

Toward Scalable Normalizing Flows for the Hubbard Model

Janik Kreit,^{a,b,*} Andrea Bulgarelli,^{a,b} Lena Funcke,^{a,b} Thomas Luu,^{b,c} Dominic Schuh,^{a,b} Simran Singh^{a,b} and Lorenzo Verzichelli^d

^a*Transdisciplinary Research Area (TRA) Matter, University of Bonn, Germany*

^b*Helmholtz Institute for Radiation and Nuclear Physics (HISKP), University of Bonn, Germany*

^c*Institute for Advanced Simulation 4 (IAS-4), Forschungszentrum Jülich, Germany*

^d*Physics Department, University of Turin & INFN - Turin unit, Italy*

E-mail: jkreit@uni-bonn.de

Normalizing flows have recently demonstrated the ability to learn the Boltzmann distribution of the Hubbard model, opening new avenues for generative modeling in condensed matter physics. In this work, we investigate the steps required to extend such simulations to larger lattice sizes and lower temperatures, with a focus on enhancing stability and efficiency. Additionally, we present the scaling behavior of stochastic normalizing flows and non-equilibrium Markov chain Monte Carlo methods for this fermionic system.

*The 42nd International Symposium on Lattice Field Theory,
2nd-8th November, 2025,
Tata Institute of Fundamental Research, Mumbai, India*

*Speaker

1. Introduction

The Hubbard model is a central framework in condensed-matter physics for describing strongly correlated electronic systems [1, 2]. It can be formulated either in terms of a Hamiltonian or an action, with the latter typically referring to the finite-temperature auxiliary-field formulation. Despite its simple structure, numerical simulations in this formulation are challenging due to the multimodal structure of the Boltzmann distribution and the presence of large potential barriers. As a consequence, standard sampling methods such as Hybrid Monte Carlo (HMC) often suffer from ergodicity violations [3], which can lead to biased estimates of physical observables.

Generative neural samplers (GNSs), including Normalizing Flows (NFs) [4, 5] and autoregressive models [6], have shown strong performance in representing Boltzmann distributions across a wide range of physical systems. Often referred to as *Boltzmann generators* [7], they have been applied in lattice quantum field theory [8–10], statistical mechanics [11, 12], string theory [13, 14], and quantum chemistry [7, 15, 16]. Beyond accelerating sampling, GNSs enable direct estimates of thermodynamic observables [9] and entanglement entropies [17, 18]. In parallel, non-equilibrium Markov chain Monte Carlo (NE-MCMC) methods [19–22] remain effective in regimes where HMC suffers from ergodicity problems and strong autocorrelations [23].

In this work, we investigate the performance of NE-MCMC methods, NFs, and SNFs for sampling the Boltzmann distribution of the Hubbard model. While NFs provide efficient generative proposals and NE-MCMC methods offer favorable scaling properties, SNFs combine both by augmenting deterministic flow transformations with stochastic updates [21, 24]. This unified framework is particularly well suited for addressing ergodicity issues in Hubbard model simulations.

Building on earlier work on symmetry-enforced normalizing flows for the Hubbard model at small volumes [25, 26], we extend these studies to larger lattice volumes and lower temperatures by improving the numerical stability of the Hubbard action. We also present initial scaling results for NE-MCMC methods, NFs, and SNFs in this setting.

The remainder of these proceedings is organized as follows. In Sec. 2, we review the Hubbard model. In Sec. 3, we introduce an algorithm to improve numerical stability at low temperatures. The methods of NE-MCMC, NFs, and SNFs are discussed in Sec. 4, and their scaling results are presented in Sec. 5. Conclusions and an outlook are given in Sec. 6.

2. Hubbard Model

The Hubbard model is defined by the Hamiltonian

$$H = -\kappa \sum_{\langle x,y \rangle} \left(a_{x,\uparrow}^\dagger a_{y,\uparrow} + a_{x,\downarrow}^\dagger a_{y,\downarrow} \right) - \frac{U}{2} \sum_x (n_{x,\uparrow} - n_{x,\downarrow})^2, \quad (1)$$

where $a_{x,\sigma}^\dagger$ ($a_{x,\sigma}$) denotes the fermionic creation (annihilation) operator for a particle with spin $\sigma \in \{\uparrow, \downarrow\}$ at lattice site x , and $n_{x,\sigma} \equiv a_{x,\sigma}^\dagger a_{x,\sigma}$ is the corresponding number operator. The first term in Eq. (1) describes nearest-neighbor hopping with amplitude κ , while the second term represents an on-site interaction with coupling strength $U > 0$. In the following, we consider hexagonal, half-filled lattices which correspond to zero chemical potential. For all simulations, we set $\kappa = 1$.

