Neural network approach to quasiparticle dispersions in doped antiferromagnets

Check for updates

Hannah Lange [1,2,3] ✉, Fabian Döschl [1,3], Juan Carrasquilla[4,5,6] & Annabelle Bohrdt[3,7]

Numerically simulating large, spinful, fermionic systems is of great interest in condensed matter physics. However, the exponential growth of the Hilbert space dimension with system size renders exact quantum state parameterizations impractical. Owing to their representative power, neural networks often allow to overcome this exponential scaling. Here, we investigate the ability of neural quantum states (NQS) to represent the bosonic and fermionic $t - J$ model – the high interaction limit of the Hubbard model – on various 1D and 2D lattices. Using autoregressive, tensorized recurrent neural networks (RNNs), we study ground state representations upon hole doping the half-filled system. Additionally, we propose a method to calculate quasiparticle dispersions, applicable to any network architecture or lattice geometry, and allowing to infer the low-energy physics from NQS. By analyzing the strengths and weaknesses of the RNN ansatz we shed light on the challenges and promises of NQS for simulating bosonic and fermionic systems.

The simulation of quantum systems has remained a persistent challenge until today, primarily due to the exponential growth of the Hilbert space, making it exceedingly difficult to parameterize the wave functions of large systems using exact methods. Since the seminal work of Carleo and Troyer[1], the idea of using neural networks to simulate quantum systems[1–5] has been applied successfully for a large number of quantum systems, leveraging various neural network architectures. These architectures include restricted Boltzmann machines[6], convolutional neural networks (CNNs)[7], group CNNs[8], autoencoders[9] as well as autoregressive neural networks such as recurrent neural networks (RNNs)[5,10–14], with neural network representations of both amplitude and phase distributions of the quantum state under consideration. These neural quantum states (NQS) make use of the innate ability of neural networks to efficiently represent probability distributions. When applying them to represent quantum systems, this ability can help to reduce the number of parameters required to encode the system.

Despite their representative power, NQS have been shown to face challenges during the training process, for example, when they are trained to minimize the energy, i.e. to represent ground states. This results from the intricate nature of the loss landscape, characterized by numerous saddle points and local minima that complicate the search for the global minimum[15]. One promising avenue to overcome this problem is the use of many uncorrelated samples during the training. This strategy is facilitated when using autoregressive neural networks[16,17], allowing to directly sample

from the wave functions' amplitudes. Autoregressive networks have already been applied in the physics context[18,19], such as for the variational simulation of spin systems[11–14].

Many works have so far focused on NQS representations of spin systems at half-filling, revealing that NQS can be used to study a variety of phenomena that are relevant to state-of-the-art research, as e.g. shown for RNN representations on various lattice geometries, including frustrated spin systems[11,20], and systems with topological order[21]. For all of these systems, the physics becomes even richer when introducing mobile impurities, e.g. holes, into the system, yielding a competition between the magnetic background and the kinetic energy of the impurity. Simulating such systems holds particular relevance for understanding high-temperature superconductivity, where the superconducting dome arises upon doping the antiferromagnetic half-filled state with holes[22]. The search for NQS that are capable of representing such spinful fermionic systems is still in its early stages. In recent years, the first NQS have been developed that obey the fermionic statistics, simulating molecules[23–27], spinless fermions[17] and spinful fermions[28–31]. Among those architectures are FermiNet[23,24], Slater-Jastrow ansätze[17,28] or variants of Jordan-Wigner transformations[25,29,32].

Here, we use an autoregressive neural network architecture, supplemented with a Jordan-Wigner transformation, to simulate ground and excited states of the high interaction limit of the Fermi-Hubbard model, believed to capture essential features of high-temperature cuprate superconductors.

[1]Ludwig-Maximilians-University Munich, Theresienstr. 37, Munich D-80333, Germany. [2]Max-Planck-Institute for Quantum Optics, Hans-Kopfermann-Str.1, Garching D-85748, Germany. [3]Munich Center for Quantum Science and Technology, Schellingstr. 4, Munich D-80799, Germany. [4]Department of Physics, 60 Saint George St., University of Toronto, Toronto, Ontario M5S 1A7, Canada. [5]Vector Institute, MaRS Centre, Toronto, Ontario M5G 1M1, Canada. [6]Department of Physics and Astronomy, University of Waterloo, Ontario N2L 3G1, Canada. [7]University of Regensburg, Universitätsstr. 31, Regensburg D-93053, Germany. ✉e-mail: hannah.lange@physik.uni-muenchen.de

Specifically, we use a tensorized 2D version of an RNN wave function[11] with gated recurrent units[33–35], proven to successfully model spin systems[10,11,20,21,36,37]. In the remainder of this paper, we will discuss the system under investigation, namely the bosonic and fermionic $t − J$ model, followed by the presentation of a scheme for the calculation of dispersion relations from any NQS architecture, tested for quasiparticle dispersions of 1D and 2D $t − J$ systems. We find that the RNN succeeds in accurately capturing the features of the considered low-energy states. Lastly, we discuss the performance of the RNN ansatz and identify the strengths and bottlenecks of this ansatz.

Results

We apply the RNN to simulate ground and excited states of the fermionic (bosonic) $t − J$ model, both in one and two dimensions. In its more generalized form, known as the fermionic (bosonic) $t − XXZ$ model, with anisotropic superexchange interactions denoted as J_z and $J_\pm$, the Hamiltonian under consideration reads as follows:

$$\mathcal{H}_{tXXZ} = -t \sum_{\langle i,j \rangle, \sigma} \mathcal{P}_G \left(\hat{c}_{i,\sigma}^\dagger \hat{c}_{j,\sigma} + \text{h.c.} \right) \mathcal{P}_G$$
$$+ J_z \sum_{\langle i,j \rangle} \left(\hat{S}_i^z \cdot \hat{S}_j^z - \frac{1}{4} \hat{n}_i \hat{n}_j \right) \qquad (1)$$
$$+ J_\pm \sum_{\langle i,j \rangle} \frac{1}{2} \left(\hat{S}_i^+ \cdot \hat{S}_j^- + \hat{S}_i^- \cdot \hat{S}_j^+ \right),$$

with the fermionic (bosonic) creation and annihilation operators $\hat{c}_{i,\sigma}^\dagger$ and $\hat{c}_{i,\sigma}$ for particles at site i with spin σ, spin (pseudospin) operators $\hat{S}_i = \sum_{\sigma,\sigma'} \hat{c}_{i,\sigma}^\dagger \frac{1}{2} \sigma_{\sigma\sigma'} \hat{c}_{i,\sigma'}$ and density operators $\hat{n}_i$[38]. Furthermore, $\mathcal{P}_G$ projects out states with more than one particle per site. Note that for a single hole, this single occupancy constraint leads to the same statistics for the bosonic and fermionic models. For $J_z = J_\pm$, Eq. (1) reduces to the $t − J$ model and for $J_\pm = 0$ to the $t − J_z$ model. Despite its relevance for the study of unconventional superconductivity, whether there are actually superconducting phases in the $t − J$ and the Fermi Hubbard model is still under debate[39–41]. While finding NQS representations of superconducting states, or – more broadly – pairing wave functions[28,30,42–44], is still a topic of current research, here, we focus on the capacity of the RNN ansatz to represent ground and excited states of the $t − J$ model and not on their superconducting properties.

In the absence of doping ($\hat{n}_i = 1$), Eq. (1) reduces to the XXZ model or, in the case of $J_z = J\pm$, the Heisenberg model. Prior studies have already utilized RNNs to simulate these spin models[20,45], with the possibility of rendering the model stoquastic by making use of the Marshall sign rule[46]. This is done by implementing the sign rule directly in the RNN architecture[20], yielding a simplified optimization procedure of the wave functions' phase. In contrast, we do not implement any bias on the phase of the quantum state, in order to make our architecture applicable to any number of holes in the system.

When the ground state at $\hat{n}_i = 1$ is doped with a single hole, the resulting mobile impurity gets dressed with a cloud of magnetic excitations. This yields the formation of a magnetic polaron, which has already been observed in ultracold atom experiments[47]. Its properties strongly depend on the spin background, see Fig. 1a and b. Upon further doping, the strong correlations in the model make the simulation of the Fermi-Hubbard or $t − J$ models numerically challenging, despite impressive numerical advances in the past years[39,48–50]: Commonly used methods all come with their specific limitations, e.g. density matrix renormalization group[51,52] is limited by the area-law of entanglement, making it challenging to apply this methods to 2D or higher dimensions. Finally, the calculation of spectral functions or the dispersion relations $E(\boldsymbol{k})$[53], as exemplary shown in Fig. 1, is of great interest for many fields in physics to reveal the emergent physics of a system under investigation. In condensed matter physics, they are typically used to infer the dominating excitations in the ground state or higher energy states, e.g. upon doping the system. This information is contained in specific features of the spectra, e.g. the bandwidth of the quasiparticle dispersion $E(\boldsymbol{k})$. However, the calculation of spectra or dispersions $E(\boldsymbol{k})$ is in general, computationally
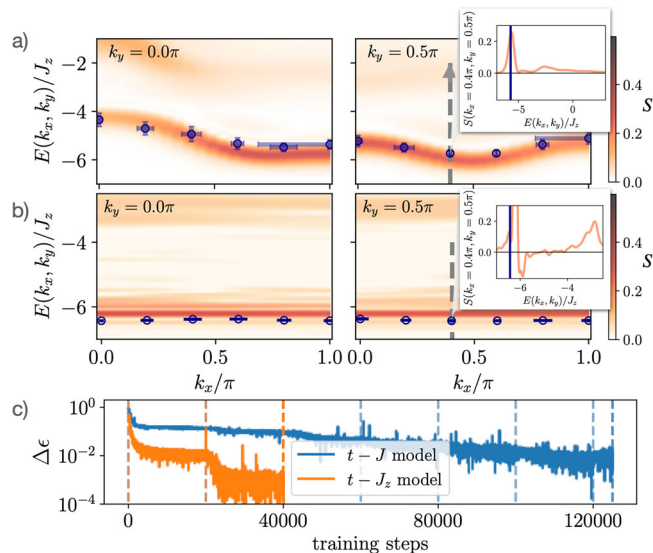

Fig. 1 | Results for the $t − J$ and $t − J_z$ square lattice with 10×4 sites, $t/J_z = 3$ and open boundaries in x, periodic boundaries in y direction. a) Quasiparticle dispersion of a single hole for the $t − J$ system obtained with the recurrent neural network (RNN) (blue markers), compared to the matrix product state (MPS) spectral function from ref. 53 with the spectral weight S indicated by the colormap and shown in the inset for $\boldsymbol{k} = (0.4\pi, 0.5\pi)$ (gray dashed lines). We average the energy over the last 100 training iterations with the standard deviation denoted by the respective error bars in blue. **b** Dispersion of the $t − J_z$ system obtained with the RNN, compared to the MPS spectral function and with the same error bars as in **a**. **c** Relative errors $\Delta\epsilon = \frac{E_{\text{RNN}} - E_{\text{DMRG}}}{|E_{\text{DMRG}}|}$ during the training, with hidden dimension $d_h = 300$. The training was restarted after 20000 steps (dashed lines), with an increased number of samples per iteration from 200 to 600 ($t − J$) or 1000 ($t − J_z$) in the last restart.

costly using conventional methods, e.g. density-matrix renormalization group (DMRG) simulations[54,55]: The former typically involves a, in general expensive, time-evolution of the state, and the latter the calculation of a global operator, the momentum $\boldsymbol{k}$, which is typically very costly for matrix-product-states. Our RNN ansatz uses $U(1) = U(1)_{\hat{N}} \times U(1)_{\hat{S}_z}$ symmetry, i.e. conserved total particle and total magnetization[10,11,20,25,45,56]. Further details on the RNN architecture can be found in Methods.

