Complex Pole Representation and Its Applications to Quantum Many-Body Problems

by

Lei Zhang

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Physics and Scientific Computing)
in the University of Michigan
2026

Doctoral Committee:

Professor Emanuel Gull, Co-Chair
Professor Kai Sun, Co-Chair
Professor Xiaoming Mao
Professor Dominika Zgid
Professor Liuyan Zhao

Lei Zhang
lzphy@umich.edu
ORCID iD: 0000-0002-8828-4639

Acknowledgments

I am deeply indebted to all the people who have supported and helped me throughout my Ph.D. journey at Michigan.

First and foremost, I would like to thank my advisor, Prof. Emanuel Gull, for his guidance, support, and encouragement throughout my Ph.D. studies. I am deeply grateful for the freedom he gave me to explore my own research ideas, as well as for the insightful advice he provided whenever I needed direction. Whether I was facing technical difficulties, developing new ideas, or making important decisions about my research, he was always willing to discuss them with me and offer thoughtful suggestions. I also sincerely appreciate his financial support, which allowed me to focus on my research and complete this dissertation. His mentorship has played an essential role in my growth as a researcher, and I feel very fortunate to have had the opportunity to work with him.

I am also grateful to Prof. Kai Sun, Prof. Xiaoming Mao, Prof. Dominika Zgid, and Prof. Liuyan Zhao for generously serving on my dissertation committee. I sincerely appreciate the time and effort they devoted to reading my dissertation, attending my defense, and providing valuable feedback on my work.

Next, I am grateful to the members of our group: Dr. Sergei Iskakov, Dr. Wei-Ting Lin, Dr. Gaurav Harsha, Dr. Xinyang Dong, Dr. Chia-Nan Yeh, Dr. Runxue Yu, Dr. Thomas Blommel, and especially Yang Yu and Dr. André Erpenbeck, for creating a vibrant group environment and for many valuable discussions and collaborations. I also thank Prof. James P. F. LeBlanc and Daria Gazizova at Memorial University of Newfoundland for their valuable collaboration.

I am also deeply grateful to my undergraduate advisor, Prof. Youjin Deng, for first introducing me to scientific research and for the encouragement and guidance that had a lasting impact on my development as a physicist. His mentorship opened the door for me to pursue research, and I would not have reached this point without his support.

I would also like to thank my former officemates, Dr. Siddhartha Sarkar, Dr. Kai

Zhang, and especially Xiaohan Wan, for their support and for the many conversations we shared. I cannot remember how many times I asked Xiaohan questions about the graduation process and postdoctoral applications. She was always willing to help and patiently answered my questions, resolving many of my confusions during this important stage of my Ph.D. journey.

Next, I am grateful to the many friends who have been part of my life. I thank Fan Zou and Taigao Ma for carrying our friendship from our undergraduate years to the present, and Muyi Zhang, Kaiwen Hu, Weizhe Zhang, Yuan Zhu, and Tianke Zhuang for celebrating traditional Chinese festivals together and making life far from home feel warmer. My special thanks go to Dongyuan Zhou, Yilun Li, Xue Su, Sheng Huang, Bobo Bai, and Shuowei Jin for their companionship and support. Although we are now scattered across different places, the time we spent together during my first few years at Michigan remains among the most precious memories of my life. Finally, I am also grateful to Chenghao Li, Xiaoya Zhao, Wenxue Liu, Warsan Salah, Xinyue Wu, Yuxuan Chen, Yuto Hino, Fiona Pratt-MacDonald, Milad Hoseinpour, and Manatsanan (Palm) Pirunjinda from the conversation circle for the weekly conversations, which brought me companionship and encouragement during the final stage of my time at Michigan.

Last but not least, I would like to thank my family in China. I am deeply grateful to my parents for raising me and supporting me throughout my education. I am also glad that, during the final year of my Ph.D., we began to understand each other better. I am also grateful to my grandparents for their unwavering love. I especially remember my maternal grandfather for his strength and optimism. After my maternal grandmother, my uncle, and my aunt passed away a long time ago, he continued to live with resilience and care for those around him. Since I was a child, he always tried to save the best things for me. He loved cream cakes. Although I eventually learned how to make many kinds of cakes, he passed away before I had the chance to return to China and make one for him. Still, his strength and persistence will continue to influence me for the rest of my life.

Table of Contents

Acknowledgments ii

List of Figures viii

List of Tables xvi

List of Algorithms xvii

List of Appendices xviii

List of Acronyms xix

Abstract xxi

Chapter 1. Introduction 1

Chapter 2. Second Quantization 6
2.1. Quantum Mechanics of a Single Particle . . . 6
2.2. Quantum Mechanics of Many Identical Particles . . . 8
2.3. Second Quantization . . . 10
2.3.1. Field operators . . . 10
2.3.2. General basis states . . . 12
2.3.3. Hamiltonian . . . 16

Chapter 3. Single-Particle Green's Functions 18
3.1. Definition . . . 18
3.1.1. Imaginary-axis Green's functions . . . 18
3.1.2. Real-axis Green's functions . . . 19
3.2. Lehmann Representation . . . 20
3.3. Analytic Properties . . . 21
3.3.1. Complex Hermitian Hamiltonian . . . 21
3.3.2. Real symmetric Hamiltonian . . . 25

Chapter 4. Prony-Like Methods **26**
4.1. Introduction . . . 26
4.2. Prony Approximation Method . . . 27
4.3. ESPRIT Algorithm . . . 28
4.3.1. Scalar-valued ESPRIT . . . 28
4.3.2. Matrix-valued extension . . . 30

Chapter 5. Complex Pole Representation for Green's Functions **37**
5.1. $\boldsymbol{G}(z)$ on the Complex Plane . . . 37
5.1.1. Complex pole representation . . . 37
5.1.2. Analysis on the complex pole representation . . . 41
5.1.3. Matrix-valued extension . . . 43
5.2. Other Green's Functions . . . 44

Chapter 6. Minimal Pole Representation and Controlled Analytic Continuation **46**
6.1. Introduction . . . 46
6.2. Theory . . . 47
6.2.1. Approximation . . . 48
6.2.2. Holomorphic mapping . . . 49
6.2.3. Extraction . . . 50
6.3. Results . . . 51
6.3.1. Error control . . . 51
6.3.2. Dependence on temperature . . . 52
6.3.3. Dependence on data points . . . 53
6.3.4. Challenging examples . . . 54
6.3.5. Versatility . . . 56
6.3.6. Noisy data . . . 57
6.3.7. Imaginary-time data . . . 58
6.4. Conclusions . . . 59

Chapter 7. Matrix-Valued Extension and Systematic Improvements **60**
7.1. Introduction . . . 60
7.2. Sources of Ambiguity in NAC . . . 62
7.3. Minimal Pole Method . . . 64
7.3.1. Approximation on the Matsubara axis . . . 65
7.3.2. Holomorphic mapping . . . 67
7.3.3. Pole extraction . . . 69
7.3.4. Restriction to real poles . . . 71
7.3.5. Combination with the Discrete Lehmann Representation . . . 73
7.3.6. Incorporation of prior knowledge . . . 73

7.3.7. Application to heating and cooling of self-consistent many-body simulations . . . 74
7.4. Results . . . 74
7.4.1. Ambiguity issues revisited . . . 74
7.4.2. Matrix-valued continuations: fermion case . . . 79
7.4.3. Matrix-valued continuations: bosonic case . . . 80
7.4.4. Matrix-valued continuations: continuous system . . . 81
7.4.5. Recovering extreme spectra with prior knowledge . . . 83
7.4.6. Application to quantum Monte Carlo data . . . 85
7.4.7. Application to scalar and matrix-valued real-materials data . 86
7.5. Conclusions . . . 89

Chapter 8. Real-Frequency Bath Fitting **90**
8.1. Introduction . . . 90
8.2. Method . . . 91
8.2.1. Minimal Pole Method . . . 92
8.2.2. Applications . . . 95
8.3. Results . . . 99
8.3.1. Approximation of typical spectral functions . . . 99
8.3.2. Reconstructing the Green's function in the complex plane . . 101
8.3.3. Decomposing model spectral densities for real-time evolution . 103
8.3.4. Decomposing structured spectra for real-time evolution . . . 109
8.4. Conclusions . . . 112

Chapter 9. Quantum Impurity Solvers **113**
9.1. Introduction . . . 113
9.2. Imaginary-Axis Impurity Solvers . . . 115
9.2.1. Hamiltonian-Based Impurity Solver . . . 115
9.2.2. Real-Pole-Based Diagrammatic Impurity Solver . . . 118
9.3. Real-Axis Impurity Solvers . . . 119
9.3.1. Liouvillian-Based Impurity Solver for Steady States . . . 119
9.3.2. Complex-Pole-Based Diagrammatic Impurity Solver . . . 128
9.4. Conclusions . . . 133

Chapter 10. Mini-Pole Tutorial **135**
10.1. Installation and Setup . . . 135
10.1.1. Dependencies . . . 135
10.1.2. Installation Commands . . . 136
10.2. Usage . . . 136
10.2.1. `MiniPole` . . . 136
10.2.2. `MiniPoleDLR` . . . 144

10.2.3. `MiniPoleRf` 145
10.2.4. `MiniPoleRfDPR` 148
10.3. Summary 149

Chapter 11. Conclusions and Outlook **150**

Appendices **153**

Bibliography **173**

List of Figures

4.1. Two representative patterns of singular-value distributions at a relative noise level of 10^{-6}. (a) The exact function can be represented exactly as a sum of three exponentials. (b) The exact function cannot be represented exactly as a finite sum of exponentials. The optimal values of M are 3 for panel (a) and approximately 6 for panel (b). 29

5.1. Illustration of $G(z)$ on the complex plane. Green triangles and lines represent the simple poles and branch cuts in the exact representation, while red crosses indicate the locations of complex poles used to approximate $G(z)$ in the upper half plane. 38

5.2. Illustration of peaks generated by complex-valued A_l and ξ_l. The black dashed line represents an ordinary Lorentzian peak ($\mathrm{Im}\, A_l = 0$), while the red solid line represents the corresponding asymmetric peak ($\mathrm{Im}\, A_l \neq 0$). Here, $\mathrm{Re}\, \xi_l$ determines the peak center, and $\mathrm{Im}\, \xi_l$ controls the half-width. For fixed width, the peak height is controlled by $\mathrm{Re}\, A_l$, which scales the peak in the vertical direction, while the asymmetric component is controlled by $\mathrm{Im}\, A_l$. . . 39

6.1. Holomorphic functions $g(z)$ and $g^{-1}(w)$ mapping the complex plane to the unit disk and an interval on the imaginary axis to the unit circle. Also shown are points on and near the real axis as triangles, along with their image under g. 49

6.2. Integrated real axis error $err(A) = \int_{\mathbb{R}} d\omega |A - A_{\mathrm{cont}}|$ for the discrete (left) and continuous (right) case as a function of control parameter ε. Also indicated is the number of poles M. Inset: exact spectrum $A(\omega)$. Other parameters are $\beta = 200$, $n_0 = 30$ (left) and 0 (right), $\Delta n = 1$ and $N_\omega = 2001$. 51

6.3. Recovered spectral functions at different temperatures $\beta \in \{1, 2, 5, 10, 20, 50\}$ and various precision levels $\{10^{-3}, 10^{-6}, 10^{-9}, 10^{-12}\}$. Left panel: evolution of spectrum shapes. Right panel: convergence of $err(A)$ as a function of precision. Number of available data points is fixed at $N_\omega = 201$. 53

6.4. Convergence of spectrum moments as a function of precision for $\beta = 1$ and 2. 54
6.5. Recovered spectral functions at different numbers of data points $N_\omega \in \{7, 9, 11, 35, 151, 1001\}$ and various precision levels $\{10^{-3}$, 10^{-6}, 10^{-9}, $10^{-12}\}$. Left panel: evolution of spectrum shapes. Right panel: convergence of $err(A)$ as a function of N_ω at different precision levels. System temperature is fixed at $\beta = 30$. 55
6.6. Continuation of continuous spectral functions. From left to right: tight binding density of states of 2d square lattice with nearest- and next-nearest neighbor hopping. Semicircular density of states. Tight-binding density of states of the anisotropic triangular lattice. 'Kondo'-like spectral function. Shown are the exact input $A(\omega)$ in black, a continuation with Maximum Entropy (blue), SOM (purple), and a Prony fit (this method) in red. Maximum Entropy parameters fine-tuned to yield best spectra possible. . . 56
6.7. Analytic continuation of non-positive spectral functions in noiseless scenarios. Black: Exact input. Red: Continuation. Panel (a/b): off-diagonal continuous (a) and discrete (b) fermion case. (c) diagonal boson case. (d) discrete off-diagonal boson case. . . . 57
6.8. Spectral functions for different levels of relative Gaussian noise δ on the imaginary axis. Upper panel: discrete case. Lower panel: continuous case. Also indicated is the number of poles M. 58
6.9. Evolution of the recovered spectrum shapes from the imaginary-time data $G(\tau)$ at various precision levels $\{10^{-3}$, 10^{-6}, 10^{-9}, $10^{-12}\}$ for: (a) a semicircular spectrum; and (b) a multiple-featured spectrum. 59

7.1. Illustration of the ill-posed nature of NAC. Left panels: unrestricted least-squares fits. Middle panels: fits using a discrete system. Right panels: fits using a continuous system. The top panels compare the exact spectrum (blue shaded region) with the recovered spectrum (red). The bottom panels show the difference between the recovered and exact Green's functions on the imaginary axis, with Matsubara points marked by orange dots. Insets: comparison between the exact and recovered integrated spectral functions, $\mathrm{int}(\boldsymbol{A})(\omega) = \int_{-\infty}^{\omega} d\omega' \boldsymbol{A}(\omega')$. 63

7.2. Left: Complex plane with real and imaginary axis (black lines), Matsubara frequencies in the interval $[i\omega_{n_0}, i\omega_{n_\omega-1}]$ (orange dots), remaining frequencies (grey dots), and pole locations of a spectral function (green triangles). Right top: Complex plane after holomorphic mapping (7.9) transforming the interval onto the unit circle, mapping the remainder of the complex plane to the interior of the unit disk, infinity to zero, and the poles to the interior of the egg-shaped image of the real axis. Bottom right: Example of diagonal and off-diagonal spectral functions belonging to shared complex poles (green triangles in other panels). 67
7.3. Illustration of the holomorphic mapping (7.21). Left: The original plane, showing Matsubara frequencies in the interval $(-\infty, -i\omega_{n_0}] \cup [i\omega_{n_0}, +\infty)$ (orange dots), remaining frequencies (grey dots), and pole locations of a spectral function (green triangles). Top right: The complex plane after the holomorphic mapping (7.21), which transforms the interval onto the unit circle, maps the remainder of the complex plane inside the unit disk, and maps the real axis onto the line segment $(-1, 0)$ to $(1, 0)$, with the origin mapped to the origin and infinity mapped to $(\pm 1, 0)$. Bottom right: Examples of diagonal and off-diagonal spectral functions corresponding to the shared real poles. Delta peaks are broadened for better visualization. 72
7.4. Left panel: interpolation of an example Green's function at and between interpolation points, obtained from a least-squares fit to DLR coefficients. Right panel: same quantity obtained using the method of Sec. 7.3.1. 75
7.5. Maximum relative deviation ε_{rel} [Eq. (7.29)] on the imaginary axis, evaluated both at and between Matsubara points, among all random configurations. Results are shown for inverse temperatures $\beta \in \{1, 10, 100, 1000\}$, orbital numbers $n_{\text{orb}} \in \{1, 2, 5\}$, and noise levels $\delta \in \{10^{-3}, 10^{-6}, 10^{-9}, 10^{-12}\}$. Blue circles denote bosonic problems, and red triangles denote fermionic problems. . 76
7.6. Three scenarios in which the exact Matsubara data are indistinguishable above the precision level 10^{-4}. MPM is performed at noise levels $\delta \in \{10^{-3}, 10^{-5}, 10^{-7}, 10^{-9}\}$. The inverse temperature is fixed at $\beta = 100$. 78

7.7. Exact solution (light purple, large) and analytic continuation of diagonal (top panel) and off-diagonal (bottom panel) fermionic Matsubara Green's functions of a Hubbard dimer. Continuation is performed using both the mapping (7.9) (MPM, purple) and the mapping (7.21) (MPM-restricted, dashed deep purple). The simulations are conducted at $\beta = 10$. 79
7.8. Analytic continuation of bosonic Matsubara Green's functions obtained from the exact diagonalization of the same discrete Hubbard dimer system as in Fig. 7.7. 81
7.9. Analytic continuation of a continuous spectral function. Top panel: fermionic case; bottom panel: bosonic case. Blue circles represent the locations of the shared poles, with their weights indicated by the color bar. The simulations are carried out at $\beta = 20$. 82
7.10. Noise robustness analysis of the MPM algorithm, performed by adding Gaussian random noise with amplitudes $\delta \in \{10^{-2}, 10^{-4}, 10^{-6}\}$. (a) Diagonal and (b) off-diagonal fermionic case. (c) Diagonal and (d) off-diagonal bosonic case. 83
7.11. An extreme example of a spectral function that is sharp, non-Lorentzian, and far from the imaginary axis. (a) Continuation performed at 64-bit floating-point precision without introducing prior knowledge. The inset in (a) shows the difference between the corresponding Green's functions on the imaginary axis $[0, i\infty)$. (b) Continuation after incorporating prior knowledge of the Gaussian peak, performed at different noise levels $\delta \in \{10^{-6}, 10^{-8}, 10^{-10}, 10^{-12}\}$. The inset in (b) shows the convergence of the recovered mean value μ and standard deviation σ of the Gaussian peak as the noise level δ decreases. 84
7.12. MPM continuation of Quantum Monte Carlo (CT-HYB) data obtained from the single impurity Anderson model embedded in an infinite-dimensional Bethe lattice. The upper panels compare the recovered and exact spectra for different numbers of Monte Carlo samples, while the lower panels show the corresponding differences. Simulations are performed at $\beta = 20$. 85

7.13. Spectral function (band structure) of Si obtained by analytic continuation of self-consistent GW data [17, 95] at inverse temperature $\beta = 700$ Ha^{-1}. We use the same input Matsubara data for all plots and, for clearer visualization, evaluate the spectral function at $\eta = 0.005$ above the real axis, introducing an artificial broadening. Left panel: scalar continuation of the diagonal elements of the Green's function. Middle panel: matrix-valued continuation of the Green's function. Right panel: matrix-valued continuation of the self-energy followed by a Dyson equation on the real axis to obtain the real-frequency Green's function. Top set of plots: continuation with the method presented in Sec. 7.3. Middle set of plots: evaluation with DLR coefficients as presented in Sec. 7.3.5. Bottom panel: evaluation with Nevanlinna [16] and Carathéodory [17] methods. 86

8.1. Left: The original complex plane z with the spectral function on the real-frequency axis (purple) and pole locations (green triangles). Right: Representations using two different mappings (M1 and M2). Top right: The complex plane u after applying the holomorphic mapping (8.3), which transforms the entire real-frequency range $(-\infty, +\infty)$ onto the unit circle. This mapping places poles from the lower (upper) half-plane into the interior (exterior) of the unit circle. Bottom right: Similar to M1, but employing the holomorphic mapping (8.4). This transformation maps the finite real-frequency interval $[\omega_{\min}, \omega_{\max}]$ onto the unit circle, positioning all poles inside the unit circle. 92

8.2. Spectral recovery using MPM (M1) for different density of states: Kondo-like (top row), bosonic multi-orbital (middle row), and semicircular (bottom row). First column: Exact spectral function $A(\omega)$. Second column: Maximum (red pentagons) and mean (blue circles) of the deviation as a function of the number of complex poles M in the lower half-plane. Third column: Distributions of the absolute difference as a function of real frequency ω. Fourth column: Distribution of complex poles in the complex plane z for $M = 9$ (red circles) and $M = 19$ (blue triangles). 100

8.3. Same setup as Fig. 8.2, but now for the real part of the retarded Green's function and the Matsubara Green's function. First column: Exact real part of the retarded Green's function, corresponding to the Hilbert transform of $\pi A(\omega)$. Second column: Error of $\mathrm{Re}[G(\omega)]$ as a function of M. Third column: Distribution of the absolute difference $|\mathrm{Re}[\hat{G}(\omega)] - \mathrm{Re}G(\omega)|$ as a function of ω. Fourth column: Distribution of the absolute difference $|\hat{G}(iy) - G(iy)|$ on the positive imaginary axis. 103

8.4. Comparison of AAA (red), ESPRIT (blue), MPM (M1) (green), and MPM (M2) (orange) for an Ohmic bath at $T = 300\,\mathrm{K}$ ($\beta\omega_c \approx 0.240$). Left panel: (a) Exact effective spectral function $A(\omega)$ and (d) exact bath correlation function $C(\omega)$. Middle panel: Relative error of (b) $A(\omega)$ as defined in Eq. (8.34) and (e) $C(t)$ as defined in Eq. (8.35). All approximations use $M = 8$ terms. The vertical line separates the inside and outside of the approximated real-time interval. The relative difference $|\hat{C}^{(L_1)}(t) - \hat{C}^{(L_2)}(t)|/|\hat{C}^{(L_1)}(0)|$ is shown by the thick gray curve. Right panel: Mean error (Eqs. (8.36) and (8.37)) as a function of M, evaluated (c) within the interval $[0, t_c]$ and (f) over the long tail $[t_c, 10t_c]$. 106