Applying a Suzuki–Trotter decomposition [27, 28] and a Hubbard–Stratonovich transformation [29] to the Hamiltonian, the model can be expressed in terms of an action S ,

$$S[\phi] = \sum_{x,t \in \Lambda} \frac{\phi_{xt}^2}{2\delta U} - \log \det M[\phi] - \log \det M[-\phi]. \quad (2)$$

Here, $\phi \in \mathbb{R}^\Lambda$ is a bosonic auxiliary field defined on the lattice $\Lambda = N_x \times N_t$, with physical sites N_x and artificial sites N_t arising from the Suzuki–Trotter decomposition. This decomposition introduces an error of $\mathcal{O}(\delta^2)$, where $\delta = \beta/N_t$ and β is the inverse temperature.

Fermionic dynamics are encoded in the fermion matrix $M \in \mathbb{R}^{N_x N_t \times N_x N_t}$, whose elements are

$$M[\phi]_{x't',xt} = \delta_{x'x} \delta_{t't} - [e^{\delta h}]_{x'x} e^{\phi_{xt}} \mathcal{B}_{t'} \delta_{t',t+1}, \quad (3)$$

where h is the hopping matrix, and $\mathcal{B}_t = 1$ for $1 < t \leq N_t$ and $\mathcal{B}_1 = -1$ implements anti-periodic temporal boundary conditions.

Owing to the sparsity of M , its determinant can be evaluated using an LU decomposition,

$$\det M = \det L \cdot \det U. \quad (4)$$

As shown in Ref. [3], the triangular structure of L and U allows an equivalent formulation:

$$\det M = \det (\mathbb{1} + A_1 A_2 \dots A_{N_t}), \quad (5)$$

where the product of matrices $A_t \in \mathbb{R}^{N_x \times N_x}$ is referred to as the *sausage of matrices*. The individual matrices are given by

$$A_t = e^{\delta h} F_t \quad \text{with} \quad [F_t]_{xx'} = e^{\phi_{x,N_t-t}} \delta_{xx'}. \quad (6)$$

3. Enhancing Numerical Stability with QR Decompositions

In Eq. (6), the matrices A_t depend exponentially on the auxiliary field ϕ and the hopping matrix h . Consequently, the product $A \equiv A_1 A_2 \dots A_{N_t}$ develops a wide range of eigenvalues, resulting in a large condition number. While this behavior is mathematically well defined, it leads to numerical instabilities due to the accumulation of rounding errors during matrix multiplications. These instabilities propagate into the evaluation of the action and can produce unstable estimates of physical observables.

To mitigate this problem, we perform intermediate QR decompositions [30] during the computation of the matrix product. Here, Q is an orthogonal matrix with unit condition number, while the triangular matrix R contains the ill-conditioned part. This approach was previously employed in Ref. [31], and we select the QR decomposition over a singular value decomposition due to its significantly lower computational cost.

Applying this procedure to the sausage of matrices, as shown in Algorithm 1, yields

$$QR = A. \quad (7)$$

Rewriting the determinant from Eq. (5) in terms of Q and R ,

$$\det (\mathbb{1} + QR) = \det Q \cdot \det (Q^T + R), \quad (8)$$

enables a numerically stable evaluation. The same technique can also be used to compute the inverse of $\mathbb{1} + QR$, which is required for several observables, including the gradient $\frac{\partial S}{\partial \phi_{xt}}$ used in HMC and in the training loop of the NF.

Algorithm 1 Stable multiplication of a chain of matrices using QR decomposition

```

1:  $Q \leftarrow \mathbb{1}_{N_x \times N_x}$ 
2:  $R \leftarrow \mathbb{1}_{N_x \times N_x}$ 
3: for each  $A_t$  in  $[A_1, A_2, \dots, A_{N_t}]$  do
4:    $R \leftarrow R \cdot A_t$ 
5:    $Q_{\text{help}}, R \leftarrow \text{QRDecomposition}(R)$ 
6:    $Q \leftarrow Q \cdot Q_{\text{help}}$ 
7: end for

```

4. Methods

4.1 Non-Equilibrium Markov Chain Monte Carlo

NE-MCMC is based on Jarzynski's equality [32], which relates the free-energy difference ΔF between an initial and a final macrostate of a statistical system undergoing an out-of-equilibrium evolution to an ensemble average,

$$\langle e^{-W} \rangle_f = e^{-\Delta F}. \quad (9)$$

Here, W denotes the work performed on the system during the non-equilibrium process, and $\langle \cdot \rangle_f$ indicates an average over all possible non-equilibrium trajectories.

