Check for updates

Efficient solutions of fermionic systems using artificial neural networks

Even M. Nordhagen[1], Jane M. Kim[2], Bryce Fore[3],
Alessandro Lovato[3] and Morten Hjorth-Jensen[2,4]*

[1]Department of Physics and Njord Center, University of Oslo, Oslo, Norway, [2]Department of Physics and Astronomy and Facility for Rare Isotope Beams, Michigan State University, East Lansing, MI, United States, [3]Physics Division, Argonne National Laboratory, Lemont, IL, United States, [4]Department of Physics and Center for Computing in Science Education, University of Oslo, Oslo, Norway

In this study, we explore the similarities and differences between variational Monte Carlo techniques that employ conventional and artificial neural network representations of the ground-state wave function for fermionic systems. Our primary focus is on shallow neural network architectures, specifically the restricted Boltzmann machine, and we examine unsupervised learning algorithms that are appropriate for modeling complex many-body correlations. We assess the advantages and drawbacks of conventional and neural network wave functions by applying them to a range of circular quantum dot systems. Our findings, which include results for systems containing up to 90 electrons, emphasize the efficient implementation of these methods on both homogeneous and heterogeneous high-performance computing facilities.

1 Introduction

Solving the Schrödinger equation for systems of many interacting bosons or fermions is classified as an NP-hard problem due to the complexity of the required many-dimensional wave function, resulting in an exponential growth of degrees of freedom. Reducing the dimensionalities of quantum mechanical many-body systems is an important aspect of modern physics, ranging from the development of efficient algorithms for studying many-body systems to exploiting the increase in computing power. To write software that can fully utilize the available resources has long been known to be an important aspect of these endeavors. Despite tremendous progress having been made in this direction, traditional many-particle methods, either quantum mechanical or classical ones, face huge dimensionality problems when applied to studies of systems with many interacting particles.

Over the last 2 decades, quantum computing and machine learning have emerged as some of the most promising approaches for studying complex physical systems where several length and energy scales are involved. Machine learning techniques and in particular neural-network quantum states Goodfellow et al. [1] have recently been applied to studies of many-body systems, see, for example, Refs. Carleo and Troyer [2]; Carrasquilla and Torlai [3]; Pfau et al. [4]; Calcavecchia et al. [5]; Carleo et al. [6]; Boehnlein et al. [7]; Adams et al. [8]; Lovato et al. [9], in various fields of physics and quantum chemistry, with very promising results. In many of these studies, results have aligned well with exact analytical solutions or are in close agreement with state-of-the-art quantum Monte Carlo calculations.

The variational and diffusion Monte Carlo algorithms are among the most popular and successful methods available for ground-state studies of quantum mechanical systems. They both rely on a suitable ansatz for the ground-state of the system, often dubbed the *trial wave function*, which is defined in terms of a set of variational parameters whose optimal values are found by minimizing the total energy of the system. Devising flexible and accurate functional forms for the trial wave functions requires prior knowledge and physical intuition about the system under investigation. However, for many systems we do not have this intuition, and as a result it is often difficult to define a good ansatz for the state function.

According to the universal approximation theorem, a deep feedforward neural network can represent any continuous function within a certain error Hornik et al. [10] — see also Refs. Murphy [11]; Hastie et al. [12]; Bishop [13]; Goodfellow et al. [1] for further discussions of deep learning methods. Since the variational state in principle can take any functional form, it is natural to replace the trial wave function with a neural network and treat it as a machine learning problem. This approach has been successfully implemented in recent works, see, for example, Refs. Pfau et al. [4]; Carleo and Troyer [2]; Cassella et al. [14]; Adams et al. [8]; Lovato et al. [9], and forms the motivation for the present study. Here, the neural network of choice was derived from so-called Gaussian-binary restricted Boltzmann machines, much inspired by the recent contributions by Carleo *et al.*, see, for example, Refs. Carleo and Troyer [2]; Carleo et al. [6]. Unlike binary-binary restricted Boltzmann machines Sieber and Gehringer [15], the approximation properties of Gaussian-binary restricted Boltzmann machines aren't yet well-understood. However, they are considered highly-expressive networks for certain problems, such as dimensionality reduction and continuous probability density estimation. Note that neural-network representations of variational states are more general, as they don't in principle require prior knowledge of the ground-state wave function, thereby opening the door to systems that have yet to be solved. Particular attention however has to be devoted to the symmetries of the problem, whose inclusion is critical to achieve accurate results.

In this work, we will focus on systems of electrons confined to move in two-dimensional harmonic oscillator systems, so-called quantum dots. These are strongly confined electrons and offer a wide variety of complex and subtle phenomena which pose severe challenges to existing many-body methods. Due to their small size, quantum dots are characterized by discrete quantum levels. For instance, the ground states of circular dots show similar shell structures and magic numbers as seen for atoms and nuclei. These structures are particularly evident in measurements of the change in electrochemical potential due to the addition of one extra electron. Here, these systems will serve as our test of the applicability of restricted Boltzmann machines as artificial neural network variational states.

The theoretical foundation and the methodology are explained in Section 2. The subsequent sections present our results with an analysis of computational methods and resources. In the last section we present our conclusions and perspectives for future work.

2 Materials and methods

For any Hamiltonian $\hat{\mathcal{H}}$ and trial wave function ψ_T, the variational principle guarantees that the expectation value of the energy E_T is greater than or equal to the true ground state energy E_0,
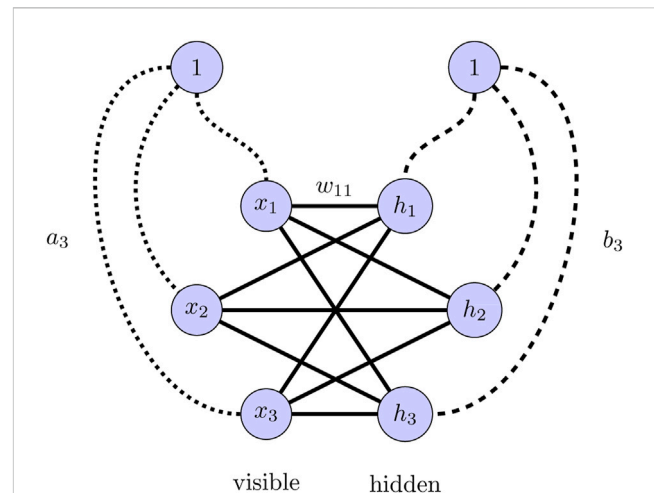

FIGURE 1
Architecture of a restricted Boltzmann machine. Inter-layer connections between the visible and the hidden layer are represented by the solid lines, where, for instance, the line connecting x_1 to h_1 represents the weight w_{11}. The dotted lines represent the visible biases, where the line going from the bias unit to the visible unit x_3 represents the bias weight a_3. The dashed lines represent the hidden biases, where the line going from the bias unit to the hidden unit h_3 represents the bias weight b_3.

$$E_0 \leq E_T = \frac{\langle \psi_T | \hat{\mathcal{H}} | \psi_T \rangle}{\langle \psi_T | \psi_T \rangle}. \tag{1}$$

Thus, approximate solutions to the time-independent Schrödinger equation can be obtained by choosing a careful parameterization of the wave function and minimizing the energy E_T with respect to the parameters. Since the integrals representing E_T are normally high dimensional, it is most efficient to evaluate them by means of Monte Carlo methods

$$E_T \approx \langle E_L \rangle = \frac{1}{n} \sum_{i=1}^{n} E_L(\boldsymbol{R}_i), \quad \boldsymbol{R}_i \sim |\psi_T(\boldsymbol{R})|^2. \tag{2}$$

This involves collecting n samples of configurations and averaging over the so-called local energies

$$E_L(\boldsymbol{R}) = \frac{1}{\psi_T(\boldsymbol{R})} \hat{\mathcal{H}} \psi_T(\boldsymbol{R}). \tag{3}$$

We apply the variational Monte Carlo (VMC) method to various circular quantum dots systems. These are systems of interacting electrons confined to move in a two-dimensional harmonic oscillator well. The (scaled)[1] Hamiltonian is given by

$$\hat{\mathcal{H}} = \frac{1}{2} \sum_i \left[-\nabla_i^2 + \omega^2 r_i^2 + \sum_{j \neq i} \frac{1}{r_{ij}} \right], \tag{4}$$

where ω is the oscillator frequency, r_i is the distance between electron i and the origin, and r_{ij} is the distance between electrons

---

1 Natural units are used with energy given in units of $\hbar$ and length given in units of $\sqrt{\hbar/m}$.

i and j. We will henceforth assume the total number of electrons N to be even and the total spin of the system to be zero.