8.5. Same setup as Fig. 8.4, except that temperature is now set to $T = 0.001\,\mathrm{K}$ ($\beta\omega_c \approx 7.19 \times 10^4$). Top panel: (a) Exact spectral function. (b) Exact bath correlation function. Bottom panel: Mean error for (c) the approximated interval and (d) the long-tail. 107

8.6. Comparison for a sub-Ohmic bath at $T = 50\,\mathrm{K}$ ($\beta\omega_c \approx 1.44$). Eight terms are used for all approximations. (a) Exact spectral function, which has a divergence at $\omega = 0$. (b) Exact bath correlation function, which exhibits an extremely slowly decaying tail for the real part. The insets show the performance of different methods in a zoomed-in region. Mean error for the approximated interval and the long tail are shown in subfigures (c) and (d), respectively. 108

8.7. Comparison of different approximation methods (solid lines) with the exact results (dashed lines) for a structured spectral function at $T = 300\,\mathrm{K}$. Results are obtained for $M = 100$. (a) Structured spectrum, (b) real part, and (c) imaginary part of the corresponding bath correlation function. The insets show the performance of different methods in magnified regions. 109

8.8. Mean error from Eqs. (8.36) and (8.37) as a function of the effective number of poles used in the computation of $C(t)$. Unfilled markers represent the original results that should not be used, while filled markers represent the results after correction. The inset shows the performance of ESPRIT (original) on a larger scale. 111

9.1. Simulated spectral function $A(\omega)$ obtained from the DMFT self-consistency loop. (a) Evolution with the number of iterations for $M = 6$. (b) Converged results for different numbers of poles $M \in \{1, 2, 3, 4, 5, 6\}$. 118
9.2. Convergence of the approximation to the effective greater spectral function $A^{>}(\omega)$. (a) Approximation with different numbers of poles M. (b) Convergence of the approximation error as a function of M. 126
9.3. Convergence of the spectral function $A(\omega)$ for $M = 3$ as a function of the number of non-Hermitian Lanczos eigenvalues k. 127
9.4. Simulated spectral function $A(\omega)$ for the Gaussian bath at $\beta = 10$ and $U = 1$, using different numbers of poles $M \in \{1, 2, 3\}$. 128
9.5. Convergence of the approximation to the input greater Green's function $G^{>}(\omega)$. (a) Approximations obtained using different numbers of poles M. (b) Convergence of the approximation error as a function of M. 131
9.6. (a) Simulated GF2 self-energy obtained using different numbers of poles M. (b) Convergence of the self-energy error as a function of M. 132
9.7. Simulated spectral function $A(\omega)$ obtained from the DMFT self-consistency loop. (a) Evolution with the number of iterations for $M = 4$. (b) Converged results for different numbers of poles $M \in \{2, 4, 6, 8, 10\}$. 133

10.1. Two typical patterns of singular-value distributions in the first ESPRIT step for (a) data with random noise at level 10^{-5} and (b) data without random noise but with systematic bias at level 10^{-6}, which may arise from loss of precision during iterative simulations. The gray lines indicate the recommended cutoffs. 140
10.2. (a) Singular-value distribution of the first ESPRIT step for an example with absolute noise level 10^{-4}, corresponding to relative noise level 10^{-5}. The dashed line shows the chosen cutoff, which can be obtained by any of the three commands introduced in the main text. (b) The recovered spectral function, which becomes closer to the exact result as the noise level decreases. 141

10.3. A series of plots generated by `p.check_valid()` for checking the validity of the intermediate steps. 143

List of Tables

A.1. Recovered $\{A_l, \xi_l\}$ for Fig. 6.2(a). 156
A.2. Recovered $\{A_l, \xi_l\}$ for Fig. 6.2(b). 157

List of Algorithms

1. Matrix-valued ESPRIT algorithm 32
2. Approximation Algorithm 66
3. Orthonormalization Algorithm 117
4. Non-interacting Bethe-lattice DMFT self-consistent loop 117
5. Bethe-lattice DMFT self-consistent loop 130
6. Non-Hermitian Lanczos Algorithm 170

List of Appendices

Appendix A. Implementation Details of the Scalar-Valued Minimal Pole Method **154**
A.1. Error Control 155
A.2. Dependence on Temperature 156
A.3. Dependence on Data Points 158
A.4. Challenging Examples 158
A.5. Versatility 161
A.6. Noisy Data 162

Appendix B. Implementation Details of the Liouvillian-Based Impurity Solver **163**
B.1. Fermionic Operators in Tensor Product Form 163
B.2. Super-Fermion Representation 164
B.2.1. Standard Formalism 164
B.2.2. Alternative Formalism 167
B.3. Explicit Form of the Liouvillian 168
B.4. Non-Hermitian Lanczos Algorithm 169
B.4.1. Theoretical Background 169
B.4.2. Green's Function Evaluation 171

List of Acronyms

AAA adaptive Antoulas–Anderson.

AMEA auxiliary master equation approach.

ARPES angle-resolved photoemission spectroscopy.

BCF bath correlation function.

CT-HYB continuous-time hybridization expansion.

DLR discrete Lehmann representation.

DMFT dynamical mean-field theory.

ED exact diagonalization.

ESPRIT Estimation of Signal Parameters via Rotational Invariance Techniques.

GF2 second-order Green's function perturbation theory.

HEOM hierarchical equations of motion.

IR intermediate representation.

MaxEnt maximum entropy.

MPM Minimal Pole Method.

NAC numerical analytic continuation.

QCD quantum chromodynamics.

QFT quantum field theory.

QMC quantum Monte Carlo.

SAC stochastic analytic continuation.

SIAM single-impurity Anderson model.

SOM stochastic optimization method.

SVD singular value decomposition.

TRIQS Toolbox for Research on Interacting Quantum Systems.

Abstract

This dissertation develops the complex pole representation as a unified framework for compactly representing Green's functions in quantum many-body simulations, with applications to numerical analytic continuation, hybridization fitting, compact representation and quantum impurity solvers, accompanied with software implementation. The central idea is to replace continuous spectral and correlation functions by a small number of complex poles and complex weights, while preserving the analytic structure needed for physical interpretation and numerical simulation.

We begin by reviewing the second-quantization formalism and the single-particle Green's functions used throughout this dissertation. Particular emphasis is placed on the Lehmann representation, which shows that finite-size Green's functions can be expressed as sums over real poles in the frequency domain. In extended or continuous systems, these real poles become dense and give rise to broadened spectral features and branch cuts, making a direct real-pole representation inefficient. This observation motivates the complex pole representation, in which continuous spectral features are approximated by poles away from the real axis. We then introduce Prony-like methods, including ESPRIT, as numerical tools for extracting minimal exponential and pole representations from sampled data. Combined with suitable holomorphic mappings, these methods provide the algorithmic foundation for recovering complex pole representations of Green's functions.

We next develop the complex pole representation for Green's functions in detail. We explain its motivation from the analytic structure of Green's functions, derive its basic form on the complex plane, and analyze its implications for causality, positivity, physical constraints, and singular spectral features. We also show that not only retarded and advanced Green's functions, but also greater, lesser, time-ordered, anti-time-ordered, and Keldysh Green's functions can be represented compactly by complex poles. This analysis provides the conceptual foundation for the main applications developed in the

following chapters.

The first main application is numerical analytic continuation of Matsubara data to the real-frequency axis. Analytic continuation is notoriously ill-conditioned, as small uncertainties on the imaginary axis can be strongly amplified on the real axis. We show that Matsubara Green's functions can be approximated by a minimal complex pole representation to within a prescribed precision, and that the resulting real-frequency spectral function converges systematically to the exact result as the input precision improves. This provides a controlled route to analytic continuation for a broad class of fermionic and bosonic response functions, including diagonal, off-diagonal, discrete, and continuous cases.

The second application extends the method to matrix-valued correlation functions and introduces several systematic improvements. In multi-orbital quantum simulations, correlation functions are matrix-valued objects whose elements share common analytic structure. Instead of continuing each matrix element independently, we construct compact pole representations with shared pole locations and matrix-valued weights. We further develop pole restrictions, constraints on the spectral shape, a combination with the discrete Lehmann representation, the incorporation of prior knowledge, and an automatic procedure for selecting the initial Matsubara point. These extensions broaden the applicability of the method while preserving the systematically improvable character of the original scalar formulation.

The third application concerns compact representations of real-frequency and real-time correlation functions. We generalize the complex pole approach to real-axis fitting and demonstrate its application to spectral densities and bath correlation functions. The resulting representation achieves accuracy comparable to state-of-the-art methods on short-time intervals, while providing a more controlled description of long-time tail behavior. This makes the method useful for real-time simulations, quantum impurity models, and open quantum systems, where continuous baths must be represented by a finite number of degrees of freedom.

The fourth application uses the complex pole representation to construct quantum impurity solvers. On the imaginary axis, we show that complex poles can be used to construct non-Hermitian Hamiltonians for non-interacting problems, while the inclusion of interactions requires additional restrictions that lead back to real-pole-based formulations. We also demonstrate the difficulties that arise when complex poles are used

directly in imaginary-axis diagrammatic calculations. On the real axis, by contrast, the complex pole representation leads to compact and efficient impurity solvers. In particular, we construct a Liouvillian-based solver for nonequilibrium steady states and a complex-pole-based diagrammatic solver for real-axis perturbation theory. These examples show that complex poles can be used not only for pre- and post-processing, but also as active computational degrees of freedom in quantum impurity simulations.

Finally, we present Mini-Pole, an open-source Python implementation of the methods developed in this dissertation. The package provides tools for analytic continuation, compact representation, hybridization fitting, and pole compression. This software component makes the complex pole framework accessible for practical use and provides a reproducible implementation of the algorithms introduced throughout the dissertation.

Taken together, these results show that the complex pole representation provides both a practical numerical method and a conceptual framework for treating Green's functions in quantum many-body theory. By connecting analytic continuation, hybridization fitting, compact representation, impurity solvers, and software implementation within a single pole-based perspective, this dissertation provides a systematically improvable approach to extracting, representing, and directly manipulating real-frequency information in quantum many-body simulations.

Chapter 1

Introduction

Understanding quantum many-body systems is one of the central goals of modern condensed matter physics, quantum chemistry, and materials science [1–3]. In these systems, interactions among many particles generate collective phenomena that cannot be captured by single-particle physics alone, such as magnetism, superconductivity, metal-insulator transitions, and strongly correlated electronic behavior. A direct description in terms of the full many-body wavefunction is often impractical, because the dimension of the many-body Hilbert space grows exponentially with system size [4]. It is therefore useful to formulate the problem in terms of reduced quantities that encode the physical information most relevant to excitations, response, and emergent behavior [1–3]. Among these reduced quantities, Green's functions play a particularly important role [5, 6]. They provide a compact language for describing single-particle excitations, response functions, correlation effects, and connections to experimentally measurable quantities.

In equilibrium finite-temperature simulations, Green's functions are often computed on the imaginary-time or imaginary-frequency axis [3], where the numerical problem is usually better conditioned and many powerful methods, including perturbative calculations [7–9] and Monte Carlo simulations [10–12], can be formulated. However, physical spectra and dynamical response functions are naturally defined on the real-frequency axis. This mismatch between the numerically convenient imaginary axis and the physically interpretable real axis is one of the central difficulties addressed in this dissertation.

The transformation from imaginary-axis data to real-frequency information is known as numerical analytic continuation (NAC). Although the underlying Green's function is

analytic away from the real axis, recovering its boundary value on the real axis from discrete and possibly noisy imaginary-axis data is an ill-conditioned problem [13]. Small uncertainties in the input data are amplified exponentially in the recovered spectral function. For this reason, analytic continuation has long been regarded as one of the most delicate post-processing steps in quantum many-body simulations. Many regularization strategies have been developed, including interpolation with Padé [14, 15] and Nevanlinna functions [16, 17], the Maximum Entropy (MaxEnt) method [13, 18–23], stochastic analytic continuation (SAC) and variants [24–31], genetic algorithms and machine learning [32–34], causal projections [35] and Prony fits [36, 37]. These methods have important advantages and have been widely used. Nevertheless, it remains difficult in practice to obtain a real-frequency spectrum that improves systematically as the precision of the imaginary-axis data increases.

We approach this problem using the complex pole representation developed in our work [38]. The key observation is that continuous spectral features can often be accurately approximated by a small number of poles away from the real axis, provided that both the pole locations and weights are allowed to be complex-valued. Instead of asking only how to continue a function from the imaginary axis to the real axis, we ask what minimal information, namely the minimal number of complex poles, is contained in the input data. This perspective leads naturally to the name of our method: the Minimal Pole Method (MPM). By reducing the problem to this minimal representation, we show that all ambiguities associated with analytic continuation are eliminated. Most importantly, unlike any other procedure known to us, MPM systematically improves the precision of the recovered spectral function as the precision of the input field-theory data is increased. This capability paves the way toward reliable uncertainty estimates of excitations, which are difficult to obtain with existing methodologies. The method is applicable not only to positive diagonal fermionic spectral functions, but also to off-diagonal, bosonic, discrete, and continuous response functions. It is also robust to noise: when the fit is performed to the known precision of the input data, the resulting spectra remain physically meaningful and improve as the uncertainty on the Matsubara axis is reduced. Compared with Ref. [38], we also include a procedure for applying the method to imaginary-time data, although the method is primarily designed for imaginary-frequency data.

In realistic many-body calculations, Green's functions, self-energies, and susceptibili-

ties are often matrices in orbital, spin, momentum, or band indices. Treating each matrix element independently may ignore important shared analytic structure and lead to inconsistent continuations. Based on this observation, we generalize MPM to a matrix-valued formulation, in which different matrix elements share the same set of complex poles while carrying distinct matrix-valued weights [39]. This structure provides a coherent representation of multi-orbital Green's functions and self-energies, enables the continuation of off-diagonal elements, and facilitates applications to real-materials data. We also introduce several systematic improvements, including restrictions to real poles when appropriate, constraints on the spectral shape, a combination with the discrete Lehmann representation, and an automatic procedure for selecting the initial Matsubara point. These developments broaden the applicability of the method while preserving its systematically improvable character. Compared with Ref. [39], this thesis also includes sections explaining the sources of ambiguity in NAC and the procedure for incorporating prior knowledge, accompanied by numerical experiments. We also add numerical results for quantum Monte Carlo data and provide a detailed explanation of why the performance of analytic continuation for real-materials data differs when it is applied to scalar-valued Green's functions, matrix-valued Green's functions, and matrix-valued self-energies, as well as a discussion of the different performance of the original method and its variant.

The same complex-pole perspective can also be transferred from the Matsubara axis to the real-frequency axis. Motivated by the success of MPM for NAC, we generalize the method to real-frequency bath fitting and compact representation [40]. Quantum impurity models, open quantum systems, and real-time simulations often require approximating a continuous bath by a finite set of discrete modes or exponentials. Traditional fitting procedures are frequently nonlinear, computationally expensive, and sensitive to local minima. The complex pole representation provides an alternative route: spectral densities, real-frequency Green's functions, and bath correlation functions can be approximated by a compact set of complex poles to a prescribed accuracy. The resulting representation achieves accuracy comparable to state-of-the-art methods on short-time intervals, while providing a more controlled description of long-time tail behavior. This makes the method useful for real-time simulations, quantum impurity models, and open quantum systems, where continuous baths must be represented by a finite number of degrees of freedom.

Beyond analytic continuation and bath fitting, the complex pole representation can also be used as an active computational ingredient in quantum impurity solvers. In this dissertation, we explore this idea in both imaginary-axis and real-axis formulations. On the imaginary axis, complex poles can be used to construct non-Hermitian Hamiltonians for non-interacting problems, but the introduction of interactions reveals a fundamental limitation: since the complex pole representation is valid only in one half of the complex plane, interacting Hamiltonian-based and diagrammatic calculations generally require additional restrictions that lead back to real-pole-based formulations. On the real axis, by contrast, the complex pole representation is directly compatible with the analytic structure of retarded, advanced, greater, lesser, time-ordered, anti-time-ordered, and Keldysh Green's functions. This enables compact real-axis impurity solvers, including a Liouvillian-based solver for nonequilibrium steady states and a complex-pole-based diagrammatic solver for real-axis perturbation theory. These examples show that complex poles can be used not only for pre- and post-processing, but also as computational degrees of freedom in many-body simulations.

To make these methods accessible and reproducible, this dissertation is accompanied by Mini-Pole, an open-source Python implementation of the main algorithms developed here. The package provides tools for analytic continuation, compact representation, hybridization fitting, and pole compression. It supports scalar-valued and matrix-valued correlation functions and includes several variants corresponding to the methods introduced throughout the dissertation. In this way, the software implementation serves not only as a practical tool, but also as a concrete realization of the complex pole framework.

The remainder of this dissertation is organized as follows. Ch. 2 introduces the second-quantization formalism used throughout the thesis. Starting from the quantum mechanics of identical particles, it introduces creation and annihilation operators, field operators, and the second-quantized Hamiltonian. Ch. 3 introduces single-particle Green's functions on both the imaginary and real axes. Particular emphasis is placed on the Lehmann representation and the analytic properties that follow from it, including real-pole representations, sum rules, and symmetry properties. Ch. 4 introduces Prony-like methods, including the Prony approximation method, the ESPRIT algorithm and its matrix-valued extension, which provide the numerical foundation for extracting minimal exponential and pole representations from sampled data.

Ch. 5 develops the complex pole representation for Green's functions. It explains

how complex-valued poles and weights can approximate broadened spectral functions, analyzes the implications for causality, positivity, physical constraints, and singular spectral features, and discusses the matrix-valued extension. Ch. 6 applies this representation to numerical analytic continuation and demonstrates controlled convergence of real-frequency spectra as the Matsubara input precision increases. Ch. 7 extends the method to matrix-valued correlation functions and introduces systematic improvements, including pole restrictions, spectral-shape constraints, combinations with the discrete Lehmann representation, the incorporation of prior knowledge, and automatic selection of the initial Matsubara point. Ch. 8 applies the same ideas to real-frequency bath fitting and shows how complex pole representations can approximate spectral functions and bath correlation functions with controlled accuracy, including their long-time tails. Ch. 9 explores the use of the complex pole representation in quantum impurity solvers. It discusses both imaginary-axis and real-axis formulations, analyzes the limitations of direct complex-pole-based imaginary-axis simulations, and presents real-axis Liouvillian-based and diagrammatic impurity solvers. Ch. 10 provides a tutorial for the Mini-Pole package, including installation, basic usage, and the main classes used for analytic continuation, compact representation, bath fitting, and pole compression. Finally, Ch. 11 summarizes the main conclusions and discusses possible future directions.

During the preparation of this dissertation, ChatGPT was used solely to assist with language polishing and grammar correction. All suggestions from this tool were independently reviewed and verified by the author before being incorporated. No scientific content, research ideas, analyses, results, figures, references, or conclusions were generated by this tool.

Chapter 2

Second Quantization

This chapter gives an introduction to second quantization in quantum field theory (QFT), which will be used throughout the rest of the thesis. Throughout this thesis, we use bold font to denote tensors, e.g., vectors and matrices, and use a hat "$\hat{}$" to denote operators, unless otherwise specified. This chapter largely follows the conventions used in Ref. [6]. We set $\hbar = -e = m_e = 1$, where e is the electron charge and m_e is its corresponding mass.

2.1. Quantum Mechanics of a Single Particle

For a single particle, its physical state can be described by a ket $|\Psi\rangle$ belonging to a *Hilbert space*, a vector space endowed with an inner product, whose evolution satisfies the *Schrödinger equation*

$$i\partial_t|\Psi\rangle = \hat{h}|\Psi\rangle \ , \tag{2.1}$$

where $\hat{h} = \hat{h}(\hat{\boldsymbol{r}}, \hat{\boldsymbol{p}}, \hat{\boldsymbol{\mathcal{S}}})$ is the one-body Hamiltonian operator, which in general depends on the position operator $\hat{\boldsymbol{r}}$, the momentum operator $\hat{\boldsymbol{p}}$, and the spin operator $\hat{\boldsymbol{\mathcal{S}}}$. For example, for a particle of mass m, charge q, and gyromagnetic ratio g moving in an external field, $\hat{h}$ has the form

$$\hat{h} = \frac{1}{2m}\left(\hat{\boldsymbol{p}} - \frac{q}{c}\boldsymbol{A}(\hat{\boldsymbol{r}})\right)^2 + q\phi(\hat{\boldsymbol{r}}) - g\mu_B \boldsymbol{B}(\hat{\boldsymbol{r}}) \cdot \hat{\boldsymbol{\mathcal{S}}} \ , \tag{2.2}$$

where ϕ is the scalar potential, $\boldsymbol{A}$ is the vector potential, $\boldsymbol{B} = \boldsymbol{\nabla} \times \boldsymbol{A}$ is the magnetic field, and μ_B is the Bohr magneton. Once the Hamiltonian and the initial ket at t_0 are

given, the dynamics of $|\Psi\rangle$ at later times $t > t_0$ is fully determined.

