Neural Network Solver for Small Quantum Clusters

Nicholas Walker,1 Samuel Kellar,1 Yi Zhang,1,2,3 and Ka-Ming Tam1,2

1Department of Physics and Astronomy, Louisiana State University, Baton Rouge, LA 70803, USA
2Center for Computation & Technology, Louisiana State University, Baton Rouge, LA 70803, USA
3Kavli Institute for Theoretical Sciences, University of Chinese Academy of Sciences, Beijing, 100190, China

(Dated: June 11, 2021)

Machine learning approaches have recently been applied to the study of various problems in physics. Most of the studies are focused on interpreting the data generated by conventional numerical methods or an existing database. An interesting question is whether it is possible to use a machine learning approach, in particular a neural network, for solving the many-body problem. In this paper, we present a solver for interacting quantum problem for small clusters based on the neural network. We study the small quantum cluster which mimics the single impurity Anderson model. We demonstrate that the neural network based solver provides quantitatively accurate results for the spectral function as compared to the exact diagonalization method. This opens the possibility of utilizing the neural network approach as an impurity solver for other many body numerical approaches, such as dynamical mean field theory.

I. INTRODUCTION

A single quantum impurity is the simplest possible quantum many body problem for which interaction plays a crucial role.12 It was invented as a model to describe diluted magnetic impurity in the otherwise non-magnetic metallic host. It has been known from the beginning that the perturbation series diverges even with an infinitesimal anti-ferromagnetic coupling strength. This became a major problem of strongly correlated systems from the 60’s.13

While the physics of a single impurity problem has been rather well studied, interest in the quantum impurity problem was revived during the 90’s. This was partly due to the interest in mapping lattice models onto impurity models.5,15 It has been shown that at infinite dimensions, the lattice models are equivalent to single impurity models in a mean-field as represented by the density of states of the host. This approximated mapping is known as the dynamical mean field theory. It has been further generalized to cluster impurity models to include the effects for finite dimensional systems.20,21

These mappings provide a systematic tractable approximation for the lattice models and have become a major paradigm in the field of strongly correlated systems.11 Combined with density functional theory, they provide one of the best available methods for the study of properties of materials in which the strong interaction is important.12

Unlike the infinite band limit which normally considered for the single impurity problem, the density of the bath, i.e. the mean-field, can be rather complicated. There is in general no analytic method for a very accurate solution. Many different methods for solving the effective impurity problem have been proposed. They can be broadly divided in the two categories: semi-analytic, and numeric.

For the semi-analytic methods, the most widely used one is iterative perturbation theory. Its idea is to interpolate the self-energy at both the weak and strong coupling limit, together with some exact constraints, such as Luttinger theorem.19 The other is the local moment approximation. It considers the perturbation on top of the strong coupling limit represented by the unrestricted Hartree Fock solution.13

Numerical methods can be further divided into two main classes, diagonalization based and the Monte Carlo based. Diagonalization methods usually require to discretize the host by a finite number of so-called bath sites. The Hamiltonian which includes the bath sites and one impurity site is diagonalized exactly.15 Another diagonalization based method is the numerical renormalization group in which the bath sites are mapped onto a one dimensional chain of sites. The hopping amplitude decreases rapidly down the chain. The model is then diagonalized iteratively as more sites are included.10 Density matrix renormalization group and coupled cluster theory have also been used as impurity solvers.17,20

On the other hand, the quantum Monte Carlo method for solving impurity problems was first proposed by Hirsh and Fye.21 The idea is to break up the time axis by the Trotter-Suzuki approximation. The interaction in each time segment is handled by the Hubbard-Stratonovich approximation.22 The Monte Carlo method is then used to sample the Hubbard-Stratonovich fields. The idea of sampling the partition function without the Trotter-Suzuki approximation has been borrowed from the Stochastic Series Expansion in the simulation of quantum spin models to a so-called continuous time quantum Monte Carlo.23 The method has seen a lot of development over the last decade. Notably, the expansion with respect to the strong coupling limit has been proposed and complicated coupling functions beyond simple Hubbard local density-density coupling term can now be studied.24

The past few years have seen tremendous development of machine learning (ML) both in terms of the algorithms and the implementation.25,26 Many of the ML approaches in physics are designed to detect phase transitions or accelerate Monte Carlo simulations. It is a
tantalizing proposal to utilize ML approaches to build a solver for quantum systems.