A simple ansatz can be built starting from the analytical solutions to the non-interacting case. The harmonic oscillator eigenfunctions are given by

$$\phi_{m,n}(x, y) \propto e^{-\omega(x^2+y^2)} H_m(\sqrt{\omega}\,x) H_n(\sqrt{\omega}\,y), \qquad (5)$$

where H_n are the Hermite polynomials of degree n. To constrain the antisymmetry of the many-body wave function, products of the lowest $N/2$ spatial states and the two spin states $\xi_{\pm}(\sigma)$ are used as a basis for a Slater determinant

$$\psi_{\text{SD}}(\boldsymbol{R}) = \det\big[\{\phi_{m,n}(x_i, y_i)\xi_k(\sigma_i)\}\big],$$

where m, n, k label the single-particle state, i labels the particle, and $\boldsymbol{R}$ contains all coordinates of the N particles. As an aside, we do not include the spin projections σ_i as explicit inputs to the wave function as we will describe how to treat them separately in Section 2.2. We then define a reference state by pulling the common exponential term out of the determinant and inserting a single variational parameter α

$$\psi_{\text{Ref}}(\boldsymbol{R}; \alpha) = e^{-\alpha\omega\sum_i\left(x_i^2+y_i^2\right)} \det\big[\{H_m(\sqrt{\omega}\,x_i) H_n(\sqrt{\omega}\,y_i)\xi_k(\sigma_i)\}\big]. \qquad (6)$$

Correlations among electrons can be handled by a Padé-Jastrow factor Drummond et al. [16],

$$g(\boldsymbol{R}; \beta) = \exp\left(\sum_{i=1}^{N}\sum_{j>i}^{N} \frac{a_{ij}r_{ij}}{1+\beta r_{ij}}\right), \qquad (7)$$

where β is a variational parameter and

$$a_{ij} = \begin{cases} 1/3 & \text{if } \sigma_i = \sigma_j \\ 1 & \text{if } \sigma_i \neq \sigma_j \end{cases},$$

in order for the Kato cusp condition to be satisfied Huang et al. [17]. The product of the Slater determinant and the Padé-Jastrow factor is commonly named the Slater-Jastrow ansatz,

$$\psi_{\text{Slater–Jastrow}}(\boldsymbol{R}; \alpha, \beta) = \psi_{\text{Ref}}(\boldsymbol{R}; \alpha) \times g(\boldsymbol{R}; \beta). \qquad (8)$$

2.1 Gaussian-binary restricted Boltzmann machine

There are many possible choices for a machine learning inspired wave function, but using an artificial neural network has recently been demonstrated to give remarkably good results in studies of quantum mechanical many-body systems Cassella et al. [14]; Adams et al. [8]; Lovato et al. [9]; Fore et al. [18]; Rigo et al. [19]. Inspired by Ref. Carleo and Troyer [2], our choice is to start from a restricted Boltzmann machine (RBM) configured for continuous inputs, illustrated in Figure 1. The inputs $\boldsymbol{x} \in \mathbb{R}^{2N}$ are the ravelled particle positions and interactions between the particles are mediated by H hidden binary nodes. After summing over all the possible values of the hidden nodes, the marginal distribution of the inputs to the Gaussian-binary RBM takes the form

$$P(\boldsymbol{R}; \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{w}) = \exp\left(-\sum_{i=1}^{2N} \frac{(x_i - a_i)^2}{2\sigma_i^2}\right)\prod_{j=1}^{H}$$
$$\times \left[1 + \exp\left(b_j + \sum_{i=1}^{2N} \frac{x_i w_{ij}}{\sigma_i^2}\right)\right]. \qquad (9)$$

Here, $\boldsymbol{a} \in \mathbb{R}^{2N}$ and $\boldsymbol{b} \in \mathbb{R}^{H}$ are the bias parameters of the input and hidden nodes, respectively. The weights between the input and hidden nodes are $\boldsymbol{w} \in \mathbb{R}^{2N \times H}$, while $\boldsymbol{\sigma} \in \mathbb{R}^{2N}$ are the widths of the Gaussian input nodes (not to be confused with the spin projections). It is possible to train these widths by reparameterizing them as $\sigma_i = \exp(s_i)$, but in this work all of the widths were fixed to $\sigma = 1/\sqrt{\omega}$ and only the biases and weights are treated as variational parameters. This allows us thus to reduce the number of parameters to the biases and weights only. See Appendix B for the derivation of the marginal probability.

Notice how the marginal distribution in Eq. 9 mimics the Gaussian parts of our aforementioned ansatzes in Eqs 6, 8. Based on such observations, our next step is to construct two corresponding ansatzes

$$\psi_{\text{RBM}}(\boldsymbol{R}; \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{w}) = P(\boldsymbol{R}; \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{w})$$
$$\times \det\big[\{H_m(\sqrt{\omega}\,x_i) H_n(\sqrt{\omega}\,y_i)\xi_k(\sigma_i)\}\big], \qquad (10)$$

and

$$\psi_{\text{RBM+PJ}}(\boldsymbol{R}; \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{w}, \beta) = P(\boldsymbol{R}; \boldsymbol{a}, \boldsymbol{b}, \boldsymbol{w}) \times g(\boldsymbol{R}; \beta)$$
$$\times \det\big[\{H_m(\sqrt{\omega}\,x_i) H_n(\sqrt{\omega}\,y_i)\xi_k(\sigma_i)\}\big]. \qquad (11)$$

The two trial wave functions above apply different levels of physical intuition. While ψ_{RBM} doesn't contain specific information about the electron-electron interactions, $\psi_{\text{RBM+PJ}}$ contains a correlation factor that explicitly upholds the cusp condition. Both ansatzes contain knowledge about the required antisymmetry and the Gaussians in the marginal distribution help localize the wave functions to satisfy the boundary conditions far from the oscillator well. Also, as the marginal distribution is positive definite, these ansatzes will never collapse into the bosonic state even if the marginal distribution is not symmetric.

2.2 Code optimization

Parallel computing is an important part of our efforts for developing an efficient VMC solver. However, increasing the available computational resources alone is often not sufficient. One should also consider developing sophisticated algorithms that deliberately minimize the number of floating point operations, cache misses, and communication between parallel processes. Reducing the number of floating point operations is important in Monte Carlo calculations, in particular for the calculations reported here where we produce a large quantity (typically close to a billion) of samples. To see this, one can consider the evaluation of the kinetic energy. The kinetic energy term of the Schrödinger equation is usually one of the more computationally expensive parts to compute in terms of computing cycles. It includes, amongst other elements, the computation of the Laplacian of the wave function. The
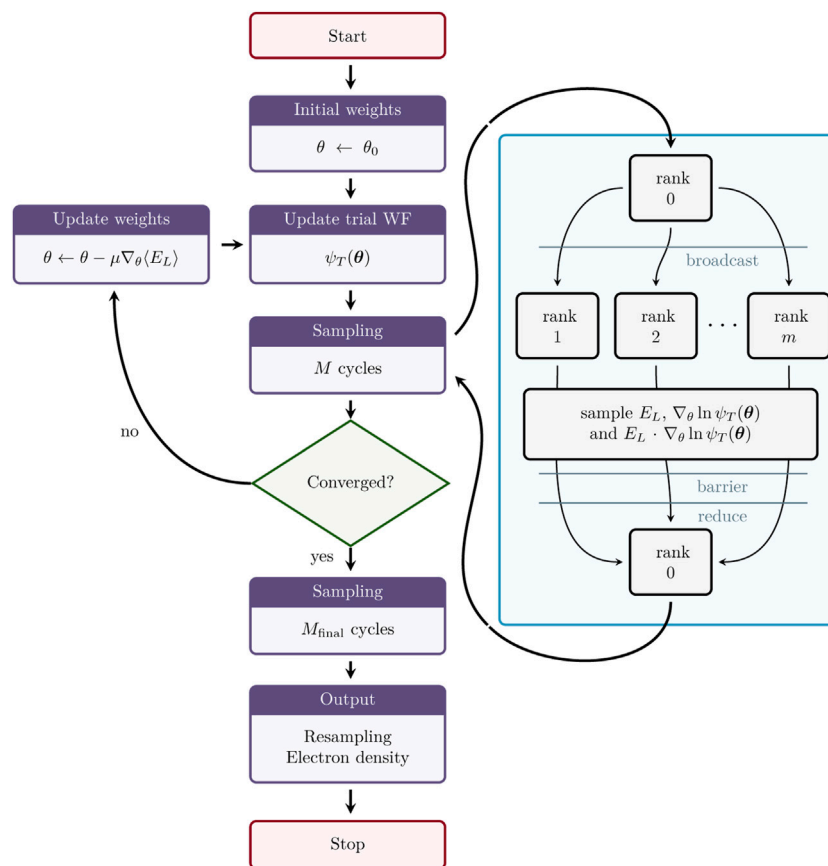
FIGURE 2
Flow chart of the solver with emphasis on the parallel processing. The sampling is parallelized across m processes, where the trial wave function (WF) is broadcast from process 0. Then E_L, $\nabla_\theta \ln \psi_T$ and $E_L \cdot \nabla_\theta \ln \psi_T$ are sampled independently on each process, and an average is taken after all processes are done with sampling.