NQS dispersion relations

Here, we calculate the dispersion relations $E(\boldsymbol{k})$ in $t − XXZ$ systems with different dimensions and different lattice geometries using NQS. The method that we propose is applicable to any NQS architecture since the momentum is enforced via a momentum constraint in the cost function, forcing the system to a specific target momentum $\boldsymbol{k}_{\text{target}}$ within the dispersion, see Fig. 2 and Methods. This is in contrast to previous works[26,57,58], where the momentum is used for the definition of the wave function coefficients. Hence, the scheme only requires the possibility to draw samples from the NQS and calculate the respective probabilities, making the calculation of $E_{\text{NQS}}(k_x, k_y)$ computationally efficient, and allows to use pre-trained NQS from the ground state as a starting point for the momentum training. Furthermore, the scheme can also be combined with spatial symmetries. This could help to improve the accuracy, e.g. when using a NQS with implemented translational invariance. Moreover, additional symmetries could also be used to calculate e.g. m_4 rotational resonances[59] or to probe the competition between the s, p or d-wave ground state energies[60]. Here, we focus on the peak positions of the quasiparticle spectra. In principle, also the spectral weights could also be accessed by calculating the overlap $\langle \psi_{\boldsymbol{k}}^{1h} | \hat{c}_{\boldsymbol{k}} | \psi_0^{0h} \rangle$ of the momentum eigenstate $\psi_{\boldsymbol{k}}^{1h}$ with the ground state ψ_0^{0h} upon removing a particle from the system. To also obtain the peak positions and spectral weights of higher energy states, usually the Green's function is computed. In the context of NQS, this has been done for a
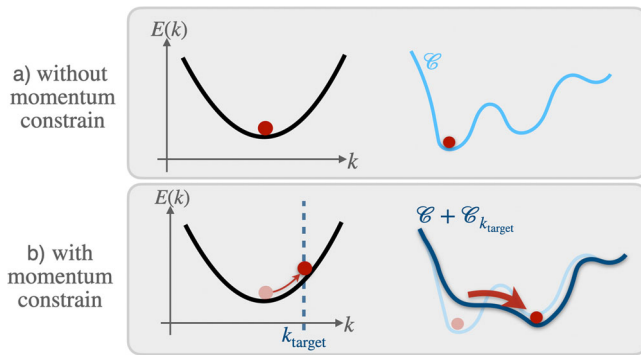
Fig. 2 | Calculating dispersion relations from NQS. Adding the momentum constraint $\mathcal{C}_{k_{target}}$ of Eq. (12), on top of the energy minimization $\mathcal{C}$ of Eq. (15) in **a**, changes the loss landscape as schematically shown in **b** and forces the neural quantum state into a higher energy state $E(k)$ with the desired momentum k_{target}.

quantum Ising model using time dependent variational Monte Carlo $(t-\text{VMC})$[61], for $J_1 - J_2$ and Heisenberg models using an extension of the stochastic reconfiguration algorithm[62,63] or for the Hubbard model using dynamical variational Monte Carlo (VMC)[64].

$t-$XXZ model in 1D. In Fig. 3a the dispersion for a single hole in an antiferromagnetic $t-$XXZ chain with 20 sites and $J_\pm = 1, J_z = 4$ and $t = 8$, obtained with a 1D RNN and exact diagonalization (ED) is shown. Note that in the case of a single hole, no holes can be exchanged and hence the bosonic and fermionic models are the same. The results for the ground state energy at $k_x = 0.5\pi$, obtained during a training with 20000 iterations, and the energies away from the ground state, shown in Fig. 3, are in relatively good agreement with the exact values from ED. However, at some values of $k_x \neq 0.5\pi$ it can be seen that the RNN is trapped in local minima close to the ground state. Overall, the RNN succeeds in capturing physical properties like the bandwidth very accurately, revealing the underlying physical excitations:

For the system under consideration, the bandwidth and the shape of the dispersion in Fig. 3a is a result of spin-charge separation in 1D systems. Spin-charge separation denotes the fact that the motion of a hole in such an antiferromagnetic (AFM) spin chain with coupling $J_\pm, J_z \ll t$ can be approximated by an almost free hole that is only weakly coupled to the spin chain. Hence, the dispersion in Fig. 3 can be approximated by two separate dispersions; i.e. holon and spinon dispersions. Hereby, the holon is the charge excitation, associated with energy scales t, and the spinon is the spin excitation associated with energy $J_\perp, J_z$. In ref. 59 it is shown that the combined dispersion is

$$E(k_x) = -2t\,\cos(k_h) + J_\pm\,\cos(2\Delta k) + J_\pm + J_z, \qquad (2)$$

where $\Delta k = k_x - k_h$, k_h is the momentum of the holon and $k_x = k_h + k_s$ is the combined momentum of the holon and spinon. Eq. (2) is denoted by the gray line in Fig. 3. Again, the agreement with the RNN is relatively good.

$t-J$ model on a square lattice. Due to the layered structure of high-T_c superconductors like cuprates[22] or nickelates[65,66], the physics of $t-J$ systems upon doping is particularly interesting in 2D. In Figs. 1 and 4, the quasiparticle dispersion for a single hole on 10×4 and $4 \times 4 t-J$ and $t-J_z$ lattices are presented. In both cases, Figs. 1b and 4b show that the ground state convergence is better for the $t-J_z$ model with relative errors on the order of $\Delta\epsilon \approx 10^{-3}$ for both system sizes, yielding a good agreement with the reference energies from DMRG (10×4 system) and ED (4×4 system) for all considered energies $E(k_x, k_y)$ away from the ground state. With a relative error of $\Delta\epsilon \approx 10^{-2}$, the error of the $t-J$ ground states is above the $t-J_z$ systems, which is also reflected in the accuracy of the dispersion $E_{RNN}(k_x, k_y)$ in Figs. 1 and 4.
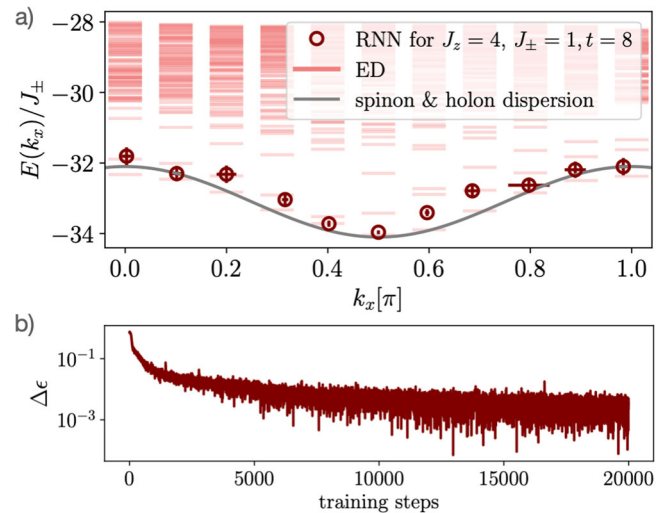

Fig. 3 | Results for the 1D $t-$XXZ system with 20 sites and $J_\pm = 1$, $J_z = 4$ and $t = 8$. a Quasiparticle dispersion for a single hole obtained with the recurrent neural network (RNN, red markers), compared to exact energies from exact diagonalization(ED, light red lines) and the combined spinon and holon dispersions from Eq. (2) (gray). We average the RNN energy over the last 100 training iterations, each with 200 samples, with the standard deviation denoted by the error bars. We show the exact low-energy excited states as well. **b** Relative error $\Delta\epsilon = \frac{E_{RNN} - E_{ED}}{|E_{ED}|}$ during the ground state training. **a** and **b** are obtained using a 1D RNN architecture with a hidden dimension $d_h = 100$.
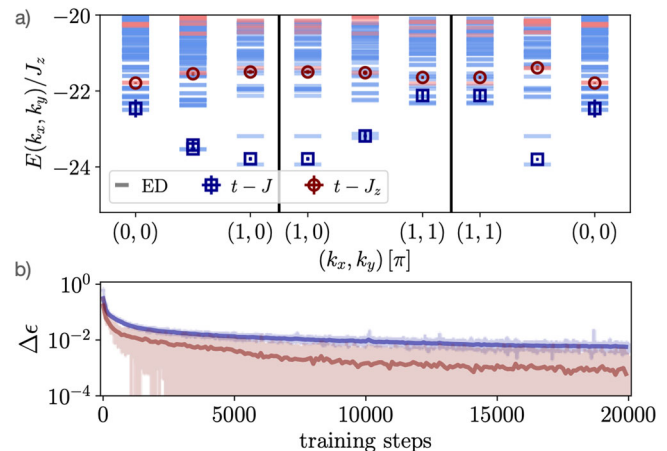

Fig. 4 | Results for the $t-J$ (blue) and $t-J_z$ (red) square lattice with 4×4 sites, $t/J = 3$ and periodic boundaries. a Quasiparticle dispersion for a single hole obtained with the recurrent neural network (RNN, blue and red markers), compared to the exact energies from exact diagonalization (ED, blue and red lines). We average the energy over the last 100 training iterations, each with 200 samples, with the standard deviation denoted by the respective error bars shown in blue and red. We show the exact low-energy excited states as well. **b** Relative error $\Delta\epsilon$ during the ground state training for $t-J$ (light blue) and $t-J_z$ (light red) square lattice ground states, with a hidden dimension $d_h = 100$ and minimum-step stochastic reconfiguration ($t-J$) and $d_h = 70$ and Adam ($t-J_z$). Thick lines are averages over 100 training iterations to guide the eye.

