ISSN 0027-1349, Moscow University Physics Bulletin, 2024, Vol. 79, Suppl. 2, pp. S639–S646. © Allerton Press, Inc., 2024.

Application of Neural Networks for Path Integrals Computation in Relativistic Quantum Mechanics

D. V. Salnikov[1,2*], V. V. Chistiakov[1], A. V. Vasiliev[1], and A. S. Ivanov[1]

[1] *Lomonosov Moscow State University, Moscow, Russia*
[2] *Institute for Nuclear Research, Russian Academy of Sciences, Moscow, Russia*
Received October 1, 2024; revised October 10, 2024; accepted October 20, 2024

Abstract—In quantum theory, the expectation value of an observable can be represented as a path integral. In general, it cannot be computed analytically. There are various approximate methods of lattice calculations, for example, the Monte Carlo method. Currently, an approach to solving this problem using neural networks is being developed. In our research, we calculated path integrals in several models of relativistic quantum mechanics using the normalizing flows algorithm. For fast calculations with high accuracy, this algorithm was used in conjunction with the Markov chain generation method.

Keywords: path integral, monte carlo calculations, normalizing flows model

DOI: 10.3103/S0027134924702096

1. INTRODUCTION

In quantum field theory, the amplitudes of processes are expressed in terms of reduction formulae as the expectation values of Heisenberg operators over the ground state [1]. Following this approach, it is necessary to compute the Green functions, defined as

$$G(t_1, ..., t_n) = \langle 0|x(t_n) \dots x(t_1)|0\rangle,$$

where x denotes the dynamic variable of a theory and $\{t_i\}_{i=1}^n$ presents an ordered set of time points.

We calculate the ground state expectation value of an observable F for a one-dimensional quantum mechanics system. We consider the Hamiltonian of the following form:

$$H = T(p) + V(x),$$

where $T(\cdot)$ and $V(\cdot)$ are the kinetic and potential energies.

The ground state expectation value can be evaluated as the Gibbs average in the low temperature limit:

$$\langle F\rangle_\beta = \frac{\sum_n e^{-\beta(E_n - E_0)}\langle n|F|n\rangle}{\sum_n e^{-\beta(E_n-E_0)}}, \qquad (1)$$

$$\langle 0|F|0\rangle \simeq \langle F\rangle_\beta, \quad \beta(E_1 - E_0) \gg 1,$$

where Z is the partition function, E_n is the energy eigenvalue corresponding to the Hamiltonian eigenstate $|n\rangle$ and β is the inverse temperature. The natural system of units $\hbar = c = 1$ is used.

On the other hand, the average (1) can be represented as a path integral with periodic boundary conditions $x(0) = x(\beta)$ [2]:

$$\langle F\rangle_\beta = \frac{1}{Z}\int D[x(t)]\, e^{-\mathcal{S}[x(t)]} F[x(t)], \qquad (2)$$

where $\mathcal{S}[\cdot]$ is the Euclidean action of the system and t is the imaginary time related to the physical one by the Wick rotation $t_{\rm ph} = -it$.

Analytical calculation of the path integral (2) is possible for a limited class of specific models, in other cases, various approximate methods are accomplished. The perturbation theory is one of such methods. However, in particular physical theories (for example QCD) the perturbation theory expansion is not applicable due to the large coupling constant of the strong interaction. Another approach to computation of the path integral (2) is a lattice field theory. In this framework, the path integral approximates to a finite dimensional integral.

The lattice calculations imply splitting of a segment $[0, \beta]$ into N nodes with step of $\tau = \frac{\beta}{N}$ and correspondence between continuous trajectory of $x(t)$ and vector $x_i = x(t_i)$ of the space $\mathbb{R}^N$.

*E-mail: salnikov@inr.ru

The path integral (2) is approximated by the following N-dimensional integral,

$$\langle F\rangle_\beta = \int d^N x\, P(x)\, F(x) + O(\tau^2), \quad x \in \mathbb{R}^N,$$
$$P(x) = \frac{e^{-S(x)}}{Z}, \quad Z = \int d^N x\, e^{-S(x)}, \qquad (3)$$

where $P(x)$ is the probability density function and $S(\cdot)$ is the lattice Euclidean action.

In particular, Green's two-point function

$$G(s) = \langle 0|x(s)x(0)|0\rangle, \quad s > 0 \qquad (4)$$

reduces to

$$G_{\text{latt}}(s) = \int d^N x\, P(x) \left[\frac{1}{N}\sum_{i=1}^{N} x_i x_{i+s}\right]. \qquad (5)$$

The integral (3) is calculated by the Monte Carlo method. We generate a sample of trajectories $\{x^{(k)}\}_{k=1}^{M}$ containing N-dimensional vectors distributed according to the probability density function $P(x)$ [3]. Thus, the integral (3) is approximately equal to the sample average:

$$\int d^N x\, P(x)\, F(x) \simeq \frac{1}{M}\sum_{k=1}^{M} F(x^{(k)}).$$

Therefore, the computation of the expectation value of an observable F is reduced to the problem of multidimensional vectors (trajectories) generating with the target distribution $P(x)$. The commonly used approach to solving this problem is the Metropolis algorithm [3].