Laplacian term in the expression for the local energy can be written as

$$\frac{\nabla_i^2 \psi_T}{\psi_T} = \nabla_i^2 \ln \psi_T + \left(\nabla_i \ln \psi_T\right)^2.$$

This way of writing the kinetic energy term is beneficial for two reasons: First, our trial wave function has normally an exponential shape, which is taken care of by the log-function. This is often the case for many other ansatzes for the trial functions. Secondly, this form allows for separating various elements of the trial wave function and reducing thereby the number of floating point operations. Evaluating the left-hand side of the above equation directly as it stands requires more floating point operations than evaluating the first and second derivatives of the logaritmic function. Moreover, with a logarithmic function and exponential functions as ansatzes for the trial wave function, this facilitates the use of automatic differentiation Neidinger [20]; Baydin et al. [21].

By writing the trial wave function as a product of various terms, here $\psi_T = \prod_j \psi_j$, the kinetic energy terms from each particle i can be written as a sum of their corresponding Laplacians and gradients

$$\frac{\nabla_i^2 \psi_T}{\psi_T} = \sum_j \nabla_i^2 \ln \psi_j + \left(\sum_j \nabla_i \ln \psi_j\right)^2. \tag{12}$$

Obtaining analytical expressions of the gradient and Laplacian for all the wave function elements is usually computationally advantageous. However, in many Monte Carlo studies they are normally evaluated numerically using automatic differentiation Neidinger [20]; Baydin et al. [21]. Nowadays, automatic differentiation algorithms are employed routinely in VMC calculations.

The computational complexity of calculating the determinant in the naïve way is proportional to $\mathcal{O}(n^3)$. This calls for a reduction of dimensionality as well as efficient evaluations of the Slater determinant. In this work we do not consider open systems and assume that all single-particle states up to the Fermi level are filled up. We can then split the Slater determinant in a spin-up and a spin-down part Pfau et al. [4] without affecting the expectation value of the energy, that is

$$\psi_{\text{RBM}} = \det\left[\{\phi_{nm}(\boldsymbol{r}_\uparrow)\xi(\sigma_\uparrow)\}\right] \times \det\left[\{\phi_{nm}(\boldsymbol{r}_\downarrow)\xi(\sigma_\downarrow)\}\right].$$

The gradient and Laplacian of the logarithm of a determinant with respect to particle i are given by
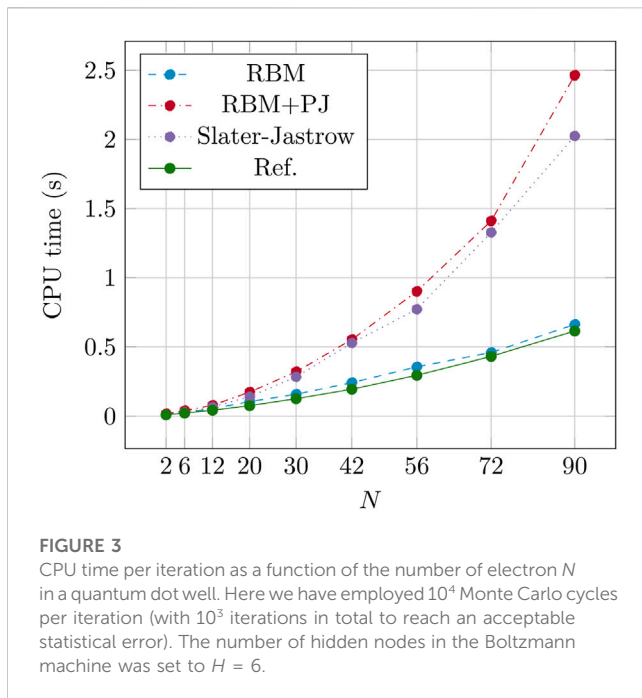
FIGURE 3
CPU time per iteration as a function of the number of electron N in a quantum dot well. Here we have employed 10^4 Monte Carlo cycles per iteration (with 10^3 iterations in total to reach an acceptable statistical error). The number of hidden nodes in the Boltzmann machine was set to $H = 6$.

TABLE 1 CPU time per iteration in seconds for various system sizes and ansatzes. The table compliments Figure 3, and computational details are given in the main text.

N	RBM	RBM + PJ	Slater-Jastrow	Ref
2	0.013517	0.017304	0.011518	0.010029
6	0.027193	0.040015	0.028200	0.022379
12	0.054788	0.078638	0.066240	0.043121
20	0.10498	0.17428	0.14075	0.075787
30	0.15861	0.32107	0.28436	0.12629
42	0.24249	0.55319	0.52828	0.19511
56	0.35651	0.90180	0.77236	0.29539
72	0.45964	1.4112	1.3279	0.43174
90	0.66245	2.4634	2.0244	0.61491

$$\nabla_i \ln \det = \sum_j \nabla_i d_{ji} d_{ij}^{-1},$$
$$\nabla_i^2 \ln \det = \sum_j \nabla_i^2 d_{ji} d_{ij}^{-1},$$

where d_{ji} is element (j, i) of the matrix and d_{ij}^{-1} is the corresponding element of the inverse matrix Hammond et al. [22]. The general solution requires inversion of the matrix, which is known to be costly. Fortunately, we can use a trick to reduce the cost: If we move one particle at a time in our sampling over configurations, this means that we change either the elements of one row or alternatively one column of the Slater matrix. In this case, there is a simple relation between the old and the new inverse matrix

$$d_{kj}^{-1} = \begin{cases} \dfrac{1}{R_i} d_{kj}^{-1} & \text{if } j = i \\[2ex] d_{kj}^{-1} - \dfrac{S_{ij}}{R_i} d_{ki}^{-1} & \text{if } j \neq i, \end{cases}$$

such that the new inverse Slater matrix can be found by a few operations when the previous is known. Here, R_i is the ratio between the new and the old determinant and S_{ij} is the cross product between columns in the new rows and the old matrix,

$$R_i = \sum_j d_{ij} d_{ji}^{-1},$$
$$S_{ij} = \sum_l d_{il} d_{lj}^{-1}.$$

By using these expressions, the entire Slater determinant matrix is inverted only once per simulation. We also avoid including spin flips in the simulations. We limit ourselves in this work to systems where we can use these approximations. For systems where all single-particle states up to the Fermi level are filled, the above serves as a useful approximation if the Hamiltonian does not contain spin-dependent terms. If not, every suggested move should include possible spin flips as well.

Like most other Monte Carlo schemes, the algorithm can be split into smaller individual parts and run efficiently in parallel. For each optimization step, the system is sampled independently in several processes, and the results from all the processes are averaged before performing the parameter optimization. In this way, we achieve near perfect parallelization with message passing interface (MPI). A flow chart of the simulation code can be found in Figure 2. Notice that the parameters are updated with respect to the gradients of the expectation value of the local energy. The latter is given by

$$\nabla_\theta \langle E_L \rangle = 2 \left(\langle E_L \nabla_\theta \ln \psi_T(\boldsymbol{\theta}) \rangle - \langle E_L \rangle \langle \nabla_\theta \ln \psi_T(\boldsymbol{\theta}) \rangle \right),$$

as discussed by Ref. Umrigar and Filippi [23], among others. When the simulations are run with a burn-in period[2], each process should have a burn-in time equal to the burn-in time for a single process. The theoretical parallelization efficiency is then given by $(t_{\text{burn-in}} + t_{\text{sample}})/(m t_{\text{burn-in}} + t_{\text{sample}})$ where m is the number of parallel processes, $t_{\text{burn-in}}$ is the burn-in time and t_{sample} is the total sampling time. Additionally, the weight optimization can't be parallelized, but has a negligible computational cost compared to the sampling. The communication can also be neglected even with low communication speed. The type of Markov-chain Monte Carlo simulations discussed here are rather simple to parallelize with almost no cost from communication between processes.
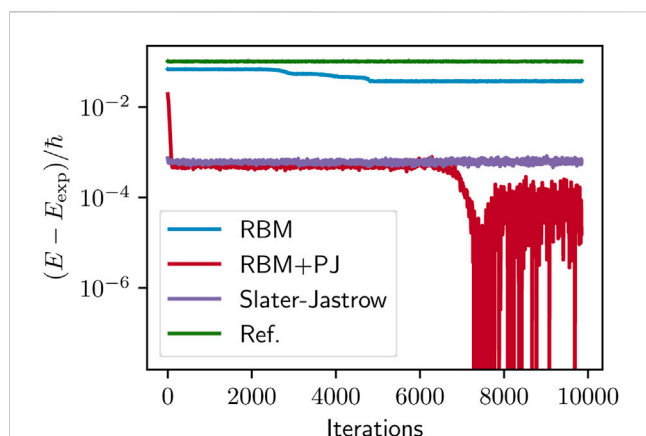
3 Results and discussions