Starting from an equilibrium Boltzmann distribution, a non-equilibrium Markov chain is constructed by updating the configuration using transition probabilities $P_{s(n)}$. The latter depends on a parameter s of the action, which is varied at each step n of the evolution. In particular, $P_{s(n)}(z_{n-1} \rightarrow z_n)$, used to generate the configuration z_n from z_{n-1} , satisfies detailed balance for the action $S_{s(n)}$ with the value of s at the n -th step. In this framework, the work W is defined as the cumulative change of the action induced by variations of s at fixed configurations,

$$W = \sum_{n=0}^{n_{\text{step}}-1} (S_{s(n+1)}[z_n] - S_{s(n)}[z_n]), \quad (10)$$

where n_{step} is the number of Monte Carlo steps along the evolution.

It is important to notice that the system is never thermalized after the first step. Equilibrium expectation values are instead recovered via a reweighting formula,

$$\langle O \rangle_p = \frac{\langle O e^{-W} \rangle_f}{\langle e^{-W} \rangle_f}, \quad (11)$$

where $\langle O \rangle_p$ denotes the expectation value with respect to the target distribution $p \propto e^{-S_s(n_{\text{step}})}$.

In the case of the Hubbard model, it is natural to define

$$S_s[\phi] = \sum_{x,t \in \Lambda} \frac{\phi_{xt}^2}{2\delta U} - s (\log \det M[\phi] + \log \det M[-\phi]) \quad (12)$$

with $s \in [0, 1]$. For $s = 0$, the distribution is Gaussian, allowing trivially ergodic sampling, while the full action in Eq. (2) is recovered at $s = 1$. In the NE-MCMC simulations, we interpolate linearly from $s(0) = 0$ to $s(n_{\text{step}}) = 1$.

4.2 (Stochastic) Normalizing Flows

SNFs, introduced by Wu et al. [33], combine deterministic NFs with stochastic NE-MCMC updates. A NF $f_n : z_n \rightarrow f_n(z_n)$ induces a transformation of probability densities according to

$$\log q_{n+1}(f_n(z_n)) = \log q_n(z_n) - \log \det \left| \frac{\partial f_n(z_n)}{\partial z} \right|, \quad (13)$$

where the NF allows an easily tractable Jacobian determinant that encodes the change in volume. A SNF is constructed by alternating blocks of flow transformations and NE-MCMC updates. The resulting chain can be represented as

$$z_0 \xrightarrow{\text{NF}} f_0(z_0) \xrightarrow{\text{MC}} z_1 \xrightarrow{\text{NF}} f_1(z_1) \xrightarrow{\text{MC}} z_2 \xrightarrow{\text{NF}} \dots \xrightarrow{\text{MC}} z_{n_{\text{step}}} \equiv \phi. \quad (14)$$

The output density $q_{n_{\text{step}}}$ is obtained by combining the change-of-variables formula in Eq. (13) with the accumulated work in Eq. (10),

$$\log q_{n_{\text{step}}}(\phi) = \log q_0(z_0) - W - \sum_{n=0}^{n_{\text{step}}-1} \log \det \left| \frac{\partial f_n(z_n)}{\partial z} \right|. \quad (15)$$

Training of the SNF is performed by minimizing the reverse Kullback–Leibler divergence [34], given by

$$\text{KL}(q_{n_{\text{step}}} || p) = \langle S_{S(n_{\text{step}})}[\phi] + \log q_{n_{\text{step}}}(\phi) \rangle_{\phi \sim q_{n_{\text{step}}}}. \quad (16)$$

5. Results

In this section, we present our results on the numerical stability of the action and the scalability of NFs, SNFs and NE-MCMC.