A possible route to build a quantum solver based on the ML approach is to identify the feature vector (input data) and the label (output data) for the problem. Then a large pool of data are generated to train the model, specifically a neural network model. An Anderson impurity problem is a good test bed for the validity of such a solver. We note that similar ideas have been explored using machine learning approaches. This paper is focused on using the kernel polynomial expansion and supervised ML, specifically, a neural network as building blocks for a quantum impurity solver.

While it is relatively cheap to solve a single impurity problem from the above methods in the modern computational facilities, the interest in the random disorder warrants a new requirement to solve a large set of single or few impurities problems for calculating the random disorder averaging. The hope is that a fast neural network based numerical solver in real frequency can expand the range of applicability of the recently developed typical medium theory for interacting strongly correlated systems, such as the Anderson-Hubbard model.

The paper is organized as follows. In the next section, we discuss mapping the continuous Green function into a finite cluster as has been done in many numerical calculations of the dynamical mean field theory. In the section III, we discuss the expansion of the spectral function in terms of the Chebyshev polynomials. In the section IV, we explain how to use the results from the section II and III as the feature vectors and labels of the neural network. In the section V, we present the spectral function calculated from the neural network approach. We conclude and discuss the future work in the last section.

II. REPRESENTING THE HOST BY FINITE NUMBER OF BATH SITES

We first identify the input and the output data of a single impurity Anderson model. For the input data, it includes the bare density of states, the chemical potential and the Hubbard interaction of the impurity site. For a system in the thermodynamic limit, the density of states is represented by a continuous function. Representing a continuous function in the neural network presents a problem. We use the idea of exact diagonalization of the Anderson model to describe the continuous bath by a finite number of poles. We first approximate the host Green function by a cluster of bath sites,

\[ G_0(i\omega_n) \approx G^c_0(i\omega_n). \]  
(1)

In the exact diagonalization method for impurity models, the continuum bath is discretized and represented by a finite number of so-called bath sites, see Fig.1. Assuming that there are \( N_b \) bath sites, each bath site is characterized by a local energy (\( \epsilon_i \)) and a hopping (\( t_i \)) term with the impurity site. Two additional variables, one for the local Hubbard interaction (\( U \)) and the other for the chemical potential (\( \epsilon_f \)), are required to describe the impurity site. Therefore, there are in total \( 2+2N_b \) variables for representing the impurity problem. The host Green function represented in a finite cluster can be written exactly as following,

\[ G^{cl}_0(i\omega_n) = (i\omega_n + \epsilon_f - \sum_{k=1}^{N_b} t_k t_k^\dagger / i\omega_n - \epsilon_k)^{-1}. \]  
(2)

The full Hamiltonian in the discretized form is represented pictorially in Fig.1. It is given as

\[ H = \sum_{i,\sigma} t_i(c_{i,\sigma}^\dagger c_{0,\sigma} + H.c.) + \sum_{i,\sigma} \epsilon_i c_{i,\sigma}^\dagger c_{i,\sigma} + U(c_{0,\uparrow}^\dagger c_{0,\downarrow} - 1/2)(c_{0,\downarrow}^\dagger c_{0,\uparrow} - 1/2) - \epsilon_f \sum_{\sigma} c_{0,\sigma}^\dagger c_{0,\sigma}. \]  
(3)

\( c_{i,\sigma}^\dagger \) and \( c_{i,\sigma} \) are the creation and annihilation operators for the site \( i \) with spin \( \sigma \) respectively. The impurity site is denoted as the 0-th site. The sum of the bath sites are from 1 to \( N_b \) and the sum of the spin is for the the up and down spins for the electrons.

Parameterizing the host Green function by a finite number of variables is a standard procedure for the exact diagonalization solver for quantum impurity problems. Many different prescriptions have been investigated in details to optimize this approximation. Conceptually, practical applications of the numerical renormalization group method also require the approximated mapping onto a finite cluster chain. Unlike the exact diagonalization method, the cluster chain can be rather large, therefore much higher accuracy can be attained in general.