In this section we compare the computational complexity, ground state energy, energy convergence, contribution from the different Hamiltonian terms and electron densities for various trial wave function ansatzes. The RBM ansatz consists of a Slater determinant with Hermite polynomials as the basis, multiplied with the RBM marginal distribution, as presented in Eq. 10 We
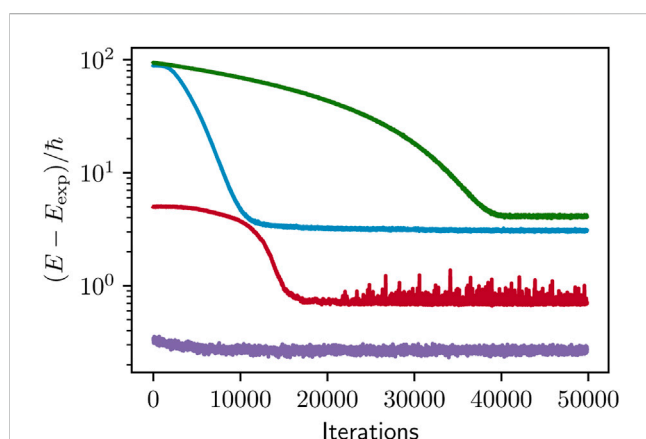
---

2 A burn-in period or time refers to the amount of time which is needed before a steady state is reached in a Markov-chain Monte Carlo simulation. When the most likely or steady state has been reached, one can start collecting samples for the stochastic evaluation of the various integrals.

TABLE 2 The ground state energy of two-dimensional quantum dots with *N* electrons and frequency *ω*. Other references include diffusion Monte Carlo results taken from Høgberget [24] and semi-analytical results obtained by Taut [25]. The energy is given in Hartree units, and the numbers in parenthesis are the statistical uncertainties in the last digit. Bold values correspond to the lowest ground-state energy obtained in this work. For abbreviations see the text.

N	ω	RBM	RBM + PJ	Slater-Jastrow	Ref	DMC (Ref. Høgberget[24])	Semi-analytical (Ref. Taut[25])
2	0.1	0.46774(5)	**0.440975(8)**	0.44129(1)	0.5279(1)	0.44079(1)	
	1/6	0.70389(7)	**0.666665(1)**	0.66710(1)	0.7693(1)		2/3
	0.28	1.07100(6)	**1.021668(7)**	1.02192(1)	1.1388(1)	1.02164(1)	
	0.5	1.72343(7)	1.659637(6)	**1.65974(1)**	1.7983(3)	1.65977(1)	
	1.0	3.0789(1)	**2.999587(5)**	2.99936(1)	3.1484(3)	3.00000(1)	3.0
6	0.1	3.6971(1)	3.5700(2)	**3.5695(1)**	3.8552(5)	3.55385(5)	
	0.28	7.9318(3)	**7.6203(2)**	7.6219(1)	8.0517(9)	7.60019(6)	
	0.5	12.2640(6)	**11.80494(7)**	11.8104(2)	12.2799(9)	11.78484(6)	
	1.0	20.5635(6)	**20.1773(1)**	20.1918(2)	20.697(1)	20.15932(8)	
12	0.1	12.6772(4)	12.3416(4)	**12.29962(9)**	12.9742(9)	12.26984(8)	
	0.28	26.389(2)	25.7266(2)	**25.7049(4)**	26.625(2)	25.63577(9)	
	0.5	40.375(1)	**39.2348(2)**	39.2421(5)	40.227(2)	39.1596(1)	
	1.0	67.620(3)	65.7911(7)	**65.7026(4)**	66.744(3)	65.7001(1)	
20	0.1	30.7906(8)	30.1444(2)	**30.0403(2)**	31.253(2)	29.9779(1)	
	0.28	63.592(3)	62.1445(5)	**62.0755(7)**	63.681(3)	61.9268(1)	
	0.5	96.356(5)	94.101(1)	**94.0433(9)**	95.755(4)	93.8752(1)	
	1.0	159.428(3)	156.104(1)	**155.8900(4)**	157.904(6)	155.8822(1)	
30	0.1	61.853(2)	60.774(2)	**60.585(1)**	62.449(4)	60.4205(2)	
	0.28	126.891(8)	124.437(2)	**124.195(2)**	126.717(5)	123.9683(2)	
	0.5	191.455(9)	187.488(2)	**187.325(3)**	189.977(6)	187.0426(2)	
	1.0	315.364(8)	308.989(2)	**308.576(1)**	311.70(2)	308.5627(2)	
42	0.1	109.767(7)	108.128(2)	**107.928(2)**	110.630(7)	107.6389(2)	
	0.28	224.257(9)	220.588(3)	**220.224(2)**	223.837(8)	219.8426(2)	
	0.5	337.43(1)	331.410(3)	**331.276(3)**	335.18(1)	330.6306(2)	
	1.0	553.40(1)	543.746(3)	**542.977(2)**	548.07(2)	542.9428(8)	
56	0.1	179.035(8)	176.659(2)	**176.221(1)**	180.08(1)	175.9553(7)	
	0.28	364.52(2)	359.456(6)	**358.470(2)**	363.81(1)	358.145(2)	
	0.5	547.20(3)	538.666(5)	**537.841(4)**	544.12(3)	537.353(2)	
	1.0	894.12(2)	881.010(5)	**879.514(3)**	887.20(5)	879.3986(6)	
72	0.1	274.12(1)	270.870(3)	**270.296(3)**	275.34(2)		
	0.28	556.63(2)	549.899(8)	**548.315(4)**	555.45(2)		
	0.5	833.85(3)	822.78(2)	**821.089(6)**	829.31(3)		
	1.0	1355.37(2)	1341.54(2)	**1339.85(1)**	1349.65(6)		
90	0.1	399.84(1)	395.486(4)	**394.621(4)**	401.19(2)		
	0.28	809.99(2)	800.504(6)	**799.187(5)**	808.35(2)		
	0.5	1211.92(5)	1198.12(3)	**1195.025(9)**	1205.65(4)		
	1.0	1973.95(5)	1948.75(2)	**1946.27(1)**	1958.58(5)		

FIGURE 4
Energy convergence for quantum dots with N = 2 and ω = 1/6. The figure shows how the ground state energy approaches the analytical value (Ref. Taut [25]) for various ansatzes.

FIGURE 5
Energy convergence for quantum dots with N = 56 and ω = 0.1. The reference value was obtained by Ref. Høgberget [24] using diffusion Monte Carlo. The labeling of the various results is the same as in the previous figure.

have not made any attempt to include optimized single-particle state functions through mean-field optimizations like Hartree-Fock theory. The second ansatz is the RBM ansatz with a correlation factor described in Eq. 7, abbreviated RBM + PJ. We have also include results obtained using a traditional Slater-Jastrow ansatz (Eq. 8), and as reference we use a plain Slater determinant (without a correlation factor, Eq. 6). The diffusion Monte Carlo (DMC) results obtained by Høgberget [24] are our main reference for the ground state energy. For quantum dots with two electrons we compare our results with corresponding analytical ones from Ref. Taut [25].

Figure 3 displays the average CPU time per iteration as a function of system size for the different wave function ansatzes. To obtain these estimates, we averaged the CPU time per iteration over 10^3 iterations. The RBM + PJ and Slater-Jastrow ansatzes are the most computationally demanding due to the Padé-Jastrow factor, which depends on the relative distance between electrons and requires updating of a matrix containing electron-electron

distances. Each process has its own memory, and the matrix is not shared across processes.

