

HONOURS PROJECT IN PHYSICS

QUANTUM SIMULATION OF QUANTUM
PHASES OF MATTER WITH INTERACTING
PHOTONS

Zejian LI
Supervisor: Dimitris G. ANGELAKIS
Co-supervisor: Valerio SCARANI

A thesis submitted to the
Physics department, National University of Singapore
in partial fulfillment for the degree of Bachelor of Science
with Honours in Physics



DECLARATION

I hereby declare that this thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in this thesis.

This thesis has also not been submitted for any degree in any university previously.

Date 05/04/2019

Signature 

Publications

- [1] Zejian Li, Jirawat Tangpanitanon and Dimitris G Angelakis. “Artificial Neural Networks for Learning Quantum Phases of Hybrid Light-Matter Systems”. Submitted to the 2019 IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC) - *Special Session on Quantum Information Processing*.

Acknowledgements

I would like to sincerely thank my supervisor Dimitris Angelakis, who accepted me to his group and guided me throughout the entire project, for the priceless discussions and ideas, which allowed me to overcome the difficulties, for the precious materials he provided and the talks, seminars and group meetings he organised, which deepened my understandings in quantum optics and many-body physics, without which this work would not have been possible.

I would also like to thank all my group-mates for their help and collaboration, with special thanks to Jirawat Tangpanitanon, who helped me solve many problems I encountered and provided guidelines for my progress; to Supanut Thanasilp and Marc-Antoine Lemonde, for the daily discussions and exchanges of thoughts, and to Wing Chi Yu who helped me on the topic of machine learning.

And many thanks to Rui Jiang for helping me in proofreading this thesis.

Finally, I would like to thank Centre for Quantum Technologies, National University of Singapore and École Polytechnique, who made this project possible.

Abstract

We are now in the midst of the second quantum revolution, where one seeks to exploit the fundamental rules of quantum mechanics so as to construct hardware that allows us to control quantum systems at an individual level, from the invention of atomic clocks to the race for quantum supremacy that is on the rise around the globe. A special type of quantum hardware called the quantum simulator, as envisioned by Richard Feynman in 1982, is one that can efficiently mimic other quantum systems by exploiting quantum parallelism and is being brought to reality thanks to recent advances in technology.

Quantum simulators can be extremely useful for solving quantum many-body problems, the computational resource required for which, if simulated on a classical computer, would grow exponentially with the number of particles in the system. Meanwhile, most of the phenomena present in real-life materials arise from many-body problems, such as the high-Tc superconductivity.

In this thesis, we will first review the concept of quantum simulation and some of its major platforms. We then study the Bose-Hubbard model, which is a simple many-body interacting model that gives rise to the Mott-to-superfluid quantum phase transition, that has been observed with a quantum simulator implemented with cold atoms in optical lattices.

Inspired by experimental advances in nonlinear quantum optics, in the second part of the thesis, we theoretically study the quantum simulator consisting of interacting photons in coupled resonator arrays, described by the Jaynes-Cummings-Hubbard model, that can exhibit quantum many-body phases of strongly correlated photons.

In recent decades, there has been tremendous development in machine learning and artificial neural networks, and we are currently witnessing a virtuous and exciting cross-fertilisation between machine learning and physical sciences. Among the various interfaces, artificial neural networks are emerging as a powerful tool for solving many-body problems, with successful applications to lattice models.

Inspired by these successes, in the last part of the thesis, we present a variational method of probing quantum phases in the Bose-Hubbard and the Jaynes-Cummings-Hubbard models based on artificial neural networks and machine learning algorithms, as well as the novel results that we have obtained. Our work provides a numerical tool for benchmarking the quantum simulator for small system sizes on a classical computer, and suggests that the choice of an optimal artificial neural network in studying quantum many-body system is problem specific.

Contents

1	Introduction	8
1.1	Quantum simulation	9
1.1.1	Major platforms for quantum simulation	9
1.2	Quantum simulation of quantum phase transition	10
1.3	Bose-Hubbard model	11
1.3.1	Strong interaction regime ($U \gg t$)	12
1.3.2	Weak interaction regime ($t \gg U$)	12
1.3.3	Mean-field theory	13
1.3.4	Exact diagonalisation	14
2	Quantum many-body phases of strongly interacting photons	17
2.1	Quantisation of light	17
2.2	Jaynes-Cummings model	19
2.3	Photon blockade effect	21
2.4	Jaynes-Cummings-Hubbard model	21
2.4.1	Mean-field theory	22
2.5	Techniques for probing quantum many-body phases	24
3	Machine learning quantum phases of matter	27
3.1	Machine learning basics	27
3.2	Machine learning the ground state of a quantum system	28
3.2.1	Artificial neural networks for strongly correlated lattice models	29
3.2.2	Variational Monte Carlo for the ground state of lattice models	33
3.3	BH and JCH phases via machine learning	35
3.3.1	Machine learning results for the BH model	35
3.3.2	Machine learning results for the JCH model	37
3.4	Discussion	38
3.5	Conclusion	39
A	Programming codes for numerical solutions	44
A.1	Python code for calculating the numerical mean-field phase diagram of the BH model	44
A.2	Matlab code for the exact diagonalisation of the BH model in grand canonical ensemble	45

A.3	Python code for the exact diagonalisation of the BH model in canonical ensemble	48
A.4	Matlab code for the exact diagonalisation of the JCH model in canonical ensemble	50
A.5	Python code for calculating the numerical mean-field phase diagram of the JCH model	54

List of Figures

1.1	Schematic of the Bose-Hubbard model	12
1.2	Mean-field phase diagrams of the Bose-Hubbard model	14
1.3	Phase diagram of Bose-Hubbard model in grand canonical ensemble from exact diagonalisation	15
1.4	Phase diagram of Bose-Hubbard model in canonical ensemble from exact diagonalisation	15
2.1	Schematic of an ideal single-mode cavity	17
2.2	Schematic of the Jaynes-Cummings-Hubbard model, the energy spectrum and the phase transition diagram from exact diagonalisation. . .	23
2.3	Mean-field phase diagrams of the Jaynes-Cummings-Hubbard model .	25
3.1	Schematics of the artificial neural networks	31
3.2	Machine learning results for the Bose-Hubbard model	36
3.3	Variation of the cost function in machine-learning the Jaynes-Cummings-Hubbard model	37
3.4	Machine learning results for the Jaynes-Cummings-Hubbard model .	38

Chapter 1

Introduction

The idea of simulating one physical system with another has been present throughout the history of human civilisation. Around 2000 years ago, the ancient Chinese and Greeks both built mechanical apparatus to simulate and predict apparent motions of celestial objects [1]. In recent decades, with the development of modern computer technologies, simulating physics with computers has opened up a novel era for science. To simulate a real-life physical system with computers, one needs to first build a mathematical model describing the system, and by changing the various parameters of the model and solving the equations using computers, predictions about the systems can be made from the solutions. Such an approach is extremely useful especially when the physical system of interest is not experimentally accessible or when the cost of trial-and-errors on the system itself is high. For example, weather forecast can be achieved by simulating the behaviour of the atmosphere [2]; the application of computational fluid dynamics has allowed efficient design of aircraft [3].

When the system of interest comes to a scale where the physics is governed by quantum mechanics, the equation describing the dynamics of the system is the well-known Schrödinger equation:

$$i\hbar \frac{d}{dt} |\psi(t)\rangle = \hat{H} |\psi(t)\rangle, \quad (1.1)$$

where i is the imaginary unit, \hbar is the reduced Planck constant, \hat{H} is the Hamiltonian of the system and $|\psi(t)\rangle$ is the state vector that contains all information of the system at instant t , living in a complex vector space known as the Hilbert space. The key challenge in solving the Schrödinger equation lies in the fact that the dimension of the Hilbert space, or in other words, the number of complex coefficients required to describe the vector $|\psi\rangle$, grows exponentially with the size of the physical system. Consider a system of N spins-1/2 particles, the dimension of the state vector is 2^N . When N is as small as 50, the storage of the vector $|\psi\rangle$ requires several petabytes of memory assuming each coefficient requires 1 byte to represent, which is on the order of magnitude of the memory capacity of Trinity, one of the largest among the existing supercomputers today [4]. This number is, however, far below the number of electrons in any real-life material which is of the order 10^{23} . The computational complexity therefore forbids the exact simulation of real-life physical materials on a

quantum mechanical scale given today’s computational resource.

1.1 Quantum simulation

Facing the problem of complexity, in 1982 Richard Feynman proposed the idea of simulating quantum mechanical systems using quantum computers, which are computers “*built of quantum mechanical elements which obey quantum mechanical laws*” [5]. It has been shown later in [6] that the quantum computers conceived by Feynman, based on quantum bits and circuits, can be programmed to simulate any local quantum system efficiently, where the time and spatial resources required for the simulation are only proportional to the time and space taken by the system of interest.

However, such universal, or “digital” quantum simulator requires the ability to fully control individual quantum particles in a many-body system as well as the interactions between them, such that the quantum bits and gate operations can be realised and performed with high fidelity, which still has a long way to go before these assumptions can be experimentally realised. A more realistic approach that is possible in the near-term future consists of building an “analogue” quantum simulator. Such simulators are specially designed quantum hardware that mimic the system to be simulated, in the sense that there exists a mapping between the Hamiltonian of the simulator and that of the system, and that the state of the simulator can also be mapped to that of the system. A good quantum simulator is one such that its state is experimentally accessible for both preparation and measurement, and that the Hamiltonian can be controlled to a certain extent. By preparing the simulator in an appropriate initial state that maps to the initial state of the system, and letting it evolve under the effect of the engineered Hamiltonian, one can probe and study the behaviour of the system by performing measurements on the simulator.

1.1.1 Major platforms for quantum simulation

To date, many platforms have been proposed and realised for performing quantum simulation. We briefly review some of the major platforms in this section.

Cold atoms

Neutral atoms can be trapped in optical lattices and cooled down to temperatures close to absolute zero, providing a versatile platform for simulating many-body physics [7]. Optical lattices are created by the interference patterns of overlapping laser beams that can create an effective periodic trapping potential for the atoms via dipole interaction. They can be engineered to create any lattice geometry including those simulating the structures in solid-state materials. For example, the Mott-to-superfluid transition in the Bose-Hubbard model can be efficiently simulated using such technique [8].

Trapped ions

Apart from neutral atoms, ions can be trapped by electric fields oscillating at radio frequency [9], laser-cooled and manipulated with precision [10]. While neutral atoms interact weakly with each other, trapped ions have much stronger interaction due to Coulomb repulsion. Various models have been simulated with trapped ions such as frustrated spins systems [11] and discrete time crystals [12]. Due to the high controllability of trapped ions, this platform is also promising for realising digital quantum computations [13].

Interacting photons

In complementary to the atom-based platforms, hybrid light-matter systems have emerged as quantum simulators, inspired by advances in cavity quantum electrodynamics (QED) in both optical and microwave regimes [14]. Photons in the cavity can be engineered to interact via coupling with the atoms, and strongly correlated photons can be achieved in coupled resonator arrays (CRAs) that can be used to simulate the Mott-to-superfluid phase transition of light [15, 16, 17]. This thesis will be mainly focusing on this platform.

Among various implementations of interacting photons, superconducting circuits have been envisioned to be a most promising physical platform [18, 19], where the quantum “particles” used for simulation are the circuit excitations. The circuit can behave as “artificial atoms” yet having more freedom in design and tunability compared to real atoms. They can be further coupled to circuit LC-resonators, that serve as “cavities” to form CRAs. On the other hand, as the circuit implementation does not necessarily conserve the total number of excitations as in the case of real atoms, they are naturally complement to the cold atom platform by accessing out-of-equilibrium physics.

1.2 Quantum simulation of quantum phase transition

We have seen that quantum simulation can be potentially helpful in studying quantum systems involving a large number of interacting particles, also known as many-body systems. As pointed out in [10], analogue quantum simulators can be less sensitive to errors compared to digital quantum simulators, since they can provide qualitative results even in the absence of full quantitative details. For example, quantum phase transitions are typically robust against local perturbations and can thus be studied using analogue quantum simulators.

One encounters phase transitions in everyday life. The source of life, liquid water, turns into vapour when it boils in a kettle, and turns into solid ice when it is in a freezer. While phases of matter are omnipresent, this subject has been at the heart of material science and many other domains of physics [20]. In a physical context, phases describe regimes of matter where physical properties are uniform throughout

the material and a phase transition refers to abrupt changes in these properties, when the system goes from one phase into another resulting from changes in external parameters. A classical phase transition, such as that of water, happens at finite temperature and are therefore caused by thermal fluctuations in the constituting particles of the material. In quantum physics, transitions between different phases of matter can occur even in the absence of thermal fluctuation, i.e. at absolute zero temperature, due to the quantum fluctuations[21] in the ground state of the system. Such phase transitions are referred to as quantum phase transitions (QPTs) that describe sharp changes in the properties of the ground state of a system at absolute zero temperature. Thanks to advances in cooling technologies, it is possible to achieve in a laboratory environment temperatures as low as several millikelvins, for example using a commercial dilution refrigerator, where the thermal fluctuation is negligible compared to quantum fluctuations, allowing for QPTs to be observed and studied.

Quantum phases of matters are of great interest of study as they can account for phenomenon such as high temperature superconductivity in cuprate superconductors [22]. However, as many-body physical problems, QPTs are in general difficult to study using classical techniques unless certain approximations are used.

This thesis will focus on the quantum simulation of QPTs in the Bose-Hubbard model and in the Jaynes-Cummings-Hubbard model. The latter can be implemented with interacting photons in coupled resonator array systems, that can be used as a quantum simulator for the QPTs. We will finally show how we can study these two models using machine learning techniques.

1.3 Bose-Hubbard model

The Bose-Hubbard (BH) model describes a system of interacting bosons on a lattice, that can be realised by cold atoms in optical lattices, as sketched in Fig. 1.1, defined by the following Hamiltonian:

$$\hat{H}^{\text{BH}} = -t \sum_{\langle i,j \rangle} \hat{b}_i^\dagger \hat{b}_j + \frac{U}{2} \sum_i \hat{n}_i (\hat{n}_i - 1) - \mu \sum_i \hat{n}_i, \quad (1.2)$$

where $\langle i, j \rangle$ denotes nearest neighbour pairs of lattice indices, \hat{b}_i^\dagger and \hat{b}_i are the bosonic creation and annihilation operators for the lattice site i and $\hat{n}_i = \hat{b}_i^\dagger \hat{b}_i$ is the corresponding bosonic number operator. The first term describes the nearest-neighbour hopping of bosons with hopping amplitude t . The second term describes the on-site interaction between bosons characterised by the parameter U , where $U > 0$ signifies that the interaction is repulsive and vice versa. The third term determines the number of particles in the ground state via the chemical potential μ .

To understand the phases present in the BH model, we start by looking at two limiting cases. In the following we consider a one-dimensional lattice with N sites.

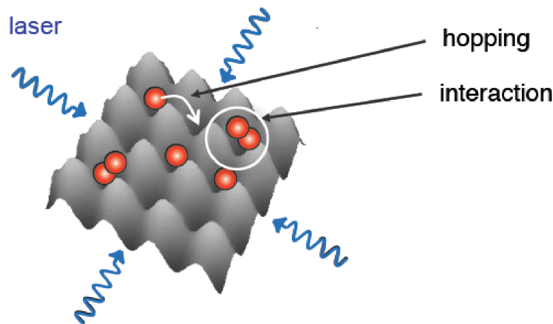


Figure 1.1: Schematic of the Bose-Hubbard model realised by cold atoms in optical lattices.