The mapping onto the finite cluster to mimic the continuous bath may represent a nuisance. Nonetheless, this is a necessity for any diagonalization based method. These methods do not mimic the situation of continuum in the time dimension as done by continuous time Quantum Monte Carlo methods. However the mapping presents an opportunity to naturally adapt to a machine learning approach in which a finite discretized set of variables is required.

Under the above approximation, the finite set of variables, \( \{ t_i \}, \{ \epsilon_i \}, U, \epsilon_f \) can be treated as the input feature vector for the machine learning algorithms. The next question is what is the desired output or label for the feature vectors. We will focus on the spectral function in this study. For this purpose, the next step is to represent the spectral function in a finite number of variables instead of a continuous function. We will show in the next section that the kernel polynomial method fulfills this goal.
III. EXPANDING THE IMPURITY GREEN FUNCTION BY CHEBYSHEV POLYNOMIALS

In this section, we briefly discuss the kernel polynomial method for the calculation of the spectral function of a quantum interacting model. Once the host parameters, the impurity interaction, and chemical potential are fixed, the ground state of the cluster is obtained using a diagonalization method for sparse matrices. We use the Lanczos approach in the present study. Once the ground state is found, the spectral function can be calculated by applying the resolvent operator, \( \frac{1}{\omega - H} \), to the ground state. A popular method is the continuous fraction expansion. The challenge is that the continuous fraction tends to be under-damped and produce spurious peaks. A more recent method is to use the orthogonal polynomial expansion. We will argue that for applications, the orthogonal polynomial expansion method tends to produce better results as we will explain later.

The zero temperature single particle retarded Green function corresponding to a generic many-body Hamiltonian is defined as

\[
G(\omega) = \langle GS | c \frac{1}{\omega + i0^+ - H} c^\dagger | GS \rangle. \tag{4}
\]

\( |GS \rangle \) is the ground state of \( H \). \( c \) and \( c^\dagger \) are the creation and annihilation operators, respectively. The spectral function is given as \( A(\omega) = -(1/\pi) \text{Im}(G(\omega)) \). It is more convenient to directly expand the spectral function given as

\[
A(\omega) = \langle GS | c(\omega - H) c^\dagger + c^\dagger \delta(\omega - H) c | GS \rangle. \tag{5}
\]

Consider the Chebyshev polynomials of the first kind defined as \( T_n(x) = \cos(n \arccos(x)) \). Two important properties are the orthogonality and the recurrence relations. The product of two Chebyshev polynomials integrated over \( x = [-1, 1] \) weighted by the function \( w_n = \frac{1 + \delta_{n,0}}{\sqrt{1 - x^2}} \)

is given as

\[
\int dx w_n(x) T_m(x) T_n(x) = \delta_{n,m}. \tag{6}
\]

The recurrence relation is given as

\[
T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x). \tag{7}
\]

The Chebyshev polynomials expansion method is based on the fact that the set of Chebyshev polynomials form an orthonormal basis as defined in the Eq. (6). Thus a function, \( f(x) \) defined within the range of \( x = [-1, 1] \) can be expanded as

\[
f(x) = \sum_{n=0}^{\infty} \mu_n T_n(x), \tag{8}
\]

and the expansion coefficient can be obtained by the inner product of the function \( f(x) \) and the Chebyshev polynomials as follow

\[
\mu_n = \int_{-1}^{1} dx f(x) T_n(x) w_n(x). \tag{9}
\]

Practical calculation involves truncation at a finite order. The truncation is found to be problematic, especially when the function, \( f(x) \), is not smooth. For our application, the function is a spectral function of a finite size cluster, which is a linear combination of a set of delta functions. For this reason, a direct application of the above formula will not provide a smooth function. This is in analogue with the Gibbs oscillations in the Fourier expansion. The remedy is to introduce a damping factor (kernel) in each coefficient of the expansion. We refer the choice of the damping factor to the review. We use the Jackson kernel given as

\[
f(x) \approx \sum_{n=0}^{N} g_n \mu_n T_n(x) \tag{10}
\]

\[
g_n = \frac{(N - n + 1) \cos(\frac{\pi n}{N+1}) + \sin(\frac{\pi n}{N+1}) \cot(\frac{\pi}{N+1})}{N+1}. \tag{11}
\]

We list the steps for calculating the coefficients as follows.