In contrast to the previous section, there is no spin-charge separation in the strict sense in two-dimensional systems. In the case $t \gg J_\pm = J_z =: J$ that we consider here ($t/J = 3$), the mobile dopant can be described by fractionalized spinons and chargons that are confined by a string-like potential that arises due to the spin background distortion when the dopant moves through the system[67,68]. Based on this idea, Laughlin[69] drew the analogy with the 1D Fermi-Hubbard or $t-J$ systems and suggested that the dispersion in the respective 2D systems can be interpreted in
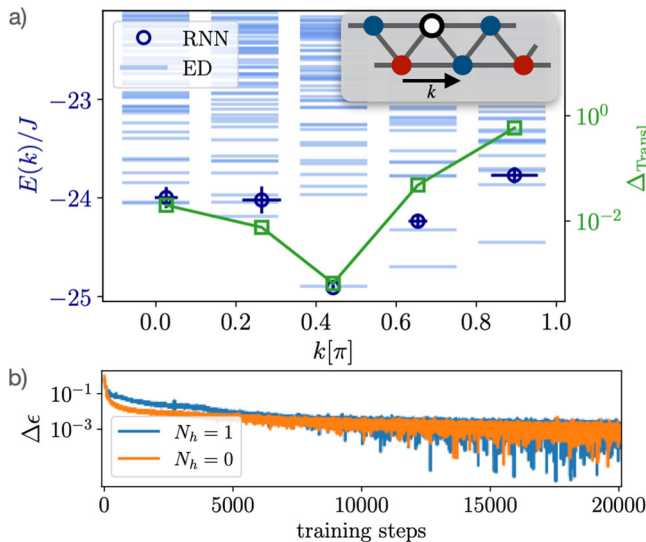
Fig. 5 | Results for the $t-J$ model on a triangular lattice with 9×2 sites, $t/J = 3$ and periodic boundaries along x direction. a Quasiparticle dispersion for a single hole obtained with the recurrent neural network (RNN, blue markers), compared to the energies from exact diagonalization (ED, light blue lines). The large error for $k = 0.998\pi$ appears together with a large violation of translational invariance $\Delta_{\text{Transl.}}$, measured by the relative difference between log-amplitudes $\log |\psi(\sigma)|^2$ and $\log |\psi(\hat{T}_y \sigma)|^2$ as defined in Eq. (3) shown in green. All values are averages over the last 100 training iterations, each with 200 samples, with the standard deviation denoted by the blue error bars. We show the exact low-energy excited states as well. **b** Relative error $\Delta\epsilon$ during the ground state training without doping (orange) and with one hole (blue).

terms of pointlike partons, spinons and chargons, that interact with each other. This *parton picture* explains that the quasiparticle dispersion for a single hole is dominated the spinon with a bandwidth on the order of $J_\pm$, with corrections by the chargon on energy scales of t[53]. This mechanism also provides the explanation for the flat dispersion for the $t-J_z$ model in contrast to the $t-J$ model, as captued by the RNN, see Figs. 1 and 4. Despite the small deviations from the dispersions calculated with ED or DMRG, our RNN architecture, succeeds in capturing the respective bandwidths of $t-J_z$ and $t-J$ models very accurately, allowing to gain valuable insights on the spinon and chargon physics from the RNN dispersions. Furthermore, the fact that node $(\pi/2, \pi/2)$ and antinode $(\pi, 0)$ are degenerate in the 4×4 system is correctly reproduced.

Lastly, we would like to mention that there is a small region of suppressed spectral weight near (π, π) in the DMRG results of the $t-J$ system[59]. This suppression yields difficulties for our RNN scheme that are further discussed in Supplementary Note 3.

$t-J$ model on a triangular lattice. On triangular lattices, the physical phenomena that are observed are distinctly different from the physics of bipartite lattices, due to the notion of frustration and the absence of particle-hole symmetry in non-bipartite lattices, among them e.g. kinetic frustration[70,71]. In particular, the underlying constituents upon doping the triangular ladder are not known[71], making the triangular lattice an intriguing system to study. Recent advancements have shown that these lattices can also be studied experimentally using optical triangular lattices[72–74] and solid-state platforms based on Moiré heterostructures[75–77].

Triangular spin systems have already been studied using RNNs[20]. Here, we consider a triangular $t-J$ ladder with length $L_x = 9$, with the quasiparticle dispersion for a single hole and the learning with and without doping shown in Fig. 5.

As suggested in ref. 20, we use variational annealing for the training for the triangular lattice that was shown to improve the performance for
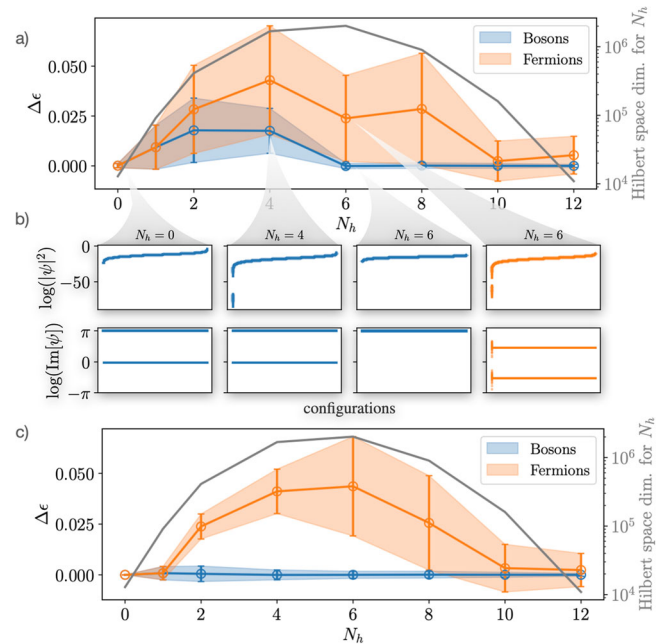

Fig. 6 | Recurrent neural network (RNN) representation for ground states of the bosonic and fermionic $t-J_{(z)}$ model with $t/J = 3$, $0 \le N_h \le 12$ for a 4×4 square lattice with open boundaries. a Relative energy error for bosons (blue) and fermions (orange) as well as the Hilbert space dimension for a fixed number of holes N_h and magnetization $\hat{S}_z$ (gray). **b** Logarithmic amplitude $\log(|\psi|^2)$ and phase $\log(\text{Im}\psi)$ distributions from exact diagonalization (ED) for exemplary bosonic (blue) and fermionic (orange) hole numbers. **c** Relative error and Hilbert space dimension for the $t-J_z$ model. We use a hidden dimension $h_d = 100$. All values obtained from the RNN are averages over the last 100 training steps (each with 200 samples), and error bars denote the respective standard deviation.

frustrated systems like the triangular Heisenberg model[20], see Methods. In Fig. 5 it can be seen that this procedure yields relatively good results for the ground states, with errors of $\Delta\epsilon \approx 0.001$ for both $N_h = 0$ and $N_h = 1$. For the dispersion shown in Fig. 5a, we consider the momentum k defined along the ladder, as shown in the inset figure. When enforcing $k \neq 0.444\pi$ away from the ground state, the exact energy gaps from ED to the first excited states strongly decrease, and the the RNN gets trapped in these states in most cases, in particular for $k > 0.444\pi$. Furthermore, the error bars of the enforced momenta are much higher compared to the other lattice geometries that were studied in Figs. 1, 3 and 4, suggesting that the RNN states partly break the translation invariance, and hence challenge the momentum optimization scheme. This is further supported by the relative difference between the wave function amplitudes $\log |\psi_\lambda(\sigma_i)|^2 = \log p_\lambda(\sigma_i)$ and the respective translated samples $\log |\psi_\lambda(\hat{T}_{\mathbf{e}_\mu} \sigma_i)|^2 = \log p_\lambda(\hat{T}_{\mathbf{e}_\mu} \sigma_i)$,

$$\Delta_{\text{Transl.}}^\mu = \frac{1}{N_s} \sum_i \frac{\left(\log p_\lambda(\sigma_i) - \log p_\lambda\left(\hat{T}_{\mathbf{e}_\mu} \sigma_i\right)\right)^2}{\left(\log p_\lambda(\sigma_i) + \log p_\lambda\left(\hat{T}_{\mathbf{e}_\mu} \sigma_i\right)\right)^2}, \quad (3)$$

which we take as a *measure* for the violation of the translational invariance at the respective momenta. Figure 5 shows that the momenta with large errors coincident with a large $\Delta_{\text{Transl.}}^y$.

Performance of the RNN ansatz
The analysis of the quasiparticle dispersions indicate that our bosonic and fermionic RNN ansätze successfully learns the dominating physics in the considered regimes. In the following, we provide a more detailed discussion on the performance of the RNN ansatz for fermionic and bosonic spin systems upon hole doping, focusing on ground states on a 4×4 square lattice.

Figure 6 shows the relative error for the ground state energies of the $t-J_{(z)}$ model obtained with our RNN ansatz upon doping the half-filled

system with N_h holes. Starting from $N_h = 0$, where the $t - J$ reduces to the Heisenberg model, our RNN reaches a relative ground state energy error $\Delta\epsilon \approx 10^{-4}$ after 20,000 training steps compared to ED. Fig. 6b shows that the respective phase and amplitude distributions are relatively simple in this case, with a low variance for the logarithmic amplitude and only two values for the phase, 0 and π. Note that when comparing to the literature of ground state representations using RNNs for the Heisenberg model[11,45], the optimization problem in our setup is more challenging due to the following reasons: (i) The RNN that we use has a local Hilbert space dimension of three states instead of two, allowing for all values of N_h in principle. (ii) Our RNN learns the sign structure without any bias, i.e. we do not implement the Marshall sign rule already in the RNN, which would only work for $N_h = 0$. (iii) We do not include the knowledge of spatial symmetries yet, which can improve the performance as shown in Methods.

Upon doping, the exact log-amplitude and phase distributions from ED can become more complicated than for the $t - J_z$ model. For example, for $N_h = 4$, the variance of the exact amplitudes becomes very large, $\sigma^b_{N_h=6}(\log|\psi|^2) = 15.91$, see Fig. 6b. This yields larger ground state energy errors than for the $t - J_z$ model, and is further complicated when including the antisymmetry in the fermionic case. Again, we make the observation that for larger hole dopings, $N_h \geq 6$ for bosons and $N_h \geq 10$ for fermions, the distributions for phase and amplitude become less complicated than in the low to intermediate doping regime, yielding a higher accuracy of the RNN wave function with errors $\Delta\epsilon \leq 10^{-4}$ for bosons and $\Delta\epsilon \leq 10^{-2}$ for fermions in the respective doping regimes.