1.3.1 Strong interaction regime ($U \gg t$)

In the limit of $U \gg t$, the hopping term becomes negligible and the Hamiltonian reduces to a sum of local Hamiltonians

$$\hat{H} = \frac{U}{2} \sum_{i=1}^N \hat{n}_i(\hat{n}_i - 1) - \mu \sum_{i=1}^N \hat{n}_i = \sum_{i=1}^N \hat{h}_i, \quad (1.3)$$

where

$$\hat{h}_i = \frac{U}{2} \hat{n}_i(\hat{n}_i - 1) - \mu \hat{n}_i \quad (1.4)$$

is a localised Hamiltonian for the site i . Clearly the eigenstate for each \hat{h}_i are the number states

$$\hat{h}_i |n_i\rangle = E_i |n_i\rangle, \quad E_i = \frac{U}{2} n_i(n_i - 1) - \mu n_i. \quad (1.5)$$

The ground state of the system is therefore such that each n_i minimises the local eigen-energy E_i :

$$n_i = \max\{0, \lfloor \frac{\mu}{U} \rfloor + 1\}, \quad (1.6)$$

resulting in well-defined integer number of particles per site, which characterises the Mott insulator (MI) phase.

1.3.2 Weak interaction regime ($t \gg U$)

In the limit of $t \gg U$, the on-site interaction is negligible and the Hamiltonian becomes

$$\hat{H} = -t \sum_{\langle i,j \rangle} \hat{b}_i^\dagger \hat{b}_j - \mu \sum_{i=1}^N \hat{n}_i, \quad (1.7)$$

which can be diagonalised using Fourier transform (assuming periodic boundary condition):

$$\hat{H} = \sum_k (\epsilon_k - \mu) \hat{b}_k^\dagger \hat{b}_k, \quad \hat{b}_k = \frac{1}{\sqrt{N}} \sum_{j=1}^N \hat{b}_j e^{-ikj}, \quad \epsilon_k = -2t \cos k, \quad (1.8)$$

where

$$k = \frac{2\pi}{N}m, \quad m \in \{0, 1, \dots, N-1\}. \quad (1.9)$$

Therefore, the ground state for fixed number of particles m will be the state where all particles condense into the $k = 0$ state in the Fourier space:

$$|\Psi_{\text{SF}}\rangle = \frac{1}{m!}(\hat{b}_{k=0}^\dagger)^m |0\rangle, \quad \hat{b}_{k=0}^\dagger = \sum_{i=1}^N \hat{b}_i^\dagger, \quad (1.10)$$

which characterises the superfluid (SF) state because of the delocalisation of particles.

1.3.3 Mean-field theory

To study the system in the regime between the aforementioned two limiting cases, one can use the mean-field theory, which is done by ignoring correlations between sites. This approximation is exact in the infinite dimensional case. The order parameter can be chosen to be

$$\psi = \langle \hat{b}_i \rangle. \quad (1.11)$$

With the decoupling approximation:

$$\hat{b}_i^\dagger \hat{b}_j = \langle \hat{b}_i^\dagger \rangle \hat{b}_j + \langle \hat{b}_j \rangle \hat{b}_i^\dagger - \langle \hat{b}_i^\dagger \rangle \langle \hat{b}_j \rangle, \quad (1.12)$$

the mean-field Hamiltonian becomes

$$\hat{H}^{\text{MF}} = \sum_i \hat{h}_i^{\text{MF}}, \quad \hat{h}_i^{\text{MF}} = -\mu \hat{n}_i + \frac{U}{2} \hat{n}_i (\hat{n}_i - 1) + tz\psi^2 - tz(\psi^* \hat{b}_i + \psi \hat{b}_i^\dagger), \quad (1.13)$$

where z is the coordination number. The ground state energy can be obtained using second order perturbation theory:

$$\begin{aligned} E_g &= -\mu n + \frac{U}{2} n(n-1) + \psi^2 (tz + (tz)^2 (\frac{n}{U(n-1) - \mu} + \frac{n+1}{\mu - Un})) + O(\psi^4) \\ &= \text{const.} + m\psi^2 + O(\psi^4). \end{aligned} \quad (1.14)$$

Using Landau theory [23] we expect the phase transition to occur at $m = 0$, which gives:

$$1 + tz (\frac{n}{U(n-1) - \mu} + \frac{n+1}{\mu - Un}) = 0. \quad (1.15)$$

The corresponding phase diagram can be obtained by plotting equation (1.15), as shown in Fig. 1.2(a). We note that the mean-field phase diagram can also be obtained numerically instead of using the perturbation treatment in finding the ground state energy. For each set of parameters U, t and μ fixed in equation (1.13), the ground state energy can be found as a function of the order parameter ψ by diagonalising the local Hamiltonian (with a certain truncation on the boson number) h_i and taking the smallest the eigenvalue E_g . Minimising E_g with respect to ψ yields the value of the order parameter physically adopted by the system. The diagram obtained using this method is shown in Fig. 1.2(b).

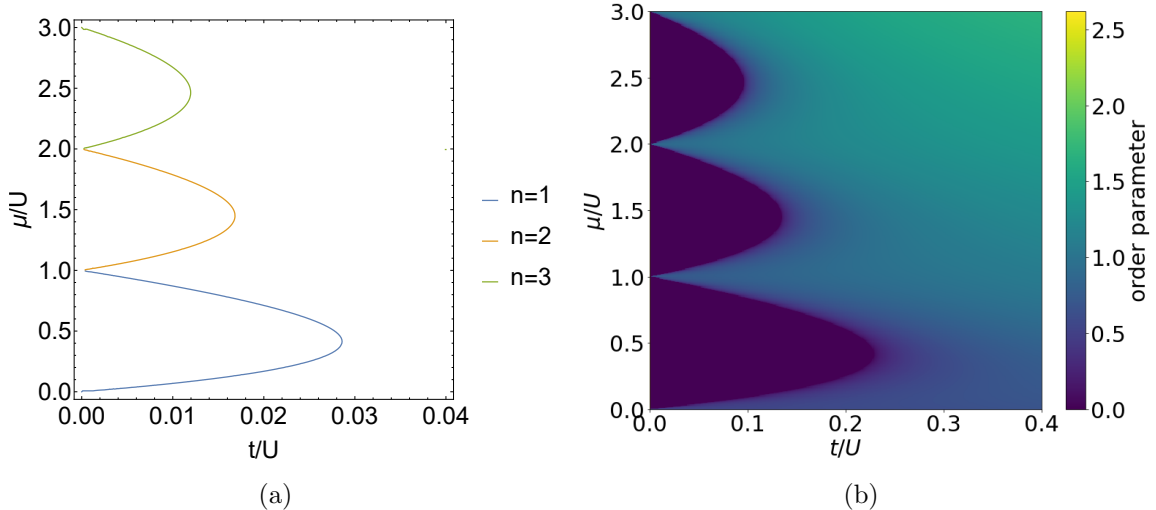


Figure 1.2: Comparison between the phase diagrams obtained using (a) analytical mean-field theory and (b) numerical mean-field theory for the 3D Bose-Hubbard model ($z = 6$), with truncation of boson number at $n_{\max} = 10$. Figure (a) is plotted according to equation 1.15, where the lobes on the left correspond to the Mott insulator phase and the region on the right corresponds to the superfluid region. Figure (b) shows the order parameter ψ as a function of t/U and μ/U . It shows the Mott insulator lobes corresponding to the regions with $\psi = 0$ and the rest corresponds to the superfluid phase. The plot is reproduced from [24] using our code attached in Appendix A.1.

1.3.4 Exact diagonalisation

The BH model can also be solved numerically by performing exact diagonalisation on the Hamiltonian with truncation on the maximum number of bosons allowed per site and the number of sites in the lattice. The order parameter is chosen to be the variance of the number of bosons on the middle site $\text{var}(\hat{n})$. A resulting phase diagram is shown in Fig. 1.3.

We note that one can define an number operator for the total number of bosons on the lattice,

$$\hat{N}_{\text{total}}^{\text{boson}} = \sum_i \hat{n}_i, \quad (1.16)$$

which commutes with \hat{H}^{BH} , implying that the eigen-spectrum of the latter can be grouped into sub-manifolds labelled by the total number of bosons. Therefore, we can also study the BH model in a canonical ensemble formalism, where we fix the total number of bosons, in which case the chemical potential term in the Hamiltonian is removed,

$$\hat{H}_{\text{CE}}^{\text{BH}} = -t \sum_{\langle i,j \rangle} \hat{b}_i^\dagger \hat{b}_j + \frac{U}{2} \sum_i \hat{n}_i (\hat{n}_i - 1), \quad (1.17)$$

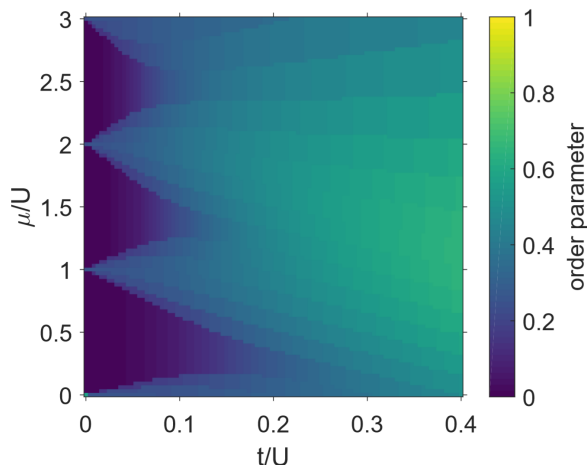


Figure 1.3: Order parameter as a function of t/U and μ/U in the 1D BH model in grand canonical ensemble, calculated using exact diagonalisation with the code attached in Appendix A.2. The simulation is done on a lattice containing 6 sites with periodic boundary condition and the maximum number of particles is 4. On the left of the figure the areas with darker colour correspond to the Mott insulator phase and the brighter area on the right corresponds to the superfluid state. Note that due to the small system size, the order parameter in the dark-coloured lobes doesn't exactly go to 0.

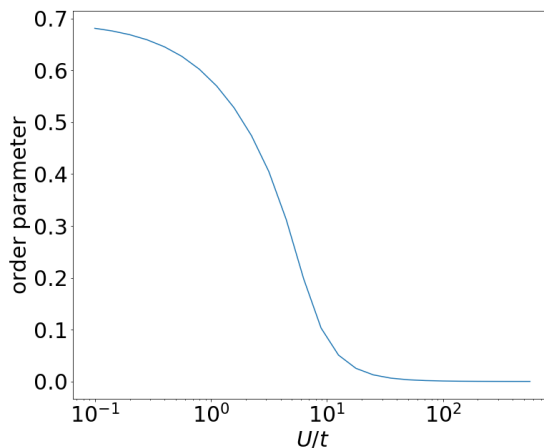


Figure 1.4: Order parameter as a function of U/t . in the 1D BH model in canonical ensemble, on a lattice of 6 sites with a total of 6 bosons, calculated using exact diagonalisation with the code attached in Appendix A.3. The maximum number of bosons allowed on one site is 3. As the interaction increases the system moves from superfluid phase to Mott insulator phase, where the order parameter drops to zero.

where the label “CE” signifies the canonical ensemble and the Hilbert space of the system is restricted to the sub-manifold with a fixed total number of bosons. The characteristics of the BH model can be studied in the unit-filling sub-manifold, where the total number of the bosons is the same as the number of sites in the lattice. By performing the same exact diagonalisation procedure and choosing the same order parameter $\text{var}(\hat{n})$, we obtain the phase transition diagram as shown in Fig. 1.4.

Chapter 2

Quantum many-body phases of strongly interacting photons

Having seen the quantum phases presented in the BH model, we will show in this chapter that by realising photon-photon interaction, the MI-SF transition can be simulated using hybrid light-matter systems. To begin with, we will introduce the canonical quantisation of light, that will give rise to the notion of photons. We will then discuss the mechanism of light-matter interaction, described by the Jaynes-Cummings model, that finally leads to the realisation of quantum many-body phases of light in coupled resonator arrays, described by the Jaynes-Cummings-Hubbard model.

2.1 Quantisation of light

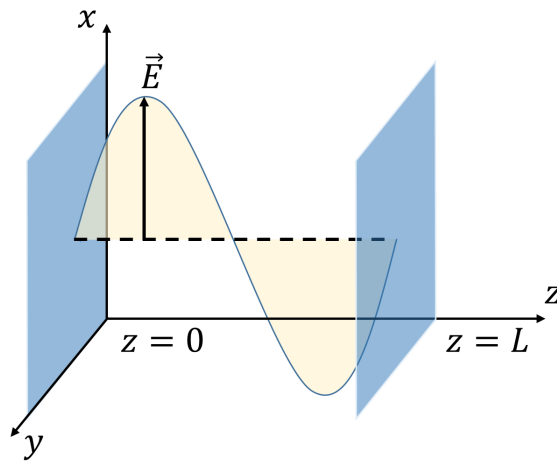


Figure 2.1: Schematic of a single-mode cavity with perfectly reflecting mirrors located at $z = 0$ and $z = L$, where the electric field \vec{E} is polarised in the x direction.

Consider a single-mode electromagnetic field confined to a one-dimensional cavity along the z axis with perfectly reflecting mirrors parallel to the xy plane and sepa-

rated by a distance L , as sketched in Fig. 2.1. The fields must vanish at those two boundaries and thus form a standing wave in the cavity. Assuming that the cavity contains neither source nor media, the electromagnetic field satisfies the Maxwell's equations without the source terms,

$$\begin{aligned}\vec{\nabla} \times \vec{E} &= -\frac{\partial \vec{B}}{\partial t} \\ \vec{\nabla} \times \vec{B} &= \mu_0 \epsilon_0 \frac{\partial \vec{E}}{\partial t} \\ \vec{\nabla} \cdot \vec{B} &= 0 \\ \vec{\nabla} \cdot \vec{E} &= 0,\end{aligned}\tag{2.1}$$

written in SI units, where μ_0 is the vacuum permeability, ϵ_0 being the vacuum permittivity and $\mu_0 \epsilon_0 = 1/c^2$ with c being the speed of light in vacuum. We further assume that the electric field is polarised along the x direction, such that $\vec{E}(\vec{r}, t) = \vec{e}_x E_x(z, t)$, implying that the magnetic field is polarised along y direction, $\vec{B}(\vec{r}, t) = \vec{e}_y B_y(z, t)$. The solution to the Maxwell's equations can be written as

$$\begin{aligned}E_x(z, t) &= \sqrt{\frac{2\omega_d^2}{V\epsilon_0}} q(t) \sin(kz), \\ B_y(z, t) &= \frac{\mu_0 \epsilon_0}{k} \sqrt{\frac{2\omega_d^2}{V\epsilon_0}} p(t) \cos(kz),\end{aligned}\tag{2.2}$$

where ω_d is the frequency of the cavity mode that satisfies the boundary condition, i.e. $\omega_d \in \{cm\pi/L \mid m = 1, 2, \dots\}$, k is the corresponding wave number $k = \omega_d/c$, V is the effective volume of the cavity, $q(t)$ is a time-dependent factor having the dimension of length and $p(t) = \dot{q}(t)$ is the time derivative of $q(t)$.