1. The input bare Green function is approximated by the bare Green function of a finite size cluster. The set of parameters \( \mu, \{t_i\}, \{\epsilon_i\} \) are obtained by minimizing the difference between the left hand side and the right hand size of the Eq. (1) according to some prescriptions.

2. The ground state \( (|GS\rangle) \) and the corresponding energy \( E_{GS} \) are obtained by Lanczos algorithm.

3. The spectrum of the Hamiltonian are scaled to within the range of [-1,1] as required by the Chebyshev expansion. \( H \Rightarrow (H - E_{GS})/a \), where \( a \) is a real positive constant. The units of energy are also scaled in terms of \( a \).
4. The expansion coefficients are given by the inner product between the spectral function and the Chebyshev polynomials.

\[ \mu_n = \langle \alpha_n | \alpha_n \rangle, \]

where \( |\alpha_0\rangle = c|GS\rangle \) and \( |\alpha_n\rangle = T_n(H)|\alpha_0\rangle \). With the \( |\alpha_0\rangle \) and the \( |\alpha_1\rangle = H|\alpha_0\rangle \) ready, all the higher order coefficients can be obtained via the recurrence relation.

\[ |\alpha_n\rangle = 2H|\alpha_{n-1}\rangle - |\alpha_{n-2}\rangle \]

5. The spectral function is obtained by feeding the coefficients into Eq. (8).

All the coefficients can be obtained by repeated use of the Eq. (13) which involves matrix vector multiplication. The matrix for interacting system is usually very sparse, and the computational complexity of the matrix vector multiplication is linear with respect to the vector length, which grows as \( 4^{N_b+1} \) assuming no reduction by symmetries is employed.

IV. FEATURE VECTORS AND THE LABELS FOR THE MACHINE LEARNING

Our strategy is to train a neural network for a large set of variables for the host, i.e., the bath sites, impurity interaction and the impurity chemical potential. The impurity solver is a function of the impurity Green function given by the bath Green function and the impurity site interaction and chemical potential, that is in total \( 2 + 2N_b \) variables for the input.

The impurity Green function can be represented by \( N \) coefficients of the Chebyshev polynomials expansion for the output. Using the above method the spectral function is effectively represented in terms of \( N \) coefficients. It allows us to naturally employed the supervised learning method by identifying the \( 2 + 2N_b \) variables as the input feature vectors, and the \( N \) variables as the output labels.

While the kernel polynomial method grows exponentially with the number of sites, the end result is represented by a finite number of coefficients which presumably does not scale exponentially with the number of sites. Once the neural network is properly trained, we can use it to predict the impurity Green function without involving a calculation which scales exponentially.

V. RESULTS

We generated 5000 samples by the KPM method for randomly chosen parameters. They are drawn uniformly from the range listed as follows.

\[ t_{i,\uparrow} = t_{i,\downarrow} = [0, 1.5] \]
\[ \epsilon_{i,\uparrow} = \epsilon_{i,\downarrow} = [-5, 5] \]
\[ U = [0, 10] \]
\[ \epsilon_f = [-2.5, 2.5] \]

We assume that the electron bath has a symmetric density of states. That is \( t_i = t_{i+N_b/2} \) and \( \epsilon_i = -\epsilon_{i+N_b/2} \) for \( i = 1 \) to \( N_b/2 \) and \( N_b \) even. This further reduces the number of variables in the feature vector to \( N_b + 2 \).

Before embarking on training the neural network, we would like to have some idea of the coefficients. We randomly pick the 32 samples and plot the coefficients in Fig. 5. There are two prominent features of the coefficients: 1. There are clear oscillations and the coefficients do not decrease monotonically; 2. For all cases shown here the coefficients essentially vanish for the orders which are around 200 or higher. Due to these two reasons we decided to train the neural network for the coefficients from order 0 to order 255.

With the above approximations, the task of solving the Anderson impurity model boils down to mapping a vector containing \( N_b + 2 \) variables to a vector containing \( N \) coefficients. For the particular case we study \( N_b = 6 \), and \( N = 256 \). Machine learning algorithms can thus be naturally applied to this mapping.

We set up an independent dense neural network for each coefficient. The neural network has 14 layers. The input layer contains \( N_{in} = N_b + 2 \) units, and the output layer contains the expansion coefficient for one specific order. The twelve hidden layers have the number of units as follows \( 2N_{in}, 2N_{in}, 4N_{in}, 4N_{in}, 8N_{in}, 8N_{in}, 8N_{in}, 8N_{in}, 8N_{in}, 4N_{in}, 4N_{in}, 2N_{in}, 2N_{in} \).