To solve Eq. (2.1) explicitly, we can utilize the so-called *position-spin kets* $|\boldsymbol{x}\rangle$, where $\boldsymbol{x} = (\boldsymbol{r}\sigma)$, and we have assumed that the particle has spin whose projection is denoted by σ. They are the eigenstates of $\hat{\boldsymbol{r}}$ and $\hat{\boldsymbol{\mathcal{S}}}_z$,

$$\hat{\boldsymbol{r}}|\boldsymbol{x}\rangle = \boldsymbol{r}|\boldsymbol{x}\rangle \ , \quad \hat{\boldsymbol{\mathcal{S}}}_z|\boldsymbol{x}\rangle = \sigma|\boldsymbol{x}\rangle \ , \tag{2.3}$$

are orthonormal,

$$\langle \boldsymbol{x}'|\boldsymbol{x}\rangle = \delta_{\sigma\sigma'}\delta(\boldsymbol{r}-\boldsymbol{r}') := \delta(\boldsymbol{x}-\boldsymbol{x}') \ , \tag{2.4}$$

and are complete,

$$\int d\boldsymbol{x}\, |\boldsymbol{x}\rangle\langle\boldsymbol{x}| = \hat{\mathbb{1}} \ , \quad \text{with } \int d\boldsymbol{x} := \sum_\sigma \int d\boldsymbol{r} \ . \tag{2.5}$$

The ket $|\Psi\rangle$ can then be expressed as

$$|\Psi\rangle = \int d\boldsymbol{x}\Psi(\boldsymbol{x})|\boldsymbol{x}\rangle \ , \tag{2.6}$$

where $\Psi(\boldsymbol{x}) := \langle\boldsymbol{x}|\Psi\rangle$ is the so-called *wavefunction* or *probability amplitude*. The Hamiltonian can be written as

$$\langle\boldsymbol{x}|\hat{h}|\boldsymbol{x}'\rangle = h_{\sigma\sigma'}(\boldsymbol{r}, -i\boldsymbol{\nabla}, \boldsymbol{S})\delta(\boldsymbol{r}-\boldsymbol{r}') \ , \tag{2.7}$$

and, for example, Eq. (2.2) has the explicit form

$$h_{\sigma\sigma'}(\boldsymbol{r}, -i\boldsymbol{\nabla}, \mathbf{S}) = \frac{\delta_{\sigma\sigma'}}{2m}\Big(-i\boldsymbol{\nabla} - \frac{q}{c}\boldsymbol{A}(\boldsymbol{r})\Big)^2 + \delta_{\sigma\sigma'}q\phi(\boldsymbol{r}) - g\mu_B\boldsymbol{B}(\boldsymbol{r})\cdot\boldsymbol{S}_{\sigma\sigma'} \ , \tag{2.8}$$

where the relation $\langle\boldsymbol{x}|\hat{\boldsymbol{p}}|\boldsymbol{x}'\rangle = -i\delta_{\sigma\sigma'}\boldsymbol{\nabla}\delta(\boldsymbol{r}-\boldsymbol{r}')$ has been utilized, and $\boldsymbol{S}$ is the matrix of the spin operator with elements $\boldsymbol{S}_{\sigma\sigma'} = \langle\sigma|\hat{\boldsymbol{\mathcal{S}}}|\sigma'\rangle$. For spin-$\frac{1}{2}$ fermions, the operator $\hat{\boldsymbol{\mathcal{S}}} = (\hat{\boldsymbol{\mathcal{S}}}^x, \hat{\boldsymbol{\mathcal{S}}}^y, \hat{\boldsymbol{\mathcal{S}}}^z)$ can be written explicitly as

$$\hat{\boldsymbol{\mathcal{S}}}^j = \frac{1}{2}\sum_{\sigma\sigma'}\hat{d}^\dagger_\sigma\boldsymbol{\sigma}^j_{\sigma\sigma'}\hat{d}_{\sigma'} \ , \quad j = x, y, z, \tag{2.9}$$

where $\hat{d}^{(\dagger)}_\sigma$ are the creation/annihilation operators of spin σ that will be introduced in

Sec. 2.3.2. Therefore, we have $\boldsymbol{S} = \frac{1}{2}\boldsymbol{\sigma}$, with $\boldsymbol{\sigma} = (\boldsymbol{\sigma}^x, \boldsymbol{\sigma}^y, \boldsymbol{\sigma}^z)$ being the *Pauli matrices* with components

$$\boldsymbol{\sigma}^x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} , \quad \boldsymbol{\sigma}^y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \quad \text{and} \quad \boldsymbol{\sigma}^z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} . \tag{2.10}$$

Finally, by substituting Eqs. (2.6) and (2.7) into Eq. (2.1), the Schrödinger equation can be explicitly expressed as

$$i\partial_t \Psi(\boldsymbol{x}) = \sum_{\sigma'} h_{\sigma\sigma'}(\boldsymbol{r}, -i\boldsymbol{\nabla}, \mathbf{S})\Psi(\boldsymbol{x}) . \tag{2.11}$$

2.2. Quantum Mechanics of Many Identical Particles

For many identical particles, since it is impossible to distinguish which particle is at which location, the position-spin kets are not always independent of each other. For example, in the two-particle case, the normalized ket $|\boldsymbol{x}\boldsymbol{x}'\rangle$ describes the same physical state as $|\boldsymbol{x}'\boldsymbol{x}\rangle$ and therefore can differ only by a phase factor,

$$|\boldsymbol{x}'\boldsymbol{x}\rangle = e^{i\alpha}|\boldsymbol{x}\boldsymbol{x}'\rangle . \tag{2.12}$$

Applying this relation twice leads to $e^{i\alpha} = \pm 1$, where particles corresponding to $e^{i\alpha} = +1$ are called *bosons*, and those corresponding to $e^{i\alpha} = -1$ are called *fermions*. In the following, we will always use upper and lower signs to refer to bosons and fermions, respectively.

Generalizing to the N-particle case, one can verify that the position-spin ket $|\boldsymbol{x}_1 \cdots \boldsymbol{x}_N\rangle$ satisfies

$$|\boldsymbol{x}_{P(1)} \cdots \boldsymbol{x}_{P(N)}\rangle = (\pm)^P |\boldsymbol{x}_1 \cdots \boldsymbol{x}_N\rangle , \tag{2.13}$$

where P is a permutation of $(1, \cdots, N)$ and $(\pm)^P$ is equal to 1 and ± 1 for even and odd

permutations, respectively. The inner product of two kets is given by

$$\begin{aligned}\langle \boldsymbol{x}'_1 \cdots \boldsymbol{x}'_N | \boldsymbol{x}_1 \cdots \boldsymbol{x}_N \rangle &= \sum_P (\pm)^P \prod_{j=1}^N \delta\big(\boldsymbol{x}'_j - \boldsymbol{x}_{P(j)}\big) \\ &= \begin{vmatrix} \delta(\boldsymbol{x}'_1 - \boldsymbol{x}_1) & \cdots & \delta(\boldsymbol{x}'_1 - \boldsymbol{x}_N) \\ \vdots & \ddots & \vdots \\ \delta(\boldsymbol{x}'_N - \boldsymbol{x}_1) & \cdots & \delta(\boldsymbol{x}'_N - \boldsymbol{x}_N) \end{vmatrix}_{\pm} , \end{aligned} \tag{2.14}$$

and the completeness relation reads

$$\frac{1}{N!} \int d\boldsymbol{x}_1 \cdots d\boldsymbol{x}_N \, |\boldsymbol{x}_1 \cdots \boldsymbol{x}_N\rangle\langle \boldsymbol{x}_1 \cdots \boldsymbol{x}_N| = \hat{\mathbb{1}} \ , \tag{2.15}$$

where the normalization factor comes from the fact that, given all distinct coordinates, there are $N!$ equivalent configurations leading to the same physical state.

The ket $|\Psi\rangle$ can be expanded in this basis as

$$|\Psi\rangle = \frac{1}{N!} \int d\boldsymbol{x}_1 \cdots d\boldsymbol{x}_N |\boldsymbol{x}_1 \cdots \boldsymbol{x}_N\rangle \Psi(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N) \ , \tag{2.16}$$

where the wavefunction $\Psi(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N) = \langle \boldsymbol{x}_1 \cdots \boldsymbol{x}_N | \Psi \rangle$ is totally symmetric and anti-symmetric for bosons and fermions, respectively. The normalization condition reads

$$1 = \langle \Psi | \Psi \rangle = \frac{1}{N!} \int d\boldsymbol{x}_1 \cdots d\boldsymbol{x}_N |\Psi(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N)|^2 . \tag{2.17}$$

In practice, one can construct the N-particle kets by (anti)symmetrizing products of one-particle kets,

$$|\boldsymbol{x}_1 \cdots \boldsymbol{x}_N\rangle = \frac{1}{\sqrt{N!}} \sum_P (\pm)^P |\boldsymbol{x}_{P(1)}\rangle \cdots |\boldsymbol{x}_{P(N)}\rangle \ , \tag{2.18}$$

which is referred to as *first quantization*. For the N-particle Hamiltonian, its non-

interacting part can be expressed as

$$\begin{aligned}\hat{H}_0 &= \sum_{j=1}^{N} \hat{h}(\hat{\boldsymbol{r}}_j, \hat{\boldsymbol{p}}_j, \hat{\boldsymbol{\mathcal{S}}}_j) \\ &= \hat{h}(\hat{\boldsymbol{r}}_1, \hat{\boldsymbol{p}}_1, \hat{\boldsymbol{\mathcal{S}}}_1) \otimes \hat{\mathbb{1}} \otimes \cdots \otimes \hat{\mathbb{1}} + \cdots + \hat{\mathbb{1}} \otimes \cdots \otimes \hat{\mathbb{1}} \otimes \hat{h}(\hat{\boldsymbol{r}}_N, \hat{\boldsymbol{p}}_N, \hat{\boldsymbol{\mathcal{S}}}_N) \ , \end{aligned} \tag{2.19}$$

where $\otimes$ stands for the tensor product of operators acting on different particles. Similarly, the interacting part can be written as

$$\hat{H}_{\text{int}} = \frac{1}{2} \sum_{i \neq j}^{N} v(\hat{\boldsymbol{r}}_i, \hat{\boldsymbol{r}}_j) \ , \tag{2.20}$$

each term of which acts only on a two-particle subspace and leaves the rest unchanged. As a result, one can proceed by performing calculations with respect to each term in Eq. (2.18), which results in cumbersome expressions with a large number of terms differing only in the sign and the order of the coordinates. Another disadvantage is that expressions here depend explicitly on the number of particles N, which is unfavorable for dealing with systems with particle-number fluctuations. As we will show in the next section, these issues can be resolved in the *second quantization* framework.

2.3. Second Quantization

2.3.1. Field operators

To deal with cases of arbitrarily many identical particles, we define a collection $\mathbb{F}$ of Hilbert spaces, known as *Fock space*, according to

$$\mathbb{F} = \{\mathbb{H}_0, \mathbb{H}_1, \cdots, \mathbb{H}_N, \cdots\}, \tag{2.21}$$

with $\mathbb{H}_N$ being the Hilbert space of N identical particles. Since $\mathbb{H}_0$ is the space with zero particles and has no degrees of freedom, there exists only one physical state, which can be described by a normalized ket $|0\rangle$.

For an arbitrary N-particle position-spin ket $|\boldsymbol{x}_1 \cdots \boldsymbol{x}_N\rangle$, it can be generated by re-

peatedly applying a so-called *field operator* $\hat{\psi}^\dagger(\boldsymbol{x})$ to $|0\rangle$,

$$|\boldsymbol{x}_1 \cdots \boldsymbol{x}_N\rangle = \hat{\psi}^\dagger(\boldsymbol{x}_N)|\boldsymbol{x}_1 \cdots \boldsymbol{x}_{N-1}\rangle = \hat{\psi}^\dagger(\boldsymbol{x}_N) \cdots \hat{\psi}^\dagger(\boldsymbol{x}_1)|0\rangle \ . \tag{2.22}$$

Defining the (anti)commutator between two operators $\hat{A}$ and $\hat{B}$ as $[\hat{A}, \hat{B}]_\mp = \hat{A}\hat{B} \mp \hat{B}\hat{A}$, the particle exchange relation (2.13) requires that

$$[\hat{\psi}^\dagger(\boldsymbol{x}), \hat{\psi}^\dagger(\boldsymbol{y})]_\mp = 0 \ . \tag{2.23}$$

After defining the adjoint operator according to

$$\langle \Psi' | \hat{\psi}(\boldsymbol{x}) | \Psi \rangle = (\langle \Psi | \hat{\psi}^\dagger(\boldsymbol{x}) | \Psi' \rangle)^*, \tag{2.24}$$

one can immediately obtain the (anti)commutation relation of ψ's,

$$[\hat{\psi}(\boldsymbol{x}), \hat{\psi}(\boldsymbol{y})]_\mp = 0 \ . \tag{2.25}$$

Its action on an N-particle position-spin ket $|\boldsymbol{y}_1 \cdots \boldsymbol{y}_N\rangle$ can be expressed as

$$\hat{\psi}(\boldsymbol{x})|\boldsymbol{y}_1 \cdots \boldsymbol{y}_N\rangle = \sum_{k=1}^{N} (\pm)^{N+k} \delta(\boldsymbol{x} - \boldsymbol{y}_k)|\boldsymbol{y}_1 \cdots \boldsymbol{y}_{k-1}\boldsymbol{y}_{k+1} \cdots \boldsymbol{y}_N\rangle, \tag{2.26}$$

which removes a particle from each coordinate successively while keeping the final result totally (anti)symmetric by adjusting the signs of the prefactors. Finally, it can be proved that the annihilation and creation operators satisfy the (anti)commutation rule

$$[\hat{\psi}(\boldsymbol{x}), \hat{\psi}^\dagger(\boldsymbol{y})]_\mp = \delta(\boldsymbol{x} - \boldsymbol{y}) \ . \tag{2.27}$$

Although the field operators are completely defined by Eqs. (2.22) and (2.26), the same information is also fully contained in the (anti)commutation relations (2.23), (2.25), and (2.27), which can be more convenient in practice since one can directly manipulate the kets using these relations, and the field operators automatically take care of the symmetry of the kets.

2.3.2. General basis states

2.3.2.1. Orthogonal basis

For a generic orthonormal one-particle ket $|n\rangle$, where the quantum number $n = (s\tau)$ consists of a generic orbital quantum number s and a spin projection τ along the same axis as that of $|\boldsymbol{x}\rangle = |\boldsymbol{r}\sigma\rangle$, the overlap between $|n\rangle$ and $|\boldsymbol{x}\rangle$ is

$$\varphi_n(\boldsymbol{x}) := \langle \boldsymbol{x}|n\rangle = \varphi_s(\boldsymbol{r})\delta_{\tau\sigma} \ , \tag{2.28}$$

with the orthonormality condition $\int d\boldsymbol{x}\, \varphi_n^*(\boldsymbol{x})\varphi_m(\boldsymbol{x}) = \delta_{nm}$. By defining the creation/annihilation operators associated with n as

$$\hat{d}_n^\dagger := \int d\boldsymbol{x}\varphi_n(\boldsymbol{x})\hat{\psi}^\dagger(\boldsymbol{x}) \ , \tag{2.29}$$

$$\hat{d}_n := \int d\boldsymbol{x}\varphi_n^*(\boldsymbol{x})\hat{\psi}(\boldsymbol{x}) \ , \tag{2.30}$$

an arbitrary N-particle ket $|n_1 \cdots n_N\rangle$ can be generated by

$$|n_1 \cdots n_N\rangle = \hat{d}_{n_N}^\dagger |n_1 \cdots n_{N-1}\rangle = \hat{d}_{n_N}^\dagger \cdots \hat{d}_{n_1}^\dagger |0\rangle \ , \tag{2.31}$$

and the actions of the annihilation operators can be expressed as

$$\hat{d}_n |n_1 \cdots n_N\rangle = \sum_{k=1}^{N} (\pm)^{N+k} \delta_{nn_k} |n_1 \cdots n_{k-1} n_{k+1} \cdots n_N\rangle \ , \tag{2.32}$$

which are completely analogous to Eqs. (2.22) and (2.26), with the index n playing the role of $\boldsymbol{x}$. Similarly, d's and $d^\dagger$'s obey the same (anti)commutation relations

$$[\hat{d}_n, \hat{d}_m^\dagger]_\mp = \delta_{mn} \ , \quad [\hat{d}_n, \hat{d}_m]_\mp = [\hat{d}_n^\dagger, \hat{d}_m^\dagger]_\mp = 0 \ . \tag{2.33}$$

As a result, the N-particle ket $|n_1 \cdots n_N\rangle$ has the symmetry property

$$|n_{P(1)} \cdots n_{P(N)}\rangle = (\pm)^P |n_1 \cdots n_N\rangle \ , \tag{2.34}$$

the inner product expression

$$\langle n'_1 \cdots n'_N | n_1 \cdots n_N \rangle = \sum_P (\pm)^P \prod_{j=1}^N \delta_{n'_j n_{P(j)}} \ , \tag{2.35}$$

and the completeness relation

$$\frac{1}{N!} \sum_{n_1,\cdots,n_N} |n_1 \cdots n_N\rangle\langle n_1 \cdots n_N| = \hat{\mathbb{1}} \ . \tag{2.36}$$

The ket $|\Psi\rangle$ can be expanded in this basis as

$$|\Psi\rangle = \frac{1}{N!} \sum_{n_1,\cdots,n_N} |n_1 \cdots n_N\rangle \Psi(n_1, \cdots, n_N) \ , \tag{2.37}$$

where the wavefunction $\Psi(n_1, \cdots, n_N) = \langle n_1 \cdots n_N | \Psi \rangle$ is totally symmetric (antisymmetric) for bosons (fermions), and the normalization condition reads

$$1 = \langle \Psi | \Psi \rangle = \frac{1}{N!} \sum_{n_1,\cdots,n_N} |\Psi(n_1, \cdots, n_N)|^2. \tag{2.38}$$

Conversely, the field operators can be recovered from

$$\hat{\psi}(\boldsymbol{x}) = \sum_n \varphi_n(\boldsymbol{x}) \hat{d}_n \ , \quad \hat{\psi}^\dagger(\boldsymbol{x}) = \sum_n \varphi^*_n(\boldsymbol{x}) \hat{d}^\dagger_n \ , \tag{2.39}$$

their kets are related by

$$|\boldsymbol{x}_1 \cdots \boldsymbol{x}_N\rangle = \sum_{n_1,\cdots,n_N} \varphi^*_{n_1}(\boldsymbol{x}_1) \cdots \varphi^*_{n_N}(\boldsymbol{x}_N) |n_1 \cdots n_N\rangle \ , \tag{2.40}$$

$$|n_1, \cdots, n_N\rangle = \int d\boldsymbol{x}_1 \cdots d\boldsymbol{x}_N \varphi_{n_1}(\boldsymbol{x}_1) \cdots \varphi_{n_N}(\boldsymbol{x}_N) |\boldsymbol{x}_1 \cdots \boldsymbol{x}_N\rangle \ , \tag{2.41}$$

and their overlaps are given by the so-called *Slater determinant*

$$\begin{aligned}\Psi_{n_1\cdots n_N}(\boldsymbol{x}_1,\cdots,\boldsymbol{x}_N) :=& \langle \boldsymbol{x}_1\cdots\boldsymbol{x}_N|n_1\cdots n_N\rangle \\ =& \sum_P (\pm)^P \varphi_{n_1}(\boldsymbol{x}_{P(1)})\cdots\varphi_{n_N}(\boldsymbol{x}_{P(N)}) \\ =& \begin{vmatrix} \varphi_{n_1}(\boldsymbol{x}_1) & \cdots & \varphi_{n_1}(\boldsymbol{x}_N) \\ \vdots & \ddots & \vdots \\ \varphi_{n_N}(\boldsymbol{x}_1) & \cdots & \varphi_{n_N}(\boldsymbol{x}_N) \end{vmatrix}_{\pm} . \end{aligned} \tag{2.42}$$

One can also transform from one orthonormal basis $\{|n\rangle\}$ to another orthonormal basis $\{|\alpha\rangle\}$ by

$$\hat{c}_\alpha = \sum_n U_{\alpha n}\hat{d}_n \ , \quad \hat{c}^\dagger_\alpha = \sum_n U^*_{\alpha n}\hat{d}^\dagger_n \ , \quad \text{with } U_{\alpha n} := \langle\alpha|n\rangle \ , \tag{2.43}$$

and one can verify that all relations introduced in this section still hold.