Lattice Monte Carlo calculation of path integral is significant for the modern quantum field theory [4] and condensed matter physics [5] where the systems under consideration have numerous dynamic variables and other approaches are not applicable.

However, the generation of trajectories using the Metropolis algorithm requires large computational and time resources in real problems, so the actual task is to build a faster algorithm for lattice calculations. Recently, a new approach where calculations are accelerated using a neural network generative algorithm of normalizing flows has been actively developing.

This approach has been demonstrated for calculations in various quantum field models: in the precisely solvable Ising model [6], in the model ϕ^4 [7] and QCD [8]. A two-dimensional lattice of $N \times N$ was studied with a relatively small $N \sim 16$ in these papers. A two-dimensional lattice $N \times N$ can be regarded as a system with N degrees of freedom with division of time interval into N nodes.

The main task of the work is to apply generative ML algorithms to problems of one-dimensional quantum mechanics and to evaluate their effectiveness for lattice calculations with numerous time interval partitioning nodes for various Hamiltonians. Unlike the mentioned works, we consider lattices not $N \times N$, but $1 \times N$, however, with many nodes $N = 256$ and a more complex target distribution $P(x)$. Since we do not consider systems with a large number of degrees of freedom, the results of this work cannot be directly applied to real calculations.

In a recent paper [9], we investigated quantum mechanical systems with one degree of freedom with a nonrelativistic Hamiltonian of the form

$$H = \frac{p^2}{2} + V(x)$$

and numerous nodes in time $N = 256$ for some set of potentials $V(x)$. In the harmonic oscillator model, the generative model accurately reproduced the target distribution; in models with higher potentials, the generative model allowed the Metropolis algorithm to be accelerated by more than 10 times. In the first paper, we investigate one-dimensional quantum mechanical systems with a more complex relativistic Hamiltonian

$$H = \sqrt{p^2 + m^2} - m + V(x)$$

for different potentials $V(x)$.

The consideration of quantum mechanical systems described by the relativistic hamiltonian is motivated by a number of practical problems in physics of elementary particles and condensed matter. The choice of the potentials $V(x)$ conditioned by the most commonly used models of interaction. As we believe, the specific type of the potential within a framework of this research is not important.

2. A RELATIVISTIC PATH INTEGRAL ON A LATTICE

An action on a lattice $S(x)$ can be represented as follows [3]

$$S(x) = \sum_{i=1}^{N} \tau\Theta(x_{i+1} - x_i) + \tau V(x_i),$$

where function of $\Theta(\cdot)$ is defined as

$$\exp\left(-\tau\Theta(\xi)\right) = \int_{-\infty}^{+\infty} \frac{dp}{2\pi} \exp\left(-\tau T(p) + ip\xi\right). \quad (6)$$

Consider relativistic systems with the following kinetic term

$$T(p) = \sqrt{p^2 + m^2} - m.$$

In this case [10], the function $\Theta(\cdot)$ reads as follows,

$$\Theta(\xi) = -\frac{1}{\tau}\ln\left[\frac{m^2\tau}{\pi}\frac{K_1(\eta)}{\eta}\right] - m, \qquad (7)$$

where η is defined as

$$\eta = m\tau\sqrt{1+\left(\frac{\xi}{\tau}\right)^2}.$$

In the limits of low and large masses, the function (7) has the following forms

$$\Theta(\xi) = \frac{m\xi^2}{2\tau^2} + \Theta(0), \quad m\tau \gg 1;$$

$$\Theta(\xi) = \frac{1}{\tau}\ln\left[1+\left(\frac{\xi}{\tau}\right)^2\right] + \Theta(0), \quad m\tau \ll 1.$$

3. NORMALIZING FLOWS

The Hamiltonian H and probability density function $P(x)$ for the nonrelativistic harmonic oscillator are

$$H = \frac{p^2}{2} + \frac{x^2}{2},$$

$$P(x) = \frac{1}{Z}\exp\left[-\frac{1}{2}(Ax,x)\right],$$

where A is a positive-definite symmetric matrix. It's not necessary to use complicated algorithms for generating a sample of trajectories with a target distribution $x \sim P(x)$. It is enough to apply a linear map $x = Cz$ with a set of vectors $\{z^{(k)}\} \sim \mathcal{N}^N(0,1)$ whose coordinates are independent and have a standard normal distribution [9].

In more complicated models, the map $x = g(z)$, which transform a sample of normally distributed vectors $z \sim \mathcal{N}^N(0,1)$ into a sample of vectors with the target distribution, $x \sim P(x)$ will be nonlinear and generally not expressed analytically. However, the problem of approximating this map by a function from some multiparametric family can be considered [11].