The impact of parallel processing on computational overhead is minimal compared to other aspects of the code, except for the burn-in period. In fact, the variation in CPU time as a result of noise is much more important than the variation due to parallel processing. Additional information on CPU time per iteration for different wave function ansatzes is presented in Table 1. The reported calculations were performed for 10^4 Monte Carlo cycles and an oscillator frequency of ω = 1.0, with similar results for other values of ω. Each benchmark simulation was run on a single core without a burn-in period. Production runs require more cycles and consequently longer CPU times per iteration. For instance, for a system size of N = 90 with 2^{20} Monte Carlo cycles run on 30 nodes (960 threads), the RBM, RBM + PJ, and Slater-Jastrow ansatzes required 0.55960, 2.7345, and 1.3001 s per iteration, respectively, with approximately 50,000 iterations needed for convergence.
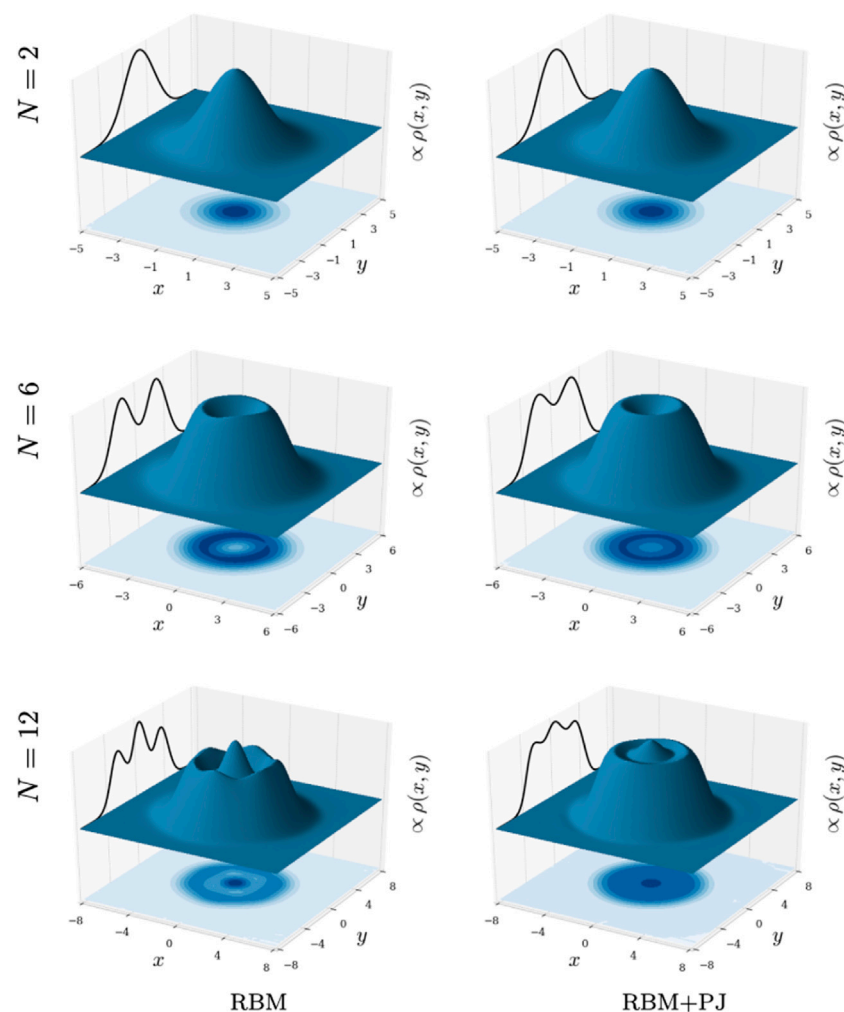
The ground state energies of two-dimensional quantum dots of various sizes and frequencies are presented in Table 2. The RBM + PJ and Slater-Jastrow ansatzes provide the lowest ground state energies as expected as the correlation factor does explicit satisfy the Kato's cusp conditions. The relative errors with respect to diffusion Monte Carlo (DMC) calculations tend to be less than 0.2% for both methods. The various results obtained with the RBM + PJ ansatz show that this ansatz dominates for small quantum dots, but is outperformed by the Slater-Jastrow ansatz for larger systems. We suspect this is due to the fact that the former ansatz is more complex and contains significantly more parameters than the latter, and has therefore a hard time finding the global minimum. The optimization could be improved with a more sophisticated optimization algorithm.

For low frequency dots ($\omega < 0.28$), the RBM produces ground state energies lower than the reference energy, but fails for large high frequency dots ($N > 6$, $\omega > 0.28$). For low frequency dots the interaction energy dominates, and the RBM manages to learn the correlations. For high frequency dots, it is the one-body part of the Hamiltonian which dominates and the interaction part is less important for the final expectation value of the energy. For these systems, the Slater determinant part of the wave function ansatz is the most important one, meaning in turn that the RBM ansatz we have used may not capture well the structure of the single-particle states that define the Slater determinants.

We also note that the reference values are similar to corresponding Hartree-Fock energies up to 30 particles Mariadason [26], which is expected as both approaches try to find the optimal single Slater determinant without a correlation factor.

Some simulations were also performed for the RBM and RBM + PJ ansatzes with sorted network inputs to enforce anti-symmetry under exchange of two particles, as suggested by Ref. Saito [27], among others. Sorted inputs showed promising results, where the ground state energy typically dropped in the fourth or fifth digit. For example, the RBM + PJ ansatz found the ground state energy of a system with N = 30 electrons and ω = 0.5 to be 187.311(1) Hartree. This is lower than the Slater-Jastrow energy. We will investigate this further in a follow-up paper [28].

For the traditional ansatzes that do not contain artificial neural networks, the learning rate determines to a large extent how fast the
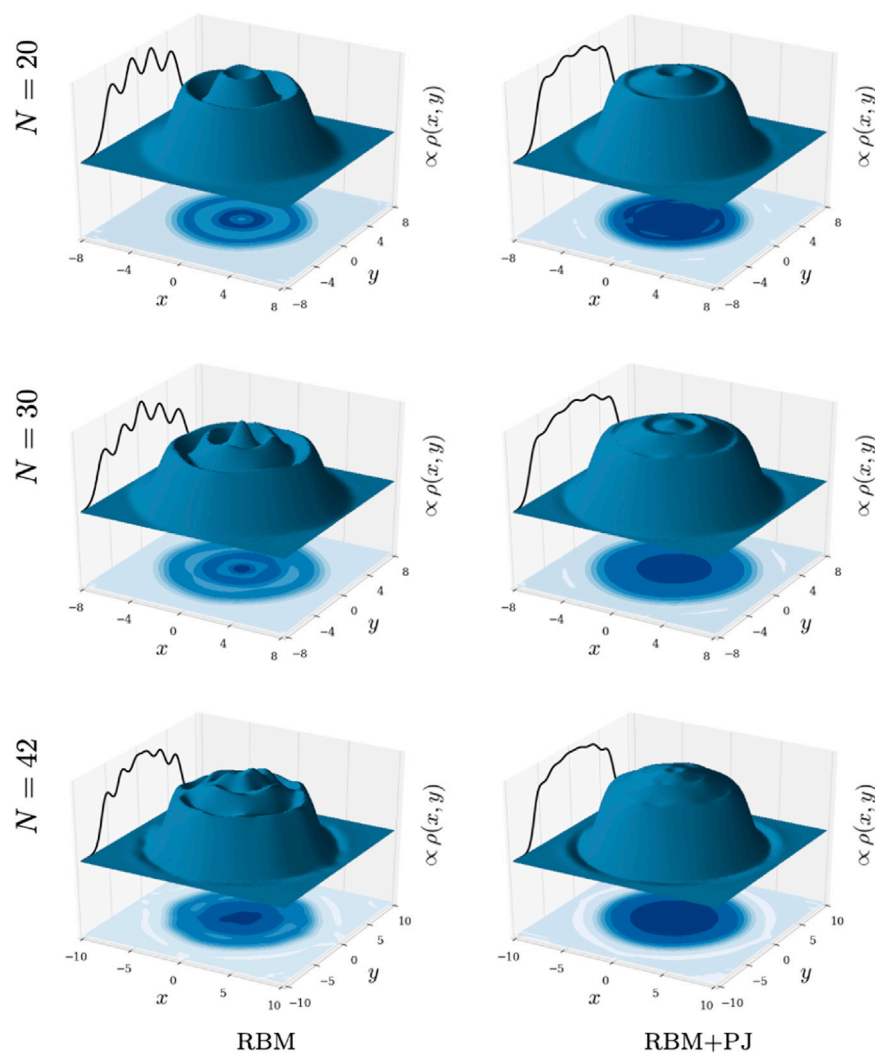
FIGURE 6
Electron density profiles, $\rho(x, y)$, for two-dimensional quantum dots with frequency $\omega = 0.5$ and $N = 2$, 6 and 12 electrons seen from the top. The surface plot and the contour plot on the xy-plane illustrate the density, and the graph on the yz-plane represents the cross-section through $x = 0$. They were obtained using RBM (left column) and RBM + PJ (right column), with $M = 2^{30}$ Monte Carlo cycles after convergence. The plots are noise-reduced using a Savitzky-Golay filter. For abbreviations and description of the natural units used, see the main text for more details.

trial wave function converges towards the true ground state wave function. However, for the RBM and the RBM + PJ approaches, the results are sensitive to the chosen learning rate values. A too large learning rate can easily cause exploding energies, and with a too small learning rate the obtained energies with a given trial function might not converge at all. Our strategy is to find the highest learning rate that does not lead to exploding energies. In general this requires efficient grid searching methods. Here, when the energies have converged, we decrease the learning rate by a factor of 10^3 and let it run until it converges again. Also, the RBMs tend to converge step-wise, making it hard to know whether or not they have converged.

In Figure 4, the convergence of the local energy is plotted for our four ansatzes for $N = 2$ electrons. Here the results are compared with analytical calculations for $N = 2$. A similar behavior is shown for $N = 56$ electrons and $\omega = 0.1$ in Figure 5. For the $N = 2$ electrons case, RBM + PJ turns out to be the most accurate ansatz with small absolute errors. During training, the Padé-Jastrow parameter is rather constant, but the adjustment of weights seems important for the model to reach the energy minimum. For larger systems, the Slater-Jastrow ansatz provides a slightly lower energy than the RBM + PJ ansatz. Since the number of inputs for the RBM increases with number of particles, the network contains additional training parameters as we increase the number of electrons. Therefore, the networks get more complex for large quantum dots, and they are naturally harder to train.