5.1 Computational Cost of the Action

As discussed in Sec. 2, the determinant of the fermion matrix can be expressed using the sausage of matrices. This formulation exploits the sparsity of the full fermion matrix, significantly reducing both computational cost and memory consumption. A scaling analysis of these quantities is shown in Fig. 1, while the key scaling behavior is summarized in Table 1. Using the sausage formalism reduces the computational cost by a factor of N_t^2 and memory usage by a factor of N_t .

5.2 Numerical Stability of the Action at Low Temperatures

The numerical instability of the action in the sausage formalism introduces a bias that becomes more severe at larger inverse temperatures β and larger numbers of temporal sites N_t . To quantify the numerical precision, we define

$$\text{Precision} \equiv \sqrt{\frac{1}{N_{\text{cfg}}} \sum_{i=1}^{N_{\text{cfg}}} \left(S[\phi^{(i)}] - S[\phi_r^{(i)}] \right)^2}, \quad (17)$$

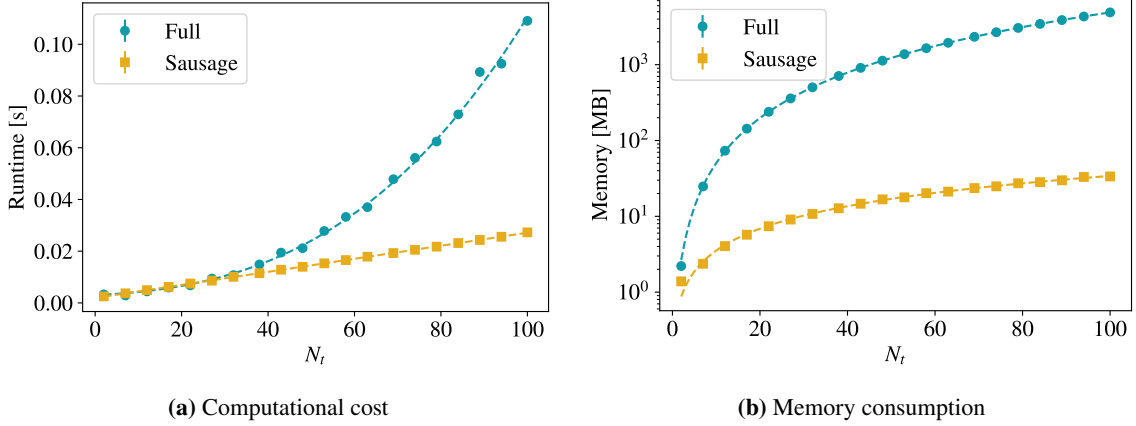


Figure 1: Scaling of computational cost (a) and memory consumption (b) for the full fermion matrix (blue) and the sausage formalism (orange) for different N_t at fixed $N_x = 4$. Dashed lines correspond to fits based on the scaling behavior from Table 1. Error bars are smaller than the marker size.

	Full	Sausage
Computational cost	$N_x^3 N_t^3$	$N_x^3 N_t$
Memory consumption	$N_x^2 N_t^2$	$N_x^2 N_t$

Table 1: Scaling behavior of computational cost and memory consumption for the full fermion matrix and the sausage formalism. Both quantities improve in terms of N_t .

which measures the root-mean-square difference between the action computed for a configuration ϕ and a time-shifted configuration ϕ_r , with $(\phi_r)_{x,t} \equiv \phi_{x,t+1}$. In exact computations, where instabilities are absent, this quantity would be zero due to time-translational symmetry.

Fig. 2 shows the numerical precision with and without QR decompositions as a function of β , while keeping δ fixed (thus increasing N_t). Without QR stabilization, the action becomes inaccurate for $\beta > 10$, whereas the QR decomposition maintains good precision up to $\beta = 100$. The vertical black line indicates graphene at room temperature, serving as a target for future simulations.

5.3 Scaling of the Training: (S)NFs

Here, we quantify the scaling of (S)NFs by measuring the computation cost, i.e., the training time, required to reach a given measure of quality. In this setup, we report the training cost until the sampler achieves a 20% acceptance rate in an accept-reject step. In Fig. 3(a), we compare RealNVP [35], a standard NF architecture, with an SNF having a fixed $n_{\text{step}} = 3$ at $U = 2$, $\beta = 5$, and $N_t = 20$. Both flows exhibit an exponential scaling with system size; however, the SNF shows an additional offset due to the extra computations of the action during the training loop.

Importantly, the present study is performed for a fixed architecture. Future work will explore how the training cost scales with the number of Monte Carlo steps and the number of parameters in the NF layers.