We search for g in the form of a composition of affine transformations:

$$g = \mathcal{A}_n \circ ... \circ \mathcal{A}_{i+1} \circ \mathcal{A}_i \circ ... \circ \mathcal{A}_1.$$

Each of $\mathcal{A}_i$ is defined as

$$\mathcal{A}(u) = u, \quad [\mathcal{A}(v)]_k = e^{\theta_{1k}(u)}v_k + \theta_{2k}(u), \qquad (8)$$

where the vector z is the direct sum of vectors of equal length $z = u \oplus v$ and $\theta : \mathbb{R}^N \to \mathbb{R}^{N/2}$ is a some multiparametric map $\theta(u) \equiv \theta(u|w)$, where $w \in \mathbb{R}^K$ are parameters. It is of the form $\theta = \sigma(w_m\sigma(...w_1u)))$, where w_k are arbitrary matrices, σ is a nonlinear transformation (activation function, LeakyReLU in this case). Thus, map of θ is a fully connected neural network.

To accelerate the neural network learning, an orthogonal transformation is also used,

$$x = Og(z).$$

The matrix O can be constructed in the following way [9].

We denote the generator of the group $\mathbb{Z}_N$ in regular representation as T

$$T_{i,k} = \delta_{i+1,k}; \quad T^\dagger = T^{-1}; \quad T^N = I.$$

The transformation O leads T to a block-diagonal form (the regular representation of $\mathbb{Z}_N$ is decomposed into the sum of irreducible ones):

$$O^\dagger TO = 1\bigoplus R_1\bigoplus ...\bigoplus R_{N/2-1}\bigoplus(-1),$$

where R_k is the rotation matrix by angle $\frac{2\pi k}{N}$.

We constructed the loss function as follows. Consider the Kullback–Leibler divergence $D_{\mathrm{KL}}(\cdot|\cdot)$, defined as

$$D_{\mathrm{KL}}(p|q) = \int dx\, p(x)\ln\frac{p(x)}{q(x)}.$$

It has the following properties:

$$D_{\mathrm{KL}}(p|q) \geq 0, \quad D_{\mathrm{KL}}(p|q) = 0 \Leftrightarrow p(x) = q(x).$$

Thus, $D_{\mathrm{KL}}(p|q)$ can be regarded as the distance between distributions p and q. We estimate the probability distribution function of the generated dataset using substitution of the initial normally distributed vectors:

$$P_g(x) = r(z)\left|\det\frac{\partial g}{\partial z}\right|^{-1}, \quad z = g^{-1}(x),$$

$$r(z) = (2\pi)^{-N/2}\exp\left[-\frac{1}{2}(z,z)\right]. \qquad (9)$$

We use shifted KL-divergence as a loss function [6, 7]

$$L[w] = D_{\mathrm{KL}}(P_g|P) - \ln Z.$$

Taking into account (9), we rewrite it in more explicit form

$$L[w] = \frac{1}{M}\sum_{k=1}^{M}\left\{S(g(z_k|w)) - \ln\left|\det\frac{\partial g(z_k|w)}{\partial z}\right|\right\}.$$

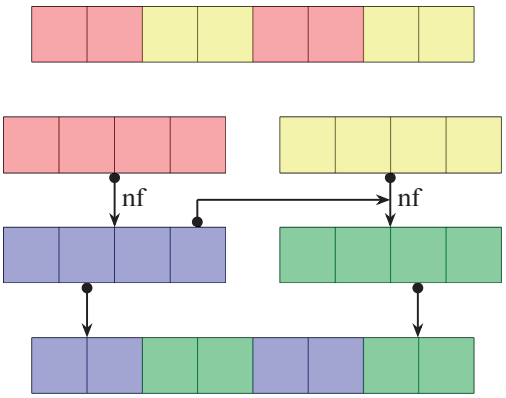

Fig. 1. Multiscale architecture. We divide the input vector $z = (z_1, ..., z_8)$ (8-dimensional in this example) into two parts, $z_{(1)} = (z_1, z_2, z_5, z_6)$ and $z_{(2)} = (z_3, z_4, z_7, z_8)$. Then, we transform them to $y_{(1)} = g_1(z_{(1)})$ and $y_{(2)} = g_2(z_{(2)}|y_{(1)})$, and concatenate $y_{(1)} = (y_1, y_2, y_5, y_6)$ and $y_{(2)} = (y_3, y_4, y_7, y_8)$ into vector $y = (y_1, y_2, y_3, y_4, y_5, y_6, y_7, y_8)$. The resulting vector is $x = Oy$.