The Hamiltonian of the single-mode field in the cavity is given by its classical field energy,

$$\begin{aligned}H^{\text{cav}} &= \frac{1}{2} \int dV [\epsilon_0 \vec{E}^2(\vec{r}, t) + \frac{1}{\mu_0} \vec{B}^2(\vec{r}, t)] \\ &= \frac{1}{2} (p^2 + \omega_d^2 q^2),\end{aligned}\tag{2.3}$$

from which one can easily verify that p and q form a pair of canonical conjugate variable as they satisfy the Hamilton's equations:

$$\begin{aligned}\frac{dp}{dt} &= -\frac{\partial H^{\text{cav}}}{\partial q} \\ \frac{dq}{dt} &= \frac{\partial H^{\text{cav}}}{\partial p}.\end{aligned}\tag{2.4}$$

To perform the canonical quantisation of the fields, one simply promote the conjugate variables $p(t)$ and $q(t)$ to operators \hat{p} and \hat{q} and impose the canonical commutation relation:

$$[\hat{q}, \hat{p}] = i\hbar.\tag{2.5}$$

We further define the non-Hermitian operators

$$\begin{aligned}\hat{a} &= \sqrt{\frac{1}{2\hbar\omega_d}}(\omega_d\hat{q} + i\hat{p}), \\ \hat{a}^\dagger &= \sqrt{\frac{1}{2\hbar\omega_d}}(\omega_d\hat{q} - i\hat{p}),\end{aligned}\tag{2.6}$$

with which we can write the Hamiltonian as

$$\hat{H}^{\text{cav}} = \hbar\omega_d(\hat{a}^\dagger\hat{a} + \frac{1}{2}),\tag{2.7}$$

where $[\hat{a}, \hat{a}^\dagger] = 1$.

We can therefore identify the Hamiltonian to that of a quantum harmonic oscillator, where \hat{a} and \hat{a}^\dagger are simply the annihilation and creation operators for photons. The eigenstates are denoted $\{|n\rangle \mid n = 0, 1, 2, \dots\}$ with the number n indicating the number of photons in the cavity, the eigen-energies being $E_n = \hbar\omega_d(n + 1/2)$. We can also define the photonic number operator to be $\hat{n} = \hat{a}^\dagger\hat{a}$ that share the same eigenstates with the Hamiltonian. We set $\hbar = 1$ from now on for convenience.

2.2 Jaynes-Cummings model

The interaction between an atom and the photons in the cavity can be described by the Jaynes-Cummings (JC) model [25] with the Hamiltonian

$$\hat{H}^{\text{JC}} = \hat{H}^{\text{cav}} + \hat{H}^{\text{atom}} + \hat{H}^{\text{int}},\tag{2.8}$$

where

$$\hat{H}^{\text{cav}} = \omega_d\hat{a}^\dagger\hat{a},\tag{2.9}$$

which is the free energy of the photons as derived from Section 2.1 with the constant term dropped. \hat{H}^{atom} is the free energy term of the atom and \hat{H}^{int} is the interaction term between the atom and the photons.

An atom can be trapped in a cavity to realise strong effective photon-photon interaction. When the atomic transition between the ground state $|g\rangle$ and the first excited state $|e\rangle$ is coupled to the cavity mode, the atom can be treated as a two-level system, such that its free energy is given by the Hamiltonian (setting the ground state energy to 0):

$$\hat{H}^{\text{atom}} = \omega_0|e\rangle\langle e|,\tag{2.10}$$

where ω_0 is the energy difference between the two energy levels. We also define the atomic raising and lowering operators as follows:

$$\hat{S}^+ = |e\rangle\langle g|, \quad \hat{S}^- = |g\rangle\langle e|.\tag{2.11}$$

The atom is coupled to the cavity field via dipole interaction, giving the interaction Hamiltonian

$$\hat{H}^{\text{int}} = -\hat{d} \cdot \hat{E}\tag{2.12}$$

where $\hat{\vec{E}} = \vec{e}_x \sqrt{\frac{\omega_d}{\epsilon_0 V}} (\hat{a} + \hat{a}^\dagger) \sin(kz)$ according to Section 2.1, and $\hat{\vec{d}}$ is the electric dipole operator of the atom. From the parity consideration $\langle e | \hat{\vec{d}} | e \rangle = 0 = \langle g | \hat{\vec{d}} | g \rangle$, we can write without loss of generality, assuming that $\langle e | \hat{\vec{d}} | g \rangle = \vec{d}$ is real,

$$\begin{aligned} \hat{\vec{d}} &= \vec{d} (|g\rangle \langle e| + |e\rangle \langle g|) \\ &= \vec{d} (\hat{S}^- + \hat{S}^+). \end{aligned} \quad (2.13)$$

Therefore, the Hamiltonian becomes

$$\hat{H}^{\text{int}} = g (\hat{S}^- + \hat{S}^+) (\hat{a} + \hat{a}^\dagger), \quad (2.14)$$

where

$$g = -(\vec{d} \cdot \vec{e}_x) \sqrt{\frac{\omega_d}{\epsilon_0 V}} \sin(kz) \quad (2.15)$$

characterises the coupling strength.

At this point the Hamiltonian contains terms that do not conserve the total number of excitations of the atom and the photons, we will now show that these terms can be omitted under certain approximation. Transforming from the Schrödinger picture into the interaction picture defined by the free Hamiltonian $\hat{H}^{\text{free}} = \hat{H}^{\text{cav}} + \hat{H}^{\text{atom}}$,

$$\begin{aligned} \hat{H}^{\text{int}}(t) &= e^{i\hat{H}^{\text{free}}t} \hat{H}^{\text{int}} e^{-i\hat{H}^{\text{free}}t} \\ &= g (\hat{S}^- \hat{a} e^{-i(\omega_d + \omega_0)t} + \hat{S}^+ \hat{a} e^{i(\omega_0 - \omega_d)t} + \text{H.C.}), \end{aligned} \quad (2.16)$$

we see that the terms associated with $\hat{S}^- \hat{a}$ and $\hat{S}^+ \hat{a}^\dagger$ oscillate a higher frequency $\omega_d + \omega_0$, while the other two terms conserve the number of total number of excitations and oscillate at a lower frequency $\omega_0 - \omega_d$. When $|\omega_0 - \omega_d| \ll |\omega_0 + \omega_d|$ (the rotating-wave approximation), we can drop the fast oscillating terms and obtain the Jaynes-Cummings Hamiltonian:

$$\hat{H}^{\text{JC}} = \omega_d \hat{a}^\dagger \hat{a} + \omega_0 \hat{S}^+ \hat{S}^- + g (\hat{S}^- \hat{a}^\dagger + \hat{S}^+ \hat{a}). \quad (2.17)$$

The eigenstates of the Hamiltonian can be shown to be

$$\begin{aligned} |n, +\rangle &= \sin \theta |g, n\rangle + \cos \theta |e, n-1\rangle, \\ |n, -\rangle &= \cos \theta |g, n\rangle - \sin \theta |e, n-1\rangle, \end{aligned} \quad (2.18)$$

with the energies

$$E_{n,\pm}^{\text{JC}} = n\omega_d + \frac{\Delta}{2} \pm \sqrt{\left(\frac{\Delta}{2g}\right)^2 + n}, \quad n = 1, 2, \dots, \quad (2.19)$$

where $\Delta = \omega_0 - \omega_d$ is the detuning, $\tan 2\theta = \frac{2g\sqrt{n}}{\Delta}$. For $n = 0$, the eigenstate is $|g, 0\rangle$ with $E_0 = 0$. These eigenstates are hybrid states of atomic and photonic excitations and are known as polaritons. We can also define a polaritonic number operator:

$$\hat{N}^{\text{pol}} = \hat{a}^\dagger \hat{a} + \hat{S}^+ \hat{S}^-, \quad (2.20)$$

that counts the total number of excitations and shares the eigenstates with the Hamiltonian:

$$\hat{N}^{\text{pol}} |n, \pm\rangle = n |n, \pm\rangle. \quad (2.21)$$

2.3 Photon blockade effect

The spectrum of the JC Hamiltonian is nonlinear in n , which accounts for the effective photon-photon interaction mediated via the atom. For a fixed number of polaritons n , the ground state of the system in the corresponding sub-manifold is $|n, -\rangle$ with energy $E_{n,-}$, which is a convex increasing function of n , implying that the presence of a photon in the cavity makes it harder to add a second photon, because of the increasing energy gap $E_{1,-} - E_0 < E_{2,-} - E_{1,-}$. This effect is called photon blockade, acting as a strong effective repulsion between photons, that has been realised in the experiment [26, 27]. Note that the detuning Δ changes the nonlinearity of the spectrum and in the limit of large detuning $\Delta \gg g$, the energies reduces to be

$$E_{n,\pm} = n\omega_d + \frac{\Delta}{2}(1 \pm 1) \pm n\frac{g^2}{\Delta}, \quad (2.22)$$

which are linear in n , in which case there is no more photon blockade effect.

2.4 Jaynes-Cummings-Hubbard model

Having seen the realisation of effective photon-photon interaction resulting from the photon blockade effect in the nonlinear cavities, we will now show that it will give rise to the Mott-to-superfluid transition of phases of light that is analogue to that in the BH model. Strongly correlated photons can arise from coupled resonator arrays (CRAs), each resonator being a nonlinear cavity with Jaynes-Cummings interaction, and the cavities are coupled via nearest-neighbour photon hopping. This system of CRA, as sketched in Fig. 2.2(a), can be described by the Jaynes-Cummings-Hubbard (JCH) model [15, 16], which is a variant of the JC model. Consider a one-dimensional chain of L coupled cavities, each one doped with a single two-level atom, the Hamiltonian consists of the sum of the local terms describing the JC interaction in each cavity and the hopping terms between nearest neighbours:

$$\begin{aligned} \hat{H}^{\text{JCH}} &= \sum_k \hat{H}_k^{\text{JC}} + \sum_{\langle j,k \rangle} A(\hat{a}_j^\dagger \hat{a}_k + \hat{a}_j \hat{a}_k^\dagger) \\ &= \sum_k [\omega_d \hat{a}_k^\dagger \hat{a}_k + \omega_0 \hat{S}_k^+ \hat{S}_k^- + g(\hat{S}_k^- \hat{a}_k^\dagger + \hat{S}_k^+ \hat{a}_k)] + \sum_{\langle j,k \rangle} A(\hat{a}_j^\dagger \hat{a}_k + \hat{a}_j \hat{a}_k^\dagger) \end{aligned} \quad (2.23)$$

where $\langle j, k \rangle$ denotes nearest neighbour pairs of indices, A is the photon hopping amplitude between adjacent cavities, \hat{a}_k^\dagger (\hat{a}_k) is the creation (annihilation) operator for the photons in cavity k and \hat{S}_k^+ (\hat{S}_k^-) is the atomic raising (lowering) operator in cavity k .

We also define the number operator for the total number of excitations (polaritons) in the system,

$$\begin{aligned} \hat{N}_{\text{total}}^{\text{Pol}} &= \sum_k \hat{N}_k^{\text{pol}} \\ &= \sum_k (\hat{a}_k^\dagger \hat{a}_k + \hat{S}_k^+ \hat{S}_k^-). \end{aligned} \quad (2.24)$$

One can show that $\hat{N}_{\text{total}}^{\text{pol}}$ commutes with the Hamiltonian \hat{H}^{JCH} , therefore the eigen-spectrum of the latter can also be grouped into sub-manifolds labelled by the total number of excitations, which is similar to the case of BH model. Note that with the assumption that ω_d and ω_0 are the largest energy scales in the system, the ground state of \hat{H}^{JCH} will be the vacuum, as illustrated in Fig. 2.2(b).

Therefore, to study the phases present in the JCH model, one can consider the ground state of the system in the Hilbert space restricted to the unit-filling sub-manifold, i.e. the number of polaritons being the same as the number of cavities, or in other words, the filling factor $\bar{n} = \langle \hat{N}_{\text{total}}^{\text{pol}} \rangle / L = 1$. At resonance ($\Delta = 0$) and in the strong coupling regime ($g \gg A$), the strong nonlinearity in each cavity results in photon blockade preventing two photons to be present in the same cavity, effectively blocking the photon hopping and leading to the Mott-like ground state, i.e.,

$$|G\rangle_{\bar{n}=1} = |1, -\rangle \otimes |1, -\rangle \otimes \cdots \otimes |1, -\rangle. \quad (2.25)$$

In the limit of large detuning and strong hopping amplitude, and cavities become linear and photon blockade effect is suppressed. The system is effectively described by the BH model without the interaction term, resulting in the superfluid phase of photons.

The Mott-to-superfluid phase transition can be studied by choosing the order parameter to be the variance of polaritonic number at a certain site (e.g. the middle site), measuring the fluctuation in the number of excitations. A diagram obtained from exact diagonalisation is shown in Fig. 2.2(c).

2.4.1 Mean-field theory

One can add a chemical potential¹ term in the JCH model, as proposed in [17], to have a more direct analogy with the BH model. The Hamiltonian can then be written as

$$\hat{H}_{\text{GCE}}^{\text{JCH}} = \hat{H}^{\text{JCH}} - \mu \hat{N}_{\text{total}}^{\text{pol}}, \quad (2.26)$$

where μ is the chemical potential and the label ‘‘GCE’’ signifies the grand canonical ensemble.

We can then proceed to calculate the mean-field phase diagram. The order parameter is chosen to be $\psi = \langle \hat{a}_k \rangle$ and by applying the decoupling approximation, the mean-field Hamiltonian reads

$$\begin{aligned} \hat{H}_{\text{MF}}^{\text{JCH}} &= \sum_k \hat{h}_{\text{MF},k}^{\text{JCH}} \\ \hat{h}_{\text{MF},k}^{\text{JCH}} &= \hat{H}_k^{\text{JC}} - zA\psi(\hat{a}_k^\dagger + \hat{a}_k) + zA|\psi|^2 - \mu \hat{N}_k^{\text{pol}}, \end{aligned} \quad (2.27)$$

where z is the coordination number. Similar to our treatment in section 1.3.3, we can find the condition for phase transition with Landau theory, where the ground state

¹While photons do not naturally have a chemical potential, it is possible to engineer one as proposed in [28].