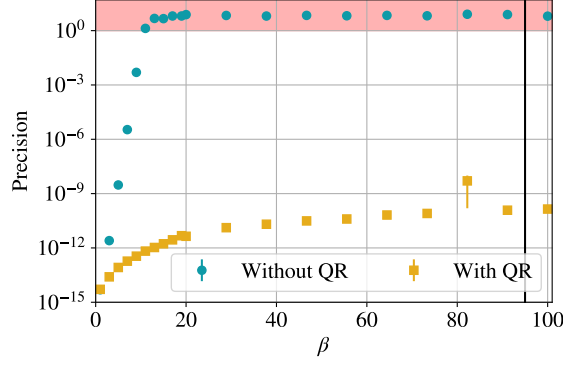


Figure 2: Precision of the Hubbard action in the sausage formalism with (orange) and without (blue) QR decomposition for different β . The black line marks graphene at room temperature, and the red shaded area indicates impracticable precision. The system is evaluated at $U = 2$, $N_x = 20$, and $\delta = 0.5$. For most data points, the error bars are smaller than the markers.

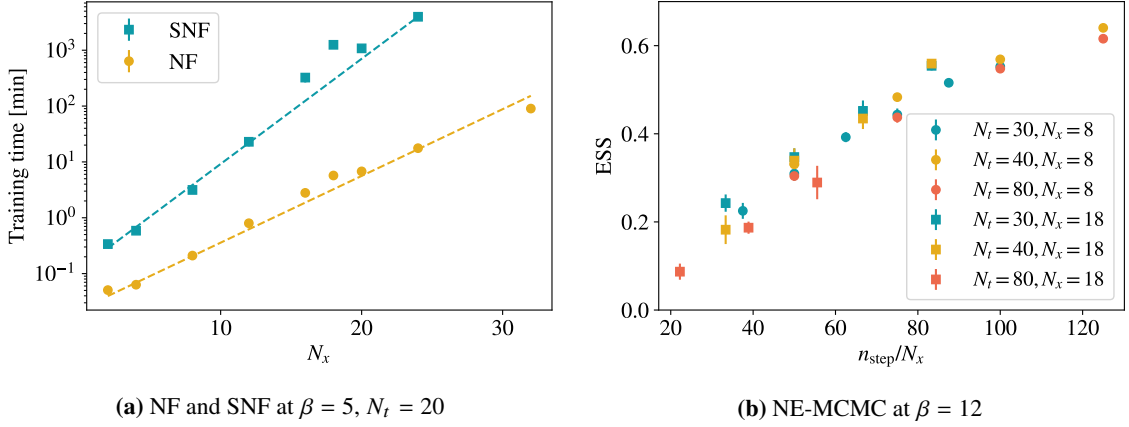


Figure 3: Scaling of the training cost of (S)NFs for a fixed architecture (left) and scaling of the sampling cost of NE-MCMC (right) at $U = 2$.

5.4 Scaling of the sampling: NE-MCMC

It has been established in various physical setups that NE-MCMC exhibits an approximately linear scaling with the lattice volume [21, 22]. More precisely, to maintain a given sampling quality (here quantified by the effective sampling size (ESS)) as the volume increases, the number of steps in the non-equilibrium evolution must grow linearly with the system size.

This scaling behavior is illustrated in Fig. 3(b), which shows the ESS of NE-MCMC simulations at $U = 2$, $\beta = 12$, and various lattice sizes $N_x \times N_t$. The data collapse indicates that the ESS is approximately a function of n_{step}/N_x , confirming the expected linear scaling with the spatial extent. Interestingly, there is no significant dependence on N_t . This is not in conflict with previous observations [21, 22], as space and time play distinct roles in the Hubbard model.

This scaling behavior is expected to carry over to SNFs, while the NF transformation reduces the computational cost. Consequently, SNFs offer a promising approach for simulating the Hubbard model on larger lattices, despite the increased training cost at fixed step size.

6. Conclusion

In this work, we have studied the numerical stability and scalability of NFs, SNFs, and NE-MCMC methods for the Hubbard model. An important step toward improved scalability is the *sausage* formalism, which enables linear scaling in the temporal lattice extent N_t when computing the fermion determinant. Numerical stability, even at low temperatures, can be ensured by performing QR decompositions between matrix multiplications.