4. MULTISCALE ARCHITECTURE

To accelerate neural network learning and reduce the number of parameters, we complicated the architecture of the model [6]. Namely, we used a multiscale architecture (Fig. 1). The number of nodes was considered to be equal to an integer power of two $N = 2^K$. We divided the subspace $\mathbb{R}^N$ into the direct sum of subspaces of smaller dimension for some number $m \leq K - 2$.

$$\mathbb{R}^{2^K} = \mathbb{R}^{2^m} \bigoplus \mathbb{R}^{2^m} \bigoplus \mathbb{R}^{2^{m+1}} \bigoplus ... \bigoplus \mathbb{R}^{2^{K-1}},$$

$$z = z_{(1)} \oplus z_{(2)} \oplus ... \oplus z_{(K-m)},$$

$$y = y_{(1)} \oplus y_{(2)} \oplus ... \oplus y_{(K-m)}.$$

The transformations g_s act in each of the subspaces

$$y_{(s)} = g_s(z_{(s)}, p_{(s)}),$$

where variables of $p_{(s)}$ contain coordinates on more "rough" scales and are defined as

$$p_{(2)} = y_{(1)}, \quad p_{(s)} = p_{(s-1)} \oplus y_{(s-1)}.$$

The variables $p_{(s)}$ are included in the transformation of g_s in the form of additional arguments of the neural network θ so that (8) is modified as follows

$$[\mathcal{A}(v)]_k = e^{\theta_{1k}(u \oplus p_{(s)})} v_k + \theta_{2k}(u \oplus p_{(s)}).$$

Finally, we concatenate $y_{(s)}$ into vector y and obtain the output vector using the orthogonal transformation $x = Oy$.

5. RESULTS

We have examined three models with relativistic kinetic term:

1. relativistic oscillator (Fig. 2):

$$H = \sqrt{p^2 + m^2} - m + \frac{m\omega^2 x^2}{2};$$

2. relativistic double well model (Fig. 3):

$$H = \sqrt{p^2 + m^2} - m + g(x^2 - x_0^2)^2;$$

3. relativistic Morse model (Fig. 4):

$$H = \sqrt{p^2 + m^2} - m + \frac{1}{2}\left[\left(e^{-\alpha x} - 1\right)^2 - 1\right].$$

The trajectories generated by the neural network approximate the target distribution of $P(x)$ with low accuracy. To clarify the result, it is necessary to take the generated trajectories as the initial ones for the Metropolis algorithm and accomplish several iterations (sweeps). The criterion for termination of the algorithm is the absence of a change in the average value of the calculated observable over several iterations. In this case, trajectories are considered to be thermalized. We compared the results of applying the Metropolis algorithm to trajectories generated by a neural network and to "cold" trajectories (with zero initial configuration: $x_i^{(k)} \equiv 0$). In the case of generated trajectories, the thermalization is on average 2–4 times faster. For various potentials and observables, the acceleration has different values. Thus, this approach is model depended.

We calculated the density matrix from the generated dataset (similarly as in [3]):

$$\rho(x, x) = \frac{1}{Z} \sum_n e^{-\beta E_n} |\psi_n(x)|^2.$$

It coincides with the ground state wave function under low temperature condition $\beta(E_1 - E_0) \gg 1$.

The thermal density matrix was also calculated in the double potential well model (Fig. 3f) In the case of this model, the coordinate distributions differ qualitatively at low and high temperatures. At low temperatures, the presence of two potential minima is essential, and at high temperatures $\beta g x_0^4 \ll 1$, the presence of two minima does not affect the coordinate distribution.

We compare the square of the wave function calculated from the generated trajectories and using the numerical solution of the Schrödinger equation. The best match is observed in the double well potential.