As we consider in total of 256 orders, we have 256 independent neural networks. Considering the coefficients at different orders separately may lose some information contained in the correlations between different orders. While it is possible to predict a few coefficients by one neural network, we do not get a good prediction for using a single neural network to predict all 256 coefficients without an elaborated fine tuning. Therefore, instead of searching for an optimal number of coefficients for one neural network, we consider each coefficient independently.

We show the spectral function in the Fig. 3 they are from the same 32 samples as that in the Fig. 2. Both the results from the direct numerical calculation based on the Lanczos method and recurrence relation and those from the neural network prediction are plotted. They basically overlap with each other. There is a slight difference for the range of energy where the spectral function is nearly zero. This is perhaps due to the incomplete cancellation among the expansion terms at different orders due to the errors from the neural networks. An improvement may be attainable if we consider the correlations of the coefficients for different orders. The input parameters of each of the 32 samples are plotted in Fig. 4.

Evidence of the capability of the neural network approach can be seen in Fig. 5. We plot the comparisons of the first 32 expansion coefficients obtained by the direct numerical calculation and the neural network prediction. 1000 samples are considered, we find that two methods give very close results. With the 1000 samples we considered, all exhibit a linear trend. This clearly shows that a neural network is capable of providing a good predic-
VI. CONCLUSION

We demonstrate that the supervised machine learning approach, specifically the neural network method, can be utilized as a solver for small quantum interacting clusters. This could be potentially useful for the statistical DMFT or typical medium theory for which a large number of impurity problems have to be solved for disorder averaging. The main strategy is to devise a finite number of variables as the feature vector and the label for the supervised machine learning method. In line with the exact diagonalization method for the single impurity Anderson model, the feature vector is represented by the hopping and the local energy of the lattice model. The output, spectral function, is represented in terms of Chebyshev polynomials with the damping kernel. The label is then represented by the coefficients of the expansion. By comparing the coefficients directly calculated by the Lanczos method and the recurrence relation and that by the neural network, we find the agreement between the results from these two methods is very good. Notably, among the 1000 samples being tested, there is no exceptional outlier. They all have good agreement with that from the direct numerical method.

For a simple impurity problem, the present method may not have an obvious benefit, as a rather large pool of samples have to be generated for training at the first place. The situation is completely different for the study of disorder models, such as those being studied by the typical medium theory, where the present method has a clear advantage. Once the neural network is trained the calculations is computationally cheap. For systems in which disorder averaging is required, this method can beat most if not all numerical methods in term of efficiency. Moreover the present approach is rather easy to be generalized for more complicated models, such as a few impurities model required in the dynamical cluster approximation. In addition, the matrix product basis has been proposed for the kernel polynomial expansion, this method can be easily adapted for it.

The ideas presented in this paper is rather generic. They can be generalized for the solutions from different solvers. For example, it can be adapted to the solutions from QMC as long as the solutions can be represented in some kind of series expansion and it can also be adapted for the coefficients from the coupled cluster theory.

VII. ACKNOWLEDGEMENT

We thank Mark Jarrell for his comments and suggestions for this project. This work is funded by the NSF Materials Theory grant DMR1728457. This work used the high performance computational resources provided by the Louisiana Optical Network Initiative (http://www.loni.org) and HPC@LSU computing. Additional support (NW and KMT) was provided by the NSF EPSCoR CIMP project under award OIA-154107. Additional support (YZ) at Louisiana State University was provided by the U.S. Department of Energy, Office of Science, Office of Basic Energy Sciences under Award No. DE-SC0017861.