2.3.2.2. Non-orthogonal basis

Suppose one has a complete but non-orthogonal basis $\{|\alpha\rangle\}$. Denote the overlap by $\chi_\alpha(\boldsymbol{x}) := \langle\boldsymbol{x}|\alpha\rangle$. One can still define the creation/annihilation operators [41]

$$\hat{c}^\dagger_\alpha := \int d\boldsymbol{x}\chi_\alpha(\boldsymbol{x})\hat{\psi}^\dagger(\boldsymbol{x}) \quad \text{and} \quad \hat{c}_\alpha := \int d\boldsymbol{x}\chi^*_\alpha(\boldsymbol{x})\hat{\psi}(\boldsymbol{x}) \tag{2.44}$$

to create/annihilate state $|\alpha\rangle$, and the (anti)commutator with the field operator still gives back the overlap,

$$[\hat{\psi}(\boldsymbol{x}), \hat{c}^\dagger_\alpha]_\mp = \int d\boldsymbol{x}'\chi_\alpha(\boldsymbol{x}')[\hat{\psi}(\boldsymbol{x}), \hat{\psi}^\dagger(\boldsymbol{x}')]_\mp = \chi_\alpha(\boldsymbol{x}) \ . \tag{2.45}$$

However, their (anti)commutation relations now become

$$[\hat{c}_\alpha, \hat{c}^\dagger_\beta]_\mp = \boldsymbol{S}_{\alpha\beta} \ , \quad [\hat{c}_\alpha, \hat{c}_\beta]_\mp = [\hat{c}^\dagger_\alpha, \hat{c}^\dagger_\beta]_\mp = 0 \ , \tag{2.46}$$

where $\boldsymbol{S}_{\alpha\beta} = \langle\alpha|\beta\rangle$ is the overlap matrix. The inverse expressions are now

$$\hat{\psi}(\boldsymbol{x}) = \sum_{\alpha,\beta} \chi_\alpha(\boldsymbol{x})(\boldsymbol{S}^{-1})_{\alpha\beta}\hat{c}_\beta \ , \quad \hat{\psi}^\dagger(\boldsymbol{x}) = \sum_{\alpha,\beta} \hat{c}^\dagger_\alpha(\boldsymbol{S}^{-1})_{\alpha\beta}\chi^*_\beta(\boldsymbol{x}) \ , \tag{2.47}$$

where we have utilized the identity operator

$$\sum_{\alpha,\beta} |\alpha\rangle(\boldsymbol{S}^{-1})_{\alpha\beta}\langle\beta| = \hat{\mathbb{1}} \ . \tag{2.48}$$

In practice, it is usually more convenient to work in an orthogonal basis $|n\rangle$. To achieve this, one needs to find a transformation matrix $\boldsymbol{T}$ such that

$$|n\rangle = \sum_\alpha |\alpha\rangle\boldsymbol{T}_{\alpha n} \quad \text{with} \quad \langle n|m\rangle = \delta_{nm} \ . \tag{2.49}$$

This implies that $\boldsymbol{T}^\dagger\boldsymbol{S}\boldsymbol{T} = \mathbb{1}$, or equivalently $\boldsymbol{S}^{-1} = \boldsymbol{T}\boldsymbol{T}^\dagger$, which does not uniquely determine $\boldsymbol{T}$.

Since $\boldsymbol{S}$ is a Hermitian and positive definite matrix, it can be diagonalized by a unitary matrix $\boldsymbol{U}$:

$$\boldsymbol{S} = \boldsymbol{U}\boldsymbol{s}\boldsymbol{U}^\dagger \ , \tag{2.50}$$

where $\boldsymbol{s}$ is a diagonal matrix with positive diagonal entries. One commonly adopted way is to use the so-called *Löwdin symmetric orthogonalization* [42]

$$\boldsymbol{T} = \boldsymbol{S}^{-1/2} = \boldsymbol{U}\boldsymbol{s}^{-1/2}\boldsymbol{U}^\dagger \ . \tag{2.51}$$

It can minimize the modification of the basis vectors due to orthogonalization, i.e., it minimizes $\sum_n ||\sum_\alpha |\alpha\rangle(\boldsymbol{T}_{\alpha n} - \delta_{\alpha n})||^2$. At the same time, numerical instability can arise when basis sets have near-linear dependence, i.e., when $\boldsymbol{s}$ has near-zero diagonal elements.

Another commonly adopted way is called *canonical orthogonalization* [43]

$$\boldsymbol{T} = \boldsymbol{U}\boldsymbol{s}^{-1/2} \ . \tag{2.52}$$

When there is near-linear dependence, one can truncate $\boldsymbol{s} = \mathrm{diag}\{s_1 \geq s_2 \geq \cdots \geq s_N\}$ at s_{n_0} according to a certain threshold and construct a truncated rectangular transfor-

mation matrix $\tilde{\boldsymbol{T}}$ of size N by n_0. With this, one obtains n_0 orthonormal basis sets by discarding $N - n_0$ sets with near-linear dependence, which circumvents the numerical instability issue mentioned above.

For simplicity, in the following we will always work with the orthogonal basis mentioned in the previous section.

2.3.3. Hamiltonian

In second quantization, one can verify that the generic Hamiltonian with non-interacting parts (2.19) and two-body interacting parts (2.20) can be expressed as [6]

$$\begin{aligned}\hat{H} &= \hat{H}_0 + \hat{H}_{\text{int}} \\ &= \int d\boldsymbol{x} d\boldsymbol{x}' \hat{\psi}^\dagger(\boldsymbol{x}) \langle \boldsymbol{x}|\hat{h}|\boldsymbol{x}'\rangle \hat{\psi}(\boldsymbol{x}') \\ &+ \frac{1}{2}\int d\boldsymbol{x} d\boldsymbol{x}' v(\boldsymbol{x},\boldsymbol{x}') \hat{\psi}^\dagger(\boldsymbol{x}) \hat{\psi}^\dagger(\boldsymbol{x}') \hat{\psi}(\boldsymbol{x}') \hat{\psi}(\boldsymbol{x}) \ . \end{aligned} \tag{2.53}$$

In a generic orbital basis $\{|n\rangle\}$, it can be re-expressed as

$$\hat{H} = \sum_{ij} h_{ij} \hat{d}_i^\dagger \hat{d}_j + \frac{1}{2}\sum_{ijkl} v_{ijkl} \hat{d}_i^\dagger \hat{d}_j^\dagger \hat{d}_k \hat{d}_l \ , \tag{2.54}$$

with

$$h_{ij} = \langle i|\hat{h}|j\rangle = \sum_{\sigma\sigma'} \int d\boldsymbol{r} \varphi_i^*(\boldsymbol{r}\sigma) h_{\sigma\sigma'}(\boldsymbol{r}, -i\boldsymbol{\nabla}, \boldsymbol{S}) \varphi_j(\boldsymbol{r}\sigma') = h_{ji}^* \ , \tag{2.55}$$

$$v_{ijkl} = \int d\boldsymbol{x} d\boldsymbol{x}' \varphi_i^*(\boldsymbol{x}) \varphi_j^*(\boldsymbol{x}') v(\boldsymbol{x},\boldsymbol{x}') \varphi_k(\boldsymbol{x}') \varphi_l(\boldsymbol{x}) \ . \tag{2.56}$$

In research, two other notations are commonly used [4]: the *physicists' notation*, defined as

$$\hat{H}_{\text{int}} = \frac{1}{2}\sum_{ijkl} v_{ijkl}^{(\text{phys})} \hat{d}_i^\dagger \hat{d}_j^\dagger \hat{d}_l \hat{d}_k \ , \tag{2.57}$$

$$\text{with} \quad v_{ijkl}^{(\text{phys})} = \int d\boldsymbol{x} d\boldsymbol{x}' \varphi_i^*(\boldsymbol{x}) \varphi_j^*(\boldsymbol{x}') v(\boldsymbol{x},\boldsymbol{x}') \varphi_l(\boldsymbol{x}') \varphi_k(\boldsymbol{x}) \ , \tag{2.58}$$

and the *chemists' notation*, defined as

$$\hat{H}_{\text{int}} = \frac{1}{2} \sum_{ijkl} v_{ijkl}^{(\text{chem})} \hat{d}_i^\dagger \hat{d}_k^\dagger \hat{d}_l \hat{d}_j \ , \tag{2.59}$$

$$\text{with} \quad v_{ijkl}^{(\text{chem})} = \int d\boldsymbol{x} d\boldsymbol{x}' \varphi_i^*(\boldsymbol{x}) \varphi_k^*(\boldsymbol{x}') v(\boldsymbol{x}, \boldsymbol{x}') \varphi_l(\boldsymbol{x}') \varphi_j(\boldsymbol{x}) \ , \tag{2.60}$$

which are related by

$$v_{ijkl}^{(\text{phys})} = v_{ikjl}^{(\text{chem})} \ . \tag{2.61}$$

In practice, it is vitally important to keep the notation consistent throughout the entire simulation.

Finally, in the absence of magnetic fields or spin-orbit coupling, Eq. (2.54) can be re-expressed as

$$\hat{H} = \sum_{s_1 s_2} \sum_{\sigma} h_{s_1 s_2} \hat{d}_{s_1\sigma}^\dagger \hat{d}_{s_2\sigma} + \frac{1}{2} \sum_{s_1 s_2 s_3 s_4} \sum_{\sigma\sigma'} v_{s_1 s_2 s_3 s_4} \hat{d}_{s_1\sigma}^\dagger \hat{d}_{s_2\sigma'}^\dagger \hat{d}_{s_3\sigma'} \hat{d}_{s_4\sigma} \ , \tag{2.62}$$

and the two notations introduced above can be modified accordingly without difficulty.

Chapter 3

Single-Particle Green's Functions

This chapter introduces the single-particle Green's functions used in this thesis. For simplicity, we consider only equilibrium and nonequilibrium steady-state cases, in which the real-time Green's functions depend on a single time variable.

3.1. Definition

3.1.1. Imaginary-axis Green's functions

To study a generic Hamiltonian $\hat{H}$ defined in Eq. (2.54), its equilibrium properties can usually be more easily simulated on the imaginary axis [2, 3]. The *imaginary-time Green's function* is defined as

$$\boldsymbol{G}_{ij}(\tau) = -\langle T_\tau \hat{d}_i(\tau)\hat{d}_j^\dagger\rangle = \begin{cases} -\langle \hat{d}_i(\tau)\hat{d}_j^\dagger\rangle\ , & \tau > 0 \\ \mp\langle \hat{d}_j^\dagger\hat{d}_i(\tau)\rangle\ , & \tau < 0 \end{cases}\ , \tag{3.1}$$

where T_τ is the time-ordering operator with respect to τ, $\hat{O}(\tau) = e^{\tau\hat{K}}\hat{O}e^{-\tau\hat{K}}$ is the imaginary-time Heisenberg representation of $\hat{O}$ defined for $-\beta \le \tau \le \beta$, with β being the inverse temperature, and $\hat{K} = \hat{H} - \mu\hat{N}$ is the grand canonical Hamiltonian. $\boldsymbol{G}_{ij}(\tau < 0)$ and $\boldsymbol{G}_{ij}(\tau > 0)$ are connected by the (anti)periodic relation

$$\begin{aligned} \boldsymbol{G}_{ij}(\tau < 0) &= \mp\frac{1}{\mathcal{Z}}\mathrm{Tr}[e^{-\beta\hat{K}}\hat{d}_j^\dagger e^{\tau\hat{K}}\hat{d}_i e^{-\tau\hat{K}}] = \mp\frac{1}{\mathcal{Z}}\mathrm{Tr}[e^{\tau\hat{K}}\hat{d}_i e^{-\tau\hat{K}}e^{-\beta\hat{K}}\hat{d}_j^\dagger] \\ &= \mp\frac{1}{\mathcal{Z}}\mathrm{Tr}[e^{-\beta\hat{K}}e^{(\beta+\tau)\hat{K}}\hat{d}_i e^{-(\beta+\tau)\hat{K}}\hat{d}_j^\dagger] = \pm\boldsymbol{G}_{ij}(\beta+\tau)\ . \end{aligned} \tag{3.2}$$

Therefore, in the following discussions, we will always assume $\tau \geq 0$ without loss of generality. At the same time, since $\boldsymbol{G}_{ij}(\tau)$ is not well-defined at the endpoints 0 and $\pm\beta$, it is advisable to use notation such as $\boldsymbol{G}_{ij}(0^+)$ and $\boldsymbol{G}_{ij}(\beta^-)$ instead of $\boldsymbol{G}_{ij}(0)$ and $\boldsymbol{G}_{ij}(\beta)$ directly.

One can perform a Fourier transform of $\boldsymbol{G}(\tau)$ to obtain the so-called *Matsubara Green's function* in imaginary frequency,

$$\boldsymbol{G}(i\omega_\nu) = \int_0^\beta d\tau e^{i\omega_\nu \tau} \boldsymbol{G}(\tau) \ , \tag{3.3}$$

$$\boldsymbol{G}(\tau) = \frac{1}{\beta} \sum_{\nu=-\infty}^{+\infty} e^{-i\omega_\nu \tau} \boldsymbol{G}(i\omega_\nu) \ , \tag{3.4}$$

where the *Matsubara frequency* $i\omega_\nu$ is $2\nu\pi i/\beta$ for bosons and $(2\nu+1)\pi i/\beta$ for fermions, with ν being an integer. The difference in $i\omega_\nu$ comes from the (anti)periodic relation (3.2).

3.1.2. Real-axis Green's functions

To obtain the physical interpretation and compare them to experimental measurements such as angle-resolved photoemission spectroscopy (ARPES), however, one needs to obtain the so-called *retarded Green's function*

$$\boldsymbol{G}^{\mathrm{Ret}}_{ij}(t) = -i\theta(t)\langle [\hat{d}_i(t), \hat{d}^\dagger_j]_\mp \rangle \ , \tag{3.5}$$

where $\hat{O}(t) = e^{i\hat{K}t}\hat{O}e^{-i\hat{K}t}$ is the real-time Heisenberg representation of the operator $\hat{O}$, and $\theta(t) = 1$ for $t > 0$ and 0 for $t < 0$ is the step function. The *spectral function* can then be obtained from the imaginary part of its Fourier transform,

$$A(\omega) = -\frac{1}{\pi}\mathrm{Im}[\mathrm{Tr}[\boldsymbol{G}^{\mathrm{Ret}}(\omega)]] \quad \text{with} \quad \boldsymbol{G}^{\mathrm{Ret}}(\omega) = \int_{-\infty}^{+\infty} dt e^{i\omega t} \boldsymbol{G}^{\mathrm{Ret}}(t) \ . \tag{3.6}$$

The Green's function (3.5) is "retarded" since it is non-vanishing only for $t > 0$. At the same time, one can also define a function in the complementary time domain, $t < 0$,

which is referred to as the *advanced Green's function*

$$\boldsymbol{G}_{ij}^{\mathrm{Adv}}(t) = +i\theta(-t)\langle[\hat{d}_i(t), \hat{d}_j^\dagger]_\mp\rangle \ . \tag{3.7}$$

The other Green's functions that will be discussed in this thesis, although they are not the main focus, include the time-ordered/anti-time-ordered Green's functions $\boldsymbol{G}^T(t)$ / $\boldsymbol{G}^{\tilde{T}}(t)$, the greater/lesser Green's functions $\boldsymbol{G}^>(t)$ / $\boldsymbol{G}^<(t)$, and the Keldysh Green's function $\boldsymbol{G}^K(t)$, which have the following definitions:

$$\boldsymbol{G}_{ij}^{T}(t) = -i\langle T_t \hat{d}_i(t)\hat{d}_j^\dagger\rangle \ , \tag{3.8}$$
$$\boldsymbol{G}_{ij}^{\tilde{T}}(t) = -i\langle T_{-t} \hat{d}_i(t)\hat{d}_j^\dagger\rangle \ , \tag{3.9}$$
$$\boldsymbol{G}_{ij}^{>}(t) = -i\langle \hat{d}_i(t)\hat{d}_j^\dagger\rangle \ , \tag{3.10}$$
$$\boldsymbol{G}_{ij}^{<}(t) = \mp i\langle \hat{d}_j^\dagger\hat{d}_i(t)\rangle \ , \tag{3.11}$$
$$\boldsymbol{G}_{ij}^{K}(t) = -i\langle[\hat{d}_i(t), \hat{d}_j^\dagger]_\pm\rangle \ . \tag{3.12}$$

3.2. Lehmann Representation

To facilitate the derivation of analytic properties in the next section, we expand the above-mentioned Green's functions in the eigenbasis of $\hat{K}$, i.e., $\hat{K}|n\rangle = E_n|n\rangle$, to obtain the so-called *Lehmann representation*. Its time-domain expressions have the following form:

$$\boldsymbol{G}_{ij}(\tau) = -\frac{1}{\mathcal{Z}}\sum_{mn} e^{-\beta E_m} e^{\tau(E_m - E_n)} v_{mn}^{(i)} v_{mn}^{(j)*} \ , \tag{3.13}$$

$$\begin{pmatrix} \boldsymbol{G}_{ij}^{\mathrm{Ret}}(t) \\ \boldsymbol{G}_{ij}^{\mathrm{Adv}}(t) \end{pmatrix} = -\frac{i}{\mathcal{Z}}\sum_{mn} \begin{pmatrix} \theta(t) \\ -\theta(-t) \end{pmatrix} (e^{-\beta E_m} \mp e^{-\beta E_n}) f_{mn}(t) v_{mn}^{(i)} v_{mn}^{(j)*} \ , \tag{3.14}$$

$$\begin{pmatrix} \boldsymbol{G}_{ij}^{T}(t) \\ \boldsymbol{G}_{ij}^{\tilde{T}}(t) \end{pmatrix} = -\frac{i}{\mathcal{Z}}\sum_{mn} \begin{pmatrix} \theta(t)e^{-\beta E_m} \pm \theta(-t)e^{-\beta E_n} \\ \theta(-t)e^{-\beta E_m} \pm \theta(t)e^{-\beta E_n} \end{pmatrix} f_{mn}(t) v_{mn}^{(i)} v_{mn}^{(j)*} \ , \tag{3.15}$$

$$\begin{pmatrix} \boldsymbol{G}_{ij}^{>}(t) \\ \boldsymbol{G}_{ij}^{<}(t) \end{pmatrix} = -\frac{i}{\mathcal{Z}}\sum_{mn} \begin{pmatrix} +e^{-\beta E_m} \\ \pm e^{-\beta E_n} \end{pmatrix} f_{mn}(t) v_{mn}^{(i)} v_{mn}^{(j)*} \ , \tag{3.16}$$

$$\boldsymbol{G}_{ij}^{K}(t) = -\frac{i}{\mathcal{Z}}\sum_{mn} (e^{-\beta E_m} \pm e^{-\beta E_n}) f_{mn}(t) v_{mn}^{(i)} v_{mn}^{(j)*} \ , \tag{3.17}$$

where $f_{mn}(t) = e^{i(E_m - E_n)t}$ and $v_{mn}^{(i)} = \langle m|\hat{d}_i|n\rangle$.

In addition, one can perform Fourier transforms to obtain the Lehmann representation in the frequency domain:

$$\boldsymbol{G}_{ij}(i\omega_\nu) = \frac{1}{\mathcal{Z}} \sum_{mn} \frac{e^{-\beta E_m} \mp e^{-\beta E_n}}{i\omega_\nu - (E_n - E_m)} v_{mn}^{(i)} v_{mn}^{(j)*} \ , \tag{3.18}$$

$$\boldsymbol{G}_{ij}^{\mathrm{Ret/Adv}}(\omega) = \frac{1}{\mathcal{Z}} \sum_{mn} \frac{e^{-\beta E_m} \mp e^{-\beta E_n}}{\omega \pm i0^+ - (E_n - E_m)} v_{mn}^{(i)} v_{mn}^{(j)*} \ , \tag{3.19}$$

$$\begin{pmatrix} \boldsymbol{G}_{ij}^{T}(\omega) \\ \boldsymbol{G}_{ij}^{\tilde{T}}(\omega) \end{pmatrix} = \frac{1}{\mathcal{Z}} \sum_{mn} \frac{\begin{pmatrix} +e^{-\beta E_m} \\ \pm e^{-\beta E_n} \end{pmatrix} v_{mn}^{(i)} v_{mn}^{(j)*}}{\omega + i0^+ - (E_n - E_m)} - \frac{\begin{pmatrix} \pm e^{-\beta E_n} \\ +e^{-\beta E_m} \end{pmatrix} v_{mn}^{(i)} v_{mn}^{(j)*}}{\omega - i0^+ - (E_n - E_m)} \ , \tag{3.20}$$

$$\begin{pmatrix} \boldsymbol{G}_{ij}^{>}(\omega) \\ \boldsymbol{G}_{ij}^{<}(\omega) \end{pmatrix} = -\frac{2\pi i}{\mathcal{Z}} \sum_{mn} \begin{pmatrix} +e^{-\beta E_m} \\ \pm e^{-\beta E_n} \end{pmatrix} \delta(\omega - (E_n - E_m)) v_{mn}^{(i)} v_{mn}^{(j)*} \ , \tag{3.21}$$

$$\boldsymbol{G}_{ij}^{K}(\omega) = -\frac{2\pi i}{\mathcal{Z}} \sum_{mn} (e^{-\beta E_m} \pm e^{-\beta E_n}) \delta(\omega - (E_n - E_m)) v_{mn}^{(i)} v_{mn}^{(j)*} \ , \tag{3.22}$$

where infinitesimal damping factors have been introduced to make the Fourier transforms convergent.