The spatial one-body density plots for $\omega = 1.0$ and the RBM and RBM + PJ ansatzes are presented in Figures 6, 7. The electron densities have a wave shape for both ansatzes, with two nodes for $N = 2$, three nodes for $N = 6$ and so on similar to those observed in

---

3   We have employed a grid of learning rates in terms of powers of ten with negative exponents. Since we have not implemented an adaptive calculation of the learning rates, we search for the optimal results using a grid of decreasing learning rates.

FIGURE 7
Electron density profiles, $\rho(x, y)$, for two-dimensional quantum dots with frequency $\omega = 0.5$ and $N = 20$, 30 and 42 electrons seen from the top. The surface plot and the contour plot on the xy-plane illustrate the density, and the graph on the yz-plane represents the cross-section through $x = 0$. They were obtained using RBM (left column) and RBM + PJ (right column), with $M = 2^{30}$ Monte Carlo cycles after convergence. The plots are noise-reduced using a Savitzky-Golay filter. For abbreviations and description of the natural units used, see main text.

Refs. Ghosal et al. [29]; Høgberget [24]. The RBM seems to exaggerate the states with more distinct peaks (higher peaks and lower wave valleys) compared to the RBM + PJ results. This can be explained by more localized electrons, which becomes even more apparent in low frequency dots (see Figure 8), as the interactions are modelled differently. The RBM would hardly be able to model the correct electron-electron distances, as the network itself is purely linear. The electron density was found to be shape-invariant for high frequency dots ($\omega > 0.28$), with decreasing spatial range as the frequency increases.

When reducing the frequency further, the interaction energy dominates over the kinetic energy and harmonic oscillator energy (see Figures 10, 11), and the electrons naturally become more localized. The Padé-Jastrow factor solves this by radially localizing the electrons and conserving the circular symmetry, such that the electrons are confined to specific orbitals. This can be seen from Figure 8, where the spatial one-body density is plotted

for low frequency dots ($\omega = 0.1$) and system sizes $N = 2$, 6 and 12. Radial localization is also what we would expect from the Hamiltonian, which is strictly circular symmetric. On the other hand, the RBM seems to localize the electrons both in radial and angular direction, with the number of electrons corresponding to the number of peaks. This is a nonphysical solution to the problem, and shows that the RBM ansatz breaks down for low frequencies. The RBM + PJ ansatz, on the other hand, confines the electrons in orbitals. Distinct peaks in radial direction is what is expected from Wigner crystallization, which we might see indications of with density parameters $r_s \approx 6.7$, $r_s \approx 1.2$, and $r_s = 0.3$ respectively for the three system sizes with the RBM + PJ ansatz. Notice for instance the small "pit" on top of the $N = 2$ plot for RBM + PJ, which isn't seen for higher oscillator frequencies. In Figure 9, we have decreased the one-body density even further to $\omega = 0.01$ for $N = 2$. As expected, the electrons become even more localized and are clearly showing Wigner crystallization effects with a density parameter of $r_s \approx 29$.
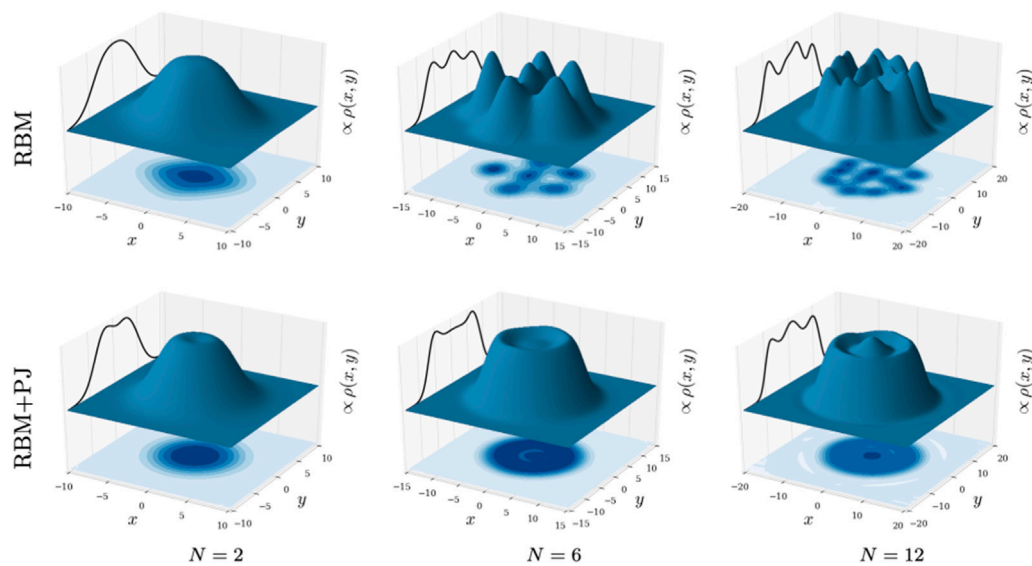
FIGURE 8
One-body density profile, $\rho(x, y)$, of two-dimensional quantum dots with frequency $\omega = 0.1$ and $N = 2$, 6 and 12 electrons seen from left to right. The ansatzes used are RBM (upper panels) and RBM + PJ (lower panels). The surface plot and the contour plot on the xy-plane illustrate the density, and the graph on the yz-plane represents the cross-section through $x = 0$. The surface plots are noise-reduced using a Savitzky-Golay filter. For abbreviations and description of the natural units used, see main text.
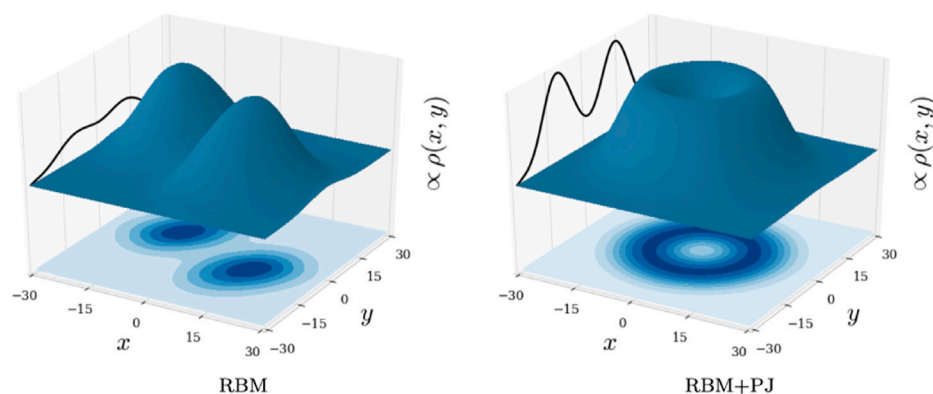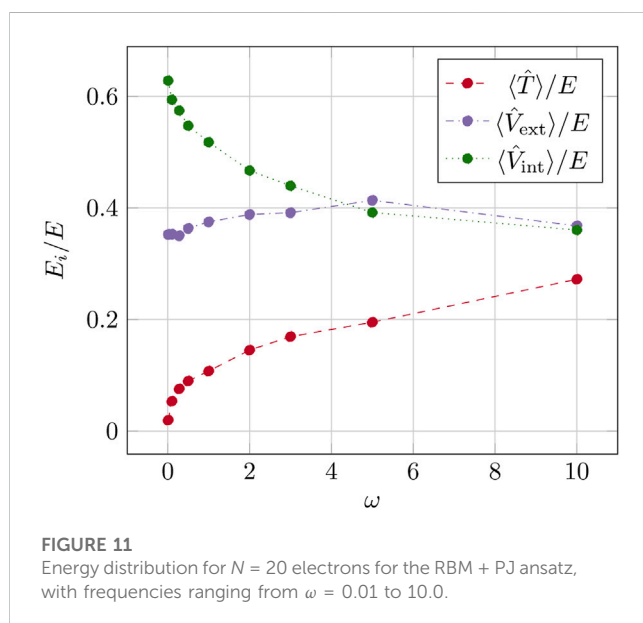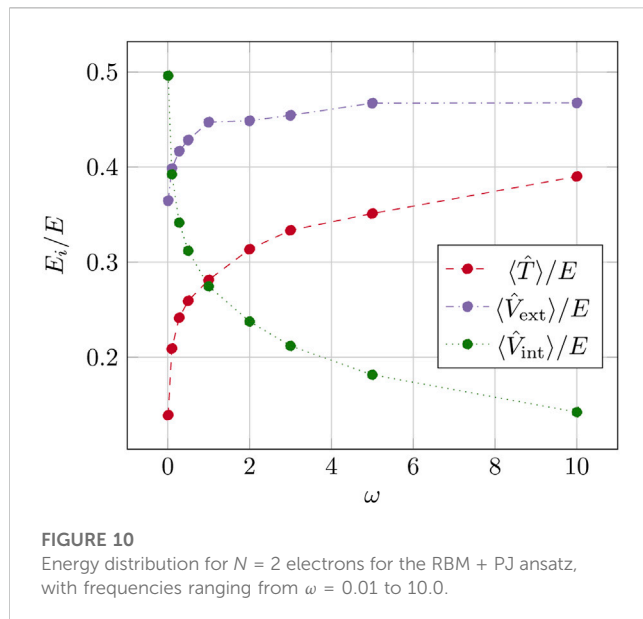

FIGURE 9
One-body density profile, $\rho(x, y)$, of two-dimensional quantum dots with frequency $\omega = 0.01$ and $N = 2$ electrons. The ansatzes used are RBM (left) and RBM + PJ (right). The surface plot and the contour plot on the xy-plane illustrate the density, and the graph on the yz-plane represents the cross-section through $x = 0$. The surface plots are noise-reduced using a Savitzky-Golay filter. For abbreviations and description of the natural units used, see main text.