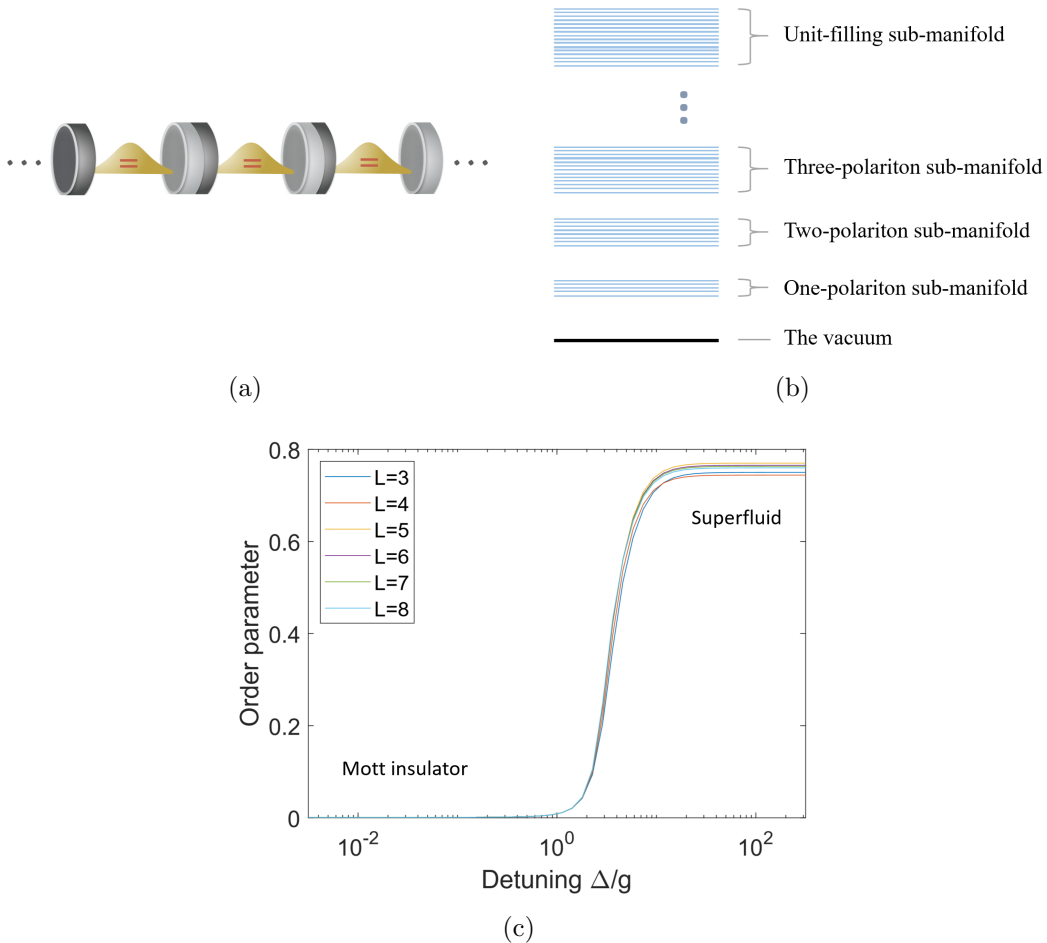


Figure 2.2: (a) Schematic of Jaynes-Cummings-Hubbard model, which describes a coupled array of single-mode cavities, each coupled to a two-level atom. (b) The energy spectrum of the JCH model, grouped into sub-manifolds labelled by polariton numbers. (c) Order parameter (Variance of \hat{N}_{pol} on the middle site) as a function of detuning of the JCH model, with $g/A = 10^2$, calculated using exact diagonalisation in canonical ensemble. For each number of sites L , the simulation is done in the unit-filling sub-manifold of polaritons, reproduced from [15] using our code attached in Appendix A.4. The truncation for the maximum number of photons allowed in each cavity is $n_{\text{max}} = 3$. As the detuning increases the system changes from Mott insulator into the superfluid phase, showing a non-zero variance in the polaritonic excitation number.

energy is found by second order perturbation theory and expressed in series in the order parameter ψ :

$$\begin{aligned}
 E_g &= E_{n,-}^{\mu} + zA\psi^2 + z^2A^2\psi^2R_n + O(\psi^4) \\
 &= \text{const.} + m\psi^2 + O(\psi^4),
 \end{aligned}
 \tag{2.28}$$

where

$$\begin{aligned}
E_{n,\pm}^\mu &= E_{n,\pm}^{\text{JC}} - \mu n, \\
R_0 &= -\left(\frac{\cos^2 \theta_1}{E_{1,-}^\mu} + \frac{\sin^2 \theta_1}{E_{1,+}^\mu}\right) \\
R_n &= \left(\frac{|\sqrt{n+1} \cos \theta_n \cos \theta_{n+1} + \sqrt{n} \sin \theta_n \sin \theta_{n+1}|^2}{E_{n,-}^\mu - E_{(n+1),-}^\mu} \right. \\
&\quad + \frac{|\sqrt{n+1} \cos \theta_n \sin \theta_{n+1} - \sqrt{n} \sin \theta_n \cos \theta_{n+1}|^2}{E_{n,-}^\mu - E_{(n+1),+}^\mu} \\
&\quad + \frac{|\sqrt{n} \cos \theta_n \cos \theta_{n-1} + \sqrt{n-1} \sin \theta_n \sin \theta_{n-1}|^2}{E_{n,-}^\mu - E_{(n-1),-}^\mu} \\
&\quad \left. + \frac{|\sqrt{n} \cos \theta_n \sin \theta_{n-1} - \sqrt{n-1} \sin \theta_n \cos \theta_{n-1}|^2}{E_{n,-}^\mu - E_{(n-1),+}^\mu}\right),
\end{aligned} \tag{2.29}$$

where $E_{n,\pm}^{\text{JC}}$ is the eigen-energy of the JC Hamiltonian and θ_n is defined by $\tan 2\theta_n = \frac{2g\sqrt{n}}{\Delta}$. The phase transition takes place at $m = 0$, which gives

$$1 + zAR_n = 0. \tag{2.30}$$

The phase diagram can be obtained by plotting equation 2.30, as shown in Fig. 2.3(a). Note that the mean-field phase diagram can also be computed numerically using the method described in Section 1.3.3, where the order parameter ψ is solved by minimising the ground state energy of the mean-field Hamiltonian, the graphical result of which is shown in Fig. 2.3. On these phase diagrams we can identify the Mott insulator lobes with order parameter being zero, and the superfluid phase with a non-zero order parameter, which are similar to the phase diagram of the BH model.

2.5 Techniques for probing quantum many-body phases

As an interlude, we briefly summarise and review some of the techniques for probing quantum many-body phases of matter.

The methods we have explored so far in this thesis are mainly the mean-field theory and exact diagonalisation. In all cases, we re-derived the results using our own developed codes as shown in the appendix. The mean-field theory approach allows for an analytical solution of the phase diagram for simple Hamiltonian, however, this approximation ignores local fluctuations and is inaccurate in low dimensions. Meanwhile, exact diagonalisation allows for an exact solution with very little approximation, yet is only feasible for small systems where the dimension of the Hilbert space is not too large.

To tackle this problem, many other numerical techniques have been developed in the past 30 years aiming to efficiently represent the state of the system using much less information compared to the size of the Hilbert space. Notable examples include

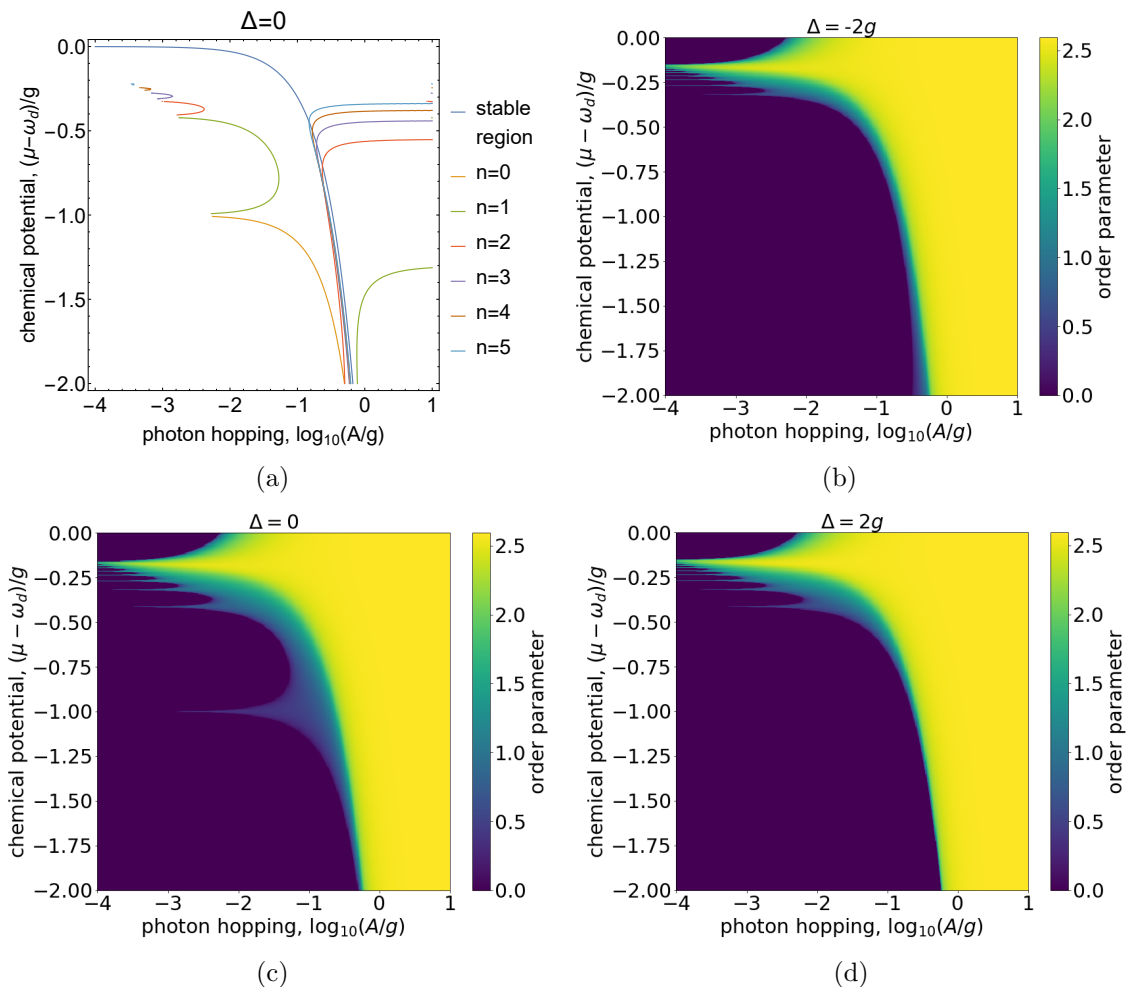


Figure 2.3: Comparison between phase diagrams for a CRA lattice with $z = 3$. (a) Analytical mean-field diagram at resonance (plot of equation 2.30 at $\Delta = 0$). (b)-(d) Numerical mean-field diagrams: order parameter as a function of photon hopping $\log_{10}(A/g)$ and chemical potential $(\mu - \omega_d)/g$, with $\Delta = 0$, $\Delta = -2g$ and $\Delta = 2g$ respectively, obtained from numerical simulation. The lobes in Figure (a) on the left (under the curve labelled “stable region”) and the dark regions in (b)(c)(d) with $\psi = 0$ correspond to the Mott insulator phase, while the other regions correspond to the superfluid phase. Note that the phase diagram obtained from analytical mean-field theory in (a) agrees well with the lobes in (b), both taken at resonance ($\Delta = 0$), and that a non-zero detuning will shift the lobes as demonstrated in (c) and (d). The results are reproduced from [29] using our code attached in Appendix A.5.

tensor networks (TNs) [30], density functional theory [31], quantum Monte-Carlo [32]. However, those methods are known to be limited to certain classes of problems. For example, tensor networks are inefficient for describing highly entangled or high-dimensional systems. Quantum Monte-Carlo suffers from the sign problem when representation fermionic systems.

Recently, artificial neuron networks have been proposed to solve quantum many-body problems in complementary to the above methods [33]. In essence, the task of finding an efficient representation of the state can be viewed as dimension reduction and feature extraction, as commonly encountered in the field of machine learning. In the next chapter, we will revise and describe in detail these algorithms that we will use to probe the quantum phases in the BH model and the JCH model.

Chapter 3

Machine learning quantum phases of matter

In recent decades, with the development in computer technology, we have witnessed more and more successful applications of machine learning in various domains, including pattern recognition, product recommendation, data mining, game playing, etc. Here we will describe its basics and show how we can use machine learning approaches to probe quantum phases of matter. Specifically, we will study the phases of the BH and the JCH models using artificial neural networks in a reinforcement learning scheme.

3.1 Machine learning basics

Formally the term “machine learning” can be defined as follows [34]: A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T , as measured by P , improves with experience E . In general, machine learning algorithms can be categorised as follows:

- Supervised learning: the machine tries to build a mathematical model through a set of labelled data [35], known as training data, where each training example in the data contains both input and the desired output, after which the machine is capable of predicting the output from new input data. Such algorithm is widely used in regression and classification [36].
- Unsupervised learning: the machine takes a set of unlabelled training data consisting of only input data without the desired output. The algorithm identifies patterns in the data, as commonly used in data clustering.
- Reinforcement learning: this type of learning algorithms does not require training data. Instead, the machine learns how to take actions in a certain environment, based on a penalty-and-reward mechanism, so as to optimise an objective. It can be, for example, used to train a game playing software against a human opponent.

Practically, the framework called the artificial neural network (ANN) is widely used in machine learning algorithms. An ANN is a set of connected nodes, that are inspired by the immense network of neurons in animal brains. They are typically powerful tools for processing complex data.

In the context of learning the quantum phases of matter, supervised (unsupervised) learning allows the machine to be trained by a set of configurations which are sampled from known (unknown) phases [37, 38, 39], while in reinforcement learning, the machine is trained using variational approaches, and the actual phases are not known prior to the training [33, 40, 41, 42, 43, 44, 45]. The latter case is more realistic because exact sampling from quantum phases are not always possible.

In reinforcement learning, an ANN can serve as a variational ansatz for the wave function of the system. Recent studies have been mostly focusing on two types of ANNs, namely the restricted Boltzmann machine (RBM) and the feed forward neural network (FFNN). [33] has shown that the RBM is capable of describing interacting quantum spin systems in both one and two dimensions. Their approach achieves accuracy higher than existing tools at that time and has sparked tremendous interests in RBM in solving quantum many-body problems. Subsequent works have shown that RBMs are capable of capturing exotic topologically-ordered phases [46, 47, 48]. These phases contain infinitely long-range entanglement and cannot be captured with the conventional TNs.

The use of FFNNs in quantum many-body problems has also been explored in parallel with RBMs. In some cases, FFNNs are shown to have faster learning time than RBMs [48, 42], including the case of Bose-Hubbard model [41, 45, 44].

3.2 Machine learning the ground state of a quantum system

The application of reinforcement learning in probing quantum phases requires a variational formalism of the problem. This can be achieved by adopting the variational method in quantum mechanics, for which we present a simple demonstration.

Suppose that for a given Hamiltonian \hat{H} , we want to find its ground state $|\epsilon_0\rangle$, which satisfies the time-independent Schrödinger equation,

$$\hat{H} |\epsilon_0\rangle = E_0 |\epsilon_0\rangle, \quad (3.1)$$

where the ground state energy E_0 is the smallest eigenvalue of H . We further assume that H admits a discrete spectrum and that the eigenstates $\{|\epsilon_n\rangle, n = 0, 1, 2, \dots\}$ form an orthonormal basis for the Hilbert space. Then for any arbitrary state $|\psi\rangle$

(which we do not assume to be normalised), the expectation value of \hat{H} is then

$$\begin{aligned}
\frac{\langle \psi | \hat{H} | \psi \rangle}{\langle \psi | \psi \rangle} &= \frac{\sum_{\epsilon_{n'}, \epsilon_{n''}} \langle \psi | \epsilon_{n'} \rangle \langle \epsilon_{n'} | \hat{H} | \epsilon_{n''} \rangle \langle \epsilon_{n''} | \psi \rangle}{\sum_{\epsilon_{n'}} |\langle \epsilon_{n'} | \psi \rangle|^2} \\
&= \frac{\sum_{\epsilon_{n'}} E_{n'} |\langle \epsilon_{n'} | \psi \rangle|^2}{\sum_{\epsilon_{n'}} |\langle \epsilon_{n'} | \psi \rangle|^2} \geq \frac{\sum_{\epsilon_{n'}} E_0 |\langle \epsilon_{n'} | \psi \rangle|^2}{\sum_{\epsilon_{n'}} |\langle \epsilon_{n'} | \psi \rangle|^2} \\
&= E_0,
\end{aligned} \tag{3.2}$$

i.e. bounded below by the ground state energy. This quantity can thus be a natural choice of the cost function in the learning procedure, which, when minimised, gives both the ground state energy and a ground state ket, which is of interest in the study of quantum phases of matter.