Our numerical results show that NFs and SNFs, when operated with a fixed number of Monte Carlo steps, exhibit an exponential scaling of training time with the spatial lattice extent N_x . In contrast, NE-MCMC shows the expected linear scaling with the lattice volume and therefore currently provides the most scalable and robust sampling strategy for the Hubbard model.

For future work, we plan to conduct a detailed analysis of the scaling behavior of SNFs when the number of Monte Carlo steps is allowed to increase with N_x . While the present results indicate unfavorable scaling at fixed Monte Carlo depth, we expect that, once this restriction is lifted, both training and sampling costs of SNFs can achieve linear scaling with the lattice volume, similar to NE-MCMC. Under these conditions, SNFs have the potential to become a competitive and fully scalable tool for ergodic sampling of the Hubbard model.

Acknowledgments

The authors thank Kim A. Nicoli for discussions during the initial phase of this project. The authors gratefully acknowledge the access to the Marvin cluster of the University of Bonn. This project was supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) as part of the CRC 1639 NuMerIQS – project no. 511713970. LV acknowledges support from the SFT Scientific Initiative of INFN and the Simons Foundation grant 994300.

References

- [1] J. Hubbard, *Electron correlations in narrow energy bands*, *Proc. Roy. Soc. A* **276** (1963) 238.
- [2] D.P. Arovas, E. Berg, S.A. Kivelson and S. Raghu, *The Hubbard Model*, *Annu. Rev. Condens. Matter Phys.* **13** (2022) 239 [2103.12097].
- [3] J.-L. Wynen, E. Berkowitz, C. Körber, T.A. Lähde and T. Luu, *Avoiding Ergodicity Problems in Lattice Discretizations of the Hubbard Model*, *Phys. Rev. B* **100** (2019) 075141 [1812.09268].
- [4] D. Rezende and S. Mohamed, *Variational Inference with Normalizing Flows*, in *Proceedings of the 32nd International Conference on Machine Learning*, vol. 37, pp. 1530–1538, PMLR, 2015 [1505.05770].
- [5] I. Kobyzev, S.J. Prince and M.A. Brubaker, *Normalizing Flows: An Introduction and Review of Current Methods*, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **43** (2021) 3964 [1908.09257].

- [6] A. van den Oord, N. Kalchbrenner and K. Kavukcuoglu, *Pixel Recurrent Neural Networks*, in *Proceedings of The 33rd International Conference on Machine Learning*, vol. 48 of *Proceedings of Machine Learning Research*, pp. 1747–1756, PMLR, 20–22 Jun, 2016 [1601.06759].
- [7] F. Noé, S. Olsson, J. Köhler and H. Wu, *Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning*, *Science* **365** (2019) eaaw1147 [1812.01729].
- [8] M.S. Albergo, G. Kanwar and P.E. Shanahan, *Flow-based generative models for Markov chain Monte Carlo in lattice field theory*, *Phys. Rev. D* **100** (2019) 034515 [1904.12072].
- [9] K.A. Nicoli, C.J. Anders, L. Funcke, T. Hartung, K. Jansen, P. Kessel et al., *Estimation of Thermodynamic Observables in Lattice Field Theories with Deep Generative Models*, *Phys. Rev. Lett.* **126** (2021) 032001 [2007.07115].
- [10] K. Cranmer, G. Kanwar, S. Racanière, D.J. Rezende and P.E. Shanahan, *Advances in machine-learning-based sampling motivated by lattice quantum chromodynamics*, *Nature Reviews Physics* **5** (2023) 526 [2309.01156].
- [11] D. Wu, L. Wang and P. Zhang, *Solving Statistical Mechanics Using Variational Autoregressive Networks*, *Phys. Rev. Lett.* **122** (2019) 080602 [1809.10606].
- [12] K.A. Nicoli, S. Nakajima, N. Strodthoff, W. Samek, K.-R. Müller and P. Kessel, *Asymptotically unbiased estimation of physical observables with neural samplers*, *Phys. Rev. E* **101** (2020) 023304 [1910.13496].
- [13] M. Caselle, E. Cellini and A. Nada, *Sampling the lattice Nambu-Goto string using Continuous Normalizing Flows*, *JHEP* **02** (2024) 048 [2307.01107].
- [14] M. Caselle, E. Cellini and A. Nada, *Numerical determination of the width and shape of the effective string using Stochastic Normalizing Flows*, *JHEP* **02** (2025) 090 [2409.15937].
- [15] N. Gebauer, M. Gastegger and K. Schütt, *Symmetry-adapted generation of 3d point sets for the targeted discovery of molecules*, in *Advances in Neural Information Processing Systems*, vol. 32, 2019 [1906.00957].
- [16] N.W.A. Gebauer, M. Gastegger, S.S.P. Hessmann, K.-R. Müller and K.T. Schütt, *Inverse design of 3d molecular structures with conditional generative neural networks*, *Nature Commun.* **13** (2022) 973 [2109.04824].
- [17] P. Białaś, P. Korcyl, T. Stebel and D. Zapolski, *Rényi entanglement entropy of a spin chain with generative neural networks*, *Phys. Rev. E* **110** (2024) 044116 [2406.06193].
- [18] A. Bulgarelli, E. Cellini, K. Jansen, S. Kühn, A. Nada, S. Nakajima et al., *Flow-Based Sampling for Entanglement Entropy and the Machine Learning of Defects*, *Phys. Rev. Lett.* **134** (2025) 151601 [2410.14466].
- [19] M. Caselle, G. Costagliola, A. Nada, M. Panero and A. Toniato, *Jarzynski’s theorem for lattice gauge theory*, *Phys. Rev. D* **94** (2016) 034503 [1604.05544].