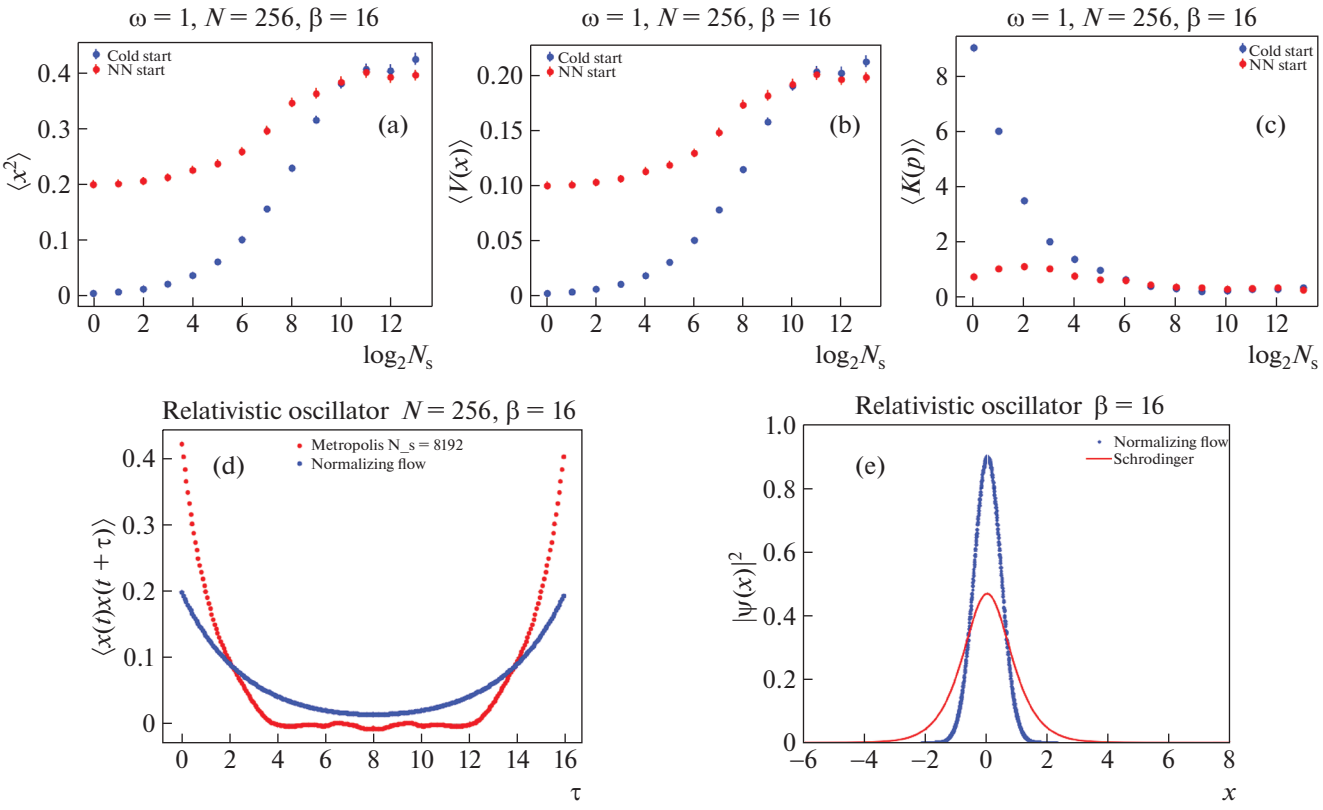

Fig. 2. Calculations were performed at the following values of the parameters: $m = \omega = 1$. The red dots correspond to the neural network approach, the blue dots correspond to the Metropolis algorithm. Figures present the value of (a) mean square coordinates $\langle x^2 \rangle$, (b) mean potential energy $\langle V(x) \rangle$, and (c) mean kinetic energy $\langle T(p) \rangle$ (c) on each sweep. The expectation values are plotted along the vertical axis, and the number of sweeps N_s on a logarithmic scale, is plotted horizontally. The figure with label (d) represents the two-point Green function calculated using the generated trajectories before (blue) and after (red) application of the Metropolis algorithm to them. The figure with label (e) compares the wave functions obtained by averaging over trajectories (blue) and the numerical solutions of the Shrödinger equation (red).

The generative model qualitatively reflects the dependence of the coordinate distribution on temperature.

The two-point Green function (4) represents dynamical properties of the system. We calculated it in path integral approach using (5). Another approach that can be used requires numerical computation of evolution operator $U(t)$

$$G(t) = \sum_n \langle 0|U^\dagger(t)x(0)U(t)|n\rangle\langle n|x(0)|0\rangle.$$

To estimate how the generative model represents dynamical properties of a system, we calculated the two-point Green function using generated trajectories. We also compared the results with the Metropolis algorithm approach (d). Despite the great difference between the Green function calculated using the generative model and the true one, there is a rapid decrease in the correlation of the coordinates of remote nodes in the models of the oscillator and the Morse potential.

The code and data are posted in the repository at the link [12].

6. DISCUSSION

The generative model approximates the target distribution of $P(x)$ with low accuracy, but this accuracy is sufficient to accelerate generation using an algorithm based on Markov chains several times. In nonrelativistic models with the kinetic term (2), which were studied in [9], it is possible to achieve greater generation accuracy.

The approximation of a path integral (2) by an integral (3) is more accurate the greater the number of nodes N. However, for models with a relativistic kinetic term (7), the quality of generation decreases with an increase in N (Fig. 5), and it cannot be improved by increasing the number of weights of the neural network. The reasons are not completely clear. This may be due to the behavior of function (7) at $\xi \gg \tau$, the presence of local minima in the loss