However, this is not a trivial process as the resource required to represent a general state in the Hilbert space still grows exponentially with the system size. One therefore demands an ansatz $|\psi(\mathcal{W})\rangle$ depending on a set of parameters \mathcal{W} , where the number of parameters is much smaller than the dimension of the Hilbert space. A good ansatz should not only be efficient in encoding the state, i.e. it captures the “correct” region in the Hilbert space that contains or is close to the ground state we are looking for, but also should allow fast evaluation of the cost function and the gradient, which will be used for updating the variational ansatz parameters during the learning procedure.

3.2.1 Artificial neural networks for strongly correlated lattice models

Inspired by the success of [33], which solved for the ground state of Ising model and Heisenberg model using ANNs as state ansatz combined with reinforcement learning scheme, we will adapt this method for studying the Bose-Hubbard model [49, 41, 44] and the Jaynes-Cummings-Hubbard model, with focus on the latter, since it is the first time to our knowledge where the ANNs have been applied to the JCH model. The essence is to represent the state ansatz using an ANN, and then optimise the cost function using Variational Monte-Carlo (VMC) [50] approach. For a given physical lattice of L sites, for which we want to find the ground state, we can expand an arbitrary state in a basis of occupation number (in the second quantisation formalism). For Bose-Hubbard model, we use the basis

$$\mathcal{B}^{\text{BH}} = \{|n_1, n_2, \dots, n_L\rangle\}, \tag{3.3}$$

where $n_i \in \{1, 2, \dots, n_{\text{max}}\}$ is the number of bosons on site i with a truncation at n_{max} . For each configuration

$$\mathcal{S}^{\text{BH}} = (n_1, n_2, \dots, n_L), \tag{3.4}$$

we will use the ANN to find the corresponding wave function $\Psi^{\text{BH}}(\mathcal{S}^{\text{BH}})$, such that the full state can be written as

$$|\Psi^{\text{BH}}\rangle = \sum_{\mathcal{S}^{\text{BH}}} \Psi^{\text{BH}}(\mathcal{S}^{\text{BH}}) |\mathcal{S}^{\text{BH}}\rangle. \tag{3.5}$$

Similarly, for the JCH model, an arbitrary state in a system with L sites can be expanded in the basis

$$\mathcal{B}^{\text{JCH}} = \{|n_1^p, n_1^a, n_2^p, n_2^a, \dots, n_L^p, n_L^a\rangle\}, \quad (3.6)$$

where $n_i^p \in \{1, 2, \dots, n_{\text{max}}\}$ is the number of photons in cavity i with a truncation at n_{max} and $n_i^a \in \{0, 1\}$ is the atomic excitation number at site i , indicates whether the atom is in state $|g\rangle$ or $|e\rangle$. For a given configuration

$$\mathcal{S}^{\text{JCH}} = (n_1^p, n_1^a, n_2^p, n_2^a, \dots, n_L^p, n_L^a), \quad (3.7)$$

we are to find the wave function Ψ^{JCH} such that the full state is

$$|\Psi^{\text{JCH}}\rangle = \sum_{\mathcal{S}^{\text{JCH}}} \Psi^{\text{JCH}}(\mathcal{S}^{\text{JCH}}) |\mathcal{S}^{\text{JCH}}\rangle. \quad (3.8)$$

In the following, we will drop the superscripts ‘‘BH’’ and ‘‘JCH’’ if the context leads to no ambiguity, e.g. we will denote write

$$|\Psi\rangle = \sum_{\mathcal{S}} \Psi(\mathcal{S}) |\mathcal{S}\rangle, \quad (3.9)$$

where $\Psi(\mathcal{S}) = \langle \mathcal{S} | \Psi \rangle$ is the wave function and \mathcal{S} denotes the tuple

$$\mathcal{S} = (v_i) = (v_1, v_2, \dots), \quad (3.10)$$

such that in the context of the BH model we have the mapping

$$v_i = n_i \quad (3.11)$$

and that in the context of the JCH model we have the mapping

$$v_{2k} = n_k^a, \quad v_{2k-1} = n_k^p. \quad (3.12)$$

We will then introduce how to represent the wave function $\Psi(\mathcal{S})$ using ANNs.

Restricted Boltzmann machine for lattice models

We first explore the Restricted Boltzmann machine (RBM) as the ANN wave function ansatz for the ground state, as shown in Fig. 3.1(a).

The wave function ansatz for the configuration \mathcal{S} is given by

$$\Psi_{\text{RBM}}(\mathcal{S}; \mathcal{W}) = \sum_{\{h_j = \pm 1\}} e^{\sum_i a_i v_i + \sum_j b_j h_j + \sum_{i,j} W_{ij} a_i b_j}, \quad (3.13)$$

where the v_i 's are the values taken by the visible units, each having a local weight a_i , the h_j 's are the values of the hidden units, each one can be either $+1$ or -1 and having a local weight b_j , the W_{ij} 's are the interaction weights between the two layers and \mathcal{W} is the set of all parameters of this ansatz, i.e.

$$\mathcal{W} = \{a_i, b_j, W_{ij} \mid i = 1, 2, \dots, N; j = 1, 2, \dots, M\} \quad (3.14)$$

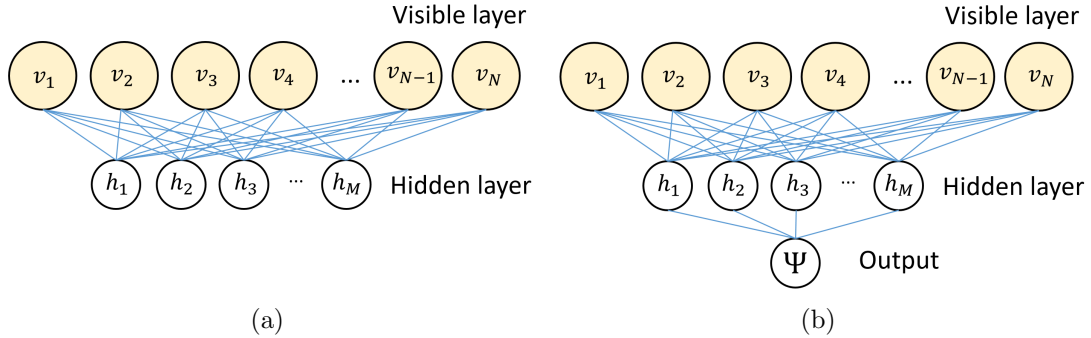


Figure 3.1: Schematics of the two ANNs we studied for the BH and the JCH lattice models: (a) Restricted Boltzmann machine, with a visible layer and a hidden layer, that is fully connected between layers. (b) Feed forward neural network, with a visible layer, a single hidden layer and an output layer consisting of one node, that is also fully connected between successive layers. In both ANNs, the visible units take values of the quantum numbers of the system of interest. The hidden units in RBM can only take values of ± 1 while those in FFNN take values determined by the visible units via the activation function.

for an RBM with N visible units and M hidden units, where all parameters are complex-valued in order to allow for the amplitude and the phase of the wave function to be encoded. The summation in equation (3.13) can be carried out analytically, giving

$$\Psi_{\text{RBM}}(\mathcal{S}; \mathcal{W}) = e^{\sum_i a_i v_i} \times \prod_{j=1}^M F_j(\mathcal{S}), \quad (3.15)$$

where

$$F_j(\mathcal{S}) = 2 \cosh \left[b_j + \sum_i W_{ij} v_i \right]. \quad (3.16)$$

One can qualitatively see that due to the full connection between the visible and the hidden layers, tracing out the hidden units allows the ansatz to capture long-range correlations between the visible units, thus making the ansatz capable of representing states with entanglement. Note that for an RBM with N visible units and M hidden units, the total number of ansatz parameters¹ is

$$|\mathcal{W}| = M + N + M \times N = \mathcal{O}(M \times N), \quad (3.17)$$

while the dimension of the Hilbert space grows exponentially with N .

One can easily calculate the log-derivatives for this ansatz with respect to the

¹As pointed out in [33], symmetries of the lattice structure can be exploited to reduce the number of parameters. For example in the presence of lattice translational symmetry, the number of parameters is $\mathcal{O}(M)$.

parameters,

$$\begin{aligned}
\frac{\partial}{\partial a_i} \log \Psi_{\text{RBM}}(\mathcal{S}) &= v_i, \\
\frac{\partial}{\partial b_j} \log \Psi_{\text{RBM}}(\mathcal{S}) &= \tanh \left(b_j + \sum_i W_{ij} v_i \right), \\
\frac{\partial}{\partial W_{ij}} \log \Psi_{\text{RBM}}(\mathcal{S}) &= v_i \tanh \left(b_j + \sum_i W_{ij} v_i \right),
\end{aligned} \tag{3.18}$$

which will be used in a gradient-based optimisation algorithm.

Feed forward neural network for lattice models

Another ANN we explored as wave function ansatz is the feed forward neural network (FFNN) with a single² hidden layer, as shown in Fig. 3.1(b). Similar to the RBM, the visible units take values of the quantum numbers $\mathcal{S} = (v_i)$, while they also serve as the input units to the network and determine the values of the hidden units h_j by

$$h_j = f \left(\sum_i W_{ij}^{(1)} v_i + b_j \right), \tag{3.19}$$

where the b_j is called the bias, the $W_{ij}^{(1)}$'s are the interaction weights between the input layer and the visible layer and f is the activation function. As suggested in [44] we adopt

$$f(x) = \log(\cosh(x)) \tag{3.20}$$

in this work. The output layer, containing only one unit, gives the value of the wave function ansatz and is determined by the hidden layer:

$$\Psi_{\text{FFNN}}(\mathcal{S}; \mathcal{W}) = \exp \left(\sum_j W_j^{(2)} h_j \right), \tag{3.21}$$

where the $W_j^{(2)}$'s are the interaction weights between the hidden layer and the output layer and \mathcal{W} is the set of all parameters,

$$\mathcal{W} = \{W_{ij}^{(1)}, W_j^{(2)}, b_j \mid i = 1, 2, \dots, N; j = 1, 2, \dots, M\}, \tag{3.22}$$

where all parameters are complex-valued, with

$$|\mathcal{W}| = N \times M + M + M = \mathcal{O}(M \times N), \tag{3.23}$$

where N is the number of visible units and M is the number of hidden units. Note that in this network information flows in one direction from the input layer to the output layer, which is where the name ‘‘feed forward’’ comes from.

²Note that in general, an FFNN can have multiple hidden layers between the visible (input) and the output layer.

We can also explicitly calculate the log-derivatives for this ansatz:

$$\begin{aligned}
\frac{\partial}{\partial W_{ij}^{(1)}} \log \Psi_{\text{FFNN}}(\mathcal{S}) &= W_j^{(2)} v_i \tanh \left(b_j + \sum_i W_{ij}^{(1)} v_i \right), \\
\frac{\partial}{\partial W_j^{(2)}} \log \Psi_{\text{FFNN}}(\mathcal{S}) &= \log \left[\cosh \left(b_j + \sum_i W_{ij}^{(1)} v_i \right) \right], \\
\frac{\partial}{\partial b_j} \log \Psi_{\text{FFNN}}(\mathcal{S}) &= W_j^{(2)} \tanh \left(b_j + \sum_i W_{ij}^{(1)} v_i \right).
\end{aligned} \tag{3.24}$$

For both RBM and FFNN, we define the parameter

$$\alpha = \frac{\text{number of hidden units}}{\text{number of visible units}} = \frac{M}{N} \tag{3.25}$$

to be the hidden unit density.

3.2.2 Variational Monte Carlo for the ground state of lattice models

With the ANN ansatz, we are now ready to find the ground state through a reinforcement learning procedure using Variational Monte-Carlo (VMC) [50] approach. As introduced in Section 3.2, the cost function can be chosen to be the energy, that is to be minimised:

$$E(\mathcal{W}) = \frac{\langle \Psi | \hat{H} | \Psi \rangle}{\langle \Psi | \Psi \rangle}, \tag{3.26}$$

where $\hat{H} \in \{\hat{H}^{\text{JCH}}, \hat{H}^{\text{BH}}\}$, $\Psi \in \{\Psi_{\text{RBM}}, \Psi_{\text{FFNN}}\}$ depending on the specific problem. For any operator $\hat{\mathcal{O}}$ its expectation value can be written, using the closure relation $\sum_{\mathcal{S}} |\mathcal{S}\rangle \langle \mathcal{S}| = \mathbb{1}$, in the form of a probabilistic sum,

$$\begin{aligned}
\langle \hat{\mathcal{O}} \rangle &= \frac{\langle \Psi | \hat{\mathcal{O}} | \Psi \rangle}{\langle \Psi | \Psi \rangle} \\
&= \frac{\sum_{\mathcal{S}, \mathcal{S}'} \langle \Psi | \mathcal{S} \rangle \langle \mathcal{S} | \hat{\mathcal{O}} | \mathcal{S}' \rangle \langle \mathcal{S}' | \Psi \rangle}{\sum_{\mathcal{S}} \langle \Psi | \mathcal{S} \rangle \langle \mathcal{S} | \Psi \rangle} \\
&= \frac{\sum_{\mathcal{S}, \mathcal{S}'} \Psi^*(\mathcal{S}) \mathcal{O}_{\mathcal{S}, \mathcal{S}'} \Psi(\mathcal{S}')}{\sum_{\mathcal{S}} |\Psi(\mathcal{S})|^2} \\
&= \frac{\sum_{\mathcal{S}} |\Psi(\mathcal{S})|^2 \sum_{\mathcal{S}'} \mathcal{O}_{\mathcal{S}, \mathcal{S}'} \frac{\Psi(\mathcal{S}')}{\Psi(\mathcal{S})}}{\sum_{\mathcal{S}} |\Psi(\mathcal{S})|^2} \\
&= \sum_{\mathcal{S}} \Pi(\mathcal{S}) \mathcal{O}_{\text{loc}}(\mathcal{S}) \\
&= \langle\langle \mathcal{O}_{\text{loc}} \rangle\rangle,
\end{aligned} \tag{3.27}$$

where $\mathcal{O}_{\mathcal{S}, \mathcal{S}'} = \langle \mathcal{S} | \hat{\mathcal{O}} | \mathcal{S}' \rangle$, $\Pi(\mathcal{S}) = \frac{|\Psi(\mathcal{S})|^2}{\sum_{\mathcal{S}} |\Psi(\mathcal{S})|^2}$, $\mathcal{O}_{\text{loc}}(\mathcal{S}) = \sum_{\mathcal{S}'} \mathcal{O}_{\mathcal{S}, \mathcal{S}'} \frac{\Psi(\mathcal{S}')}{\Psi(\mathcal{S})}$ and $\langle\langle \mathcal{O}_{\text{loc}} \rangle\rangle$ denotes the statistical expectation value of \mathcal{O}_{loc} over the probability

distribution $\Pi(\mathcal{S})$. In particular, for the Hamiltonian operator \hat{H} , we can write the cost function as

$$E(\mathcal{W}) = \langle \hat{H} \rangle = \langle\langle H_{\text{loc}} \rangle\rangle, \quad (3.28)$$

where $H_{\text{loc}}(\mathcal{S}) = \sum_{\mathcal{S}'} H_{\mathcal{S},\mathcal{S}'} \frac{\Psi(\mathcal{S}')}{\Psi(\mathcal{S})}$.