- [20] C. Bonanno, A. Nada and D. VDACCHINO, *Mitigating topological freezing using out-of-equilibrium simulations*, *JHEP* **04** (2024) 126 [2402.06561].
- [21] A. Bulgarelli, E. Cellini and A. Nada, *Scaling of stochastic normalizing flows in SU(3) lattice gauge theory*, *Phys. Rev. D* **111** (2025) 074517 [2412.00200].
- [22] C. Bonanno, A. Bulgarelli, E. Cellini, A. Nada, D. Panfalone, D. VDACCHINO et al., *Scaling flow-based approaches for topology sampling in SU(3) gauge theory*, 2510.25704.
- [23] J.P. Nilmeier, G.E. Crooks, D.D.L. Minh and J.D. Chodera, *Nonequilibrium candidate monte carlo is an efficient tool for equilibrium simulation*, *Proceedings of the National Academy of Sciences* **108** (2011) E1009 [1105.2278].
- [24] M. Caselle, E. Cellini, A. Nada and M. Panero, *Stochastic normalizing flows as non-equilibrium transformations*, *JHEP* **07** (2022) 015 [2201.08862].
- [25] D. Schuh, J. Kreit, E. Berkowitz, L. Funcke, T. Luu, K.A. Nicoli et al., *Simulating Correlated Electrons with Symmetry-Enforced Normalizing Flows*, 2506.17015.
- [26] J. Kreit, D. Schuh, K.A. Nicoli and L. Funcke, *SESaMo: Symmetry-Enforcing Stochastic Modulation for Normalizing Flows*, 2505.19619.
- [27] H.F. Trotter, *On the product of semi-groups of operators*, *Proc. Amer. Math. Soc.* **10** (1959) 545.
- [28] M. Suzuki, *Relationship between d-Dimensional Quantal Spin Systems and (d+1)-Dimensional Ising Systems: Equivalence, Critical Exponents and Systematic Approximants of the Partition Function and Spin Correlations*, *Prog. Theor. Phys.* **56** (1976) 1454.
- [29] J. Hubbard, *Calculation of Partition Functions*, *Phys. Rev. Lett.* **3** (1959) 77.
- [30] W. Gander, *Algorithms for the qr decomposition*, *Res. Rep* **80** (1980) 1251.
- [31] C. Bauer, *Fast and stable determinant quantum monte carlo*, *SciPost Phys. Core* **2** (2020) 011 [2003.05286].
- [32] C. Jarzynski, *Nonequilibrium Equality for Free Energy Differences*, *Phys. Rev. Lett.* **78** (1997) 2690 [cond-mat/9610209].
- [33] H. Wu, J. Köhler and F. Noe, *Stochastic Normalizing Flows*, in *Advances in Neural Information Processing Systems*, vol. 33, pp. 5933–5944, 2020 [2002.06707].
- [34] S. Kullback and R.A. Leibler, *On information and sufficiency*, *The annals of mathematical statistics* **22** (1951) 79.
- [35] L. Dinh, J. Sohl-Dickstein and S. Bengio, *Density estimation using Real NVP*, *ICLR* (2017) [1605.08803].