Our results show that in the low doping regime of the $t - J$ model, both fermionic systems and bosonic systems are difficult to learn, see Fig. 6. This suggests that not only the fermionic sign structure poses difficulties: Firstly, the Hilbert space dimension in the finite doping regime indicated in gray in Fig. 6a and c becomes much larger than for spin systems, challenging both the RNN ansatz and its training. Second, the frustrated motion of holes in the AFM Heisenberg background can potentially cause problems. When these holes move through the system, the spin background is affected, giving rise to an effective $J_1 - J_2$ spin model with nearest and next-nearest spin exchange interactions and is hence more difficult to learn[78]. For the $t - J_z$ model, we observe that, probably due to the lack of spin dynamics resulting from the absence of spin-flip terms, the relative errors are comparably low in the bosonic case. Furthermore, for all states with high $\log|\psi|^2$ variance, there is a significant amount of configurations σ with a large negative log amplitude, i.e. $|\psi(\sigma)|^2 \approx 0$. This makes an accurate determination of expectation values extremely costly and can affect the training process. For example, in ref. 79 it was shown that this yields higher variances for the gradients determined by stochastic reconfiguration. Lastly, Fig. 6 shows that the performance decreases for fermions, which is in agreement with the fact that already on a mean-field level, fermionic Slater determinants are much more complicated than the bosonic states.

In the Methods section, we provide a detailed discussion on these challenges are encountered during training our $t - J$ RNN architecture, yielding the relatively large errors encountered e.g. in Fig. 6. Besides the increased Hilbert space dimension and the small amplitudes for certain configurations discussed above, we discuss the learning plateau associated with a local minimum that is encountered for all considered optimization routines—including annealing[20], minimum-step stochastic reconfiguration (minSR)[80] and the recently proposed stochastic reconfiguration (SR) variant based on a linear algebra trick[81]—and the fact that SR algorithms have problems with autoregressive architectures[82]; the complicated interplay between phase and amplitude optimization[15]; and the difficulty to implement constraints on the symmetry sector under consideration, e.g. the particle number, magnetization and spatial symmetries directly into the RNN architecture[11,45]. Many of these challenges are inherent to the simulation of both bosonic and fermionic systems. Our results indicate that the bottleneck for simulating fermionic spinful systems is the training and not only the expressivity of the ansatz, and point the way to possible improvements concerning the ansatz and the training procedure.

Conclusions

To conclude, we present a neural network architecture, based on RNNs[11], to simulate ground states of the fermionic and bosonic $t - J$ model upon finite hole doping. We show that, despite many challenges due to the increased complexity of the learning problem compared to spin systems, the RNN succeeds in capturing physical properties like the shape of the dispersion, indicating the dominating emergent excitations of the systems. In order to calculate the dispersion, we present a method that can be used with any NQS ansatz and for any lattice geometry and map out quasiparticle dispersion using the RNN ansatz for several different lattice geometries, including 1D and 2D systems. Moreover, it enables an extremely efficient calculation of dispersion relations compared to conventional methods like DMRG[54], which usually require a time-evolution of the state[55]. The dispersion scheme yields a good agreement when comparing to exact diagonalization or DMRG results, and is expected to perform even better for a better ground state convergence. In principle, it can also be combined with a translationally symmetric NQS ansatz to improve the accuracy. Furthermore, the scheme could be combined with additional symmetries, e.g. rotational symmetries, enabling the calculation of m_4 rotational spectra[83].

Methods
The RNN ansatz and its bottlenecks
Given these relatively high errors on the ground state energies in some cases, we test potential bottlenecks of our approach in the Methods section, namely: (i) Difficulties in learning either the phase or the amplitude, by considering the partial learning problems separately. (ii) The optimization procedure. (iii) The optimization landscape. (iv) The expressivity of the RNN ansatz, compared to the complexity of the learning problem.

The partial learning problem. One potential bottleneck of our approach is the way the RNN wave function is split into amplitude and phase. In order to test if there are problems with the optimization of the phase or amplitude alone, we consider their learning problems separately as suggested e.g. in refs. 15,84.
1. Phase training: We sample from the exact ground state distribution $|\psi|^2$, calculated with ED, and optimize only the phase.
2. Amplitude training: Given the correct phase distribution from ED, we optimize only the logarithmic amplitude to check if the ground-state probability amplitudes can be learned.

Figure 7 shows the results of amplitude and phase trainings (dark and light blue), compared to the full training of both amplitude and phase (red).
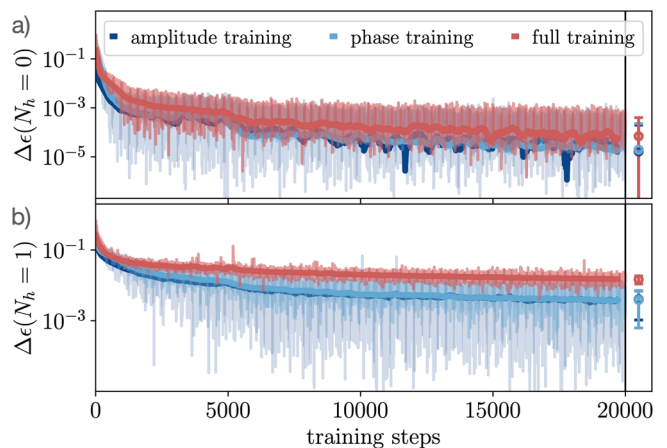

Fig. 7 | Partial training. Separate amplitude (dark blue) and phase (light blue) training, for ground states of the $t - J$ model on a 4×4 square lattice with $t/J = 3$, open boundaries and without holes $N_h = 0$ (**a**) and with a single hole $N_h = 1$ (**b**), compared to the full training in red. Thick lines and markers represent averages over 100 epochs (each with 200 samples) to guide the eye, with the respective standard deviation denoted by the error bars. We use a hidden dimension of $h_d = 70$.
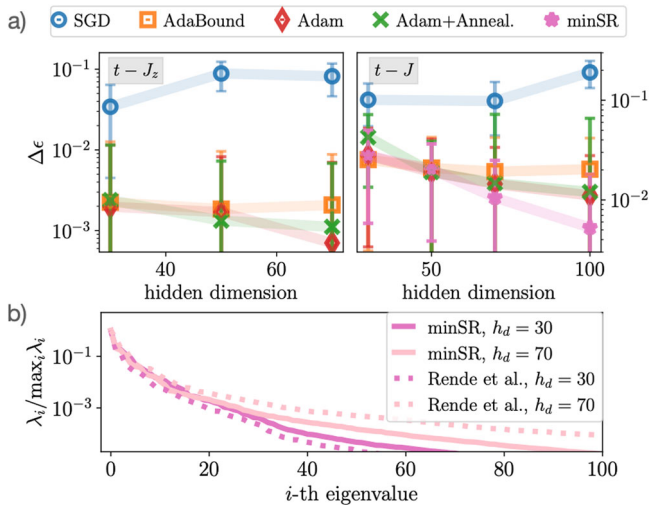
Fig. 8 | Testing different optimizers. a Optimization results for the $t-J_z$ model (left) and the $t-J$ model (right) on a 4×4 square lattice with $t/J_z = 3$, both for a single hole $N_h = 1$ and periodic boundaries, using stochastic gradient descent (SGD), AdaBound, Adam, Adam+Annealing and minimum-step stochastic reconfiguration (minSR), and 200 samples (1000 samples for minSR) in each variational Monte Carlo (VMC) step. All values are averages over the last 100 training steps, error bars denote the respective standard deviation. **b** Eigenvalues of the T-matrix (minSR algorithm[80], solid lines) and of the $X^T X$ matrix (stochastic reconfiguration variant of Rende et al.[81], dotted lines) before the training, for the 4×4 $t-J$ system with one hole and open boundaries and hidden dimensions $h_d = 30, 70$, using 1000 samples.

For all considered systems, the results of the partial trainings are closer to the exact ground state, e.g. for open boundaries and $N_h = 1$, the relative error is decreased from $\Delta\epsilon = 0.0147(37)$ to $\Delta\epsilon = 0.0040(30)$ for the amplitude training and $\Delta\epsilon = 0.0039(33)$ for the phase training. However, for all considered cases we observe the same problem as in the full training: the RNN gets stuck in a plateau that survives up to 20000 training steps. Although the relative error of the plateau decreases when considering the partial learning problems, the improvement is surprisingly low given the amount of information that is added to the training. Furthermore, whether the amplitude or phase training is more problematic remains unclear. Even for the phase training, for which the training samples are generated from the exact distribution $|\psi|^2$ calculated with ED, the improvement is not significantly larger than for the amplitude training. This is in agreement with the results of Bukov et al.[15]

Comparison of optimizers. As a next test, we compare the optimization results of different optimizers in Fig. 8a, namely Stochastic gradient descent (SGD), adaptive methods like AdaBound[85] and Adam[86], and more advanced methods such as Adam+Annealing[20] and the recently developed variant of stochastic reconfiguration (SR), minimum-step SR (minSR)[80]. We show the optimization results for the $t-J_z$ model on the left and the $t-J$ model on the right, both for $N_h = 1$.

Typically, Adam is used for RNN wave function optimization[11,20,45,87], adapting the learning rate in each VMC update. For 200 samples used in each optimization step, Adam yields relative errors on the order of $\Delta\epsilon \approx 10^{-3}$ for the $t-J_z$ model and $\Delta\epsilon \approx 10^{-2}$ for the $t-J$ model. AdaBound, employing dynamic bounds on learning rates, yielding a gradual transition from Adam to SGD during the training, has similar results.