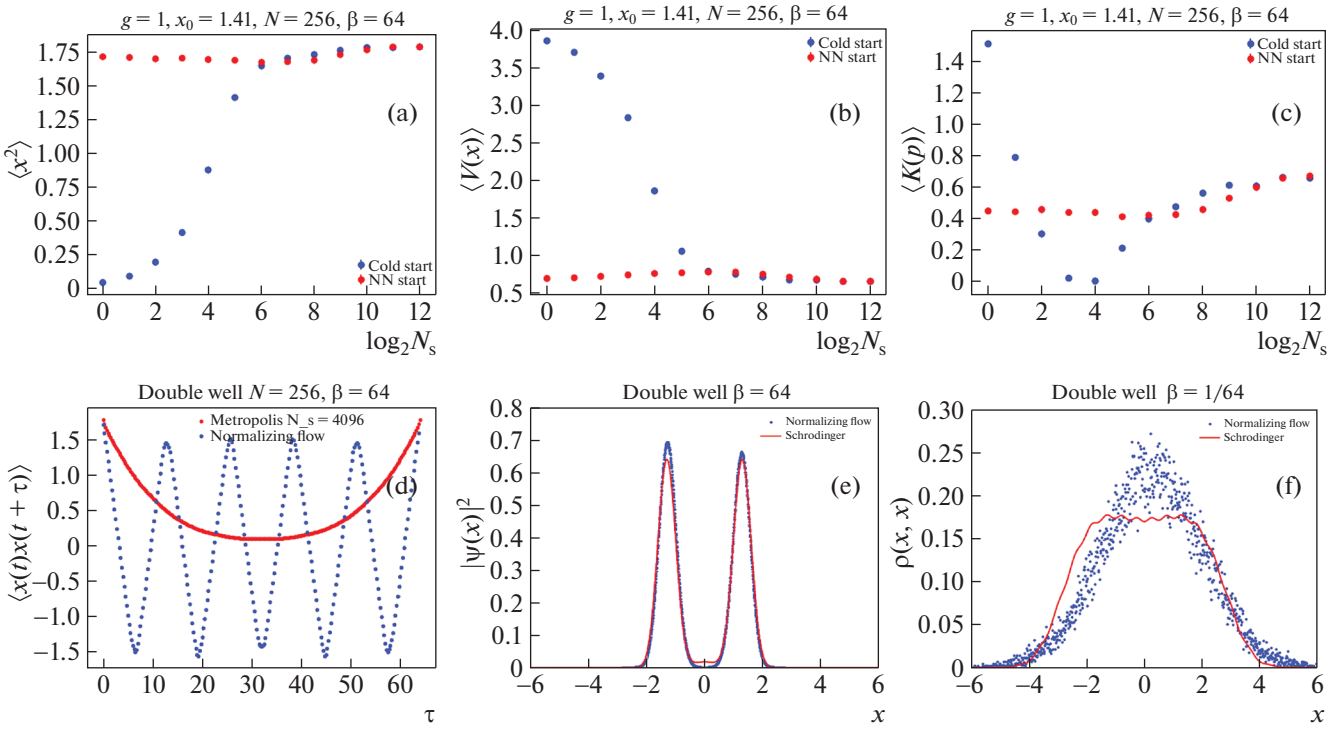

Fig. 3. Calculations were performed at the values of the parameters: $g = 1$, $x_0 = 1.41$. The red dots correspond to the neural network approach, the blue dots correspond to the Metropolis algorithm. Figures present the value of (a) mean square coordinates $\langle x^2 \rangle$, (b) mean potential energy $\langle V(x) \rangle$, and (c) and mean kinetic energy $\langle T(p) \rangle$ on each sweep. The expectation values are plotted along the vertical axis, and the number of sweeps N_s on a logarithmic scale is plotted horizontally. The figure with label (d) represents the two-point Green function calculated using the generated trajectories before (blue) and after (red) application of the Metropolis algorithm to them. The figure with label (e) compares the wave functions obtained by averaging over trajectories (blue) and the numerical solutions of the Shrödinger equation (red).

function. Future research is expected to solve this problem by improving the architecture of the generative model.

7. CONCLUSIONS

The neural network generative algorithm of normalizing flows allows generating trajectories with a distribution close to the target, which accelerates the calculation of the average values but this acceleration may depend significantly on the model. Generative model represents temperature dependence of density matrix (in the double well potential). In the models with complicated kinetic term, the quality of generation decreases significantly under rising of the number of nodes N. This dependence is not observed for the nonrelativistic (Gaussian) kinetic term [9].

We tested the approach on one-dimensional quantum mechanical systems, but the Monte Carlo method has a significant advantage in problems with numerous degrees of freedom. In the following papers, it is proposed to extend this approach to $(1 + d)$-dimensional quantum field theory.

APPENDIX

NUMERICAL SOLUTION OF THE SCHRÖDINGER EQUATION

Consider an algorithm for the numerical solution of the stationary Schrödinger equation:

$$H\psi = E\psi$$

with the Hamiltonian of the relativistic particle. We search for a solution of $\psi(x)$ in the form of linear combination of eigenfunctions of a nonrelativistic harmonic oscillator:

$$\psi(x) = \sum_k c_k \varphi_k(x), \qquad \text{(A1)}$$

$$\varphi_n(x) = \frac{1}{\pi^{1/4}} \frac{1}{\sqrt{2^n n!}} \exp\left(-\frac{x^2}{2}\right) H_n(x).$$

For a numerical solution, we limit ourselves to a finite number of $S \approx 30$ functions φ_k.