Similarly, one can show that the gradient of the cost function can be written as the expectation value of some random variable. By denoting

$$D_k(\mathcal{S}) = \frac{\partial}{\partial p_k} \log \Psi(\mathcal{S}), \quad p_k \in \mathcal{W}, \quad (3.29)$$

we have

$$\frac{\partial}{\partial p_k} E(\mathcal{W}) = \langle\langle G_k \rangle\rangle, \quad (3.30)$$

where G_k is the gradient estimator defined as

$$G_k(\mathcal{S}) = 2\text{Re}[(H_{\text{loc}}(\mathcal{S}) - \langle\langle H_{\text{loc}} \rangle\rangle) D_k^*(\mathcal{S})]. \quad (3.31)$$

Stochastic sampling

The estimation of the cost function and its gradient can be achieved via stochastic sampling over the probability distribution $\Pi(\mathcal{S})$, as shown in equation (3.28,3.30). The sampling can be done by a Markov chain Monte Carlo (MCMC) process, where we generate a Markov chain of the visible unit configurations $\mathcal{S}^{(1)} \rightarrow \mathcal{S}^{(2)} \rightarrow \dots \mathcal{S}^{(k)} \dots$ sampling the quantity $|\Psi(\mathcal{S})|^2$ that the probability is proportional to. This can be done through the Metropolis-Hastings algorithm [51], where at each step of the Markov chain, an update of configuration from \mathcal{S} to \mathcal{S}' is randomly proposed with probability $Q(\mathcal{S}'|\mathcal{S})$ and the new configuration is accepted with probability

$$\begin{aligned} A(\mathcal{S} \rightarrow \mathcal{S}') &= \min \left\{ 1, \frac{\Pi(\mathcal{S}') Q(\mathcal{S}|\mathcal{S}')}{\Pi(\mathcal{S}) Q(\mathcal{S}'|\mathcal{S})} \right\} \\ &= \min \left\{ 1, \frac{|\Psi(\mathcal{S}')|^2 Q(\mathcal{S}|\mathcal{S}')}{|\Psi(\mathcal{S})|^2 Q(\mathcal{S}'|\mathcal{S})} \right\}, \end{aligned} \quad (3.32)$$

where $Q(\mathcal{S}|\mathcal{S}')$ is the probability of proposing \mathcal{S} as the updated configuration starting from \mathcal{S}' . In practice, if one works in canonical ensemble formalism and wants to find the ground state in the sub-manifold with a fixed total quantum number, the proposed update \mathcal{S}' should be such that

$$\langle \mathcal{S}' | \hat{H} | \mathcal{S} \rangle \neq 0, \quad (3.33)$$

since both Hamiltonians in our study (\hat{H}^{BH} and \hat{H}^{JCH}) conserve the total quantum number.

Training protocol

We can now briefly summarise the training protocol of the reinforcement learning scheme for finding the ground state. For a given problem (Hamiltonian \hat{H}) and a chosen ANN ansatz ($|\Psi\rangle$), the following steps are performed:

1. Randomly initialise parameters \mathcal{W} of the ansatz $|\Psi(\mathcal{W})\rangle$.
2. Repeat:
 - (a) Calculate cost function $E(\mathcal{W})$.
 - (b) Calculate gradient³ of the cost function, i.e. $\frac{\partial}{\partial p_k} E(\mathcal{W})$, $\forall p_k \in \mathcal{W}$.
 - (c) Update parameters based on the gradient⁴. For example, using the steepest gradient descent method, one performs the update

$$p_k \leftarrow p_k - \eta \frac{\partial}{\partial p_k} E(\mathcal{W}), \quad \forall p_k \in \mathcal{W},$$

where η is called the learning rate.

until convergence of $E(\mathcal{W})$.

Note that the calculations in step (a) and (b) are performed via stochastic sampling as introduced in Section 3.2.2. If the cost function ends up converging to the global minimum⁵, its value will yield the ground state energy of the Hamiltonian \hat{H} and the resulting state $|\Psi(\mathcal{W})\rangle$ will be a ground state, on which we can perform measurements of other observables via stochastic sampling, such as the order parameter.

3.3 BH and JCH phases via machine learning

We applied the machine learning method⁶ of finding the ground state to both Bose-Hubbard model and Jaynes-Cummings-Hubbard model for specific finite size systems. First we solved the BH model with RBM for different parameters in the Hamiltonian across the phase transition as a benchmark for this ANN ansatz. We then solved the JCH model using both RBM and FFNN and compared the results with exact diagonalisation. The results are novel to our knowledge and have not been published before.

3.3.1 Machine learning results for the BH model

We study a one-dimensional chain of $L = 5$ sites of the BH model with periodic boundary condition in canonical ensemble with unit-filling of bosons, where a truncation of boson numbers at each site is set at $n_{\max} = 3$, for which the exact diagonalisation can

³While the method presented in Section 3.2.2 provides an estimate for the gradient, there exist other methods such as stochastic reconfiguration [52] that provides better stability, which is adopted in our calculations.

⁴Note that there are many gradient based optimisation algorithms, steepest gradient descent being one of them. In practice we have adopted the Adamax [53] for a more stable convergence.

⁵Note that it is possible for the value to converge to a local minimum, the reason can be the incapability of the ansatz in capturing the correct state or due to inappropriate hyper-parameters in the learning scheme.

⁶The program is based on and modified from NetKet[54].

be easily carried out to find the ground state energy E_0 and the energy gap $E_1 - E_0$, as a benchmark for the machine learning results.

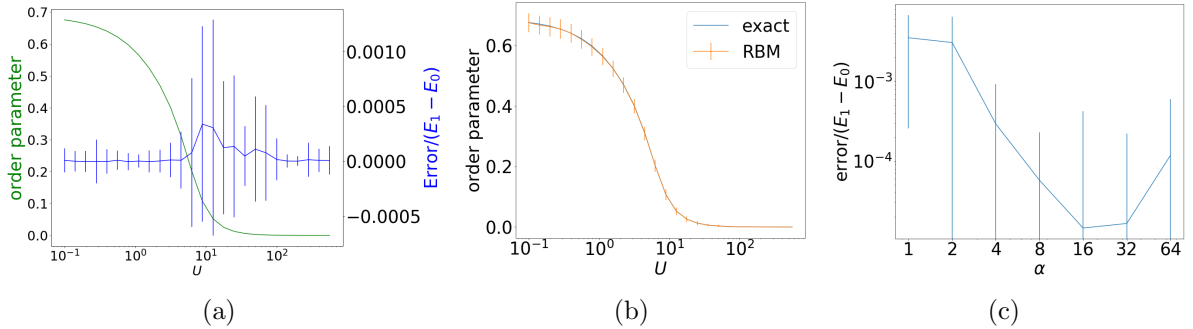


Figure 3.2: (a) Relative error in the ground state energy (right axis) for different values of U/t , plotted together with the order parameter (left axis) obtained from exact diagonalisation. The error bars are determined from the standard deviation in the cost function over the last 500 iterations. (b) Order parameter measured from the RBM ansatz (yellow curve) compared to the exact value (blue curve). The values and the error bars are determined from the average and the standard deviation of the order parameter respectively over the last 500 iterations. (c) Relative error in the ground state energy for fixed U/t with respect to hidden unit density. The error bars are determined from the standard deviation in the cost function over the last 500 iterations.

We first study the performance of the learning algorithm with the RBM ansatz in different phases of the system, where the optimisation is performed for different values of U/t ranging from the SF to MI phase of the system. To learn the ground state of the system, we fix the hidden unit density at $\alpha = 4$ (i.e. 5 visible units and 20 hidden units in the RBM). Each time we perform 4000 iterations of parameter updates and in each iteration 1000 samples are taken in the Monte-Carlo sampling to estimate the cost function and the gradient. The final learning outcome of the ground state energy is taken to be the average of the cost function over the last 500 iterations. We investigate the relative error, defined as the difference between the learning result of the energy and the exact value divided by the energy gap, as shown in Fig. 3.2(a). One can notice that the error increases at the phase transition, where the order parameter changes between zero and non-zero, while it remains relatively small for both MI and SF phase far from the phase transition. For the same values of U/t , we also perform measurements of the order parameter $\text{var}(\hat{n})$, as shown in Fig. 3.2(b), demonstrating that one can reconstruct the order parameter and thus the phase transition diagram from the ground state learnt by the ANN.

We then investigated the effect of varying the hidden unit density for a fixed value of U/t . In particular, this value is chosen to be the one corresponding to the largest error in Fig. 3.2(a). The result is shown in Fig. 3.2(c), where the error decreases and then saturates around a certain level as the hidden unit density increases.

3.3.2 Machine learning results for the JCH model

We study the JCH model consisting of a one-dimensional chain of L cavities with open boundary condition in canonical ensemble with unit-filling of excitations, L ranging from 3 to 8 with a truncation of photon number per cavity at $n_{\max} = 3$. To study the different phases of the system, we focus on the regime $g/A = 10^2$, as suggested in [15]. From Fig. 2.2(c) we pick 3 values of Δ to benchmark the performance of both RBM and FFNN:

- $\Delta = 10^{-1}g$, where the system is in Mott insulator phase,
- $\Delta = 10^{0.5}g$, where the system is at transition between the two phases,
- $\Delta = 10^2g$, where the system is in superfluid phase.

The exact ground state energy E_0 as well as the gap $E_1 - E_0$ are then calculated for these values of Δ using exact diagonalisation. Note that ω_d does not affect the order parameter or the value of the gap, so it can be fixed to an arbitrary value.

Throughout the learning process with the two ANNs, for each data point we perform 10000 iterations of parameter updates and in each iteration 1000 samples are drawn in the Monte-Carlo sampling. An example of the variation of the cost function with respect to the number of iterations is shown in Fig. 3.3, where the line corresponding to RBM exhibits convergence.

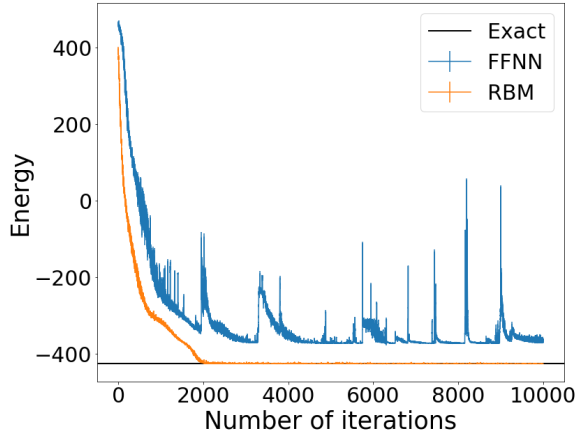


Figure 3.3: Cost function with respect to the iterations numbers for RBM and FFNN while learning the ground state of the JCH model with $L = 5$, $\Delta = 10^{-1}g$ and $\alpha = 4$, in Mott insulator phase. The error bar is the standard deviation of the energy and is invisible in this graph. While the energy given by RBM converges to the exact ground state energy as mark by the black horizontal line, FFNN exhibits significant instability.

The final learning outcome of the ground state energy is taken to be the average of the cost function over the last 500 iterations. The relative error, defined as the difference between the learning result of energy and the exact value divided by the gap, is investigated for different system sizes and different hidden unit densities, as

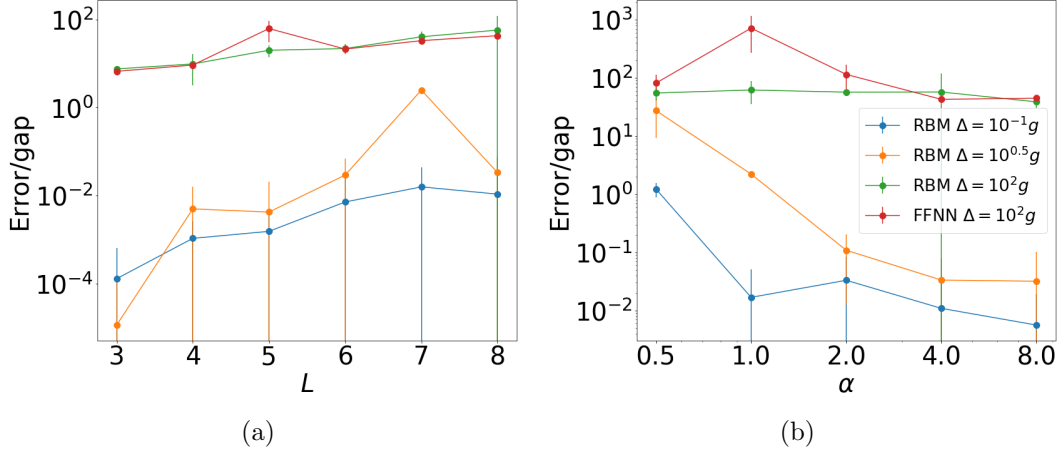


Figure 3.4: Relative error in the ground state energy of the JCH model obtained using RBM and FFNN. (a) Error with respect to system size L for fixed hidden unit density $\alpha = 4$. (b) Error with respect to hidden unit density for fixed system size $L = 8$. The error bars are determined from the standard deviation in the cost function over the last 500 iterations. The legend is shared by both plots.

shown in Fig. 3.4, where the error increases with system size L and that the error significantly grows as the system goes from MI phase ($\Delta = 10^{-1}g$) to SF phase ($\Delta = 10^2g$). In the MI phase, the error in RBM results can be significantly reduced by increasing the hidden unit density, while in the SF phase, both ANNs have errors greater than the gap, regardless of the hidden unit density. Since we noticed from Fig. 3.3 that the energy calculated for MI phase using FFNN will not converge in general, the calculation with FFNN is only done for $\Delta = 10^2g$, where the system is in SF phase.

3.4 Discussion

The results above show that for the BH model, the RBM neural network ansatz is capable of accurately representing the ground state and finding the order parameter in both MI and SF phase. Although the error increases at the phase transition between MI and SF, it remains several orders of magnitude smaller than the energy gap between the ground state and the first excited state. Therefore, the RBM is a sensible choice for studying the phases of the BH model.

In the study of the JCH model, only the MI phase can be accurately represented by the RBM ansatz, and that the error can be reduced by using more hidden units in the RBM, while FFNN fails to converge within the same number of iterations, although it has been shown to work in the case of the BH model in [45]. This could be due to the localised polaritonic states of the MI phase $|\psi\rangle \simeq \prod_K \frac{1}{\sqrt{2}}(|g, 1\rangle_k - |e, 0\rangle_k)$ having richer structure compared to the BH model, the latter having no atomic degree of freedom, that FFNN fails to capture. On the other hand, in the SF phase, the error

in the results given by both ANNs stay higher than the energy gap regardless of the number of hidden units. This could be again due to the atom-photon coupling that makes the state structure more complicated for RBM and FFNN to capture. One would expect that in the limit of strong hopping and weak coupling ($g \ll A$) both ANNs should be able to accurately represent the ground state in SF phase, since in this limit the atom and photons decouple and the model trivially reduces to the BH model.

3.5 Conclusion

We have reviewed and studied the Bose-Hubbard model and the Jaynes-Cummings-Hubbard model, the former being a simple model for quantum phase transition that is a phenomenon of great interest in many scientific domains, the latter describing a system of strongly interacting photons in coupled resonator arrays that can realise quantum many-body phases of light, which can serve as a quantum simulator for the Mott-to-superfluid transition.