Another modification of the Adam training is the use of variational annealing, shown to improve the performance for frustrated systems[20]. The idea of annealing is to avoid getting stuck in local minima by including an artificial temperature T in the learning process. In order to do so, the variational free energy of the model,

$$F_\lambda = \langle H_\lambda \rangle - T(n_{\text{step}}) \cdot S \tag{4}$$

instead of the energy (13) is minimized. Here, the averaged Hamiltonian $\langle H_\lambda \rangle$ is given by $\langle H_\lambda \rangle = \sum_\sigma |\psi_\lambda(\sigma)|^2 H_\lambda(\sigma)$. Furthermore, S denotes the Shannon entropy

$$S = -\sum_\sigma |\psi_\lambda(\sigma)|^2 \log\left[|\psi_\lambda(\sigma)|^2\right]. \tag{5}$$

The minimization procedure that we use starts with a warmup phase with a constant temperature T_0, before decreasing the temperature $T(t) = T_0(1 - (t - t_{\text{warmup}})/\tau)$ linearly with the minimization steps t. Typically, we use $\tau = 5000$ and stop the training after $t_{\text{final}} = 20000$ training iterations, but tests up to $\tau = 20000$ and $t_{\text{final}} = 40000$ did not yield any improvements. Figure 8a shows that for the square lattice, the use of annealing does not bring any advantage within the error bars.

Lastly, we apply two recently developed variant of stochastic reconfiguration (SR): minSR[80] and the linear algebra trick by Rende et al.[81]. Both methos are introduced later in the text, see Eqs. (17) and (18). For a stable training, we ensure non-exploding gradients by adding a diagonal offset $\delta(t)$ to the diagonals of the T-matrix, with $\delta(t)$ exponentially decaying from 1 to 10^{-10}. After determining the gradients using Eq. (17), we apply the Adam update rule, which we empirically find to perform better than the GD update. Moreover, since it is crucial to use enough samples for a sufficiently good approximation of the gradients in SR, typically more samples than for the other optimization routines are needed. Here, we use 1000 samples in each minSR update and find that the results on the one-hole $t-J$ ground state errors improve below the values obtained with Adam, see Fig. 8a on the right. However, we show in the Supplementary Note 2 that a comparison with Adam using the same number of samples does not lead to a conclusive result, which optimization routine is better, similar to the SR results in ref. 15.

The reason behind this can be understood when considering the spectrum of the T-matrix of the minSR algorithm: Similar to the results of ref. 82 for the S-matrix of the SR algorithm, Fig. 8b shows that the eigenvalues of T, λ_i, decrease extremely rapidly, in particular at the beginning of the training, indicating a very flat optimization landscape. This is a typical problem of autoregressive architectures[82] and causes uncontrolled, high values of T^{-1} and consequently also of the gradients $\delta\theta$, see Eq. (17). Furthermore, the shape of the spectrum does not have any feature that indicates that the spectrum could be cut off at a specific eigenvalue, making a regularization very difficult. Hence, the diagonal offset $\delta(t)$ must be chosen relatively large, yielding parameter updates that are very similar to the plain vanilla Adam optimization as long as $\delta(t)$ is larger than many of the T-eigenvalues. The spectrum of the $(X^T X)$ matrix of the SR variant by Rende et al.[81], see Eq. (18), exhibits the same problem.

When comparing the results for different hidden dimensions, e.g. for minSR in Fig. 8a (right), it may suggest that a hidden dimension $h_d > 100$ could in principle improve the results further. However, we will show later that for such a large number of parameters, it is even possible, by restricting to a fixed number of holes and hence reducing the Hilbert space dimension to $\ll 3^{N_{\text{sites}}}$, to encode the wave function using exact methods.

Symmetries. The RNN ansatz we use has implemented $U(1) = U(1)_{\hat{N}} \times U(1)_{\hat{S}_z}$ symmetry, i.e. a conserved total particle number and a total magnetization $\hat{S}_z = 0(0.5)$ for even (odd) particle numbers[11,25]. This is done by calculating the current particle number $N_p(i)$ (magnetization $S_z(i)$) after the i-th RNN cell during the sampling process and assigning a zero conditional probability if $N_p(i) = N_{\text{target}}$ ($S_z(i) = S_{z,\text{target}}$) for all sites $j > i$ that are considered afterwards, see Supplementary Note 1C. For even (odd) particle numbers, we use $S_{z,\text{target}} = 0$ ($S_{z,\text{target}} = 0.5$). As a next test, we employ additional spatial symmetries: For a symmetry operation $\mathcal{T}$ according to the lattice symmetry, we know that

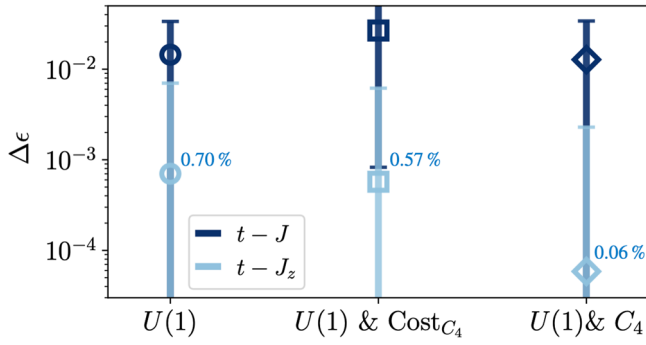$$|\psi(\sigma)|^2 = |\psi(\mathcal{T}\sigma)|^2 \tag{6}$$

Fig. 9 | The effect of U(1) and spatial symmetries. Relative error for $t-J$ (dark blue) and $t-J_z$ (light blue) models on a 4×4 square lattice with one hole, $t/J_z = 3$ and periodic boundaries, averaged over 100 training steps with 200 samples per step (error bars denote the respective standard deviation). We use a recurrent neural networks (RNN) with implemented $U(1) = U(1)_{\hat{N}} \times U(1)_{\hat{S}_z}$ symmetry, $U(1)$ and C_4 symmetry, implemented via the cost function and the RNN ansatz, with a hidden dimension of $h_d = 70$. For the $t-J_z$ model, we provide the relative energy errors as numbers in light blue.
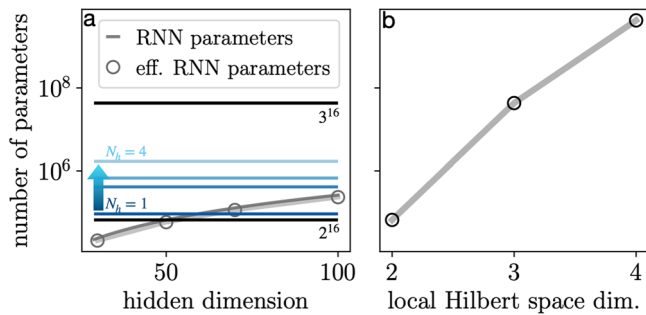

Fig. 10 | Number of parameters for the exact wave function of a 4×4 system compared to the recurrent neural network (RNN) ansatz. a We compare the number of parameters of the exact wave function using $U(1)_{\hat{N}} \times U(1)_{\hat{S}_z}$ symmetry for $0 \leq N_h \leq 4$ holes (blue) to the Hilbert space dimension 3^{16} that we want to learn with the RNN ansatz. The effective number of parameters of the RNN ansatz, i.e. the number of parameters that is kept when enforcing the $U(1)$ symmetry, with hidden dimension $30 \leq h_d \leq 100$ is denoted by the gray markers. **b** Hilbert space dimension for a local dimension of 2 (Heisenberg model), 3 ($t-J$ model) and 4 (Fermi-Hubbard model).

for the exact ground state. For rotational C_4 symmetry of the square lattice, we employ this constraint (i) in the training, by implementing it in the cost function, or (ii) in the RNN ansatz as in ref. 11.

The constraint in the cost function that we use in (i) is calculated by rotating all samples drawn from $|\psi_\lambda|^2$ according to C_4 in each VMC step, calculating $p_\lambda(\mathcal{T}_i\sigma) = |\psi_\lambda(\mathcal{T}_i\sigma)|^2$ for all $\{\mathcal{T}_i\}_i$ and adding the squared difference $\gamma(t)\sum_\sigma \left(|\psi_\lambda(\sigma)|^2 - |\psi_\lambda(\mathcal{T}_i\sigma)|^2\right)^2$ with a prefactor $\gamma(t) = \gamma_0\log_{10}(1 + 9(t - t_{\text{warmup}})/\tau)$ to the cost function. Typically, we use long decay times on the order of $\tau = 5000$ steps.

For (ii), we assign

$$p_\lambda(\sigma) = \frac{1}{|\{\mathcal{T}_i\}_i|}\sum_{\mathcal{T}=1,\{\mathcal{T}_i\}_i} |\psi_\lambda(\mathcal{T}\sigma)|^2 \qquad (7)$$

for all operations $\mathcal{T}_i$ in the symmetry group, similar to ref. 11. Note that this symmetrization scheme keeps the autoregressive property of the RNN, as pointed out e.g. in ref. 88.

The optimization results using (i) and (ii) are shown in Fig. 9 for the $t-J$ and $t-J_z$ model on a 4×4 square lattice. It can be seen that constraining the RNN wave function directly via (ii) is more succesful than via the cost function (i): Using (ii), we get an order of magnitude lower relative

errors compared to the results without spatial symmetries for the $t-J_z$ model. This possibly results from the fact that the additional constraint on the symmetry leads to barriers in the loss landscape in the regions where the symmetry is violated. Even when increasing the symmetry constraint gradually during the training, as described above, these barriers can prevent getting close to the minimum.

The $t-J$ model results do not improve significantly for both symmetry implementations (i) and (ii), with an error on the order of $\Delta\epsilon \approx 10^{-2}$ with and without spatial symmetries. Hence, we conclude that applying symmetries does only help to improve the accuracy if the ground state can already be learned sufficiently well, as for the $t-J_z$ model.

For systems with sufficiently high convergence, also rotational symmetries like s, p or d-wave symmetries could be enforced to probe the competition between the ground state energies in the respective symmetry sectors[60], which is highly relevant for the study of high-T_c superconductivity. In addition, also low-energy excited states for these symmetry sectors could be calculated by making use of the dispersion scheme presented in this work, e.g. m_4 rotational spectra[89].

Complexity of the learning problem. Lastly, we consider the complexity of our learning problem and compare it to the expressivity of our RNN ansatz in terms of the number of parameters that are encoded in the RNN. In Fig. 10 on the left, we show the number of parameters used in the RNN ansatz for the $4\times4\,t-J$ square lattice for hidden dimensions $30 \leq h_d \leq 100$. The number of parameters encoded in the ansatz is slightly lower than the number of parameters that is actually used (gray circles on the left). This is due to the way we encode the $U(1)$ symmetry in our approach, resulting in a small fraction of weights that are not updated since the respective probabilities are set to zero to obey the $U(1)$ symmetry, see Supplementary Note 1C. Furthermore, we show the dimension of the Hilbert space for the same system (with variable particle number) 3^{16} in black. For the small system size that we consider in Fig. 10, this Hilbert space dimension is two orders of magnitude larger than the number of RNN parameters. For the 10×4 system in Fig. 1 however, our RNN representation has 13 orders of magnitude less parameters than the Hilbert space with dimension 3^{40} that is learned.