3.3. Analytic Properties

3.3.1. Complex Hermitian Hamiltonian

For a generic complex-valued Hermitian Hamiltonian $\hat{H}$, different matrix elements of $\boldsymbol{G}(\tau > 0)$ are connected by

$$\begin{aligned} \boldsymbol{G}_{ji}^{*}(\tau) &= -(\langle \hat{d}_j(\tau) \hat{d}_i^\dagger \rangle)^* = -\frac{1}{\mathcal{Z}} \mathrm{Tr}[\hat{d}_i e^{-\tau \hat{K}} \hat{d}_j^\dagger e^{-(\beta-\tau)\hat{K}}] \\ &= -\frac{1}{\mathcal{Z}} \mathrm{Tr}[e^{-(\beta-\tau)\hat{K}} \hat{d}_i e^{-\tau \hat{K}} \hat{d}_j^\dagger] = -\langle \hat{d}_i(\tau) \hat{d}_j^\dagger \rangle = \boldsymbol{G}_{ij}(\tau) \ . \end{aligned} \tag{3.23}$$

Therefore, we have

$$[\boldsymbol{G}(\tau)]^\dagger = \boldsymbol{G}(\tau) \ . \tag{3.24}$$

At the endpoints, it has the properties

$$\boldsymbol{G}_{ij}(0^+) \mp \boldsymbol{G}_{ij}(\beta^-) = \boldsymbol{G}_{ij}(0^+) - \boldsymbol{G}_{ij}(0^-) = -\langle [d_i, d_j^\dagger]_\mp \rangle = -\delta_{ij} \ . \tag{3.25}$$

At intermediate points $0 < \tau < \beta$, according to Eq. (3.13), its derivative for diagonal elements can be expressed as

$$\frac{d^k \boldsymbol{G}_{ii}(\tau)}{d\tau^k} = -\frac{1}{\mathcal{Z}} \sum_{mn} |v_{mn}^{(i)}|^2 e^{-\beta E_m} e^{\tau(E_m - E_n)} (E_m - E_n)^k \ , \tag{3.26}$$

so that its even derivatives are always negative,

$$\frac{d^k \boldsymbol{G}_{ii}(\tau)}{d\tau^k} < 0 \text{ at any even } k \ . \tag{3.27}$$

To explore the analytic properties of $\boldsymbol{G}(i\omega_n)$ and $\boldsymbol{G}^{\mathrm{Ret/Adv}}(\omega)$, we again utilize the Lehmann representations (3.18) and (3.19), which clearly show that we can define a generic Green's function on the whole complex plane $z \in \mathbb{Z}$

$$\boldsymbol{G}_{ij}(z) = \frac{1}{\mathcal{Z}} \sum_{mn} \frac{e^{-\beta E_m} \mp e^{-\beta E_n}}{z - (E_n - E_m)} v_{mn}^{(i)} v_{mn}^{(j)*} \ , \tag{3.28}$$

so that

$$\boldsymbol{G}^{\mathrm{Ret/Adv}}(\omega) = \boldsymbol{G}(z = \omega \pm i0^+) \quad \text{and} \quad \boldsymbol{G}(i\omega_\nu) = \boldsymbol{G}(z = i\omega_\nu) \ . \tag{3.29}$$

Therefore, one only needs to study the properties of $\boldsymbol{G}(z)$.

Denote $E_n - E_m \in \mathbb{R}$ by $\xi_l^{(r)}$, where the superscript "r" stands for "real-valued", and the indices m and n have been combined into a single index l. Then Eq. (3.28) can be written as

$$\boldsymbol{G}(z) = \sum_l \frac{\boldsymbol{A}_l^{(r)}}{z - \xi_l^{(r)}} \ , \quad \xi_l^{(r)} \in \mathbb{R} \ , \tag{3.30}$$

where $[\boldsymbol{A}_l^{(r)}]_{ij} = \frac{1}{\mathcal{Z}}(e^{-\beta E_m} \mp e^{-\beta E_n}) v_{mn}^{(i)} v_{mn}^{(j)*}$ is the rank-1 matrix (excluding the $E_m = E_n$ cases for bosons) corresponding to the weight of the simple pole $\xi_l^{(r)}$. Note that the superscript "r" in $\boldsymbol{A}_l^{(r)}$ indicates that it is the weight corresponding to the real-valued pole $\xi_l^{(r)}$; it does not mean that the matrix itself is real-valued. For later convenience, we can combine different $\xi_l^{(r)}$'s that have the same value and discard those with zero

weights $\boldsymbol{A}_l^{(r)} = 0$. As a result, the form of $\boldsymbol{G}(z)$ is unchanged compared to Eq. (3.30), but the summation is now over all distinct $\xi_l^{(r)}$'s with non-zero weights $\boldsymbol{A}_l^{(r)} \neq 0$. Note that after this combination, $\boldsymbol{A}_l^{(r)}$ may no longer be a rank-1 matrix.

Denote $\boldsymbol{v}_{mn} = (v_{mn}^{(1)}, \cdots, v_{mn}^{(N)})^T$ as the column vector consisting of $v_{mn}^{(i)}$. Then we have

$$\boldsymbol{A}_l^{(r)} = \frac{1}{\mathcal{Z}} \sum_{\{m,n:E_m - E_n = \xi_l^{(r)}\}} (e^{-\beta E_m} \mp e^{-\beta E_n}) \boldsymbol{v}_{mn} \boldsymbol{v}_{mn}^\dagger \ , \tag{3.31}$$

which is always a Hermitian matrix, i.e., $[\boldsymbol{A}_l^{(r)}]^\dagger = \boldsymbol{A}_l^{(r)}$. In addition, since $\mathrm{sgn}(\xi_l^{(r)}) \times (e^{-\beta E_m} - e^{-\beta E_n})$ and $e^{-\beta E_m} + e^{-\beta E_n}$ are always non-negative, we can arrive at the conclusion that $\mathrm{sgn}(\xi_l^{(r)}) \boldsymbol{A}_l^{(r)}$ for bosons and $\boldsymbol{A}_l^{(r)}$ for fermions are always positive-semidefinite matrices. In addition, $\boldsymbol{A}_l^{(r)}$ satisfies the sum rules

$$\begin{aligned}
\sum_l [\boldsymbol{A}_l^{(r)}]_{ij} &= \frac{1}{\mathcal{Z}} \sum_{mn} (e^{-\beta E_m} \mp e^{-\beta E_n}) \langle m|\hat{d}_i|n\rangle \langle n|\hat{d}_j^\dagger|m\rangle \\
&= \frac{1}{\mathcal{Z}} \left(\sum_m e^{-\beta E_m} \langle m|\hat{d}_i \hat{d}_j^\dagger|m\rangle \mp \sum_n e^{-\beta E_n} \langle n|\hat{d}_j^\dagger \hat{d}_i|n\rangle \right) \\
&= \frac{1}{\mathcal{Z}} \sum_m e^{-\beta E_m} \langle m|[\hat{d}_i, \hat{d}_j^\dagger]_\mp|m\rangle \\
&= \delta_{ij} \ ,
\end{aligned} \tag{3.32}$$

so that

$$\sum_l \boldsymbol{A}_l^{(r)} = \mathbb{1} \ . \tag{3.33}$$

Note that in this derivation, the (anti)commutation rule $[\hat{d}_i, \hat{d}_j^\dagger]_\mp = \delta_{ij}$ has been utilized. In practice, people sometimes also refer to even (odd) products of fermionic operators as bosonic (fermionic) operators. In that case, the above-mentioned (anti)commutation relation may no longer hold, and one thus needs to modify Eq. (3.32) accordingly. For example, for bosonic operators $\hat{n}_i = \hat{d}_i^\dagger \hat{d}_i$, where $\hat{d}_i^{(\dagger)}$ are fermionic operators, the commutation relation becomes $[\hat{n}_i, \hat{n}_j]_- = 0$, and therefore $\sum_l \boldsymbol{A}_l^{(r)} = 0$.

As for $\boldsymbol{G}(z)$, it has the up-down symmetry

$$[\boldsymbol{G}(z)]^\dagger = \sum_l \frac{[\boldsymbol{A}_l^{(r)}]^\dagger}{z^* - \xi_l^{(r)*}} = \sum_l \frac{\boldsymbol{A}_l^{(r)}}{z^* - \xi_l^{(r)}} = \boldsymbol{G}(z^*) \ , \tag{3.34}$$

which implies $[\boldsymbol{G}(i\omega_\nu)]^\dagger = \boldsymbol{G}(-i\omega_\nu)$ after setting $z = i\omega_\nu$ and $[\boldsymbol{G}^{\rm Ret}(\omega)]^\dagger = \boldsymbol{G}^{\rm Adv}(\omega)$ after setting $z = \omega + i0^+$. In addition, it has the following asymptotic behavior:

$$\boldsymbol{G}(z) = \frac{\sum_l \boldsymbol{A}_l^{(r)}}{z} + \mathcal{O}\left(\frac{1}{z^2}\right) = \frac{\mathbb{1}}{z} + \mathcal{O}\left(\frac{1}{z^2}\right) \ , \quad z \to \infty \ . \tag{3.35}$$

By applying the so-called *Sokhotski-Plemelj theorem* $\frac{1}{x \pm i0^+} = \mp i\pi\delta(x) + \mathcal{P}(\frac{1}{x})$ to Eq. (3.30), where $\mathcal{P}(x)$ is the Cauchy principal value, we can arrive at the expression

$$\boldsymbol{G}(\omega \pm i0^+) = \sum_l \boldsymbol{A}_l^{(r)} \left[\mathcal{P}\left(\frac{1}{\omega - \xi_l^{(r)}}\right) \mp i\pi\delta(\omega - \xi_l^{(r)}) \right] \ . \tag{3.36}$$

Define the Hilbert transform as $\mathcal{H}f(\omega) = \frac{1}{\pi}\mathcal{P}\int_{-\infty}^{+\infty} d\omega' \frac{f(\omega')}{\omega - \omega'}$. By utilizing the properties of the Hilbert transform, $\mathcal{H}\delta(x) = \frac{1}{\pi x}$ and $\mathcal{H}\mathcal{H}f \equiv -f$, the real and imaginary parts of the retarded/advanced Green's function satisfy the so-called *Kramers-Kronig relations* [44–46]

$$\mathrm{Im}\boldsymbol{G}^{\rm Ret/Adv}(\omega) = \pm\mathcal{H}\mathrm{Re}\boldsymbol{G}^{\rm Ret/Adv}(\omega) \ . \tag{3.37}$$

At the same time, we can also define the so-called *spectral density matrix*

$$\boldsymbol{A}(\omega) = \sum_l \boldsymbol{A}_l^{(r)} \delta(\omega - \xi_l^{(r)}) \ , \tag{3.38}$$

which is connected to the spectral function $A(\omega)$ (3.6) by

$$A(\omega) = \mathrm{Tr}[\boldsymbol{A}(\omega)] = \sum_l \mathrm{Tr}[\boldsymbol{A}_l^{(r)}] \delta(\omega - \xi_l^{(r)}) \ , \tag{3.39}$$

to $\boldsymbol{G}(z)$ by

$$\boldsymbol{G}(z) = \int_{-\infty}^{+\infty} d\omega \frac{\boldsymbol{A}(\omega)}{z - \omega} = \sum_l \frac{\boldsymbol{A}_l^{(r)}}{z - \xi_l^{(r)}} \ , \tag{3.40}$$

and to $\boldsymbol{G}(\tau)$ by

$$\boldsymbol{G}(\tau) = -\int_{-\infty}^{+\infty} d\omega \frac{e^{-\omega\tau}}{1 \mp e^{-\beta\omega}} \boldsymbol{A}(\omega) = -\sum_l \boldsymbol{A}_l^{(r)} \frac{e^{-\xi_l^{(r)}\tau}}{1 \mp e^{-\beta\xi_l^{(r)}}} \ . \tag{3.41}$$

Finally, by utilizing the Lehmann representation introduced in Sec. 3.2, one can verify

that the greater/lesser Green's functions $\boldsymbol{G}^{>}(\omega)$ / $\boldsymbol{G}^{<}(\omega)$ and the Keldysh Green's function $\boldsymbol{G}^{K}(\omega)$ can also be expressed in terms of the spectral density matrix,

$$\boldsymbol{G}^{>}(\omega) = -2\pi i \boldsymbol{A}(\omega)(1 \pm n(\omega)) = -2\pi i \sum_l \boldsymbol{A}_l^{(r)} (1 \pm n(\xi_l^{(r)})) \delta(\omega - \xi_l^{(r)}), \tag{3.42}$$

$$\boldsymbol{G}^{<}(\omega) = \mp 2\pi i \boldsymbol{A}(\omega) n(\omega) = \mp 2\pi i \sum_l \boldsymbol{A}_l^{(r)} n(\xi_l^{(r)}) \delta(\omega - \xi_l^{(r)}), \tag{3.43}$$

$$\boldsymbol{G}^{K}(\omega) = -2\pi i \boldsymbol{A}(\omega)(1 \pm 2n(\omega)) = -2\pi i \sum_l \boldsymbol{A}_l^{(r)} (1 \pm 2n(\xi_l^{(r)})) \delta(\omega - \xi_l^{(r)}). \tag{3.44}$$

3.3.2. Real symmetric Hamiltonian

In practice, one often deals with real-valued h_{ij} and v_{ijkl} in the Hamiltonian (2.54). In this case, we can adopt the basis (2.31), i.e., without other complex-valued prefactors of norm 1, so that the matrix representations of both $\hat{H}$ and $\hat{d}^{(\dagger)}$ are real-valued. Furthermore, for a real symmetric matrix representation of $\hat{H}$, we can always restrict the eigenvectors to be real-valued. As a result, the whole calculation can be performed using real numbers.

The analytic properties introduced in the previous section can also be simplified accordingly. For example, $\boldsymbol{G}(\tau)$, $\boldsymbol{A}_l^{(r)}$, and $\boldsymbol{A}(\omega)$ are now all real symmetric, and $\boldsymbol{G}(z)$ is now symmetric,

$$[\boldsymbol{G}(z)]^T = \sum_l \frac{[\boldsymbol{A}_l^{(r)}]^T}{z - \xi_l^{(r)}} = \sum_l \frac{\boldsymbol{A}_l^{(r)}}{z - \xi_l^{(r)}} = \boldsymbol{G}(z) \ , \tag{3.45}$$

although $\mathrm{Im}[\boldsymbol{G}(z)]$ may be non-zero. In addition, the spectral density matrix can now be obtained from

$$\boldsymbol{A}(\omega) = -\frac{1}{\pi} \mathrm{Im}[\boldsymbol{G}(\omega + i0^+)] \ . \tag{3.46}$$

Note that this equation is not always true for a generic complex Hermitian Hamiltonian, since the off-diagonal elements of $\boldsymbol{A}_l^{(r)}$ could be non-real.

For simplicity, we will assume that the matrix representation of $\hat{H}$ is real symmetric throughout this thesis, unless otherwise specified.

Chapter 4

Prony-Like Methods

In this chapter, we introduce Prony-like methods, which provide an essential foundation for the remaining chapters and will be used repeatedly throughout this thesis.

4.1. Introduction

Prony's interpolation method [47] is a signal-processing technique originally proposed to decompose a uniformly sampled real-time signal $f(t)$ into a sum of decaying oscillatory signals. It constructs an interpolation of the form

$$f(t) = \sum_i A_i e^{-\alpha_i t} \cos(\omega_i t + \phi_i) \,, \tag{4.1}$$

estimating the amplitude A_i, damping factor α_i, angular frequency ω_i and phase ϕ_i of a damped sinusoidal signal. After these real-valued variables are incorporated into the complex-valued variables R_i and s_i, the equation can be rewritten as

$$f(t) = \sum_{i=1}^{M} R_i e^{s_i t} \,, \tag{4.2}$$

where M is a positive integer. Thus, Prony's method decomposes functions into sums of complex exponentials. In practice, Prony's interpolation method is numerically unstable [48].

Rather than an interpolation problem, we consider here the related approximation

problem: given N values of $f(t)$ sampled uniformly on a finite interval $[a, b]$ and given a target precision $\varepsilon > 0$, find a set of complex weights R_i and complex nodes z_i such that

$$\left| f_k - \sum_{i=1}^{M} R_i z_i^k \right| \leq \varepsilon \quad \text{for any } 0 \leq k \leq N-1 \,, \tag{4.3}$$

where $f_k = f(a + k\Delta t)$, $z_i = e^{s_i \Delta t}$, with $\Delta t = (b-a)/(N-1)$, and the extra prefactor $e^{s_i a}$ has been absorbed into R_i.

Several numerical methods have been developed to estimate M, R_i and z_i from possibly noise-contaminated data f_k. These include the Prony approximation method [49, 50], the Matrix Pencil Method [51–53], and the Estimation of Signal Parameters via Rotational Invariance Techniques (ESPRIT) [53, 54]. All of these methods can be understood as Prony-like methods [53].

4.2. Prony Approximation Method

Suppose the number of sampling points $N = 2N_0 + 1$ is odd. According to Ref. [49], we can construct an $(N_0+1) \times (N_0+1)$ complex-valued symmetric Hankel matrix $\boldsymbol{H}$ with elements $\boldsymbol{H}_{ij} = f_{i+j}$ for $0 \leq i, j \leq N_0$. Let $\{\sigma, \boldsymbol{u}\}$ be the c-eigenpair of the matrix, that is, $\boldsymbol{H}\boldsymbol{u} = \sigma \bar{\boldsymbol{u}}$. Assuming that the polynomial $P_{\boldsymbol{u}}(z) = \sum_{n=0}^{N_0} u_n z^n$ has N_0 distinct roots $\{z_1, \cdots, z_{N_0}\}$, which is always numerically achievable, and after choosing an integer $L \geq 2N_0 + 1$, there exists a unique vector $\{R_1, \cdots, R_{N_0}\}$ such that

$$f_k = \sum_{i=1}^{N_0} R_i z_i^k + \sigma d_k^{(L)} \text{ for all } 0 \leq k \leq N-1 \,, \tag{4.4}$$

where $d_k^{(L)} = \frac{1}{L} \sum_{l=0}^{L-1} \tilde{d}_l^{(L)} \exp(2\pi i k l / L)$ is the inverse discrete Fourier transform of the unit-norm variables $\tilde{d}_l^{(L)} = P_{\bar{\boldsymbol{u}}}(\exp(-2\pi i l/L)) / P_{\boldsymbol{u}}(\exp(2\pi i l/L))$ and thus has norm no larger than 1. Therefore, we obtain an approximation whose error is always controlled by σ, namely,

$$\left| f_k - \sum_{i=1}^{N_0} R_i z_i^k \right| \leq \sigma \text{ for all } 0 \leq k \leq N-1 \,. \tag{4.5}$$

Consequently, for any prescribed accuracy ε, as long as we can find suitable σ, the approximation can be made controlled.

One of the most powerful features of this method is that when $f(t)$ is a damping function that approaches 0 as $t \to +\infty$, only $M \propto \ln(1/\sigma)$ nodes have weights larger than σ [49]. Therefore, $\sum_{i=1}^{M} R_i z_i^k$ can also provide good approximations to f_k with error control. In practice, we find that solving either the $N_0 \times N_0$ or the $M \times M$ system in Eq. (4.4) to determine the weights is unstable. However, if $f(t)$ is a damping function, the M significant nodes can be predetermined by retaining only the nodes inside the unit disk. Solving the overdetermined $N \times M$ problem

$$f_k \approx \sum_{i=1}^{M} R_i z_i^k \text{ for } 0 \le k \le N-1 \tag{4.6}$$

by least-square fits is then stable. In addition, although σ should be a c-eigenvalue according to Ref. [49], which can be obtained from the Autonne–Takagi factorization [55], we find that using the singular value from singular value decomposition (SVD) gives equally good approximations.

4.3. ESPRIT Algorithm

4.3.1. Scalar-valued ESPRIT

The ESPRIT algorithm [53, 54] is a method for estimating signal parameters through the rotational invariance property of the Hankel matrix. It utilizes the SVD of an $(N-L) \times (L+1)$ Hankel matrix

$$\boldsymbol{H} = \begin{pmatrix} f(t_0) & f(t_1) & \cdots & f(t_L) \\ f(t_1) & f(t_2) & \cdots & f(t_{L+1}) \\ \vdots & \vdots & \ddots & \vdots \\ f(t_{N-L-1}) & f(t_{N-L}) & \cdots & f(t_{N-1}) \end{pmatrix} \tag{4.7}$$

expressed as

$$\boldsymbol{H} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{W} \ , \tag{4.8}$$

where $\boldsymbol{U}$ and $\boldsymbol{W}$ are unitary matrices of dimensions $(N-L)\times(N-L)$ and $(L+1)\times(L+1)$, respectively. The matrix $\boldsymbol{\Sigma}$ is a rectangular diagonal matrix of size $(N-L)\times(L+1)$, whose diagonal elements are ordered as $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{L+1} \geq 0$. In practice, L is typically chosen between $N/3$ and $N/2$ to minimize variance [52]. Throughout our implementation, we set $L = 2N/5$ unless otherwise stated.

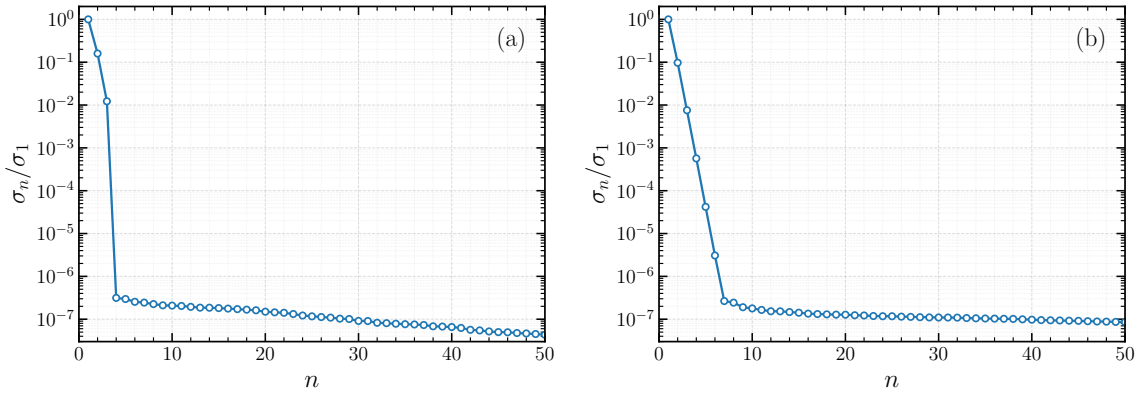

Figure 4.1: Two representative patterns of singular-value distributions at a relative noise level of 10^{-6}. (a) The exact function can be represented exactly as a sum of three exponentials. (b) The exact function cannot be represented exactly as a finite sum of exponentials. The optimal values of M are 3 for panel (a) and approximately 6 for panel (b).