These two models can be studied using various numerical and analytical techniques, and the one that we have focused on in this work is machine learning, where the ground state ansatz is represented by artificial neural networks, to our knowledge a novel approach so far. We have studied the BH model using restricted Boltzmann machine, and the JCH model using both restricted Boltzmann machine and feed forward neural network, and examined the performance of these neural network ansätze in a reinforcement learning scheme, for different phases of the models by comparing the learning outcome with exact diagonalisation results. We have found that the RBM is capable of accurately representing the ground state in both MI and SF phases of the BH model while it can only represent accurately the ground state of the JCH model in the MI phase, and that the errors in these cases can be reduced by increasing the hidden unit density in the network. The failure of FFNN in the MI phase of the JCH model together with the error of both RBM and FFNN in the SF phase of JCH suggest that the choice of an ANN for studying the quantum phases in a many-body system is problem-specific.

Nevertheless, we have gained a glimpse of the power of artificial neural network and machine learning. Their entry to the domain of quantum simulation and quantum many-body physics is more recent as compared to others fields of their applications, yet one can expect that this emerging interdisciplinary field of research to sparkle more interest and success in the future.

Bibliography

- [1] Nataraja Sarma. “Diffusion of astronomy in the ancient world”. In: *Endeavour* 24.4 (2000), pp. 157–164.
- [2] Tilmann Gneiting and Adrian E Raftery. “Weather forecasting with ensemble methods”. In: *Science* 310.5746 (2005), pp. 248–249.
- [3] Antony Jameson. “Computational aerodynamics for aircraft design”. In: *Science* 245.4916 (1989), pp. 361–371.
- [4] Erich Strohmaier et al. *TOP500 list*. 2015.
- [5] Richard P Feynman. “Simulating physics with computers”. In: *International journal of theoretical physics* 21.6 (1982), pp. 467–488.
- [6] Seth Lloyd. “Universal quantum simulators”. In: *Science* (1996), pp. 1073–1078.
- [7] Immanuel Bloch, Jean Dalibard, and Wilhelm Zwerger. “Many-body physics with ultracold gases”. In: *Reviews of modern physics* 80.3 (2008), p. 885.
- [8] Markus Greiner et al. “Quantum phase transition from a superfluid to a Mott insulator in a gas of ultracold atoms”. In: *nature* 415.6867 (2002), p. 39.
- [9] Wolfgang Paul and Helmut Steinwedel. “Ein neues massenspektrometer ohne magnetfeld”. In: *Zeitschrift für Naturforschung A* 8.7 (1953), pp. 448–450.
- [10] Iulia M Georgescu, Sahel Ashhab, and Franco Nori. “Quantum simulation”. In: *Reviews of Modern Physics* 86.1 (2014), p. 153.
- [11] Kihwan Kim et al. “Quantum simulation of frustrated Ising spins with trapped ions”. In: *Nature* 465.7298 (2010), p. 590.
- [12] J Zhang et al. “Observation of a discrete time crystal”. In: *Nature* 543.7644 (2017), p. 217.
- [13] Hartmut Häffner, Christian F Roos, and Rainer Blatt. “Quantum computing with trapped ions”. In: *Physics reports* 469.4 (2008), pp. 155–203.
- [14] Herbert Walther et al. “Cavity quantum electrodynamics”. In: *Reports on Progress in Physics* 69.5 (2006), p. 1325.
- [15] Dimitris G Angelakis, Marcelo Franca Santos, and Sougato Bose. “Photon-blockade-induced Mott transitions and X Y spin models in coupled cavity arrays”. In: *Physical Review A* 76.3 (2007), p. 031805.

- [16] Michael J Hartmann, Fernando GSL Brandao, and Martin B Plenio. “Strongly interacting polaritons in coupled arrays of cavities”. In: *Nature Physics* 2.12 (2006), p. 849.
- [17] Andrew D Greentree et al. “Quantum phase transitions of light”. In: *Nature Physics* 2.12 (2006), p. 856.
- [18] Dimitris G Angelakis. “Quantum Simulations with Photons and Polaritons”. In: *Quantum Science and Technology (Springer, 2017)* (2017).
- [19] Andrew A Houck, Hakan E Türeci, and Jens Koch. “On-chip quantum simulation with superconducting circuits”. In: *Nature Physics* 8.4 (2012), p. 292.
- [20] S Sachdev. “Quantum Phase Transitions (Cambridge University Press, Cambridge, 1999).” In: ().
- [21] Subir Sachdev. “Quantum phase transitions”. In: *Handbook of Magnetism and Advanced Magnetic Materials* (2007).
- [22] Toni Schneider and Johannes Maria Singer. *Phase transition approach to high temperature superconductivity: Universal properties of cuprate superconductors*. Imperial College Press London, 2000.
- [23] Lev Davidovich Landau. “On the theory of phase transitions”. In: *Ukr. J. Phys.* 11 (1937), pp. 19–32.
- [24] Matthew PA Fisher et al. “Boson localization and the superfluid-insulator transition”. In: *Physical Review B* 40.1 (1989), p. 546.
- [25] DF Walls and GJ Milburn. “Quantum Optics Springer”. In: *New York* (1994).
- [26] Kevin M Birnbaum et al. “Photon blockade in an optical cavity with one trapped atom”. In: *Nature* 436.7047 (2005), p. 87.
- [27] Andreas Wallraff et al. “Strong coupling of a single photon to a superconducting qubit using circuit quantum electrodynamics”. In: *Nature* 431.7005 (2004), p. 162.
- [28] M Hafezi, P Adhikari, and JM Taylor. “Chemical potential for light by parametric coupling”. In: *Physical Review B* 92.17 (2015), p. 174305.
- [29] Jens Koch and Karyn Le Hur. “Superfluid–Mott-insulator transition of light in the Jaynes-Cummings lattice”. In: *Physical Review A* 80.2 (2009), p. 023811.
- [30] Ulrich Schollwöck. “The density-matrix renormalization group in the age of matrix product states”. In: *Annals of Physics* 326.1 (2011), pp. 96–192.
- [31] Robert G Parr. “Density functional theory of atoms and molecules”. In: *Horizons of Quantum Chemistry*. Springer, 1980, pp. 5–15.
- [32] Wei-Min Zhang, Robert Gilmore, et al. “Coherent states: theory and some applications”. In: *Reviews of Modern Physics* 62.4 (1990), p. 867.
- [33] Giuseppe Carleo and Matthias Troyer. “Solving the quantum many-body problem with artificial neural networks”. In: *Science* 355.6325 (2017), pp. 602–606.
- [34] T. Mitchell. *Machine Learning*. McGraw Hill, 1997.

- [35] Stuart J Russell and Peter Norvig. *Artificial intelligence: a modern approach*. Malaysia; Pearson Education Limited, 2016.
- [36] Ethem Alpaydin. *Introduction to machine learning*. MIT press, 2009.
- [37] Juan Carrasquilla and Roger G Melko. “Machine learning phases of matter”. In: *Nature Physics* 13.5 (2017), p. 431.
- [38] Nobuyuki Yoshioka, Yutaka Akagi, and Hosho Katsura. “Learning disordered topological phases by statistical recovery of symmetry”. In: *Physical Review B* 97.20 (2018), p. 205110.
- [39] Patrick Huembeli, Alexandre Dauphin, and Peter Wittek. “Identifying quantum phase transitions with adversarial neural networks”. In: *Physical Review B* 97.13 (2018), p. 134109.
- [40] Giuseppe Carleo, Yusuke Nomura, and Masatoshi Imada. “Constructing exact representations of quantum many-body systems with deep neural networks”. In: *arXiv preprint arXiv:1802.09558* (2018).
- [41] Hiroki Saito. “Solving the Bose–Hubbard model with machine learning”. In: *Journal of the Physical Society of Japan* 86.9 (2017), p. 093001.
- [42] Zi Cai and Jinguo Liu. “Approximating quantum many-body wave functions using artificial neural networks”. In: *Physical Review B* 97.3 (2018), p. 035116.
- [43] Xiao Liang et al. “Solving frustrated quantum many-particle models with convolutional neural networks”. In: *Physical Review B* 98.10 (2018), p. 104426.
- [44] Kenny Choo et al. “Symmetries and Many-Body Excitations with Neural-Network Quantum States”. In: *Physical review letters* 121.16 (2018), p. 167204.
- [45] Hiroki Saito and Masaya Kato. “Machine learning technique to find quantum many-body ground states of bosons on a lattice”. In: *Journal of the Physical Society of Japan* 87.1 (2017), p. 014001.
- [46] Ivan Glasser et al. “Neural-network quantum states, string-bond states, and chiral topological states”. In: *Physical Review X* 8.1 (2018), p. 011006.
- [47] Raphael Kaubruegger, Lorenzo Pastori, and Jan Carl Budich. “Chiral topological phases from artificial neural networks”. In: *Physical Review B* 97.19 (2018), p. 195136.
- [48] Sirui Lu, Xun Gao, and L-M Duan. “Efficient Representation of Topologically Ordered States with Restricted Boltzmann Machines”. In: *arXiv preprint arXiv:1810.02352* (2018).
- [49] Yusuke Nomura et al. “Restricted Boltzmann machine learning for solving strongly correlated quantum systems”. In: *Physical Review B* 96.20 (2017), p. 205152.
- [50] Sandro Sorella. “Wave function optimization in the variational Monte Carlo method”. In: *Physical Review B* 71.24 (2005), p. 241103.
- [51] Nicholas Metropolis et al. “Equation of state calculations by fast computing machines”. In: *The journal of chemical physics* 21.6 (1953), pp. 1087–1092.

- [52] Sandro Sorella, Michele Casula, and Dario Rocca. “Weak binding between two aromatic rings: Feeling the van der Waals attraction by quantum Monte Carlo methods”. In: *The Journal of chemical physics* 127.1 (2007), p. 014105.
- [53] Diederik P Kingma and Jimmy Ba. “Adam: A method for stochastic optimization”. In: *arXiv preprint arXiv:1412.6980* (2014).
- [54] Giuseppe Carleo et al. *NetKet*. <http://www.netket.org>. 2018.

Appendix A

Programming codes for numerical solutions

A.1 Python code for calculating the numerical mean-field phase diagram of the BH model

```
# -*- coding: utf-8 -*-
"""
Created on Wed Oct 10 15:41:45 2018

@author: zejia
"""
import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import minimize_scalar

class BLMF:
    def e_g(ham):
        w,v=np.linalg.eigh(ham)
        return w[0]

    def __init__(self,n_max):
        self.n_max=n_max#total excitation
        self.create_local_operators()

    def create_local_operators(self):
        #photonic
        self.a=np.diag(np.sqrt(np.arange(self.n_max)+1),1)
        self.a_dag=self.a.T
        self.n=self.a_dag.dot(self.a)
        self.I=np.identity(self.n_max+1)

    def compute_hamiltonian(self,t,mu,z,psi):
        ham=-mu*self.n+0.5*self.n.dot(self.n-self.I)-t*z*psi*(self.a+
            self.a_dag)+t*z*psi**2*self.I
        return ham
```

```

def find_psi(self, t, mu, z):
    def e_min(psi):
        return BLMF.e_g(self.compute_hamiltonian(t, mu, z, psi))
    #x0=10
    res=minimize_scalar(e_min)
    return res.x

def main():
    test=BLMF(10)
    #coordination number
    z=6
    #resolution
    size=500
    ti=np.linspace(0,0.05,size)
    mui=np.linspace(0,3,size)
    global p
    p=np.zeros((size,size))

    for k1 in range(size):
        t=ti[k1]
        for k2 in range(size):
            mu=mui[k2]
            p[k2,k1]=np.abs(test.find_psi(t,mu,z))
    plt.figure(figsize=(12,10))
    plt.imshow(p,origin="lower",extent=[0,0.4,0,3],aspect="auto")
    cb=plt.colorbar()
    cb.set_label("order_parameter",size=20)
    cb.ax.tick_params(labelsize=20)
    plt.xlabel(r"$t/U$",fontsize=20)
    plt.ylabel(r"$\mu/U$",fontsize=20)
    plt.tick_params(labelsize=20)
    np.savez('pos_HB_MFT_N='+str(test.n_max)+'_z='+str(z)+'_res='+str(
        size)+'_dat',hop=ti,mu=mui,par=p)
    #plt.savefig('pos_HB_MFT_N='+str(test.n_max)+'_z='+str(z)+'_res='+
        str(size)+'_pdf')

if __name__ == "__main__":
    main()

```

A.2 Matlab code for the exact diagonalisation of the BH model in grand canonical ensemble

```

classdef BH<handle
    properties
        isPeriodic_
        L_,n_max_,mid_site_
        a_,a_dag_,n_
        A_,A_dag_,N_
    end
end

```

```

    I_one_site_
    dim_
    t_ ,mu_
    ham_
    order_parameter_
end

methods
function obj=BH(L,n_max,isPeriodic)
    obj.isPeriodic_=isPeriodic;
    obj.L_=L;
    obj.n_max_=n_max;
    obj.dim_=(n_max+1)^L;
    obj.create_local_operators();
    obj.create_lattice_operators();
    obj.mid_site_=round(obj.L_/2);
end

function create_local_operators(obj)
    obj.a_=sparse(sqrt(diag(1:obj.n_max_,1)));
    obj.a_dag_=transpose(obj.a_);
    obj.n_=obj.a_dag_*obj.a_;
    obj.I_one_site_=sparse(eye(obj.n_max_+1));
end

function res=create_lattice_operator(obj,op)
    res=cell(1,obj.L_);
    for i=1:obj.L_
        tmp=1;
        for j=1:i-1
            tmp=kron(tmp,obj.I_one_site_);
        end
        tmp=kron(tmp,op);
        for j=i+1:obj.L_
            tmp=kron(tmp,obj.I_one_site_);
        end
        res{i}=tmp;
    end
end

function create_lattice_operators(obj)
    obj.A_=obj.create_lattice_operator(obj.a_);
    obj.A_dag_=obj.create_lattice_operator(obj.a_dag_);
    obj.N_=obj.create_lattice_operator(obj.n_);
end

function update_hamiltonian(obj,t,mu)
    res=sparse(obj.dim_,obj.dim_);
    obj.t_=t;
    obj.mu_=mu;
    for i=1:obj.L_-1
        res=res-t*(obj.A_dag_{i}*obj.A_{i+1}+obj.A_{i}*obj.
            A_dag_{i+1});
    end
end

```



```

        if obj.isPeriodic_
            res=res-t*(obj.A_dag_{1}*obj.A_{obj.L_}+obj.A_{1}*obj.
                A_dag_{obj.L_});
        end
        for i=1:obj.L_
            res=res+0.5*obj.N_{i}*(obj.N_{i}-sparse(eye(obj.dim_)));
        end
        for i=1:obj.L_
            res=res-obj.mu_*obj.N_{i};
        end
        obj.ham_=res;

    end

    function update_order_parameter(obj)
        [v,~]=eigs(obj.ham_,1,'smallestreal');
        obj.order_parameter_=BH.var(v,obj.N_{obj.mid_site_});
    end

end

methods (Static)
    function res=expect(ket,op)
        res=transpose(ket)*op*ket;
    end

    function res=var(ket,op)
        res=BH.expect(ket,op*op)-(BH.expect(ket,op))^2;
    end
end
end

L=6;N=4;
test=BH(L,N,true);
size=100;
ti=linspace(0,0.4,size);
mui=linspace(0,3,size);
pars=zeros(size,size);
for k1=1:size
    t=ti(k1)
    for k2=1:size
        mu=mui(k2);
        test.update_hamiltonian(t,mu);
        test.update_order_parameter();
        pars(k2,k1)=test.order_parameter_;
    end
end
end
figure()
name=['BH_ed_L=',num2str(L),'_N=',num2str(N),'_res=',num2str(size),'.mat'];
save(name,'pars');
imshow(flipud(pars),'InitialMagnification','fit')