For the RBM ansatz, the electrons are strongly localized (in all directions) and the electron densities barely overlap. For the RBM + PJ ansatz, we observe strong orbital confinement where the Wigner crystallization wasn't the target of this study, but the framework seems capable of a profound study of this phenomenon.

To understand the behavior of the one-body density for various frequencies, we investigate the expectation values of the kinetic energy ($\langle \hat{T} \rangle$), the harmonic oscillator potential energy ($\langle \hat{V}_{\text{ext}} \rangle$) and the two-body interaction energy ($\langle \hat{V}_{\text{ext}} \rangle$). The energy distribution is plotted for the RBM + PJ ansatzes for $N = 2$ and $N = 20$ (Figures 10,

11) for frequencies $\omega \in \{0.01, 0.1, 0.28, 0.5, 1.0, 2.0, 3.0, 5.0,$ and $10.0\}$. For large values of the oscillator frequency it is the one-body part of the Hamiltonian which dominates in absolute value (kinetic energy and harmonic oscillator potential energy) compared with the expectation value of the two-body interaction. For such frequencies we notice also that the results can almost be interpreted in terms of the virial theorem. This theorem provides a useful relation between the kinetic and potential energy Fock [30]. For circular quantum dots without the two-electron interaction, the theorem reads $2\langle \hat{T} \rangle = 2\langle \hat{V}_{\text{ext}} \rangle$. From our results we notice that, due to the

FIGURE 10
Energy distribution for $N = 2$ electrons for the RBM + PJ ansatz, with frequencies ranging from $\omega = 0.01$ to 10.0.

FIGURE 11
Energy distribution for $N = 20$ electrons for the RBM + PJ ansatz, with frequencies ranging from $\omega = 0.01$ to 10.0.

Coulomb interaction with a finite range factor, this effect would disappear. Figures 10, 11 show this behavior rather clearly.

4 Computational details

The quantum dot systems were studied using a general framework for variational Monte Carlo simulations (The code is available on github.com/evenmn/VMaChine). Importance sampling was used to accelerate the simulations Metropolis et al. [31]. To minimize the local energy, we applied the Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.999$ as suggested by Ref. Kingma and Ba [32]. We use an adaptive number of cycles, starting from 2^{20} and then increased to 2^{24} for the ten last iterations after the energy had converged, and further to 2^{30} for the very last iteration to reduce the statistical uncertainty and noise in electron density. The particle step length was chosen to get an acceptance ratio close to 99.5%, spawning from 10^1 for the smallest and weakest systems to 10^{-3} for the large and narrow oscillators. The optimal learning rate was found by grid searches, and varied from 10^1 to 10^{-5}. Both the step length and the learning rate depend strongly on the system and the trial wave function.

For the RBM and RBM + PJ ansatzes, only the raw particle positions were input to the RBM. This choice was made for performance reasons, as inputting processed positions like the electron-electron distances would lead to significantly larger computational efforts, with an increased network complexity.

The Gaussian parameter was initialized to $\alpha = 1.0$, which is the analytical optimum without interaction. We use Xavier initialization for the RBM weights, putting them all close to zero Glorot and Bengio [33]. The special case with all weights set to zero corresponds to our reference ansatz with $\alpha = 1.0$. For all the RBMs but the most narrow ones ($\omega = 1.0$), the number of hidden nodes was set to 6, giving 40 to 1272 parameters for the various system sizes. For $\omega = 1.0$, we used $H = 12$ hidden nodes to achieve a lower energy.

All the simulations were run on Intel Xeon E5-2670 CPUs. In total the computational cost of this project was of the order of 10^6 CPU hours with largest amount of cycles spent, for obvious reasons, on the largest systems.

5 Conclusion and perspectives

We found that the RBM ansatz gives a significantly lower ground state energy than the reference ansatz for low frequency quantum dots. This may indicate that the RBM manages to capture some of the electron-electron correlations. Based on the one-body density plots, the RBM found the electrons to be localized both angularly and radially compared to the ansatzes containing a Padé-Jastrow factor, which confine electrons radially only. For high frequency dots, the RBM fails in the sense that the obtained ground state energy is larger than the reference energy. This can be explained by the fact that when the interactions get less important, the reference ansatz is a good guess. The RBM + PJ ansatz gives energies close to the DMC energies for small quantum dots. The ansatz performance needs to be seen in light of the computational cost, as the ansatzes containing a Padé-Jastrow factor are far more computationally intensive.

From the one-body density and energy distribution plots, it is apparent that the RBM ansatz isn't able to capture the

interaction energy, the ratio between kinetic and harmonic oscillator energies are not exactly equal to two for large frequencies. However, for such frequencies the interaction energy plays a less prominent role, resulting thus in a ratio which is close to the ideal value. When we decrease the frequency however, and the system becomes more dilute with an increase of the mean distance between particles, one notices a somewhat counter intuitive behavior. The expectation value of the kinetic energy and the harmonic oscillator potential energy decrease (and the electrons become more localized as seen in the one-body densities above). However, many-body correlations increase in importance with decreasing frequency. This is reflected in the increased role of the expectation value of the two-body Coulomb interaction, an effect which is simply due to the infinite range of the Coulomb interaction. If we were to multiply the

correlations at the same level as the Padé-Jastrow factor. Because of the linearity of the network, it is impossible for it to compute the distance between the particles, which is crucial to model the interactions correctly. One solution to this could be to input the electron-electron distance into the network, as discussed by Pfau et al. [4], but this would increase the computational intensity. Also, despite including a Slater determinant, the trial wave function is not necessarily anti-symmetric when including an RBM. We performed some simulations where anti-symmetry was forced by sorting the network inputs, which showed promising results in terms of the ground state energy. However, although promising results can be obtained, RBMs are less flexible than general neural networks that make fewer assumptions about the specific mathematical forms of the trial functions. We have encoded explicitly the anti-symmetry via a Slater determinant. Furthermore, two-body correlations are constructed using a Jastrow factor. An RBM with Gaussian distributions is capable of capturing the one-body part of the problem, but is less flexible in finding two-body or more complicated many-body correlations. Although the RBM results reported here are promising compared with existing VMC calculations, recent results with neural networks like those presented in, for example, Refs. Pfau et al. [4]; Cassella et al. [14]; Lovato et al. [9]; Adams et al. [8]; Carrasquilla and Torlai [3] offer much more flexible and promising research venues for deep learning methods applied to many-body problems. Results obtained with deep neural networks for these systems will be presented in a future work [28].

Data availability statement

The original contributions presented in the study are included in the article/supplementary materials, further inquiries can be directed to the corresponding author.

Author contributions

Funding

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Goodfellow I, Bengio Y, Courville A. *Deep learning*. Cambridge, Massachusetts): The MIT Press (2016).

2. Carleo G, Troyer M. Solving the quantum many-body problem with artificial neural networks. *Science* (2017) 355:602–6. doi:10.1126/science.aag2302

3. Carrasquilla J, Torlai G. Neural networks in quantum many-body physics: A hands-on tutorial (2021). Available at: https://arxiv.org/abs/2101.11099 (Accessed January 26, 2021).

4. Pfau D, Spencer JS, Matthews AGDG, Foulkes WMC. *Ab initio* solution of the many-electron Schrödinger equation with deep neural networks. *Phys Rev Res* (2020) 2: 033429. doi:10.1103/PhysRevResearch.2.033429

5. Calcavecchia F, Pederiva F, Kalos MH, Kühne TD. Sign problem of the fermionic shadow wave function. *Phys Rev E* (2014) 90:053304. doi:10.1103/PhysRevE.90.053304

6. Carleo G, Cirac I, Cranmer K, Daudet L, Schuld M, Tishby N, et al. Machine learning and the physical sciences. *Rev Mod Phys* (2019) 91:045002. doi:10.1103/RevModPhys.91.045002

7. Boehnlein A, Diefenthaler M, Sato N, Schram M, Ziegler V, Fanelli C, et al. *Colloquium*: Machine learning in nuclear physics. *Rev Moddern Phys* (2022) 94:031003. doi:10.1103/RevModPhys.94.031003

8. Adams C, Carleo G, Lovato A, Rocco N. Variational Monte Carlo calculations of A≤4 nuclei with an artificial neural-network correlator ansatz. *Phys Rev Lett* (2021) 127: 022502. doi:10.1103/PhysRevLett.127.022502

9. Lovato A, Adams C, Carleo G, Rocco N (2022). Hidden-nucleons neural-network quantum states for the nuclear many-body problem. Available at: https://arxiv.org/abs/2206.10021 (Accessed June 20, 2022).