---

1. J. Kondo, Prog. Theor. Phys. 32, 37 (1964), URL https://link.aps.org/doi/10.1103/PTP.32.37
2. P. W. Anderson, J. Phys. C: Solid State Physics 3, 2436 (1970), URL https://doi.org/10.1088/0022-3719/2/F3/12/F2006
3. K. G. Wilson, Rev. Mod. Phys. 47, 773 (1975), URL https://link.aps.org/doi/10.1103/RevModPhys.47.773
4. E. Müller-Hartmann, Z. Phys. B 76, 211 (1989), URL https://doi.org/10.1007/BF01312686
5. E. Müller-Hartmann, Z. Phys. B 74, 507 (1989), URL https://doi.org/10.1007/BF01311397
6. W. Metzner and D. Vollhardt, Phys. Rev. Lett. 62, 324 (1989), URL https://link.aps.org/doi/10.1103/PhysRevLett.62.324
7. A. J. Bray and M. A. Moore, J. Phys. C: Solid State Physics 13, L655 (1980), URL https://doi.org/10.1088/0022-3719/2/F13/2/F005
8. A. Georges, G. Kotliar, W. Krauth, and M. J. Rozenberg, Rev. Mod. Phys. 68, 13 (1996), URL https://link.aps.org/doi/10.1103/RevModPhys.68.13
9. M. H. Hettler, M. Mukherjee, M. Jarrell, and H. R. Krishnamurthy, Phys. Rev. B 61, 12739 (2000), URL https://link.aps.org/doi/10.1103/PhysRevB.61.12739
10. G. Biroli and G. Kotliar, Phys. Rev. B 65, 155112 (2002), URL https://link.aps.org/doi/10.1103/PhysRevB.65.155112
11. T. Maier, M. Jarrell, T. Pruschke, and M. H. Hettler, Rev. Mod. Phys. 77, 1027 (2005), URL https://link.aps.org/doi/10.1103/RevModPhys.77.1027
12. G. Kotliar, S. Y. Savrasov, K. Haule, V. S. Oudovenko, O. Parcollet, and C. A. Marianetti, Rev. Mod. Phys. 78, 865 (2006), URL https://link.aps.org/doi/10.1103/RevModPhys.78.865
13. H. Kajueter and G. Kotliar, Phys. Rev. Lett. 77, 131 (1996), URL https://link.aps.org/doi/10.1103/PhysRevLett.77.131
14. D. E. Logan and M. T. Glossop, J. Phys. Condens. Matter 12, 985 (2000), URL https://doi.org/10.1088/0953-8984/2/F2/2/F320
15. M. Caffarel and W. Krauth, Phys. Rev. Lett. 72, 1545 (1994), URL https://link.aps.org/doi/10.1103/PhysRevLett.72.1545
16. H. R. Krishna-murthy, J. W. Wilkins, and K. G. Wilson,
T. Zhu, C. A. Jiménez-Hoyos, J. McClain, T. C. 18
Y. Nez Fernández and K. Hallberg, Front. Phys. 17
20
21
J. E. Hirsch and R. M. Fye, Phys. Rev. Lett. 22
J. E. Hirsch, Phys. Rev. B 28
P. Werner and A. J. Millis, Phys. Rev. B 24
V. Dobrosavljević, A. A. Pastor, and B. K. Nikolić, EPL 27
L. Wang, Phys. Rev. B 26
L. Huang and L. Wang, Phys. Rev. B 31
H. Terletska, Y. Zhang, L. Chioncel, D. Vollhardt, and C. E. Ekuma, H. Terletska, K.-M. Tam, N. S. Vidhyadhiraja, J. Moreno, and M. Jarrell, Phys. Rev. B 92, 201114 (2015), URL https://link.aps.org/doi/10.1103/PhysRevB.92.201114
M. Ulmke, V. Janiš, and D. Vollhardt, Phys. Rev. B 51, 10411 (1995), URL https://link.aps.org/doi/10.1103/PhysRevB.51.10411
D. Semmler, K. Byczuk, and W. Hofstetter, Phys. Rev. B 84, 115113 (2011), URL https://link.aps.org/doi/10.1103/PhysRevB.84.115113
K. Byczuk, W. Hofstetter, and D. Vollhardt, Phys. Rev. Lett. 94, 056404 (2005), URL https://link.aps.org/doi/10.1103/PhysRevLett.94.056404
I. de Vega, U. Schollwöck, and F. A. Wolf, Phys. Rev. B 92, 155126 (2015), URL https://link.aps.org/doi/10.1103/PhysRevB.92.155126
A. Liesbich and H. Ishiida, J. Phys. Soc. Jpn. 24, 053201 (2011), URL https://doi.org/10.1088%2F0953-8984%2F24%2F5%2F053201
D. Medvedeva, S. Iskakov, F. Krien, V. V. Mazurenko, and A. I. Lichtenstein, Phys. Rev. B 96, 235149 (2017), URL https://link.aps.org/doi/10.1103/PhysRevB.96.235149
Y. Nagai and H. Shinaoka, Journal of the Physical Society of Japan 88, 064004 (2019), URL https://doi.org/10.1088%2F0953-8984%2F88%2F6%2F064004