```

A.3 Python code for the exact diagonalisation of the BH model in canonical ensemble

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
Created on Thu Jan 24 12:07:06 2019

@author: zejian
"""
import numpy as np
import matplotlib.pyplot as plt
from scipy.sparse import csr_matrix
from scipy.sparse import kron, eye
import scipy.sparse as sparse

class BH:

    def __init__(self, L, n_max, canonical=False, M=0):
        self.L=L
        self.n_max=n_max
        self.isCanonical=canonical
        self.dim=(n_max+1)**L
        #total number of bosons
        self.create_local_operators()
        self.create_lattice_operators()
        if(canonical):
            self.M=M
            self.find_keep_sites()

    def create_local_operators(self):
        self.a=csr_matrix(np.diag(np.sqrt(np.arange(self.n_max)+1),1))
        self.a_dag=self.a.transpose()
        self.n=self.a_dag.dot(self.a)
        self.I_one_site=eye(self.n_max+1)

    def create_lattice_operator(self, op_loc, op_lat):
        L=self.L
        for i in range(L):
            tmp=1
            for j in range(i):
                tmp=kron(tmp, self.I_one_site)
            tmp=kron(tmp, op_loc)
            for j in range(L-i-1):
                tmp=kron(tmp, self.I_one_site)
            op_lat.append(tmp.tocsr())

    def create_lattice_operators(self):
        self.A=[]
        self.A_dag=[]
        self.N=[]
        self.create_lattice_operator(self.a, self.A)
        self.create_lattice_operator(self.a_dag, self.A_dag)
        self.create_lattice_operator(self.n, self.N)
```

```

        self.N_tot=0
        for i in range(self.L):
            self.N_tot=self.N_tot+self.N[i]
def update_hamiltonian(self , t ,mu,U=1,periodic=True):
    A=self.A
    A_dag=self.A_dag
    N=self.N
    L=self.L
    H=0
    if(self.isCanonical):
        mu=0
    lmax=L-1
    if(periodic):
        lmax=L
    for i in range(lmax):
        H=H-t*(A_dag[i].dot(A[(i+1)%L])+A_dag[(i+1)%L].dot(A[i]))
            +0.5*U*N[i].dot(N[i]-eye(self.dim))-mu*N[i]
    N=self.N[0]
    if(self.isCanonical):
        H=H[self.sites_keep ,:][: , self.sites_keep]
        N=N[self.sites_keep ,:][: , self.sites_keep]
    H_dense=H.todense()
    w,v=np.linalg.eigh(H_dense)
    g=csr_matrix(v[:,0])
    #order parameter p=var(n)
    self.order_parameter=BH.variance(g,N)
    self.ground_energy=w[0]
    self.gap=w[1]-w[0]

def find_keep_sites(self):
    """
    keep the basis vectors with the correct number of excitations
    """
    M=self.M
    self.sites_keep=[]
    dim=self.dim
    for i in range(dim):
        m=self.N_tot[i,i]
        if(m>M-0.5 and m<M+0.5):
            self.sites_keep.append(i)

def variance(ket , operator):
    return BH.expectation(ket , operator.dot(operator))-BH.expectation
        (ket , operator)**2

def expectation(ket , operator):
    return ket.T.dot(operator).dot(ket)[0,0]

def main():
    global test , u_range , pars , engs
    L=3
    n_boson=3

```

```

n_max=3
test=BH(L, n_max, canonical=True, M=n_boson)
size=26
u_range=np.logspace(-1, 2.75, size)
pars=np.zeros(size)
engs=np.zeros(size)
gaps=np.zeros(size)
for i in range(size):
    test.update_hamiltonian(1, 0, U=u_range[i])
    pars[i]=test.order_parameter
    engs[i]=test.ground_energy
    gaps[i]=test.gap
plt.figure(figsize=(12, 10))
plt.plot(u_range, pars)
plt.xscale('log')
plt.xlabel(r'$U/t$', fontsize=30)
plt.ylabel('order_parameter', fontsize=30)
plt.tick_params(labelsize=30)

# plt.figure()
# plt.plot(u_range, engs)
# plt.xscale('log')
np.savez('HBL'+str(L)+'M'+str(n_boson)+'N'+str(n_max)+'_1.dat', pars=
        pars, engs=engs, u_range=u_range, gaps=gaps)

#

if __name__ == "__main__":
    main()

```

A.4 Matlab code for the exact diagonalisation of the JCH model in canonical ensemble

```

classdef jch < handle
    properties
        isPeriodic
        L, n_max, n_boson
        a, a_dag, n_photon, I_one_photon
        s_p, s_m, s_z, I_one_atom
        A, A_dag, N_photon
        N_polariton, N_tot_polariton
        S_p, S_m, S_z
        I_one_site
        sites_keep
        ham_crop
        mid_site
        dim
        om_d, delta, J, g
        gs_energy, gap, order_parameter
    end

    methods

```

```

function obj=jch(L,n_boson,n_max,isPeriodic)
    obj.dim=((2*(n_max+1))^L);
    obj.isPeriodic=isPeriodic;
    obj.L=L;
    obj.n_max=n_max;
    obj.n_boson=n_boson;
    obj.create_local_operators();
    disp('local_operators_created');
    obj.create_lattice_operators();
    disp('lattice_operators_created');
    obj.find_keep_sites();
    disp('unit_fill_subspace_found');
    obj.mid_site=round(obj.L/2);
end

function create_local_operators(obj)
    %local operators
    obj.a=sparse(sqrt(diag(1:obj.n_max,1)));
    obj.a_dag=transpose(obj.a);
    obj.n_photon=obj.a_dag*obj.a;
    obj.I_one_photon=sparse(eye(obj.n_max+1));

    obj.s_p=sparse(diag(1,-1));
    obj.s_m=transpose(obj.s_p);
    obj.s_z=obj.s_p*obj.s_m;
    obj.I_one_atom=sparse(eye(2));
    obj.I_one_site=kron(obj.I_one_atom,obj.I_one_photon);

    obj.a=kron(obj.I_one_atom,obj.a);
    obj.a_dag=kron(obj.I_one_atom,obj.a_dag);
    obj.n_photon=kron(obj.I_one_atom,obj.n_photon);

    obj.s_p=kron(obj.s_p,obj.I_one_photon);
    obj.s_m=kron(obj.s_m,obj.I_one_photon);
    obj.s_z=kron(obj.s_z,obj.I_one_photon);
end

function res=create_lattice_operator(obj,op)
    res=cell(1,obj.L);
    for i=1:obj.L
        tmp=1;
        for j=1:i-1
            tmp=kron(tmp,obj.I_one_site);
        end
        tmp=kron(tmp,op);
        for j=i+1:obj.L
            tmp=kron(tmp,obj.I_one_site);
        end
        res{i}=tmp;
    end
end

function create_lattice_operators(obj)
    obj.A=obj.create_lattice_operator(obj.a);

```

```

obj.A_dag=obj.create_lattice_operator(obj.a_dag);
obj.N_photon=obj.create_lattice_operator(obj.n_photon);
obj.S_p=obj.create_lattice_operator(obj.s_p);
obj.S_m=obj.create_lattice_operator(obj.s_m);
obj.S_z=obj.create_lattice_operator(obj.s_z);

obj.N_polariton=cell(1,obj.L);
for i=1:obj.L
    obj.N_polariton{i}=obj.N_photon{i}+obj.S_z{i};
end
obj.N_tot_polariton=sparse(obj.dim,obj.dim);
for i=1:obj.L
    obj.N_tot_polariton=obj.N_tot_polariton+obj.N_polariton{
        i};
end

end

function find_keep_sites(obj)
M=obj.n_boson;
dia_M=round(diag(obj.N_tot_polariton));
obj.sites_keep=find(dia_M==M);
end

function res=reduce_operator(obj,op)
res=op(obj.sites_keep,:);
res=res(:,obj.sites_keep);
res=sparse(res);
end

function obj=update_hamiltonian(obj,om_d,delta,g,J)

res=sparse(obj.dim,obj.dim);
om_0=om_d+delta;
obj.om_d=om_d;
obj.g=g;
obj.J=J;
obj.delta=delta;
for i=1:obj.L
    res=res+om_d*obj.N_photon{i}+om_0*obj.S_z{i};
    res=res+g*(obj.A_dag{i}*(obj.S_m{i})+obj.A{i}*(obj.S_p{i}
        )));
end
for i=1:obj.L-1
    res=res+J*(obj.A_dag{i}*(obj.A{i+1})+obj.A{i}*(obj.A_dag
        {i+1}));
end
if obj.isPeriodic
    res=res+J*(obj.A_dag{1}*(obj.A{obj.L})+obj.A{1}*(obj.
        A_dag{obj.L}));
end
end

```

```

        res=obj.reduce_operator(res);
        %obj.ham_crop=full(res);
        obj.ham_crop=res;
    end
    function obj=update_order_parameter_sparse(obj)
        n_crop=obj.reduce_operator(obj.N_polariton{obj.mid_site});
        [v,d]=eigs(obj.ham_crop,1,'smallestreal');
        obj.gs_energy=d(1);
        gs=v(:,1);
        obj.order_parameter=jch.var(gs,n_crop);
    end
    function obj=update_order_parameter(obj)
        n_crop=obj.reduce_operator(obj.N_polariton{obj.mid_site});
        hc=full(obj.ham_crop);
        [v,d]=eig(hc);
        [v,d]=sortem(v,d);
        d=diag(d);
        obj.gap=d(2)-d(1);
        obj.gs_energy=d(1);
        gs=v(:,1);
        obj.order_parameter=jch.var(gs,n_crop);
    end
    function obj=find_ground_state_energy_sparse(obj)
        d=eigs(obj.ham_crop,2,'smallestreal');
        disp(d)
        obj.gap=d(2)-d(1);
        obj.gs_energy=d(1);
    end

    function obj=find_ground_state_energy(obj)
        hc=full(obj.ham_crop);
        [v,d]=eig(hc);
        [~,d]=sortem(v,d);
        d=diag(d);
        obj.gap=d(2)-d(1);
        obj.gs_energy=d(1);
    end

end

methods (Static)
    function res=expect(ket,op)
        res=transpose(ket)*op*ket;
    end

    function res=var(ket,op)
        res=jch.expect(ket,op*op)-(jch.expect(ket,op))^2;
    end
end

end

%
%
```

```

L=5;
J=10(-5);
g=10(-3);
om_d=10;
n_max=3;n_boson=L;
n_sample=50;
test=jch(L,n_boson,n_max,0);
delta=logspace(-2.5,2.5,n_sample)*g;
pars=zeros(n_sample,1);
eig_energies=zeros(n_sample,1);
% gaps=zeros(n_sample,1);
for k=1:n_sample
    d=delta(k);
    disp(d);
    test.update_hamiltonian(om_d,d,g,J);
    test.update_order_parameter();
    pars(k)=test.order_parameter;
    eig_energies(k)=test.gs_energy;
%     gaps(k)=test.gap;
end
deltaByG=delta/g;
name=['plotParL',sprintf('%d',L),'M',sprintf('%d',n_boson),'nMax',
    sprintf('%d',n_max),'nSample',sprintf('%d',n_sample),'.mat'];
% save(name,'pars','eig_energies','deltaByG')
figure()
plot(delta/g,pars)
set(gca,'XScale','log')

```

A.5 Python code for calculating the numerical mean-field phase diagram of the JCH model

```

# -*- coding: utf-8 -*-
"""
Created on Wed Oct 10 15:41:45 2018

@author: zejia
"""
import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import minimize_scalar

class JCHMF:
    def e_g(ham):
        w,v=np.linalg.eigh(ham)
        return w[0]

    def __init__(self,n_max):
        self.M=n_max#total excitation
        self.create_local_operators()
        self.create_lattice_operators()

    def create_local_operators(self):

```



```

#photonic
self.a=np.diag(np.sqrt(np.arange(self.M)+1),1)
self.a_dag=self.a.T
self.n_photon=self.a_dag.dot(self.a)
#atomic
self.s_p=np.diag(np.sqrt(np.arange(1)+1),-1)
self.s_m=np.diag(np.sqrt(np.arange(1)+1),+1)
self.s_z=self.s_p.dot(self.s_m)
#identities
self.I_one_atom=np.identity(2)
self.I_one_photon=np.identity(self.M+1)

def create_lattice_operators(self):
#photonic
self.A=np.kron(self.I_one_atom,self.a)
self.A_dag=np.kron(self.I_one_atom,self.a_dag)
self.N_photon=np.kron(self.I_one_atom,self.n_photon)

#atomic
self.S_p=np.kron(self.s_p,self.I_one_photon)
self.S_m=np.kron(self.s_m,self.I_one_photon)
self.S_z=np.kron(self.s_z,self.I_one_photon)

#polaritonic
self.N_polariton=np.add(self.N_photon,self.S_z)

def compute_hamiltonian(self,om_0,om_d,g,k,z,psi,mu):
ham=0
ham=ham+om_0*self.S_z+om_d*self.N_photon+g*(self.S_p.dot(self.A)
+self.S_m.dot(self.A_dag))
ham=ham-z*k*psi*(self.A_dag+self.A)+z*k*np.absolute(psi)**2*np.
identity(2*(self.M+1))-mu*self.N_polariton
return ham

def find_psi(self,om_0,om_d,g,k,z,mu):
def e_min(psi):
return JCHMF.e_g(self.compute_hamiltonian(om_0,om_d,g,k,z,
psi,mu))
#x0=10
res=minimize_scalar(e_min)
return res.x

def main():
for delta in [-2,0,2]:
test=JCHMF(10)
om_0=1
om_d=om_0-delta
g=1
k=0
z=3

```

```

size=500
ki=np.logspace(-4,1,size)
mui=om_d+np.linspace(-2,0,size)
p=np.zeros((size,size))
for n1 in range(size):
    k=ki[n1]
    for n2 in range(size):
        mu=mui[n2]
        p[n2,n1]=np.abs(test.find_psi(om_0,om_d,g,k,z,mu))
plt.figure(figsize=(12,10))
plt.imshow(p,origin="lower",aspect="auto",extent=[-4,1,-2,0],)
cb=plt.colorbar()
cb.set_label("order parameter",size=20)
cb.ax.tick_params(labelsize=20)
plt.xlabel(r"photon hopping,  $\log_{10}(A/g)$ ",fontsize=20)
plt.ylabel(r"chemical potential,  $\mu - \omega_d/g$ ",fontsize=20)
plt.tick_params(labelsize=20)
np.savez('JCHMF_M='+str(test.M)+'_delta='+str(delta)+'_z='+str(z)+'_res='+str(size)+'.dat',logHop=ki,mu=mui,par=p)
#plt.savefig('JCHMF_M='+str(test.M)+'_delta='+str(delta)+'_z='+str(z)+'_res='+str(size)+'.pdf')

```

```

if __name__ == "__main__":
    main()

```