Given a target tolerance ε, the number of exponentials M is estimated as the smallest index satisfying $\sigma_{M+1} < \varepsilon_{\text{abs}}$ or $\sigma_{M+1}/\sigma_1 < \varepsilon_{\text{rel}}$, where ε_{abs} and ε_{rel} are the absolute and relative precisions, respectively. As illustrated in Fig. 4.1, when ε is unknown, one may also attempt to determine ε and σ_{M+1} automatically from the distribution of singular values. This truncation ensures that a minimal number of exponentials is obtained. Next, the nodes z_i are determined as the eigenvalues of the matrix

$$\boldsymbol{F} = (\boldsymbol{W}_0^T)^+ \boldsymbol{W}_1^T \ , \tag{4.9}$$

where T denotes the transpose, $+$ represents the pseudo-inverse, and $\boldsymbol{W}_0$ and $\boldsymbol{W}_1$ are extracted from the matrix $\boldsymbol{W}$ as

$$\boldsymbol{W}_s = \boldsymbol{W}(1 : M, 1+s : L+s), \quad s = 0, 1, \tag{4.10}$$

where both the row and column indices run from 1 to $L+1$. Finally, the weights R_i are obtained by solving the overdetermined least-squares Vandermonde system

$$\begin{pmatrix} f(t_0) \\ f(t_1) \\ \vdots \\ f(t_{N-1}) \end{pmatrix} = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ z_1 & z_2 & \cdots & z_M \\ \vdots & \vdots & \ddots & \vdots \\ z_1^{N-1} & z_2^{N-1} & \cdots & z_M \end{pmatrix} \begin{pmatrix} R_1 \\ R_2 \\ \vdots \\ R_M \end{pmatrix} . \tag{4.11}$$

Thus, all sampled data are approximated within the tolerance ε using a minimal number of exponentials.

The ESPRIT algorithm is closely related to the Matrix Pencil Method [51–53], another well-known algorithm in signal processing. The main difference is that the Matrix Pencil Method employs a QR decomposition of the Hankel matrix rather than an SVD decomposition [56]. For this reason, we do not provide a separate introduction to the Matrix Pencil Method in this thesis.

4.3.2. Matrix-valued extension

4.3.2.1. Description

In cases where the sampled function is matrix-valued, a generalization of ESPRIT to matrix-valued functions [39] can be applied. We consider a generalization of Eq. (4.3) to the following matrix-valued form:

$$\left\| \boldsymbol{f}_k - \sum_{i=1}^{M} \boldsymbol{R}_i z_i^k \right\| \leq \varepsilon \quad \text{for any } 0 \leq k \leq N-1 \, , \tag{4.12}$$

where the norm $\|\cdot\|$ of a matrix $\boldsymbol{f}$ is defined as $\max_{ij} |f_{ij}|$, both $\boldsymbol{f}_k$ and $\boldsymbol{R}_i$ are $n_{\text{orb}} \times n_{\text{orb}}$ matrices, and all elements of $\boldsymbol{R}_i$ share the same node z_i. In this section, we summarize the main aspects of the algorithm. Detailed derivations are provided in the next section for completeness.

Instead of directly operating on matrix-valued objects, we first flatten the $n_{\text{orb}} \times n_{\text{orb}}$ matrices $\boldsymbol{f}_k$ and $\boldsymbol{R}_i$ into column vectors $\vec{f}_k$ and $\vec{R}_i$ of size n_{orb}^2. The generalized ESPRIT method then introduces an auxiliary parameter L, which is typically chosen between $N/3$ and $N/2$ to minimize variance [52], and utilizes the SVD of the $n_{\text{orb}}^2(N-L)$ by

$L+1$ matrix

$$\boldsymbol{H} = \begin{pmatrix} \vec{f}_0 & \vec{f}_1 & \cdots & \vec{f}_L \\ \vec{f}_1 & \vec{f}_2 & \cdots & \vec{f}_{L+1} \\ \vdots & \vdots & \ddots & \vdots \\ \vec{f}_{N-L-1} & \vec{f}_{N-L} & \cdots & \vec{f}_{N-1} \end{pmatrix} \tag{4.13}$$

as

$$\boldsymbol{H} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{W}, \tag{4.14}$$

where $\boldsymbol{U}$ and $\boldsymbol{W}$ are $n_{\text{orb}}^2(N-L)$ by $n_{\text{orb}}^2(N-L)$ and $L+1$ by $L+1$ unitary matrices, respectively, and $\boldsymbol{\Sigma}$ is a rectangular diagonal matrix of size $n_{\text{orb}}^2(N-L)$ by $L+1$ with diagonal elements $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{L+1} \geq 0$.

The number of nodes M is estimated as the smallest number such that σ_{M+1} is below the prescribed precision ε, i.e.,

$$\sigma_{M+1} < \varepsilon_{\text{abs}} \quad \text{or} \quad \sigma_{M+1}/\sigma_1 < \varepsilon_{\text{rel}} \ , \tag{4.15}$$

where ε_{abs} and ε_{rel} are the absolute and relative precisions, respectively, which are limited either by the noise level of $\vec{f}$ or by machine precision. The truncation to the minimum number of relevant singular values ensures that a minimal number of exponentials is obtained, thereby regularizing the continuation problem. The nodes z_i are then obtained as the eigenvalues of the matrix $\boldsymbol{F}$:

$$\boldsymbol{F} = (\boldsymbol{W}_0^T)^+ \boldsymbol{W}_1^T \ , \tag{4.16}$$

where $+$ denotes the pseudo-inverse, T denotes the transpose, and $\boldsymbol{W}_0$ and $\boldsymbol{W}_1$ are obtained from the first M rows of $\boldsymbol{W}$ by deleting the last and first columns, respectively, as mentioned before. Explicitly, they are expressed as

$$\boldsymbol{W}_s = \boldsymbol{W}(1:M, 1+s:L+s), \ s = 0, 1. \tag{4.17}$$

Finally, the vectorized weights $\vec{R}_i$ are recovered by solving the overdetermined least-

squares Vandermonde system

$$\begin{pmatrix} \vec{f}_0^T \\ \vec{f}_1^T \\ \vdots \\ \vec{f}_{N-1}^T \end{pmatrix} = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ z_1 & z_2 & \cdots & z_M \\ \vdots & \vdots & \ddots & \vdots \\ z_1^{N-1} & z_2^{N-1} & \cdots & z_M^{N-1} \end{pmatrix} \begin{pmatrix} \vec{R}_1^T \\ \vec{R}_2^T \\ \vdots \\ \vec{R}_M^T \end{pmatrix} \tag{4.18}$$

and are reshaped into $n_{\text{orb}} \times n_{\text{orb}}$ matrices $\boldsymbol{R}_i$. In this way, all sampled data $\boldsymbol{f}_k$ are approximated within the error tolerance ε by the smallest number of exponentials. By varying k continuously over $[0, N-1]$, we obtain an approximation over the continuous interval in the form of Eq. (4.2),

$$\boldsymbol{f}(t) \approx \sum_{i=1}^{M} \boldsymbol{R}_i \exp\left\{ \frac{\ln z_i}{\Delta t}(t-a) \right\}, \quad t \in [a, b] \, . \tag{4.19}$$

To summarize, the matrix-valued ESPRIT algorithm works as follows:

Algorithm 1 Matrix-valued ESPRIT algorithm

Require: Input data $\{\boldsymbol{f}_0, \boldsymbol{f}_1, \cdots, \boldsymbol{f}_{N-1}\}$ sampled on a uniform grid.

Ensure: A minimal number M, corresponding nodes $\{z_1, \ldots, z_M\}$ and matrix-valued weights $\{\boldsymbol{R}_1, \ldots, \boldsymbol{R}_M\}$.

1: Construct the matrix $\boldsymbol{H}$ according to Eq. (4.13).
2: Compute the SVD of $\boldsymbol{H}$ as described in Eq. (4.14).
3: Determine the truncation rank M using Eq. (4.15).
4: Form the matrix $\boldsymbol{F}$ using Eq. (4.16).
5: Compute the eigenvalues $\{z_1, \ldots, z_M\}$ of $\boldsymbol{F}$.
6: Solve Eq. (4.18) to obtain $\{\boldsymbol{R}_1, \ldots, \boldsymbol{R}_M\}$.

4.3.2.2. Proof

To conclude this chapter, we provide detailed mathematical derivations of the matrix-valued ESPRIT algorithm. The generalization of ESPRIT to matrix-valued systems [39] follows a similar idea to that presented in Ref. [36] as a matrix-valued generalization of the scalar-valued Prony method of Ref. [37].

The objective is to approximate a matrix-valued function using a linear combination of shared exponentials based on its samples on a uniform grid. For simplicity, we flatten

the $n_{\text{orb}} \times n_{\text{orb}}$ matrices into column vectors of length n_{orb}^2 (denoted by $\rightarrow$) and express the approximation as

$$\vec{f}_k = \vec{f}_k^{\text{exact}} + \vec{n}_k \approx \sum_{i=1}^{M} \vec{R}_i z_i^k + \vec{n}_k \text{ for } k = 0, 1, \cdots, N-1 \,, \tag{4.20}$$

where $\vec{f}$ is the sampled signal, $\vec{f}^{\text{exact}}$ is the exact signal, $\vec{n}$ is the noise, z_i is the i-th exponential shared by all matrix elements, and $\vec{R}_i$ are the corresponding weights.

Noise-free case: In the noise-free case, i.e., $\vec{n} \equiv 0$, define the matrix

$$\boldsymbol{H} = \begin{pmatrix} \vec{f}_0 & \vec{f}_1 & \cdots & \vec{f}_L \\ \vec{f}_1 & \vec{f}_2 & \cdots & \vec{f}_{L+1} \\ \vdots & \vdots & \ddots & \vdots \\ \vec{f}_{N-L-1} & \vec{f}_{N-L} & \cdots & \vec{f}_{N-1} \end{pmatrix}_{n_{\text{orb}}^2 (N-L) \times (L+1)} \,. \tag{4.21}$$

If we extract two submatrices from $\boldsymbol{H}$ by deleting the last and first columns, respectively:

$$\boldsymbol{H}_0 = \boldsymbol{H}[:, 1:L] = \begin{pmatrix} \vec{f}_0 & \vec{f}_1 & \cdots & \vec{f}_{L-1} \\ \vec{f}_1 & \vec{f}_2 & \cdots & \vec{f}_L \\ \vdots & \vdots & \ddots & \vdots \\ \vec{f}_{N-L-1} & \vec{f}_{N-L} & \cdots & \vec{f}_{N-2} \end{pmatrix}_{n_{\text{orb}}^2 (N-L) \times L} \,, \tag{4.22}$$

$$\boldsymbol{H}_1 = \boldsymbol{H}[:, 2:(L+1)] = \begin{pmatrix} \vec{f}_1 & \vec{f}_2 & \cdots & \vec{f}_L \\ \vec{f}_2 & \vec{f}_3 & \cdots & \vec{f}_{L+1} \\ \vdots & \vdots & \ddots & \vdots \\ \vec{f}_{N-L} & \vec{f}_{N-L+1} & \cdots & \vec{f}_{N-1} \end{pmatrix}_{n_{\text{orb}}^2 (N-L) \times L} \,, \tag{4.23}$$

then it can be verified that

$$\boldsymbol{H}_0 = \boldsymbol{Z}_1 \boldsymbol{R} \boldsymbol{Z}_2 \tag{4.24}$$

$$\boldsymbol{H}_1 = \boldsymbol{Z}_1 \boldsymbol{R} \boldsymbol{Z}_0 \boldsymbol{Z}_2 \tag{4.25}$$

with the following definitions:

$$\boldsymbol{Z}_1 = \begin{pmatrix} \mathbb{1} & \mathbb{1} & \cdots & \mathbb{1} \\ z_1\mathbb{1} & z_2\mathbb{1} & \cdots & z_M\mathbb{1} \\ \vdots & \vdots & \ddots & \vdots \\ z_1^{N-L-1}\mathbb{1} & z_2^{N-L-1}\mathbb{1} & \cdots & z_M^{N-L-1}\mathbb{1} \end{pmatrix}_{n_{\text{orb}}^2(N-L)\times n_{\text{orb}}^2 M} , \tag{4.26}$$

$$\boldsymbol{R} = \begin{pmatrix} \vec{R}_1 & & & \\ & \vec{R}_2 & & \\ & & \ddots & \\ & & & \vec{R}_M \end{pmatrix}_{n_{\text{orb}}^2 M\times M} , \tag{4.27}$$

$$\boldsymbol{Z}_0 = \begin{pmatrix} z_1 & & & \\ & z_2 & & \\ & & \ddots & \\ & & & z_M \end{pmatrix}_{M\times M} , \tag{4.28}$$

$$\boldsymbol{Z}_2 = \begin{pmatrix} 1 & z_1 & \cdots & z_1^{L-1} \\ 1 & z_2 & \cdots & z_2^{L-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & z_M & \cdots & z_M^{L-1} \end{pmatrix}_{M\times L} , \tag{4.29}$$

where $\mathbb{1}$ in Eq. (4.26) denotes an identity matrix of size $n_{\text{orb}}^2 \times n_{\text{orb}}^2$.

As a result, we have

$$\boldsymbol{H}_1 - \lambda\boldsymbol{H}_0 = \boldsymbol{Z}_1\boldsymbol{R}(\boldsymbol{Z}_0 - \lambda\mathbb{1}_{M\times M})\boldsymbol{Z}_2, \tag{4.30}$$

so that solving for z_i is equivalent to solving the ordinary eigenvalue problem

$$\boldsymbol{H}_0^{+}\boldsymbol{H}_1 - \lambda\mathbb{1} , \tag{4.31}$$

where $+$ represents the pseudo-inverse, as mentioned above.

Noisy case: In the noisy case, we apply the SVD to the matrix $\boldsymbol{H}$ as follows:

$$\boldsymbol{H} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\dagger} \,, \tag{4.32}$$

where $\dagger$ denotes the Hermitian transpose, $\boldsymbol{U}$ is an $n_{\text{orb}}^2(N-L) \times n_{\text{orb}}^2(N-L)$ matrix, $\boldsymbol{\Sigma} = \text{diag}(\sigma_1, \sigma_2, \ldots, \sigma_{L+1})$ is an $n_{\text{orb}}^2(N-L) \times (L+1)$ matrix, and $\boldsymbol{V} = [v_1, \ldots, v_{L+1}]$ is an $(L+1) \times (L+1)$ matrix. To filter the noise, we truncate at

$$\sigma_{M+1} \approx \text{absolute noise level} \quad \text{or} \quad \sigma_{M+1}/\sigma_1 \approx \text{relative noise level} \,. \tag{4.33}$$

Thus, we have

$$\boldsymbol{H}_0 = \boldsymbol{U}\boldsymbol{\Sigma}'\boldsymbol{V}_0'^{\dagger} \tag{4.34}$$

$$\boldsymbol{H}_1 = \boldsymbol{U}\boldsymbol{\Sigma}'\boldsymbol{V}_1'^{\dagger} \,, \tag{4.35}$$

where $\boldsymbol{\Sigma}' = \text{diag}(\sigma_1, \sigma_2, \cdots, \sigma_M)$ is an $n_{\text{orb}}^2(N-L) \times M$ matrix, and $\boldsymbol{V}_0'$ and $\boldsymbol{V}_1'$ are obtained from $\boldsymbol{V}' = [v_1, v_2, \cdots, v_M]$ by deleting the last and first rows, respectively. Specifically, $\boldsymbol{V}_0' = \boldsymbol{V}'[1:L, 1:M]$ and $\boldsymbol{V}_1' = \boldsymbol{V}'[2:(L+1), 1:M]$. As a result, solving for the eigenvalues of $\boldsymbol{H}_1 - \lambda\boldsymbol{H}_0$ is equivalent to solving for the eigenvalues of $\boldsymbol{V}_1'^{\dagger} - \lambda\boldsymbol{V}_0'^{\dagger}$, which ultimately reduces to solving for the eigenvalues of

$$\boldsymbol{V}_1'^{\dagger}(\boldsymbol{V}_0'^{\dagger})^{+} - \lambda \mathbb{1}_{M\times M} \,, \tag{4.36}$$

which is equivalent to solving for the eigenvalues of Eq. (4.16) since $\boldsymbol{W} = \boldsymbol{V}^{\dagger}$ and the eigenvalues of an arbitrary matrix $\boldsymbol{A}$ are the same as those of $\boldsymbol{A}^T$.

After extracting the nodes $z_1, \ldots, z_M$ from Eq. (4.36), the corresponding weights can be computed from

$$\begin{pmatrix} \vec{f}_0^{\,T} \\ \vec{f}_1^{\,T} \\ \vdots \\ \vec{f}_{N-1}^{\,T} \end{pmatrix}_{N\times n_{\text{orb}}^2} = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ z_1 & z_2 & \cdots & z_M \\ \vdots & \vdots & \ddots & \vdots \\ z_1^{N-1} & z_2^{N-1} & \cdots & z_M^{N-1} \end{pmatrix}_{N\times M} \begin{pmatrix} \vec{R}_1^{\,T} \\ \vec{R}_2^{\,T} \\ \vdots \\ \vec{R}_M^{\,T} \end{pmatrix}_{M\times n_{\text{orb}}^2} \,, \tag{4.37}$$

which can then be reshaped into $n_{\text{orb}} \times n_{\text{orb}}$ matrices $\boldsymbol{R}_i$.

Further generalization: Before concluding this section, we point out that similar ideas can be applied to generalize other Prony-like methods. Furthermore, the generalization is not limited to matrix-valued cases. Implementations for both vector-valued and tensor-valued cases are straightforward, as only the vectorized form is used in both the Hankel matrix (4.21) and the Vandermonde system (4.37).

Chapter 5

Complex Pole Representation for Green's Functions

In this chapter, we provide a detailed analysis of the complex pole representation for Green's functions, which motivates the developments in the following chapters.

5.1. $G(z)$ on the Complex Plane

5.1.1. Complex pole representation

As discussed in Ch. 3, Matsubara, retarded, and advanced Green's functions can be uniformly expressed in terms of a generic Green's function defined on the whole complex plane; see Eq. (3.30). For simplicity, we first focus on the single-orbital case without spin-orbit coupling. In this case, $G(z)$ is scalar-valued. According to Eq. (3.30), $G(z)$ has singularities only on the real axis. At any other point z with $\mathrm{Im}z \neq 0$, $G(z)$ is analytic. For a finite-size system, there are only finitely many real-valued simple poles, corresponding to delta peaks in the spectral function, and these can be represented numerically in a straightforward way. However, the more interesting cases, which are the main focus of this thesis, are infinitely large systems. In practice, such systems may arise, for example, by imposing periodic boundary conditions on a lattice or by coupling an impurity to an infinitely large bath. In these cases, the singularities are usually continuous. This means that there are infinitely many real poles, and the sum over poles in Eq. (3.30) should be replaced by an integral, leading to branch cuts on the

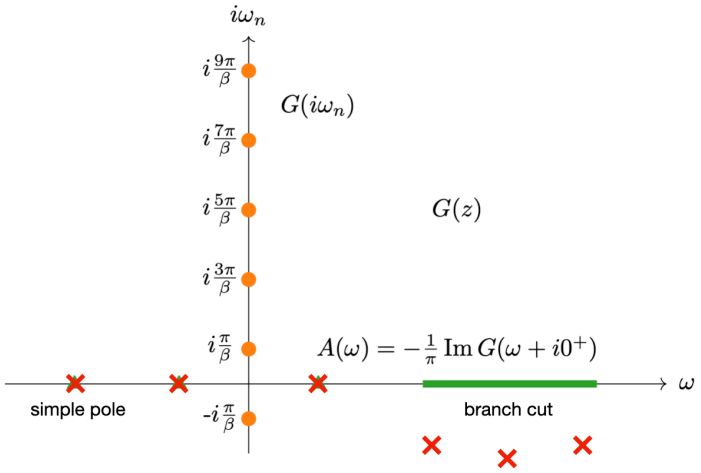

Figure 5.1: Illustration of $G(z)$ on the complex plane. Green triangles and lines represent the simple poles and branch cuts in the exact representation, while red crosses indicate the locations of complex poles used to approximate $G(z)$ in the upper half plane.

real-frequency axis. Consequently, the spectral function becomes broadened. Although poles on the real axis no longer provide good approximations, one may expect that a small number of poles away from the real axis can still approximate the spectral function well. This expectation is also motivated by the concept of scattering resonances [57].