10. Hornik K, Stinchcombe M, White H. Multilayer feedforward networks are universal approximators. *Neural Networks* (1989) 2:359–66. doi:10.1016/0893-6080(89)90020-8

11. Murphy KP. *Machine learning: A probabilistic perspective*. Cambdrige, Massachusetts): The MIT Press (2012).

12. Hastie T, Tibshirani R, Friedman J. *The elements of statistical learning: Data mining, inference and prediction*. Berlin): Springer Verlag (2009).

13. Bishop CM. *Pattern recognition and machine learning*. Berlin): Springer Verlag (2006).

14. Cassella G, Sutterud H, Azadi S, Drummond ND, Pfau D, Spencer JS, et al. Discovering quantum phase transitions with fermionic neural networks. *Phys Rev Lett* (2023) 130:036401. doi:10.1103/PhysRevLett.130.036401

15. Sieber J, Gehringer J. Quantitative universal approximation bounds for deep belief networks (2022). Available at: https://arxiv.org/abs/2208.09033 (Accessed August 18, 2022).

16. Drummond ND, Towler MD, Needs RJ. Jastrow correlation factor for atoms, molecules, and solids. *Phys Rev B* (2004) 70:235119. doi:10.1103/physrevb.70.235119

17. Huang C-J, Filippi C, Umrigar CJ. Spin contamination in quantum Monte Carlo wave functions. *J Chem Phys* (1998) 108:8838–47. doi:10.1063/1.476330

18. Fore B, Kim JM, Carleo G, Hjorth-Jensen M, Lovato A. Dilute neutron star matter from neural-network quantum states. *arXiv* [Preprint] (2022). doi:10.48550/arXiv.2212.04436

19. Rigo M, Hall B, Hjorth-Jensen M, Lovato A, Pederiva F. Solving the nuclear pairing model with neural network quantum states. *Phys Rev E* (2023) 107:025310. doi:10.1103/PhysRevE.107.025310

20. Neidinger RD. Introduction to automatic differentiation and MATLAB object-oriented programming. *SIAM Rev* (2010) 52:545–63. doi:10.1137/080743627

21. Baydin AG, Pearlmutter BA, Andreyevich Radul A, Siskind JM. Automatic differentiation in machine learning: A survey. *J Machine Learn Res* (2018) 18:1. doi:10.5555/3122009.3242010

22. Hammond BL, Lester WA, Reynolds PJ. *Monte Carlo methods in ab initio quantum chemistry*. Singapore: World Scientific (1994).

23. Umrigar CJ, Filippi C. Energy and variance optimization of many-body wave functions. *Phys Rev Lett* (2005) 94:150201. doi:10.1103/physrevlett.94.150201

24. Høgberget J. Quantum Monte Carlo studies of generalized many-body systems. Master's thesis. Oslo: University of Oslo (2013).

25. Taut M. Two electrons in a homogeneous magnetic field: Particular analytical solutions. *J Phys A* (1994) 27:1045–55. doi:10.1088/0305-4470/27/3/040

26. Mariadason AA. Quantum many-body simulations of double dot system. Master's thesis. Oslo: University of Oslo (2018).

27. Saito H. Method to solve quantum few-body problems with artificial neural networks. *J Phys Soc Jpn* (2018) 87:074002. doi:10.7566/jpsj.87.074002

28. Kim J, Fore B, Nordhagen EM, Lovato A, Hjorth-Jensen M. *Deep learning and confined electrons in two dimensions* (2022).

29. Ghosal A, Güçlü AD, Umrigar CJ, Ullmo D, Baranger HU. Incipient Wigner localization in circular quantum dots. *Phys Rev B* (2007) 76:085341. doi:10.1103/physrevb.76.085341

30. Fock V. *Z für Physik* (1930) 63:855–8. doi:10.1007/BF01339281

31. Metropolis N, Rosenbluth AW, Rosenbluth MN, Teller AH, Teller E. Equation of state calculations by fast computing machines. *J Chem Phys* (1953) 21:1087–92. doi:10.1063/1.1699114

32. Kingma DP, Ba J. Adam: A method for stochastic optimization (2014). Available at: https://arxiv.org/abs/1412.6980 (Accessed December 22, 2014).

33. Glorot X, Bengio Y. Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics; May 13-15, 2010; Sardinia, Italy (2010).

Appendix A Derivation of RBM distributions

In this , we will derive the marginal and conditional distributions of a Gaussian-binary restricted Boltzmann machine with the system energy

$$E(\boldsymbol{x}, \boldsymbol{h}) = \sum_{i=1}^{2N} \frac{(x_i - a_i)^2}{2\sigma_i^2} - \sum_{j=1}^{H} b_j h_j - \sum_{i=1}^{2N} \sum_{j=1}^{H} \frac{x_i w_{ij} h_j}{\sigma_i^2}.$$

There are $2N$ visible units x_i with related bias weights a_i and H hidden units h_j with related bias weights b_j. w_{ij} are the weights connecting the visible units to the hidden units. The joint probability distribution is given by the Boltzmann distribution

$$P(\boldsymbol{x}, \boldsymbol{h}) = \frac{1}{Z} \exp\left(-\beta E(\boldsymbol{x}, \boldsymbol{h})\right),$$

where Z is the partition function,

$$Z = \iint d\boldsymbol{x} d\boldsymbol{h} P(\boldsymbol{x}, \boldsymbol{h}),$$

and $\beta = 1/k_B T$ is the reciprocal temperature that will be fixed to 1. As the marginal and conditional distributions are closely related both for the visible and hidden layer, we present the distributions in sections respective for the two layers.

Appendix B Distributions of visible units

The distributions of the visible units are used to find properties related to the visible units. If we recall a restricted Boltzmann machine, the transformation between the visible units and the hidden units is $f_j(\boldsymbol{x}; \boldsymbol{\theta}) = b_j + \sum_{i=1}^{2N} w_{ij} x_i / \sigma_i^2$. By this expression, we can express the joint probability distribution as

$$
\begin{aligned}
P(\boldsymbol{x}, \boldsymbol{h}) &= \frac{1}{Z} \exp\left(-\sum_{i=1}^{2N} \frac{(x_i - a_i)^2}{2\sigma_i^2} + \sum_{j=1}^{H} b_j h_j + \sum_{i=1}^{2N} \sum_{j=1}^{H} \frac{x_i w_{ij} h_j}{\sigma_i^2}\right), \\
&= \frac{1}{Z} \exp\left(-\sum_{i=1}^{2N} \frac{(x_i - a_i)^2}{2\sigma^2} + \sum_{j=1}^{H} h_j f_j(\boldsymbol{x}; \boldsymbol{\theta})\right).
\end{aligned}
\tag{B1}
$$

The marginal distribution of the visible units is given by the sum over all possible hidden states, $\{\boldsymbol{h}\} \in \{0, 1\}$:

$$P(\boldsymbol{x}) = \sum_{\{\boldsymbol{h}\}} P(\boldsymbol{x}, \boldsymbol{h}),$$

as the hidden units can take binary values only. By inserting the expression of the joint probability distribution from Eq. B1, we obtain

$$
\begin{aligned}
P(\boldsymbol{x}) &= \frac{1}{Z} \sum_{\{\boldsymbol{h}\}} \exp\left(-\sum_{i=1}^{2N} \frac{(x_i - a_i)^2}{2\sigma^2} + \sum_{j=1}^{H} h_j f_j(\boldsymbol{x}; \boldsymbol{\theta})\right), \\
&= \frac{1}{Z} \exp\left(-\sum_{i=1}^{2N} \frac{(x_i - a_i)^2}{2\sigma^2}\right) \times \left(\sum_{\{\boldsymbol{h}\}} \prod_{j=1}^{H} \exp(h_j f_j)\right), \\
&= \frac{1}{Z} \exp\left(-\sum_{i=1}^{2N} \frac{(x_i - a_i)^2}{2\sigma^2}\right) \prod_{j=1}^{H} \sum_{h_j=0}^{1} \exp(h_j f_j), \\
&= \frac{1}{Z} \exp\left(-\sum_{i=1}^{2N} \frac{(x_i - a_i)^2}{2\sigma^2}\right) \prod_{j=1}^{H} \left[1 + \exp(f_j(\boldsymbol{x}; \boldsymbol{\theta}))\right].
\end{aligned}
$$

This is what we will use as the marginal distribution of the visible units.