The operator H is approximated by a finite matrix with elements:

$$\mathcal{H}_{nm} = \int dx\, \varphi_n \left(\mathcal{F}^{-1} T(p) \mathcal{F} + V(x)\right) \varphi_m,$$

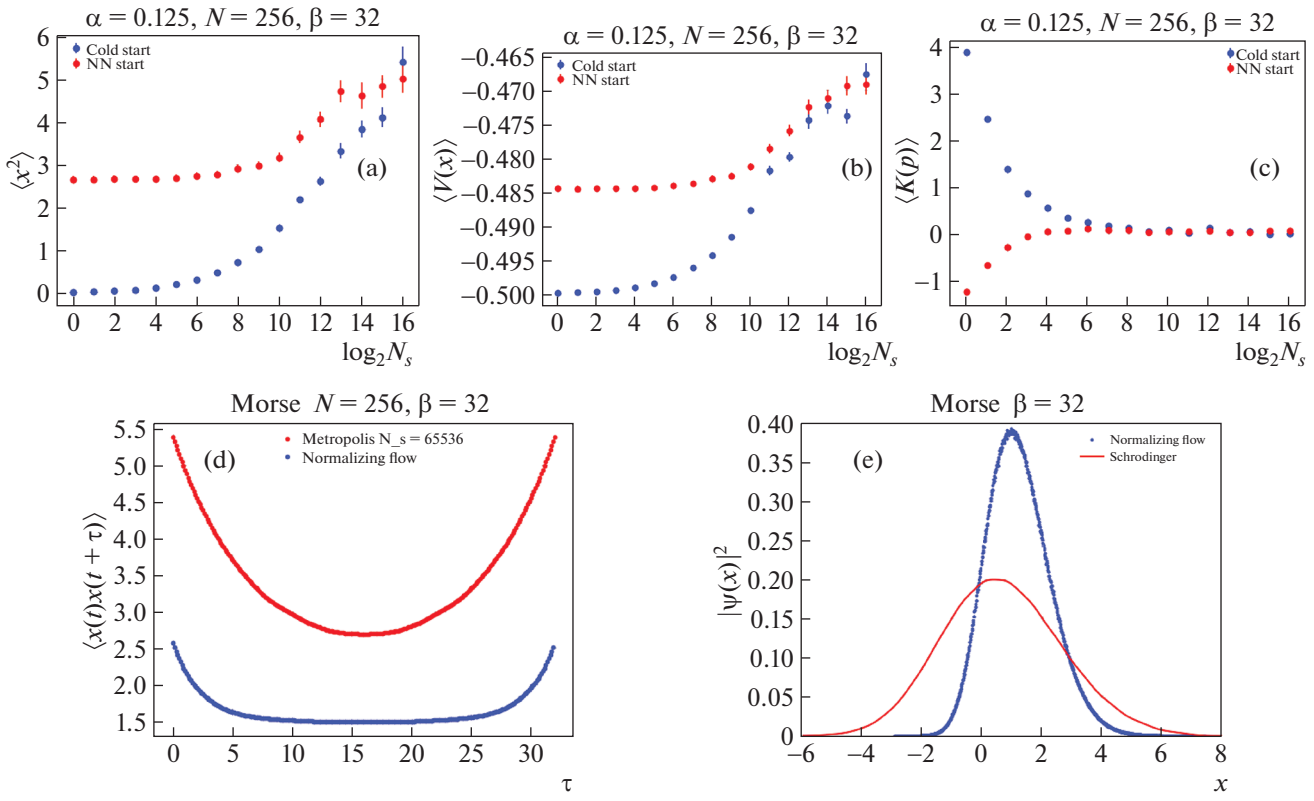

Fig. 4. Calculations were performed at the values of the parameter: $\alpha = 0.125$. The red dots correspond to the neural network approach, the blue dots correspond to the Metropolis algorithm. Figures present the value of (a) mean square coordinates $\langle x^2 \rangle$, (b) mean potential energy $\langle V(x) \rangle$, and (c) and mean kinetic energy $\langle T(p) \rangle$ on each sweep. The expectation values are plotted along the vertical axis, and the number of sweeps N_s on a logarithmic scale is plotted horizontally. The figure with label (d) represents the two-point Green function calculated using the generated trajectories before (blue) and after (red) application of the Metropolis algorithm to them. The figure with label (e) compares the wave functions obtained by averaging over trajectories (blue) and the numerical solutions of the Shrödinger equation (red).