The Hilbert space dimension $3^{N_{\text{sites}}}$ considered so far in this section allows for three states per site – spin up, down and hole –, i.e. for a variable number of holes in the system. For a fixed number of holes, the number of parameters to describe the exact state is given by all D combinations to distribute a fixed number of holes N_h, particles $N_\uparrow$ and $N_\downarrow$ on the N_{sites} sites, i.e.

$$D = \frac{N_{\text{sites}}!}{(N_\downarrow!)(N_{\text{sites}} - N_\downarrow)!} \cdot \frac{(N_\uparrow + N_h)!}{(N_\uparrow!N_h!)}, \qquad (8)$$

shown also in Fig. 6a and c (gray), with $D \ll 3^{N_{\text{sites}}}$, as shown by the blue lines in Fig. 10 for $1 \leq N_h \leq 4$. In fact, for $N_h = 1$ our RNNs encode even more parameters than this exact parameterization when $h_d > 70$. This reveals one main problem of our RNN ansatz, namely the way how the $U(1)$ symmetries are encoded: The fixed particle and magnetization Hilbert space dimension D is typically much smaller than the dimension of the RNN, since setting the RNN conditionals to zero during the sampling path corresponds to a dimension $3^x2^y1^z$ with $x + y + z = N_{\text{sites}}$ and typically y and z small. For future studies, we envision an RNN ansatz for a fixed number of holes in the spirit of Eq. (8), reducing the dimension of the parameter space that needs to be learned and hence facilitating the learning problem.

Lastly, we would like to point out that the learning problem that we consider here is more complex than for spin systems that are typically considered with this architecture[11,36,37,45], as can be seen when comparing the Hilbert space dimensions for local dimensions $d = 2$ as for spin systems, vs. $d = 3$ as for the $t-J$ model in Fig. 10 on the right. For larger systems, this difference increases, e.g. for the 10×4 system in Fig. 1 the Hilbert space dimension increases by seven orders of magnitude when going from a spin

to a $t - J$ system (with flexible number of holes). This problem becomes even more pronounced when the Fermi-Hubbard model with local dimension $d = 4$ would be considered.

NQS dispersion relations

Our dispersion scheme relies on an additional term in the cost function that penalizes momenta of the NQS away from the target momentum. The momentum $\boldsymbol{k}_{\mathrm{NQS}}$ of the NQS wave function is calculated from the translation operator $\hat{T}_{\boldsymbol{R}}$, which translates a state $\psi(\boldsymbol{r})$ by the respective vector $\boldsymbol{R}$, i.e. $\hat{T}_{\boldsymbol{R}}\psi(\boldsymbol{r}) = \psi(\boldsymbol{r} - \boldsymbol{R})$. Furthermore, it can be written as[90]

$$\hat{T}_{\boldsymbol{R}} = \exp\left(-i\boldsymbol{R} \cdot \hat{\boldsymbol{k}}\right), \tag{9}$$

with the momentum operator $\hat{\boldsymbol{k}}$. To determine the expectation value $\boldsymbol{k}_{\mathrm{NQS}} = (k_x, k_y)$ using samples $\boldsymbol{\sigma}$ drawn from the NQS wave function, we calculate the expectation value of $\hat{T}_{\boldsymbol{R}}$. For example, for a square lattice, this is done by translating all snapshots by $\boldsymbol{R} = \boldsymbol{e}_x$ and $\boldsymbol{R} = \boldsymbol{e}_y$ with $|\boldsymbol{e}_\mu| = a$ for lattice distance a and $\mu = x, y$. Then, we calculate the respective translated states, $\psi_\lambda(\hat{T}_{\boldsymbol{e}_\mu}\boldsymbol{\sigma})$, to determine the expectation value

$$\langle\psi_\lambda|\hat{T}_{\boldsymbol{e}_\mu}|\psi_\lambda\rangle = \exp\left(-i\boldsymbol{e}_\mu \cdot \boldsymbol{k}_{\mathrm{NQS}}\right) \approx \frac{1}{N_s}\sum_i \frac{\psi_\lambda(\hat{T}_{\boldsymbol{e}_\mu}\sigma_i)}{\psi_\lambda(\sigma_i)}, \tag{10}$$

with the first equality due to the translational invariance of the ground state of a square lattice, which we assume to be (approximately) present for our NQS ground states, see also Supplementary Note 3. Hence,

$$\boldsymbol{k}_{\mathrm{NQS}}^\mu = \frac{i}{a}\log\langle\psi_\lambda|\hat{T}_{\boldsymbol{e}_\mu}|\psi_\lambda\rangle. \tag{11}$$

Using a sufficiently converged NQS ground state wave function as initial state, we train using VMC with an additional term in the loss function,

$$\mathcal{C}(\boldsymbol{k}_{\mathrm{target}}) = \gamma(t)\sum_\mu \left(\boldsymbol{k}_{\mathrm{NQS}}^\mu - \boldsymbol{k}_{\mathrm{target}}^\mu\right)^2, \tag{12}$$

with the RNN momentum $\boldsymbol{k}_{\mathrm{NQS}}$ and the target momentum $\boldsymbol{k}_{\mathrm{target}}$. We use a prefactor $\gamma(t) = \gamma_0 \log_{10}(1 + 9(t - t_{\mathrm{warmup}})/\tau)$ that is turned on with typically $\tau = 100, \ldots, 1000$ and $\gamma_0 = 1, \ldots, 10$ and gradually lifts all areas in the loss landscape that correspond to a NQS wave function with momentum $\boldsymbol{k}_{\mathrm{NQS}} \neq \boldsymbol{k}_{\mathrm{target}}$, forcing the NQS to a higher energy state at momentum $\boldsymbol{k}_{\mathrm{NQS}} = \boldsymbol{k}_{\mathrm{target}}$, see Fig. 2.

For $\boldsymbol{k}_{\mathrm{target}}$ far away from the ground state momentum, we observe empirically that the imaginary part of $\boldsymbol{k}_{\mathrm{NQS}}$ can become large, on the same order as the real part, in particular if the ground state accuracy was not sufficiently high. In these cases, the RNN ends up in states that are not eigenstates of the momentum operator. In order to prevent our RNN wave function to get trapped in these states we apply an additional constraint in the loss function in these cases, penalizing large imaginary parts of the momentum, Im $\boldsymbol{k}_{\mathrm{NQS}}$.

Architecture and training

In the present paper we use a recurrent neural network (RNN)[91] to represent a quantum state defined on a 2D lattice with $N_{\mathrm{sites}} = N_x \cdot N_y$ positions occupied by N_p particles. RNNs and similar generative architectures combined with variational energy minimization have already been applied successfully for spin systems[5,11,36,45]. One of the advantages of these architectures is their autoregressive property, which allows extremely efficient independent sampling from the RNN wave function[19,92], which is important for the training procedure.

In order to represent fermionic wave functions, we start from the same approach as for bosonic spin systems and use an RNN architecture consisting of N_{sites} (tensorized) gated recurrent units (GRUs), each one representing one site of the system. The information is passed from the first cell, corresponding to the first lattice site, to the last site in a recurrent fashion, see Supplementary Note 1a.

In order to find the ground state of the system under consideration, we use the variational Monte Carlo (VMC) minimization of the energy[92,93]. VMC has already been used in a wide range of machine learning applications (see e.g. refs. 6,18 for an overview). In VMC, the expectation value of the energy of the RNN trial wave function,

$$\langle E_\lambda\rangle = \sum_\sigma |\psi_\lambda(\sigma)|^2 E_\lambda^{\mathrm{loc}}(\sigma) \approx \frac{1}{N_s}\sum_i E_\lambda^{\mathrm{loc}}(\sigma_i), \tag{13}$$

is minimized. Here, we have defined the local energy

$$E_\lambda^{\mathrm{loc}}(\sigma) = \frac{\langle\sigma|\mathcal{H}|\psi_\lambda\rangle}{\langle\sigma|\psi_\lambda\rangle}. \tag{14}$$

As shown e.g. in refs. 11,29 one can use the cost function

$$\mathcal{C} = \frac{1}{N_s}\sum_i \underbrace{\left[E_\lambda^{loc}(\sigma_i) - \langle E_\lambda^{loc}\rangle\right]}_{=:-\sqrt{N_s}\bar{\epsilon}(\sigma_i)} \tag{15}$$

to minimize both the local energy as well as the variance of the local energy to make the training more stable. In Eq. (15), we have defined $\bar{\epsilon}(\sigma_i) := -\frac{1}{\sqrt{N_s}}\left[E_\lambda^{loc}(\sigma_i) - \langle E_\lambda^{loc}\rangle\right]$, where N_s denotes the number of samples.

One of the main difficulties of neural network quantum states is the optimization of Eq. (15), due to its typically rugged landscape with many local minima and saddle points[15]. If not stated differently, we use the Adam optimizer[86] for the optimization of Eq. (15), following previous works on NQS using RNNs[10,11,45]. To improve the optimization, often stochastic reconfiguration (SR)[94,95] is used. Here, we use two recently proposed, SR variants, namely minimum-step stochastic reconfiguration (minSR) and the SR variant based on a linear algebra trick by Rende et al.[81]. In contrast to conventional SR, these variants enable the use of a large numbers of NQS parameters, see Supplementary Note 2. In minSR, each parameter λ_k of the neural network is optimized individually according to

$$\bar{O}_{\sigma_i k}\,\delta\lambda_k = \bar{\epsilon}(\sigma_i), \tag{16}$$

where σ_i denote the sample configurations, k the parameter index, $O_{\sigma_i k} = \frac{1}{\psi(\sigma_i)}\frac{\partial\psi(\sigma_i)}{\partial\lambda_k}$ and $\bar{O}_{\sigma_i k} = (O_{\sigma_i k} - \langle O_{\sigma_i k}\rangle)/\sqrt{N_s}$. Eq. (16) is solved by

$$\delta\lambda_k = \bar{O}_{k\sigma_i'}^\dagger(T^{-1})_{\sigma_i'\sigma_i}\bar{\epsilon}(\sigma_i), \tag{17}$$

with $T = \bar{O}\bar{O}^\dagger$[80]. In the version of Rende et al.,

$$\delta\lambda_k = X_{k\sigma_i'}(X^T X)_{\sigma_i'\sigma_i}^{-1} f_{\sigma_i}, \tag{18}$$

with $X = \mathrm{Concat}(\mathrm{Re}\,\bar{O}, \mathrm{Im}\,\bar{O})$ and $f_{\sigma_i} = \mathrm{Concat}(\mathrm{Re}\,\bar{\epsilon}(\sigma_i), -\mathrm{Im}\,\bar{\epsilon}(\sigma_i))$[81].

