galerkin finite element method applied to neutron diffusion equation. *Nuclear Engineering and Design*, 421:113103.

Meneses, A. A. M., Machado, M. D., and Schirru, R. (2009). Particle swarm optimization applied to the nuclear reload problem of a pressurized water reactor. *Progress in Nuclear Energy*, 51(2):319–326.

Meneses, A. A. M., Silva, P. V., Nast, F. N., Araujo, L. M., and Schirru, R. (2020). Application of cuckoo search algorithm to loading pattern optimization problems. *Annals of Nuclear Energy*, 139:107214.

Nichol, A. and Dhariwal, P. (2021). Improved denoising diffusion probabilistic models. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8162–8171.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695.

Sheskin, D. J. (2011). *Handbook of Parametric and Nonparametric Statistical Procedures, Fifth Edition*. CRC Press.

von Platen, P., Patil, S., Lozhkov, A., Cuenca, P., Lambert, N., Rasul, K., Davaadorj, M., Nair, D., Paul, S., Berman, W., Xu, Y., Liu, S., and Wolf, T. (2022). Diffusers: State-of-the-art diffusion models. GitHub repository.