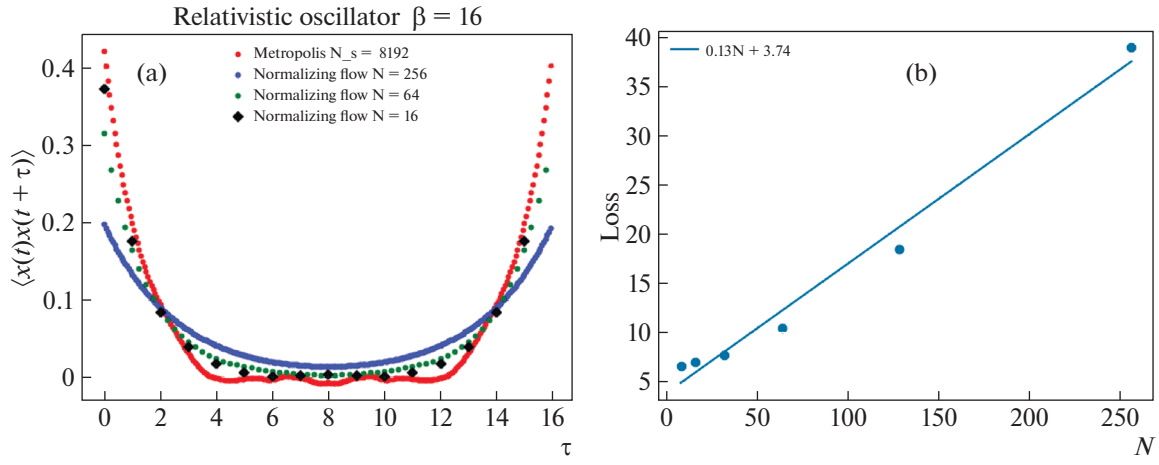

Fig. 5. (a) Two-point Green function for various number of nodes (namely, $N = 16, 64, 256$) and for the case of the Metropolis algorithm application, with the number of nodes equals to $N = 256$ and the number of sweeps equals to $N_s = 8192$. (b) Dependence of the loss function value on the number of nodes N in the case of neural network generation of trajectories.

where the operator $\mathcal{F}$ represents the Fourier transform:

$$\mathcal{F}\varphi \equiv \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} \varphi(x) e^{ipx}.$$

The expansion coefficients in (A1) are the eigenvectors of the matrix $\mathcal{H}$

$$\mathcal{H}c = Ec, \quad c \equiv ||c_1, ..., c_n||^T.$$

FUNDING

The works of DS and VC were supported by grants from the Foundation for the Development of Theoretical and Mathematical Physics "BASIS" nos. 23-2-2-40-1 and 23-2-9-18-1.

CONFLICT OF INTEREST

The authors of this work declare that they have no conflicts of interest.

REFERENCES

1. N. N. Bogoliubov and D. V. Shirkov, *Intoduction to the Theory of Quantized Fields*, Ed. by R. E. Marshak, Interscience Monographs in Physics and Astronomy, Vol. 3 (Interscience Publishers, New York, 1959).
2. J. Zinn-Justin, *Path Integrals in Quantum Mechanics*, Oxford Graduate Texts (Oxford University Press, Oxford, 2004). https://doi.org/10.1093/acprof:oso/9780198566748.001.0001
3. M. Creutz and B. Freedman, Annals of Physics **132**, 427 (1981). https://doi.org/10.1016/0003-4916(81)90074-9
4. S. Navas, C. Amsler, T. Gutsche, C. Hanhart, J. J. Hernández-Rey, C. Lourenço, A. Masoni, M. Mikhasenko, R. E. Mitchell, C. Patrignani, C. Schwanda, S. Spanier, G. Venanzoni, C. Z. Yuan, K. Agashe, G. Aielli, B. C. Allanach, J. Alvarez-Muñiz, M. Antonelli, E. C. Aschenauer, D. M. Asner, K. Assamagan, H. Baer, Sw. Banerjee, R. M. Barnett, L. Baudis, C. W. Bauer, J. J. Beatty, J. Beringer, A. Bettini, O. Biebel, K. M. Black, E. Blucher, R. Bonventre, R. A. Briere, A. Buckley, V. D. Burkert, M. A. Bychkov, R. N. Cahn, Z. Cao, M. Carena, G. Casarosa, A. Ceccucci, A. Cerri, R. S. Chivukula, G. Cowan, K. Cranmer, V. Crede, O. Cremonesi, G. D'Ambrosio, T. Damour, et al. (Particle Data Group Collaboration), Phys. Rev. D **110**, 30001 (2024). https://doi.org/10.1103/physrevd.110.030001
5. K. A. Velizhanin and A. Saxena, Phys. Rev. B **92**, 195305 (2015). https://doi.org/10.1103/physrevb.92.195305
6. S.-H. Li and L. Wang, Phys. Rev. Lett. **121**, 260601 (2018). https://doi.org/10.1103/physrevlett.121.260601
7. M. S. Albergo, G. Kanwar, and P. E. Shanahan, Phys. Rev. D **100**, 34515 (2019). https://doi.org/10.1103/physrevd.100.034515
8. M. S. Albergo, D. Boyda, D. C. Hackett, G. Kanwar, K. Cranmer, S. Racanière, D. J. Rezende, and P. E. Shanahan, arXiv Preprint (2021). https://doi.org/10.48550/arXiv.2101.08176
9. V. V. Chistiakov, A. S. Ivanov, D. V. Salnikov, and A. V. Vasiliev, Phys. Elem. Part. At. Nucl. (2025) (in review).
10. A. Ivanov, A. Novoselov, and O. Pavlovsky, Int. J. Mod. Phys. C **27**, 1650133 (2016). https://doi.org/10.1142/s0129183116501333
11. G. Papamakarios, E. Nalisnick, D. J. Rezende, S. Mohamed, and B. Lakshminarayanan, J. Mach. Learn. Res. **22**, 2617 (2021). https://jmlr.org/papers/v22/19-1028.html.
12. V. V. Chistiakov, NF_QFT (GitHub, 2024). https://github.com/Vsevolod2001/NF_QFT.