Fermionic RNN Wave Functions. The architecture introduced above is per se bosonic. When considering fermionic systems, we need to take the antisymmetry of the wave function into account. This antisymmetry is included during the variational Monte Carlo steps when calculating the local energy introduced in Eq. (14). We can expand the local energy to

$$E_{loc}(\sigma_i) = \sum_{\sigma_i'} \frac{\langle\sigma_i|H|\sigma_i'\rangle\langle\sigma_i'|\psi_\lambda\rangle}{\langle\sigma_i|\psi_\lambda\rangle}. \tag{19}$$

In this sum, we multiply each term with a factor $(-1)^P$ if σ' is connected to σ by P two-particle permutations, as suggested in ref. 29. In order to do so, we
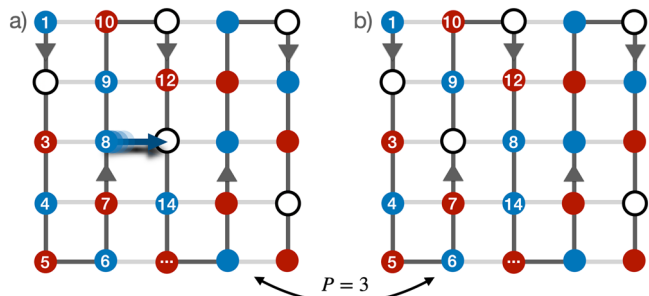
Fig. 11 | Jordan-Wigner strings on the level of NQS expectation values. a) A typical configuration σ for a 5×5 system with five holes and ten spin up (red) and spin down (blue) particles each. Sites are labeled in a 1D manner, as denoted by the white numbers. An exemplary hopping process (arrow in a) to the nearest neighbor in horizontal direction ends in the configuration σ' (**b**) and effectively exchanges P particles, here $P = 3$. The respective sign of σ' relative to σ is calculated using Eq. (19).

take the permutations along the sampling path into account. For the $t - XXZ$ Hamiltonian under consideration we only need to consider the hopping term for calculating the antisymmetric signs. An example is shown in Fig. 11. This procedure is equivalent to the implementation of Jordan-Wigner strings as e.g. in ref. 25.

Data availability
The data of all figures of this paper is provided in https://github.com/HannahLange/Fermionic-RNNs/in the `data` folder. Other data are available from the corresponding author on request.

Code availability
The code used for this paper is provided here: https://github.com/HannahLange/Fermionic-RNNs/in the `src` and `src_dispersion` folders.

References
1. Carleo, G. & Troyer, M. Solving the quantum many-body problem with artificial neural networks. *Science* **355**, 602–606 (2017).
2. Torlai, G. & Melko, R. G. Learning thermodynamics with boltzmann machines. *Phys. Rev. B* **94**, 165134 (2016).
3. Torlai, G. et al. Neural-network quantum state tomography. *Nat. Phys.* **14**, 447–450 (2018).
4. Torlai, G. et al. Integrating neural networks with a quantum simulator for state reconstruction. *Phys. Rev. Lett.* **123**, 230504 (2019).
5. Carrasquilla, J., Torlai, G., Melko, R. G. & Aolita, L. Reconstructing quantum states with generative models. *Nat. Mach. Intell.* **1**, 155–161 (2019).
6. Melko, R. G., Carleo, G., Carrasquilla, J. & Cirac, J. I. Restricted boltzmann machines in quantum physics. *Nat. Phys.* **15**, 1745–2481 (2019).
7. Schmale, T., Reh, M. & Gärttner, M. Efficient quantum state tomography with convolutional neural networks. *npj Quantum Inf.* **8**, 115 (2022).
8. Roth, C. & MacDonald, A. H. Group convolutional neural networks improve quantum state accuracy http://arxiv.org/abs/2104.05085 (2021).
9. Rocchetto, A., Grant, E., Strelchuk, S., Carleo, G. & Severini, S. Learning hard quantum distributions with variational autoencoders. *npj Quantum Inf.* **4**, 28 (2018).
10. Morawetz, S., De Vlugt, Isaac, J. S., Carrasquilla, J. & Melko, R. G. U(1)-symmetric recurrent neural networks for quantum state reconstruction. *Phys. Rev. A* **104**, 012401 (2021).
11. Hibat-Allah, M., Ganahl, M., Hayward, L. E., Melko, R. G. & Carrasquilla, J. Recurrent neural network wave functions. *Phys. Rev. Res.* **2**, 023358 (2020).
12. Sharir, O., Levine, Y., Wies, N., Carleo, G. & Shashua, A. Deep autoregressive models for the efficient variational simulation of many-body quantum systems. *Phys. Rev. Lett.* **124**, 020503 (2020).
13. Luo, D. et al. Gauge-invariant and anyonic-symmetric autoregressive neural network for quantum lattice models. https://doi.org/10.1103/PhysRevResearch.5.013216 (2023).
14. Luo, D., Chen, Z., Carrasquilla, J. & Clark, B. K. Autoregressive neural network for simulating open quantum systems via a probabilistic formulation. *Phys. Rev. Lett.* **128**, 090501 (2022).
15. Bukov, M., Schmitt, M. & Dupont, M. Learning the ground state of a non-stoquastic quantum Hamiltonian in a rugged neural network landscape. *SciPost Phys.* **10**, 147 (2021).
16. Uria, B., Côté, Marc-Alexandre, Gregor, K., Murray, I. & Larochelle, H. Neural autoregressive distribution estimation. *J. Mach. Learn. Res.* **17**, 1–37 (2016).
17. Humeniuk, S., Wan, Y. & Wang, L. Autoregressive neural Slater-Jastrow ansatz for variational Monte Carlo simulation. https://doi.org/10.21468/SciPostPhys.14.6.171 (2023).
18. Carrasquilla, J. & Torlai, G. How to use neural networks to investigate quantum many-body physics. *PRX Quantum* **2**, 040201 (2021).
19. Wu, D., Wang, L. & Zhang, P. Solving statistical mechanics using variational autoregressive networks. *Phys. Rev. Lett.* **122**, 080602 (2019).
20. Hibat-Allah, M., Melko, R. G. & Carrasquilla, J. Supplementing recurrent neural network wave functions with symmetry and annealing to improve accuracy. https://doi.org/10.48550/ARXIV.2207.14314 (2022).
21. Hibat-Allah, M., Melko, R. G. & Carrasquilla, J. Investigating topological order using recurrent neural networks. https://doi.org/10.1103/PhysRevB.108.075152 (2023).
22. Keimer, B., Kivelson, S. A., Norman, M. R., Uchida, S. & Zaanen, J. From quantum matter to high-temperature superconductivity in copper oxides. *Nature* **518**, 179–186 (2015).
23. Pfau, D., Spencer, J. S., Matthews, Alexander, G. D. G. & Foulkes, W. M. C. Ab initio solution of the many-electron schrödinger equation with deep neural networks. *Phys. Rev. Res.* **2**, 033429 (2020).
24. Spencer, J. S., Pfau, D., Botev, A. & Foulkes, W. M. C. Better, faster fermionic neural networks. https://doi.org/10.48550/ARXIV.2011.07125 (2020).
25. Barrett, T. D., Malyshev, A. & Lvovsky, A. I. Autoregressive neural-network wavefunctions for ab initio quantum chemistry. *Nat. Mach. Intell.* **4**, 2522–5839 (2022).
26. Choo, K., Carleo, G., Regnault, N. & Neupert, T. Symmetries and many-body excitations with neural-network quantum states. *Phys. Rev. Lett.* **121**, 167204 (2018).
27. Hermann, J., Schätzle, Z. & Noé, F. Deep-neural-network solution of the electronic schrödinger equation. *Nat. Chem.* **12**, 1755–4349 (2020).
28. Nomura, Y., Darmawan, A. S., Yamaji, Y. & Imada, M. Restricted boltzmann machine learning for solving strongly correlated quantum systems. *Phys. Rev. B* **96**, 205152 (2017).
29. Inui, K., Kato, Y. & Motome, Y. Determinant-free fermionic wave function using feed-forward neural networks. *Phys. Rev. Res.* **3**, 043126 (2021).
30. Luo, D. & Clark, B. K. Backflow transformations via neural networks for quantum many-body wave functions. *Phys. Rev. Lett.* **122**, 226401 (2019).
31. Robledo Moreno, J. Carleo, G., Georges, A. & Stokes, J. Fermionic wave functions from neural-network constrained hidden states. *Proc. Natl Acad. Sci.* **119**, e2122059119 (2022).
32. Yoshioka, N., Mizukami, W. & Nori, F. Solving quasiparticle band spectra of real solids using neural-network quantum states. *Commun. Phys.* **4**, 2399–3650 (2021).

33. Bengio, Y., Simard, P. & Frasconi, P. Learning long-term dependencies with gradient descent is difficult. *IEEE Trans. Neural Netw.* **5**, 157–166 (1994).

34. Schäfer, A. M., Udluft, S. & Zimmermann, H.-G. Learning long term dependencies with recurrent neural networks, in *Artificial Neural Networks – ICANN 2006*, edited by Kollias, S. D., Stafylopatis, A., Duch, Włodzisław, & Oja, E. (Springer Berlin Heidelberg, Berlin, Heidelberg, 2006) pp. 71–80.

35. Pascanu, R., Mikolov, T. & Bengio, Y. On the difficulty of training recurrent neural networks, *Proc. 30th Int. Conf. Mach. Learn.* **28**, 1310–1318 (2013).

36. Czischek, S., Moss, M. S., Radzihovsky, M., Merali, E. & Melko, R. G. Data-enhanced variational monte carlo simulations for rydberg atom arrays. *Phys. Rev. B* **105**, 205108 (2022).

37. Moss, M. S. et al. Enhancing variational monte carlo using a programmable quantum simulator. http://arxiv.org/abs/2308.02647 (2023).

38. Auerbach, A. *Interacting Electrons and Quantum Magnetism -* (Springer Science & Business Media, Berlin Heidelberg, 2012).

39. Qin, M. et al. Absence of superconductivity in the pure two-dimensional hubbard model. *Phys. Rev. X* **10**, 031016 (2020).

40. White, S. R. & Scalapino, D. J. Competition between stripes and pairing in a $t − t' − j$ model. *Phys. Rev. B* **60**, R753–R756 (1999).