Specifically, we construct the following ansatz to approximate $G(z)$ in the upper half plane:

$$G(z) = \sum_{l=1}^{M} \frac{A_l}{z - \xi_l} \quad \text{for} \quad \mathrm{Im}z > 0\ , \tag{5.1}$$

where A_l and ξ_l are now complex-valued. The retarded Green's function can be obtained from $G^{\mathrm{Ret}}(\omega) = G(\omega + i0^+)$. One can verify that all poles with $\mathrm{Im}\xi_l \leq 0$ give the correct Kramers–Kronig relation (3.37) for $G^{\mathrm{Ret}}(\omega)$, whereas poles with $\mathrm{Im}\xi_l > 0$ do not. Therefore, the poles must be restricted to the lower half plane, including the real axis. Fig. 5.1 illustrates the locations of the actual singularities and the complex poles in the approximant (5.1) for $\mathrm{Im}z > 0$.

As will be illustrated in Ch. 8, once complex-valued pole weights are allowed, continuous spectra can indeed be approximated by a small number of poles with controlled error. Here, we provide some qualitative justification for this approximation. We note

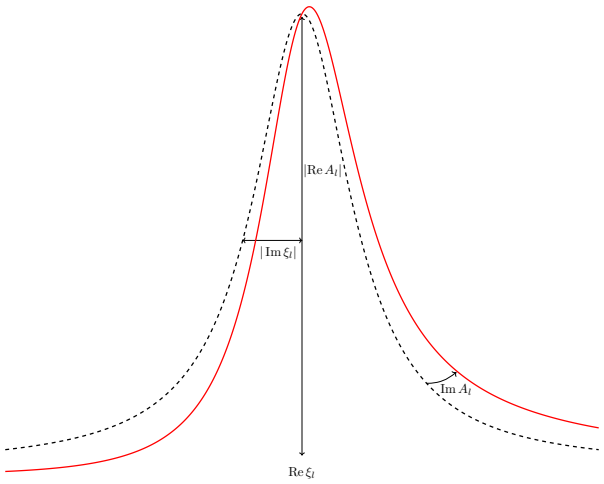

Figure 5.2: Illustration of peaks generated by complex-valued A_l and ξ_l. The black dashed line represents an ordinary Lorentzian peak ($\mathrm{Im}\, A_l = 0$), while the red solid line represents the corresponding asymmetric peak ($\mathrm{Im}\, A_l \neq 0$). Here, $\mathrm{Re}\, \xi_l$ determines the peak center, and $\mathrm{Im}\, \xi_l$ controls the half-width. For fixed width, the peak height is controlled by $\mathrm{Re}\, A_l$, which scales the peak in the vertical direction, while the asymmetric component is controlled by $\mathrm{Im}\, A_l$.

that a broadened peak has four main features: the location of its center, its width, its height, and its asymmetric component with respect to the center. Substituting Eq. (5.1) into Eq. (3.6), we obtain

$$A(\omega) = \sum_l \frac{1}{\pi} \frac{(\mathrm{Re}\, A_l)(-\mathrm{Im}\, \xi_l) + (-\mathrm{Im}\, A_l)(\omega - \mathrm{Re}\, \xi_l)}{(\omega - \mathrm{Re}\, \xi_l)^2 + (\mathrm{Im}\, \xi_l)^2} \,. \tag{5.2}$$

As shown in Fig. 5.2, each part of A_l and ξ_l has a specific role: $\mathrm{Re}\, \xi_l$ determines the location of the center, $\mathrm{Im}\, \xi_l$ determines the degree of broadening, i.e., the width, $\mathrm{Re}\, A_l$ scales the peak and thus controls its height, and finally, $\mathrm{Im}\, A_l$ determines the asymmetric part of the peak, which describes the deviation from a standard Lorentzian peak. Since the main features of a peak can all be captured by a complex-valued weight and location, it is reasonable to expect that continuous spectral functions can be well approximated

by a small number of such peaks, represented by poles with complex-valued weights and locations. We denote these peaks as *generalized Lorentzians*. Note that allowing $\mathrm{Im}A_l \neq 0$ is essential here. If we set $\mathrm{Im}A_l = 0$, the peaks reduce to ordinary Lorentzians, which are not well suited for approximating spectral functions. One reason is that Lorentzians have slowly decaying tails. The broader the Lorentzian peak is, the more slowly its tail decays, and therefore it cannot efficiently approximate spectral functions with broadened peaks and fast-decaying tails. In contrast, allowing $\mathrm{Im}A_l \neq 0$ introduces negative components into each individual peak. Although a single such peak does not necessarily preserve the positivity of the spectral function, a sum of such peaks can still do so. Moreover, these peaks can cancel one another in the long-tail region, so that spectral functions with broadened peaks and fast-decaying features can be accurately represented.

A similar analysis can be carried out for $G(z)$ with $\mathrm{Im}z < 0$, where only poles in the upper half plane are allowed. Combining the two cases, the full expression of $G(z)$ on the whole complex plane can be written as

$$G(z) = \theta(\mathrm{Im}z) \sum_{l=1}^{M} \frac{A_l^{(\mathrm{dn})}}{z - \xi_l^{(\mathrm{dn})}} + \theta(-\mathrm{Im}z) \sum_{l=1}^{M} \frac{A_l^{(\mathrm{up})}}{z - \xi_l^{(\mathrm{up})}} \,, \tag{5.3}$$

where $\xi_l^{(\mathrm{dn})}$ and $\xi_l^{(\mathrm{up})}$ represent poles in the lower and upper half planes, respectively, and the up-down symmetry (3.34) enforces $[A_l^{(\mathrm{up})}]^* = A_l^{(\mathrm{dn})}$ and $[\xi_l^{(\mathrm{up})}]^* = \xi_l^{(\mathrm{dn})}$.

Before concluding this section, we mention that although the complex pole representation works well for $G(z)$, it does not work well for $G(\tau)$ because of the step function in Eq. (5.3). More specifically, although real poles lead $G(\tau)$ to a sum of real exponentials; see Eq. (3.41), complex poles do not lead to a sum of complex exponentials. Suppose one could use a sum of complex exponentials to fit $G(\tau)$. The symmetry would then require that, after Fourier transformation, they correspond to complex-conjugate pairs of complex poles. However, doing so would violate the Kramers–Kronig relation (3.37), and evaluating the spectral function from Eq. (3.6) would give zero as long as the poles have nonzero imaginary parts. This invalidates the complex pole representation for $G(\tau)$.

5.1.2. Analysis on the complex pole representation

In this section, we examine the complex pole representation (5.3) in more detail. We begin with *causality*: $G^{\mathrm{Ret}}(t)$ vanishes for $t < 0$, whereas $G^{\mathrm{Adv}}(t)$ vanishes for $t > 0$. This property is automatically guaranteed in the complex pole representation. According to the residue theorem, to evaluate $G^{\mathrm{Ret}}(t)$ for $t < 0$ or $G^{\mathrm{Adv}}(t)$ for $t > 0$, one must close the real-frequency contour with an infinitely large semicircle in the upper half plane or lower half plane, respectively. Since $G^{\mathrm{Ret}}(\omega)$ and $G^{\mathrm{Adv}}(\omega)$ contain poles only in the lower and upper half planes, respectively, the corresponding contour integral vanishes. Therefore, causality is automatically satisfied.

Next, we take the fermionic case as an example to examine the positivity of the spectral function in this complex pole representation. According to Ch. 3, the spectral function $A(\omega)$ is non-negative for any frequency ω. This property is readily achieved in the real pole representation (3.38), since it is sufficient to require each nonzero weight $A_l^{(r)} > 0$. This representation is also robust against small perturbations, since as long as the perturbation satisfies $|\Delta A_l^{(r)}| < A_l^{(r)}$, each peak remains positive. The situation becomes more complicated in the complex pole representation (5.2), because different peaks may overlap, and although a single peak may contain negative parts, the total spectral function can still be non-negative everywhere on the real-frequency axis. However, positivity is no longer robust against small perturbations.

Let us demonstrate this point with a simple example. Suppose there are two poles $\xi_1^{(\mathrm{dn})} = -x - iy$, $\xi_2^{(\mathrm{dn})} = x - iy$, $A_1^{(\mathrm{dn})} = a - ib$ and $A_2^{(\mathrm{dn})} = a + ib$, with $x,\ y,\ a,\ b$ being positive numbers. It is straightforward to verify that each peak is not always positive, but the total spectral function

$$A(\omega) = \frac{2}{\pi} \frac{(ay - bx)\omega^2 + (ay + bx)(x^2 + y^2)}{[(\omega + x)^2 + y^2][(\omega - x)^2 + y^2]} \tag{5.4}$$

is always positive as long as $\frac{b}{a} \leq \frac{y}{x}$. However, if, for example, we perturb $A_2^{(\mathrm{dn})}$ to $(a + \Delta a) + i(b + \Delta b)$, then the spectral function changes to

$$A(\omega) = \frac{1}{\pi} \frac{-\Delta b \times \omega^3 + \mathcal{O}(\omega^2)}{[(\omega + x)^2 + y^2][(\omega - x)^2 + y^2]} \,. \tag{5.5}$$

Since ω^3 is an odd function, as long as $\Delta b \neq 0$, there always exists a frequency ω_0

such that $A(\omega_0) < 0$. Note that this conclusion is independent of the magnitude of Δb. Thus, positivity of $A(\omega)$ in the complex pole representation is not robust. In simulations, since we cannot distinguish $\Delta b = 0$ from $0 < |\Delta b| < \varepsilon$, where ε is the machine precision, we cannot determine whether $A(\omega)$ is strictly positive or contains potentially very small negative parts. This suggests that there is no numerically robust criterion for determining whether a given set of complex poles $\{A_l, \xi_l\}$ produces a strictly positive spectral function.

The above discussion shows that it is not meaningful, for example, to fix $\{\xi_l\}$ and optimize $\{A_l\}$ so that $A(\omega)$ is strictly positive on an optimization grid, because $A(\omega)$ may easily become negative at other points. Based on this observation, we only aim to find a spectral function that is close to the exact spectral function, $|A^{\text{approx}}(\omega) - A^{\text{exact}}(\omega)| < \varepsilon$, regardless of whether it is strictly positive. Here, ε is a small positive number controlling the error. As long as this approximation is controlled, the spectral function represented by the complex pole representation is *visually positive*, i.e., it cannot be distinguished by eye from a strictly positive spectral function. Since we do not wish to impose additional restrictions to enforce positivity, we seek a procedure that *automatically* leads to a visually positive spectral function. To achieve this, we use the *maximum modulus principle* in complex analysis, which states that if two functions are analytic in a connected open set, their maximal absolute difference must occur on the boundary. Taking the upper half plane as the connected open set, this implies that if we want the spectral function to be controlled, the approximated $G(z)$ must also be controlled at all points in the upper half plane. Suppose we are only able to sample $G(z)$ on a discrete grid, such as the Matsubara points. Then, to obtain a controlled approximation for the spectral function, one must ensure that $|G^{\text{approx}}(iy) - G^{\text{exact}}(iy)|$ is controlled not only at $iy = i\omega_\nu$, but also between adjacent sampling points, i.e., for $iy \in (i\omega_\nu, i\omega_{\nu+1})$. If this condition is not satisfied, the recovered spectral function cannot be controlled. At first glance, this condition may seem impossible to achieve, since one can always add another function to $G(z)$ whose zeros are located at the sampling points. This would leave the sampling data unchanged while modifying $G(z)$ between the sampling points. Does this mean that the spectral function can never be recovered in a controlled way? No, because we only care about those $G(z)$'s with *physical meaning*.

More specifically, for fermions, we have

$$-\mathrm{Im}G(iy) = \sum_l \frac{y}{y^2 + \xi_l^{(r)2}} A_l^{(r)} \ . \tag{5.6}$$

Since $A_l^{(r)}$ are positive, the imaginary part of $G(iy)$ with $y > 0$ is simply a sum of positive contributions in the same direction, with no cancellations. This implies a sense of "least oscillation". A similar statement holds for bosons, since in that case we have

$$-\mathrm{Re}G(iy) = \sum_l \frac{|\xi_l^{(r)}|}{y^2 + \xi_l^{(r)2}} \mathrm{sgn}(\xi_l^{(r)}) A_l^{(r)} \ , \tag{5.7}$$

and $\mathrm{sgn}(\xi_l^{(r)}) A_l^{(r)}$ are positive. This observation motivates a technique for determining which Matsubara points are undersampled and which are oversampled. Although the sampling may be insufficient near the initial frequency, the slowly decaying nature of $G(z)$, c.f. Eq. (3.35), ensures that the Matsubara points quickly become oversampled as $i\omega_\nu$ increases. For these oversampled points, as shown later, one can achieve a controlled approximation between adjacent sampling points.

Before concluding this section, we mention another common concern for real-frequency approximations. Sometimes the spectral function contains singularities, causing any representation composed of smooth functions, including the complex pole representation, to fail exactly at the singular points. However, as we show in later chapters, when complex poles are used to approximate *spectral moments instead of spectral shapes*, the approximated spectral function automatically wraps around the singularity and reproduces equally good spectral moments. Therefore, calculations based on these complex poles can still work accurately.

The considerations in this section serve as the main motivations for the projects presented in later chapters, where more technical details will be discussed.

5.1.3. Matrix-valued extension

Before concluding the discussion of the complex pole representation of the generalized Green's function on the whole complex plane, we present its matrix-valued extension for reference and for later convenience.

For a system with multiple orbitals, $\boldsymbol{G}(z)$ becomes a matrix. In this case, one can, of course, approximate each matrix element using Eq. (5.3). However, in practice, we find that it is usually beneficial to allow different elements to share the same set of complex poles while having their own complex-valued weights. This leads to the ansatz

$$\boldsymbol{G}(z) = \theta(\mathrm{Im}z) \sum_{l=1}^{M} \frac{\boldsymbol{A}_l^{(\mathrm{dn})}}{z - \xi_l^{(\mathrm{dn})}} + \theta(-\mathrm{Im}z) \sum_{l=1}^{M} \frac{\boldsymbol{A}_l^{(\mathrm{up})}}{z - \xi_l^{(\mathrm{up})}} \ , \tag{5.8}$$

where the symmetry relations become $[\boldsymbol{A}_l^{(\mathrm{up})}]^\dagger = \boldsymbol{A}_l^{(\mathrm{dn})}$ and $[\xi_l^{(\mathrm{up})}]^* = \xi_l^{(\mathrm{dn})}$. The analysis discussed in Sec. 5.1.2 still largely applies, with positive $\mathrm{sgn}(\xi_l^{(r)})A_l^{(r)}$ or $A_l^{(r)}$ replaced by positive semidefinite $\mathrm{sgn}(\xi_l^{(r)})\boldsymbol{A}_l^{(r)}$ or $\boldsymbol{A}_l^{(r)}$, respectively.

The retarded and advanced Green's functions can then be written explicitly as

$$\boldsymbol{G}^{\mathrm{Ret}}(\omega) = \sum_{l=1}^{M} \frac{\boldsymbol{A}_l^{(\mathrm{dn})}}{\omega - \xi_l^{(\mathrm{dn})}} \ , \qquad \boldsymbol{G}^{\mathrm{Ret}}(t) = -i\theta(t) \sum_{l=1}^{M} \boldsymbol{A}_l^{(\mathrm{dn})} e^{-i\xi_l^{(\mathrm{dn})} t} \ , \tag{5.9}$$

$$\boldsymbol{G}^{\mathrm{Adv}}(\omega) = \sum_{l=1}^{M} \frac{\boldsymbol{A}_l^{(\mathrm{up})}}{\omega - \xi_l^{(\mathrm{up})}} \ , \qquad \boldsymbol{G}^{\mathrm{Adv}}(t) = i\theta(-t) \sum_{l=1}^{M} \boldsymbol{A}_l^{(\mathrm{up})} e^{-i\xi_l^{(\mathrm{up})} t} \ . \tag{5.10}$$

5.2. Other Green's Functions

Following a similar idea, other Green's functions can also be well approximated in terms of complex poles and complex weights:

$$\boldsymbol{G}^{>}(\omega) = \sum_{l=1}^{M_1} \left(\frac{\boldsymbol{A}_l^{>(\mathrm{dn})}}{\omega - \xi_l^{>(\mathrm{dn})}} + \frac{\boldsymbol{A}_l^{>(\mathrm{up})}}{\omega - \xi_l^{>(\mathrm{up})}} \right) \ , \tag{5.11}$$

$$\boldsymbol{G}^{<}(\omega) = \sum_{l=1}^{M_2} \left(\frac{\boldsymbol{A}_l^{<(\mathrm{dn})}}{\omega - \xi_l^{<(\mathrm{dn})}} + \frac{\boldsymbol{A}_l^{<(\mathrm{up})}}{\omega - \xi_l^{<(\mathrm{up})}} \right) \ , \tag{5.12}$$

$$\boldsymbol{G}^{T}(\omega) = \sum_{l=1}^{M_1} \frac{\boldsymbol{A}_l^{>(\mathrm{dn})}}{\omega - \xi_l^{>(\mathrm{dn})}} + \sum_{l=1}^{M_2} \frac{\boldsymbol{A}_l^{<(\mathrm{up})}}{\omega - \xi_l^{<(\mathrm{up})}} \ , \tag{5.13}$$

$$\boldsymbol{G}^{\tilde{T}}(\omega) = \sum_{l=1}^{M_2} \frac{\boldsymbol{A}_l^{<(\mathrm{dn})}}{\omega - \xi_l^{<(\mathrm{dn})}} + \sum_{l=1}^{M_1} \frac{\boldsymbol{A}_l^{>(\mathrm{up})}}{\omega - \xi_l^{>(\mathrm{up})}} \ , \tag{5.14}$$

$$\boldsymbol{G}^K(\omega) = \sum_{l=1}^{M_3} \left(\frac{\boldsymbol{A}_l^{K(\mathrm{dn})}}{\omega - \xi_l^{K(\mathrm{dn})}} + \frac{\boldsymbol{A}_l^{K(\mathrm{up})}}{\omega - \xi_l^{K(\mathrm{up})}} \right) , \tag{5.15}$$

where $[\boldsymbol{A}_l^{\bullet(\mathrm{up})}]^\dagger = -\boldsymbol{A}_l^{\bullet(\mathrm{dn})}$ and $[\xi_l^{\bullet(\mathrm{up})}]^* = \xi_l^{\bullet(\mathrm{dn})}$ for any $\bullet \in \{>,\ <,\ K\}$. In deriving the expressions for $\boldsymbol{G}^T(\omega)$ and $\boldsymbol{G}^{\tilde{T}}(\omega)$, we have used the relations $\boldsymbol{G}^T(t) = \theta(t)\boldsymbol{G}^>(t) + \theta(-t)\boldsymbol{G}^<(t)$ and $\boldsymbol{G}^{\tilde{T}}(t) = \theta(t)\boldsymbol{G}^<(t) + \theta(-t)\boldsymbol{G}^>(t)$, together with the fact that only $\{\boldsymbol{A}_l^{\bullet(\mathrm{dn})}, \xi_l^{\bullet(\mathrm{dn})}\}$ contributes to $\boldsymbol{G}^\bullet(t)$ for $t > 0$, while only $\{\boldsymbol{A}_l^{\bullet(\mathrm{up})}, \xi_l^{\bullet(\mathrm{up})}\}$ contributes for $t < 0$. When $\xi_l^{>(\mathrm{dn})}$, $\xi_l^{<(\mathrm{dn})}$ and $\xi_l^{K(\mathrm{dn})}$ all coincide, their weights satisfy $\boldsymbol{A}_l^{K(\mathrm{dn})} = \boldsymbol{A}_l^{>(\mathrm{dn})} + \boldsymbol{A}_l^{<(\mathrm{dn})}$, or equivalently $\boldsymbol{A}_l^{K(\mathrm{up})} = \boldsymbol{A}_l^{>(\mathrm{up})} + \boldsymbol{A}_l^{<(\mathrm{up})}$, according to the relation $\boldsymbol{G}^K(t) = \boldsymbol{G}^>(t) + \boldsymbol{G}^<(t)$.

One can also perform a Fourier transform to obtain the corresponding expressions in the real-time domain:

$$\boldsymbol{G}^>(t) = -i\theta(t) \sum_{l=1}^{M_1} \boldsymbol{A}_l^{>(\mathrm{dn})} e^{-i\xi_l^{>(\mathrm{dn})}t} + i\theta(-t) \sum_{l=1}^{M_1} \boldsymbol{A}_l^{>(\mathrm{up})} e^{-i\xi_l^{>(\mathrm{up})}t} , \tag{5.16}$$

$$\boldsymbol{G}^<(t) = -i\theta(t) \sum_{l=1}^{M_2} \boldsymbol{A}_l^{<(\mathrm{dn})} e^{-i\xi_l^{<(\mathrm{dn})}t} + i\theta(-t) \sum_{l=1}^{M_2} \boldsymbol{A}_l^{<(\mathrm{up})} e^{-i\xi_l^{<(\mathrm{up})}t} , \tag{5.17}$$

$$\boldsymbol{G}^T(t) = -i\theta(t) \sum_{l=1}^{M_1} \boldsymbol{A}_l^{>(\mathrm{dn})} e^{-i\xi_l^{>(\mathrm{dn})}t} + i\theta(-t) \sum_{l=1}^{M_2} \boldsymbol{A}_l^{<(\mathrm{up})} e^{-i\xi_l^{<(\mathrm{up})}t} , \tag{5.18}$$

$$\boldsymbol{G}^{\tilde{T}}(t) = -i\theta(t) \sum_{l=1}^{M_2} \boldsymbol{A}_l^{<(\mathrm{dn})} e^{-i\xi_l^{<(\mathrm{dn})}t} + i\theta(-t) \sum_{l=1}^{M_1} \boldsymbol{A}_l^{>(\mathrm{up})} e^{-i\xi_l^{>(\mathrm{up})}t} , \tag{5.19}$$

$$\boldsymbol{G}^K(t) = -i\theta(t) \sum_{l=1}^{M_3} \boldsymbol{A}_l^{K(\mathrm{dn})} e^{-i\xi_l^{K(\mathrm{dn})}t} + i\theta(-t) \sum_{l=1}^{M_3} \boldsymbol{A}_l^{K(\mathrm{up})} e^{-i\xi_l^{K(\mathrm{up})}t} . \tag{5.20}$$

As a result, for an infinitely large system, all real-axis Green's functions can be compactly expressed as sums of a small number of complex poles with shared pole locations. They can therefore be incorporated into real-axis simulations, reducing the configuration space from infinitely many degrees of freedom to a small finite number of degrees of freedom while still producing results of comparable accuracy within the prescribed precision ε. This has promising applications to general real-axis simulations and serves as one of the main motivations of Chs. 8 and 9.