D. Sénéchal, Phys. Rev. B 81, 215125 (2010), URL https://link.aps.org/doi/10.1103/PhysRevB.81.215125
A. Weiße, G. Wellein, A. Alvermann, and H. Fehske, Rev. Mod. Phys. 78, 275 (2006), URL https://link.aps.org/doi/10.1103/RevModPhys.78.275
H. Q. Lin, Phys. Rev. B 42, 6561 (1990), URL https://link.aps.org/doi/10.1103/PhysRevB.42.6561
H. Lin and J. Gubernatis, Comput. Phys. 7, 400 (1993), URL https://aip.scitation.org/doi/abs/10.1063/1.4823192
F. A. Wolf, I. P. McCulloch, O. Parcollet, and U. Schollwöck, Phys. Rev. B 90, 115124 (2014), URL https://link.aps.org/doi/10.1103/PhysRevB.90.115124
R. Silver and H. Röder, Int. Mod. Phys. C 05, 735 (1994), URL https://link.aps.org/doi/10.1142/S0129183194000842
R. Silver, H. Röder, A. Voter, and J. Kress, J. Comput. Phys. 124, 115 (1996), ISSN 0021-9991, URL http://www.sciencedirect.com/science/article/pii/S0021999196900480
R. N. Silver and H. Röder, Phys. Rev. E 56, 4822 (1997), URL https://link.aps.org/doi/10.1103/PhysRevE.56.4822
A. Alvermann and H. Fehske, Phys. Rev. B 77, 045125 (2008), URL https://link.aps.org/doi/10.1103/PhysRevB.77.045125
F. A. Wolf, J. A. Justiniano, I. P. McCulloch, and U. Schollwöck, Phys. Rev. B 91, 115144 (2015), URL https://link.aps.org/doi/10.1103/PhysRevB.91.115144
L. Huang, Chin. Phys. B 25, 117101 (2016), URL https://iopscience.iop.org/article/10.1088/1674-1056/25/11/117101
L. Boehnke, H. Hafermann, M. Ferrero, F. Lechermann, and O. Parcollet, Phys. Rev. B 84, 075145 (2011), URL https://link.aps.org/doi/10.1103/PhysRevB.84.075145
FIG. 2: The coefficients of the Chebyshev polynomial expansion for 32 randomly chosen parameter sets for the finite cluster. Only the first 256 coefficients are shown, as the coefficients for the higher order terms are vanishingly small. Only the coefficients directly calculated from the kernel polynomial method (KPM) are shown here. The coefficients obtained from the neural network match very closely with the ones from the KPM and would not be visible by laying them on the same plot and thus they are omitted. We will demonstrate the quality of the coefficients in the Fig. 2. The magnitude of the coefficients for the last five coefficients are smaller than $10^{-5}$. 
FIG. 3: The spectral function, $A(\omega)$ is plotted for 32 randomly chosen parameter sets for the finite cluster. The figures correspond to the coefficient as shown in the Fig. 2. Both the results from the KPM and from the neural network are shown. They match each other very closely, and visually overlap on top of each others. A closer inspection reveals that there are slight oscillations in the spectral function when the weights are very small. This may due to the in-exact cancellations of different orders from the coefficients generated by the neural network method. In general these oscillations are rather small and only appear when the spectral weight drops to near zero.
FIG. 4: The input parameters of the 32 density of states plotted in Fig. 3. $i = 1$ corresponds to $U$, $i = 2$ corresponds to $\epsilon_f$, $i = 3, 4, 5$ correspond to $\epsilon_1, \epsilon_2, \epsilon_3$ and $i = 6, 7, 8$ correspond to $t_1, t_2, t_3$. 
FIG. 5: Comparisons of the first 32 coefficients as computed by the KPM and the neural network method. 1000 samples are plotted in each figure. The figures are ordered from left to right and top to bottom from order the 0-th to the order 31-th.