41. Himeda, A., Kato, T. & Ogata, M. Stripe states with spatially oscillating d-wave superconductivity in the two-dimensional $t − t' − J$ model. *Phys. Rev. Lett.* **88**, 117001 (2002).

42. Luo, D., Dai, D. D. & Fu, L. Pairing-based graph neural network for simulating quantum materials. http://arxiv.org/abs/2311.02143 [cond-mat.str-el] (2023).

43. Kim, J. et al. Neural-network quantum states for ultra-cold fermi gases. http://arxiv.org/abs/2305.08831 (2023).

44. Lou, Wan Tong et al. Neural wave functions for superfluids http://arxiv.org/abs/2305.06989 (2024).

45. Roth, C. Iterative retraining of quantum spin models using recurrent neural networks. https://arxiv.org/abs/2003.06228 (2020).

46. Marshall, W. Antiferromagnetism. https://royalsocietypublishing.org/doi/10.1098/rspa.1955.0200 (1955).

47. Koepsell, J. et al. Imaging magnetic polarons in the doped fermi–hubbard model. *Nature* **572**, 358–362 (2019).

48. Schäfer, T. et al. Tracking the footprints of spin fluctuations: A multimethod, multimessenger study of the two-dimensional hubbard model. *Phys. Rev. X* **11**, 011058 (2021).

49. Xu, H. et al. Coexistence of superconductivity with partially filled stripes in the hubbard model. *Science* **384**, eadh7691 (2024).

50. Arovas, D. P., Berg, E., Kivelson, S. A. & Raghu, S. The hubbard model. *Annu. Rev. Condens. Matter Phys.* **13**, 239–274 (2022).

51. Schollwöck, U. The density-matrix renormalization group in the age of matrix product states. *Ann. Phys.* **326**, 96–192 (2011).

52. White, S. R. Density matrix formulation for quantum renormalization groups. *Phys. Rev. Lett.* **69**, 2863–2866 (1992).

53. Bohrdt, A., Demler, E., Pollmann, F., Knap, M. & Grusdt, F. Parton theory of angle-resolved photoemission spectroscopy spectra in antiferromagnetic mott insulators. *Phys. Rev. B* **102**, 035139 (2020).

54. Vanderstraeten, L., Mariën, Michaël, Verstraete, F. & Haegeman, J. Excitations and the tangent space of projected entangled-pair states. *Phys. Rev. B* **92**, 201111 (2015).

55. Van Damme, M., Vanhove, R., Haegeman, J., Verstraete, F. & Vanderstraeten, L. Efficient matrix product state methods for extracting spectral information on rings and cylinders. *Phys. Rev. B* **104**, 115142 (2021).

56. Malyshev, A., Arrazola, J. M., & Lvovsky, A. I. Autoregressive neural quantum states with quantum number symmetries. http://arxiv.org/abs/2310.04166 (2023).

57. Nomura, Y. Machine learning quantum states — extensions to fermion-boson coupled systems and excited-state calculations. *J. Phys. Soc. Jpn.* **89**, 054706 (2020).

58. Viteritti, LucianoLoris, Ferrari, F. & Becca, F. Accuracy of restricted Boltzmann machines for the one-dimensional $J_1 − J_2$ Heisenberg model. *SciPost Phys.* **12**, 166 (2022).

59. Bohrdt, A., Greif, D., Demler, E., Knap, M. & Grusdt, F. Angle-resolved photoemission spectroscopy with quantum gas microscopes. *Phys. Rev. B* **97**, 125117 (2018).

60. Leung, P. W. Low-energy states with different symmetries in the t-j model with two holes on a 32-site lattice. *Phys. Rev. B* **65**, 205101 (2002).

61. Mendes-Santos, T., Schmitt, M. & Heyl, M. Highly resolved spectral functions of two-dimensional systems with neural quantum states. *Phys. Rev. Lett.* **131**, 046501 (2023).

62. Hendry, D. & Feiguin, A. E. Machine learning approach to dynamical properties of quantum many-body systems. *Phys. Rev. B* **100**, 245123 (2019).

63. Hendry, D., Chen, H., Weinberg, P. & Feiguin, A. E. Chebyshev expansion of spectral functions using restricted boltzmann machines. *Phys. Rev. B* **104**, 205130 (2021).

64. Charlebois, M. & Imada, M. Single-particle spectral function formulated and calculated by variational monte carlo method with application to d-wave superconducting state. *Phys. Rev. X* **10**, 041023 (2020).

65. Li, D. et al. Superconductivity in an infinite-layer nickelate. *Nature* **572**, 624–627 (2019).

66. Sun, H. et al. Realization of a bosonic antiferromagnet. *Nat. Phys.* **17**, 990–994 (2021).

67. Béran, P., Poilblanc, D. & Laughlin, R. B. Evidence for composite nature of quasiparticles in the 2d t-j model. *Nucl. Phys. B* **473**, 707–720 (1996).

68. Grusdt, F. et al. Parton theory of magnetic polarons: Mesonic resonances and signatures in dynamics. *Phys. Rev. X* **8**, 011046 (2018).

69. Laughlin, R. B. Evidence for quasiparticle decay in photoemission from underdoped cuprates. *Phys. Rev. Lett.* **79**, 1726–1729 (1997).

70. Haerter, J. O. & Shastry, B. S. Kinetic antiferromagnetism in the triangular lattice. *Phys. Rev. Lett.* **95**, 087202 (2005).

71. Schlömer, H., Schollwöck, U., Bohrdt, A. & Grusdt, F. Kinetic-to-magnetic frustration crossover and linear confinement in the doped triangular $t − j$ model. http://arxiv.org/abs/2305.02342 (2023).

72. Struck, J. et al. Quantum simulation of frustrated classical magnetism in triangular optical lattices. *Science* **333**, 996–999 (2011).

73. Tang, Y. et al. Simulation of hubbard model physics in wse2/ws2 moirésuperlattices. *Nature* **579**, 353–358 (2020).

74. Xu, M. et al. Frustration- and doping-induced magnetism in a fermi–hubbard simulator. *Nature* **620**, 971–976 (2023).

75. Yamamoto, R., Ozawa, H., Nak, D. C., Nakamura, I. & Fukuhara, T. Single-site-resolved imaging of ultracold atoms in a triangular optical lattice. *N. J. Phys.* **22**, 123028 (2020).

76. Wu, F., Lovorn, T., Tutuc, E. & MacDonald, A. H. Hubbard model physics in transition metal dichalcogenide moiré bands. *Phys. Rev. Lett.* **121**, 026402 (2018).

77. Davydova, M., Zhang, Y. & Fu, L. Itinerant spin polaron and metallic ferromagnetism in semiconductor moiré superlattices. *Phys. Rev. B* **107**, 224420 (2023).

78. Schlömer, H. et al. Quantifying hole-motion-induced frustration in doped antiferromagnets by hamiltonian reconstruction. https://doi.org/10.1038/s43246-023-00382-3 (2023).

79. Sinibaldi, A., Giuliani, C., Carleo, G. & Vicentini, F. Unbiasing time-dependent variational monte carlo by projected quantum evolution. http://arxiv.org/abs/2305.14294 (2023).

80. Chen, A. & Heyl, M., Efficient optimization of deep neural quantum states toward machine precision. http://arxiv.org/abs/2302.01941 arXiv:2302.01941 (2023).

81. Rende, R., Viteritti, Luciano Loris, Bardone, L., Becca, F. & Goldt, S. A simple linear algebra identity to optimize large-scale neural network quantum states. http://arxiv.org/abs/2310.05715 (2023).

82. Donatella, K., Denis, Z., Le Boité, A. & Ciuti, C. Dynamics with autoregressive neural quantum states: Application to critical quench dynamics. *Phys. Rev. A* **108**, 022210 (2023).
83. Bohrdt, A., Demler, E. & Grusdt, F. Rotational resonances and regge-like trajectories in lightly doped antiferromagnets. *Phys. Rev. Lett.* **127**, 197004 (2021).
84. Wang, J.-Q., He, R.-Q. & Lu, Z.-Y. Variational optimization of the amplitude of neural-network quantum many-body ground states. http://arxiv.org/abs/2308.09664 (2023).
85. Luo, L., Xiong, Y., Liu, Y. & Sun, X. Adaptive gradient methods with dynamic bound of learning rate, in *Proc. 7th Int. Conf. Learn. Represent.* (New Orleans, Louisiana, 2019).
86. Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization http://arxiv.org/abs/1412.6980 (2017).
87. Hibat-Allah, M., Inack, E. M., Wiersema, R., Melko, R. G. & Carrasquilla, J. Variational neural annealing. *Nat. Mach. Intell.* **3**, 2522–5839 (2021).
88. Reh, M., Schmitt, M. & Gärttner, M. Optimizing design choices for neural quantum states. *Phys. Rev. B* **107**, 195115 (2023).
89. Bohrdt, A., Demler, E. & Grusdt, F. Dichotomy of heavy and light pairs of holes in the tj model. https://doi.org/10.1038/s41467-023-43453-2 (2023).
90. Shankar, R. *Principles of quantum mechanics* (Plenum, New York, NY, 1980). https://cds.cern.ch/record/102017.
91. Hochreiter, S. & Schmidhuber, J. ürgen Long Short-Term Memory. *Neural Comput.* **9**, 1735–1780 (1997).
92. Goodfellow, I., Bengio, Y. & Courville, A. *Deep Learning* (MIT Press, 2016) http://www.deeplearningbook.org.
93. Becca, F. & Sorella, S. *Quantum Monte Carlo Approaches for Correlated Systems* (Cambridge University Press, 2017). https://doi.org/10.1017/9781316417041.
94. Stokes, J., Izaac, J., Killoran, N. & Carleo, G. Quantum Natural Gradient. *Quantum* **4**, 269 (2020).
95. Sorella, S. Green function monte carlo with stochastic reconfiguration. *Phys. Rev. Lett.* **80**, 4558–4561 (1998).

Acknowledgements

Author contributions

H.L. performed all calculations and wrote large parts of the manuscript. A.B. supervised the work. H.L., F.D., J.C and A.B. jointly analyzed the results and contributed to writing the manuscript.

Funding

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42005-024-01678-7.

Correspondence and requests for materials should be addressed to Hannah Lange.

Peer review information *Communications Physics* thanks Michele Casula, Douglas Hendry and the other, anonymous, reviewer(s) for their contribution to the peer review of this work. A peer review file is available.

Reprints and permissions information is available at http://www.nature.com/reprints

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.