Chapter 6

Minimal Pole Representation and Controlled Analytic Continuation

This chapter is based on *Lei Zhang and Emanuel Gull, "Minimal Pole Representation and Controlled Analytic Continuation of Matsubara Response Functions", Phys. Rev. B 110, 035154 (2024)* [38].

6.1. Introduction

Quantum field theory simulations at finite temperature are typically performed on the imaginary axis [3]. In a post-processing step, real-frequency information is obtained via analytic continuation for physical interpretation. Simulations that require continuation range from perturbative calculations [7–9] to lattice [10] and continuous-time [12] quantum Monte Carlo and lattice QCD [58–60] simulations, as well as algorithms for the simulation of bosonic systems [61] including He [32, 62], supersolids [63], and warm dense matter [64].

Due to the ill-conditioned nature of the numerical analytic continuation (NAC) step [13], a variety of methods have been developed. Among these are Padé [14] continued fraction fits of Matsubara data [15, 19, 65–68], an interpolation with Nevanlinna functions [16, 17], the Maximum Entropy (MaxEnt) method [13, 18, 20–23, 69–73], sparse modeling [30, 74], stochastic analytic continuation (SAC) and variants [24–31], genetic algorithms and machine learning [32–34], causal projections [35] and Prony fits [36, 37]. In all of these methods, it is difficult in practice to systematically converge the spectral function, even given high-precision Matsubara data.

In this chapter, we revisit the continuation problem from the perspective of a compact low-rank representation of response functions in terms of a pole expansion that approximates Matsubara data within a predetermined precision ε. Remarkably, as we show below, the spectral function systematically converges to the exact answer as the precision of the Matsubara fit is increased. Even 'difficult' spectral functions containing both sharp and smooth features at low and at high energies are well approximated.

The method is generally applicable to all response functions, including diagonal and off-diagonal fermionic and bosonic response functions of continuous and discrete systems. Examining the application of the methodology to data polluted with stochastic noise we find, similarly, that a fit to within the known precision of the input data results in physically reasonable spectral functions that are systematically improved as the uncertainty on the Matsubara axis is reduced.

This chapter is organized as follows. In Sec. 6.2, we introduce the theory of the minimal pole representation in three steps: the approximation to the Matsubara input in Sec. 6.2.1, the holomorphic mapping in Sec. 6.2.2, and the extraction of pole information in Sec. 6.2.3. The numerical results are presented in Sec. 6.3. We demonstrate the error control of our method in Sec. 6.3.1. Then, we analyze the dependence of the performance on system temperature and the number of available data points in Sec. 6.3.2 and Sec. 6.3.3, respectively. In Sec. 6.3.4, we test our method on challenging examples with both sharp and broadened features, followed by an illustration of the method's versatility in Sec. 6.3.5. In Sec. 6.3.6, we show its robustness to noise. Finally, Sec. 6.4 contains a concluding discussion.

6.2. Theory

As mentioned in Ch. 5, we construct an approximation of Matsubara data in the upper half of the complex plane by

$$G(z) = \sum_{l=1}^{M} \frac{A_l}{z - \xi_l}, \tag{6.1}$$

where the $\xi_l \in \mathbb{C}$ denote M pole locations in the lower half of the plane and $A_l \in \mathbb{C}$ the corresponding complex weights, in three steps. First, we approximate Matsubara

data on a finite interval of the non-negative imaginary axis using Prony's approximation method [47, 49]. Second, we map this interval onto the unit circle using a holomorphic mapping and evaluate the moments of the approximated function numerically. Finally, we use Prony's approximation for a second time to extract a compact representation in terms of pole weights and locations, map the poles back onto the original domain, and evaluate the spectral function.

Prony's method has previously been used to study the analytic continuation problem [36, 37]. The major differences to this work are that Ref. [37] employs a different approximation procedure, either a causal projection onto a finite real-axis grid or a spline interpolation, and different grids and maps, as well as a different solution method of the Prony problem. The methodology does not yield the systematic error control observed here.

6.2.1. Approximation

Our input data consists of an odd number $N_\omega = 2N + 1$ of Matsubara points $G(i\omega_n)$ that are uniformly spaced, starting from a non-negative frequency ω_{n_0} with spacing Δn, i.e., $\{i\omega_{n_0}, i\omega_{n_0+\Delta n}, \cdots, i\omega_{n_0+(N_\omega-1)\Delta n}\}$.

As mentioned in Ch. 4, Prony's interpolation method [47] interpolates G_k as a sum of exponentials $G_k = \sum_{i=0}^{N-1} w_i \gamma_i^k$, where $G_k = G(i\omega_{n_0+k\Delta n})$, $0 \leq k \leq 2N$, w_i denote complex weights and γ_i corresponding nodes. Although Prony's interpolation method is unstable [48], we can employ a Prony approximation [49] of G between $i\omega_{n_0}$ and $i\omega_{n_0+(N_\omega-1)\Delta n}$. For physical Matsubara functions, which decay in magnitude to zero for $i\omega_n \to i\infty$, only $K \propto \log(1/\varepsilon)$ out of all N nodes in the Prony approximation have weights $|w_i| > \varepsilon$ [49]. More importantly, K significant nodes w_i can be predetermined [49] such that the solution to the overdetermined problem for finding weights w_i is stable and yields an accurate solution to the Prony approximation problem

$$\left| G_k - \sum_{i=0}^{K-1} w_i \gamma_i^k \right| \leq \varepsilon \text{ for all } 0 \leq k \leq 2N \tag{6.2}$$

for a predefined tolerance $\varepsilon > 0$ via singular value decomposition. By varying k continuously over the interval $[0, 2N]$, we obtain an approximation of Matsubara data on the continuous interval $[i\omega_{n_0}, i\omega_{n_0+(N_\omega-1)\Delta n}]$. This form of approximation employs a

minimum number of exponential sums and is essential for regularizing the problem.

6.2.2. Holomorphic mapping

We then apply a holomorphic transform $g(z)$, which is a combination of linear transform and an inverse Joukowsky transform [75] and is illustrated in Fig. 6.1, to map the complex plane to the closed unit disk $\bar{D}$.

$$\begin{cases} w &= g(z) &= z_\mathrm{s} - \sqrt{z_\mathrm{s}^2 + 1} \text{ with } z_\mathrm{s} = \frac{z - i\omega_\mathrm{m}}{\Delta\omega_\mathrm{h}} \\ z &= g^{-1}(w) &= \frac{\Delta\omega_\mathrm{h}}{2}(w - \frac{1}{w}) + i\omega_\mathrm{m} \end{cases}, \tag{6.3}$$

where $\omega_\mathrm{m} = (\omega_{n_0} + \omega_{n_0+(N_\omega-1)\Delta n})/2$ is the frequency in the middle of the approximated interval, $\Delta\omega_\mathrm{h} = (\omega_{n_0+(N_\omega-1)\Delta n} - \omega_{n_0})/2$ is half of the segment length, and the branch of the square root in the first equation is chosen such that $|w| \leq 1$. The approximated Matsubara interval forms the unit circle, with $g(i\omega_{n_0}) = -i$, $g(i\omega_{n_0+(N_\omega-1)\Delta n}) = +i$, and any other point splits into two copies with identical y values. The real axis is mapped onto a closed contour contained in the unit disk with ∞ mapped to the origin.

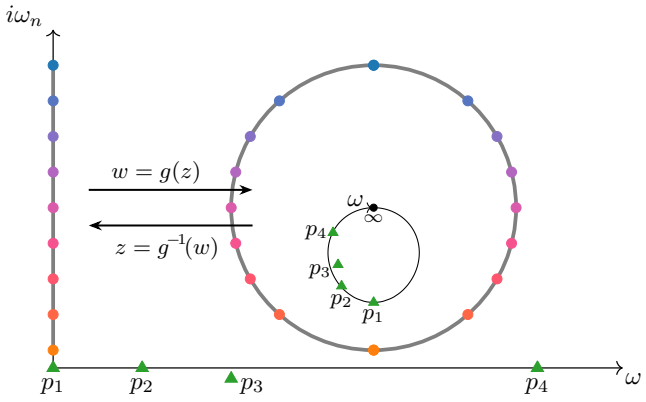

Figure 6.1: Holomorphic functions $g(z)$ and $g^{-1}(w)$ mapping the complex plane to the unit disk and an interval on the imaginary axis to the unit circle. Also shown are points on and near the real axis as triangles, along with their image under g.

Since the transformed response function $\tilde{G}(w)$ corresponds to Eq. (6.1) as $\tilde{G}(w) =$

$G(z)$ and takes the form

$$\tilde{G}(w) = \sum_{l=1}^{M} \frac{\tilde{A}_l}{w - \tilde{\xi}_l} + \text{ analytic part }, \tag{6.4}$$

the integrals over the unit circle

$$h_k := \frac{1}{2\pi i} \int_{\partial \bar{D}} \tilde{G}(w) w^k dw \tag{6.5}$$

yield its moments and, via the residue theorem, pole information [36, 37]:

$$h_k = \sum_l \tilde{A}_l \tilde{\xi}_l^k, \ k \geq 0 \, . \tag{6.6}$$

Additional simplification of Eq. (6.5) yields

$$h_k = \begin{cases} \frac{i}{\pi} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} G(i(\omega_\mathrm{m} + \Delta\omega_\mathrm{h} \sin\theta)) \sin(k+1)\theta d\theta \\ \frac{1}{\pi} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} G(i(\omega_\mathrm{m} + \Delta\omega_\mathrm{h} \sin\theta)) \cos(k+1)\theta d\theta \end{cases} , \tag{6.7}$$

for k even and odd, respectively. Using the continuous representation of G obtained in the last step and numerical quadrature, these moments are obtained to high precision. Note that since all $\tilde{\xi}_l$ lie within the unit circle, the moments h_k decay quickly as a function of k and can be truncated for $h_k \ll \varepsilon$.

6.2.3. Extraction

Eq. (6.6) forms a second Prony problem. With Eq. (6.2), M significant $\tilde{A}_l$ and $\tilde{\xi}_l$ are extracted and the resulting poles and weights are recovered as

$$\xi_l = g^{-1}(\tilde{\xi}_l) = \frac{\Delta\omega_\mathrm{h}}{2}(\tilde{\xi}_l - \frac{1}{\tilde{\xi}_l}) + i\omega_\mathrm{m} \, , \tag{6.8}$$

$$A_l = \mathrm{Res}[G(z), \xi_l] = \frac{\Delta\omega_\mathrm{h}}{2}(1 + \frac{1}{\tilde{\xi}_l^2})\tilde{A}_l \, . \tag{6.9}$$

Eqs. (6.8) and (6.9) yield a minimal pole approximation of the form of Eq. (6.1) that is accurate to within ε and reveals the analytic structure of the function. To evaluate

the corresponding spectral function $A(\omega) = -\frac{1}{\pi}\mathrm{Im}G(\omega)$, we evaluate Eq. (6.1) for ω along $\mathbb{R} + i0^+$. By lowering ε, the precision can be systematically increased, at the cost of adding additional poles. For the cases examined, this pole representation is much more compact than comparable schemes [76–82] which typically do not yield a systematically improvable representation of the spectral function and may violate the analytic properties of the response function.

6.3. Results

In this section, we conduct simulations to verify the effectiveness of our method. Appendix A contains a pedagogical implementation of this procedure that, given a set of Matsubara points and a tolerance ε, produces a compact representation of the response function and its corresponding spectral function. An open source implementation is also available as part of the Green software package [83–85].

6.3.1. Error control

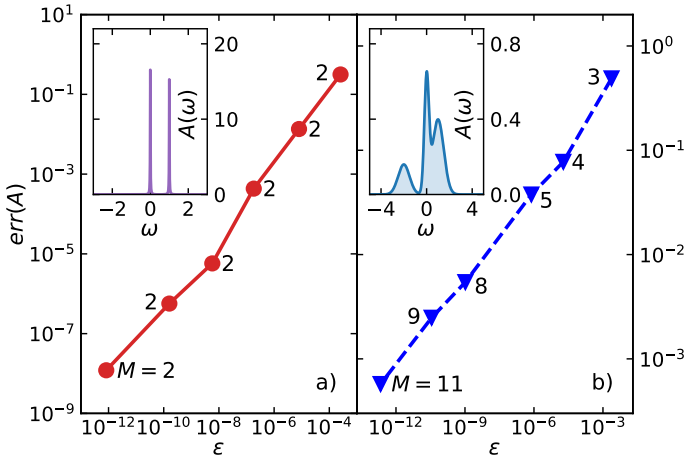

Figure 6.2: Integrated real axis error $err(A) = \int_{\mathbb{R}} d\omega |A - A_{\mathbf{cont}}|$ for the discrete (left) and continuous (right) case as a function of control parameter ε. Also indicated is the number of poles M. Inset: exact spectrum $A(\omega)$. Other parameters are $\beta = 200$, $n_0 = 30$ (left) and 0 (right), $\Delta n = 1$ and $N_\omega = 2001$.

We start our discussion with an examination of the convergence of the spectrum as a function of the error control parameter ε. For a discrete (Fig. 6.2a) and continuous (Fig. 6.2b) case we define a spectral function $A(\omega)$ on the real axis, transform it to the Matsubara axis, and continue it back to the real axis within precision ε as A_{cont}. We then show $err(A) = \int d\omega |A - A_{\mathrm{cont}}|$ as a function of ε. In striking difference to the 'ill-conditioned' nature of a direct analytic continuation, we observe that A_{cont} rapidly converges to A as ε is decreased. We observe this convergence behavior regardless of the form of the spectrum. The approximation is indeed compact: in the discrete case, only two poles are needed irrespective of the precision. In the continuum case, increasing the precision of the difference of the integral to 10^{-3} requires an increase of the number of poles from $M = 3$ to $M = 11$. Appendix A contains the precise analytical form of the functions examined along a list of the poles.

6.3.2. Dependence on temperature

To investigate how performance depends on system temperature, we use an asymmetric spectrum with multiple features as an example and conduct simulations across a wide range of temperatures and precision levels. Explicitly, data points at precision 10^{-m} are obtained by

$$G_{\mathrm{prec}}(i\omega_n) = \mathrm{round}(G_{\mathrm{exact}}(i\omega_n) \times 10^m) \times 10^{-m} \ . \tag{6.10}$$

As shown in Fig. 6.3, the results improve as precision increases, regardless of the temperature. However, as temperature increases, the convergence rate becomes slower and thus higher precision is needed to resolve the same feature, which is an inherent difficulty of the analytic continuation problem.

Although the three-peak feature cannot be resolved at the current precision for $\beta = 1$ and 2, we argue that there is still an improvement in the results. To illustrate this, we calculate the difference between the exact spectrum moments A_k and the recovered ones $\hat{A}_k$. These are calculated using the following equations:

$$A_k = \int_{-\infty}^{\infty} d\omega A_{\mathrm{exact}}(\omega)\omega^k \quad \text{and} \quad \hat{A}_k = \sum_l A_l \xi_l^k \ . \tag{6.11}$$

As shown in Fig. 6.4, the recovered moments become more and more accurate when the precision increases. It is expected that as the precision is improved further, different

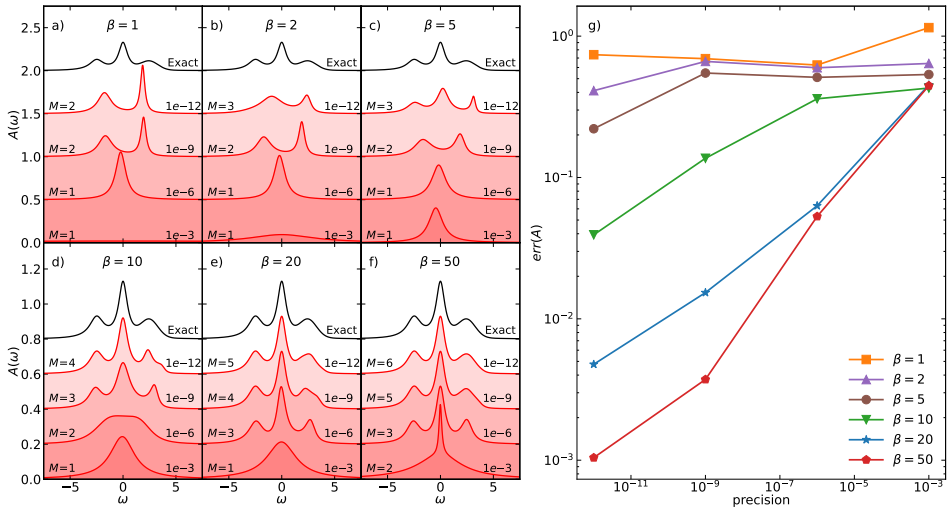

Figure 6.3: Recovered spectral functions at different temperatures $\beta \in \{1, 2, 5, 10, 20, 50\}$ **and various precision levels** $\{10^{-3}, 10^{-6}, 10^{-9}, 10^{-12}\}$**. Left panel: evolution of spectrum shapes. Right panel: convergence of** $err(A)$ **as a function of precision. Number of available data points is fixed at** $N_\omega = 201$**.**

peaks can eventually be resolved.

6.3.3. Dependence on data points

To explore the dependence on available data points, we fix the system at a moderate temperature ($\beta = 30$) and perform simulations for the model in Sec. 6.3.2 with a varying number of data points $N_\omega \in \{7, 9, 11, 35, 151, 1001\}$. As shown in Fig. 6.5, while seven points are not enough to achieve convergence, nine points are sufficient to resolve the three-peak feature. For $N_\omega \geq 11$, we observe a similar convergence behavior, which is insensitive to the number of available points. Specifically, increasing the number of data points does not necessarily accelerate convergence. We note that the reason for this phenomenon is due to the fact that the holomorphic mapping eliminates the effects from the absence of data points in the long tail. This happens because the information of mapped poles is fully contained by function values on the unit circle, which are only mapped from the finite interval $[i\omega_{n_0}, i\omega_{n_0+(N_\omega-1)\Delta n}]$. When there are too many data points, the convergence speed slightly slows down, as the mapped poles become too close to each other. Given $N_\omega = 2N+1$ data points, our method can recover at most N poles,

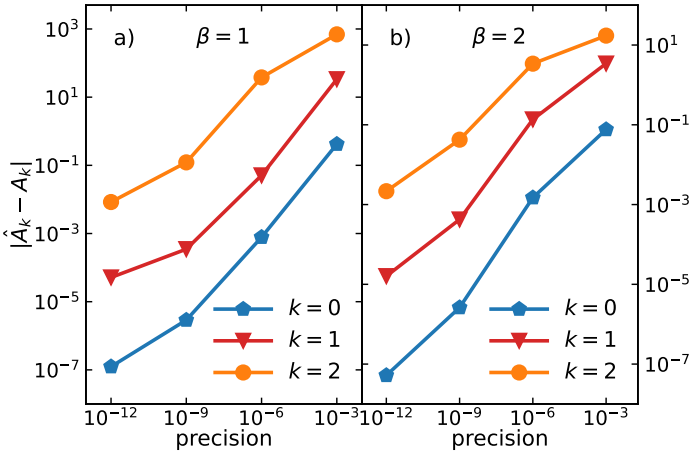

Figure 6.4: Convergence of spectrum moments as a function of precision for $\beta = 1$ and 2.

leading to failures when there are too few points to capture all poles. We estimate the threshold for the required number of points to be $N_{\text{min}} \gtrapprox 2N_{\text{pole}} + 1$, where N_{pole} is the number of poles needed to resolve the fine structure of the spectrum. When $N_\omega \geq N_{\text{min}}$, convergence should always be observed at sufficient precision.

Since the improvement is independent of temperature and highly robust to the number of available data points, we fix the system temperature at $\beta = 200$ and the number of data points at $N_\omega = 2001$ in the following discussions.

6.3.4. Challenging examples

In Fig. 6.6, we analyze the performance of the method for four continuous noiseless scenarios: A continuous spectral function with sharp band edges and a van Hove singularity, as it is encountered in a 2d tight binding calculation of the square lattice with nearest- and next-nearest-neighbor hopping (left panel); a 'semicircular' density of states with square-root singularities as encountered in the non-interacting infinite coordination number Bethe lattice with nearest neighbor hopping (middle panel); a tight-binding band structure of an anisotropic triangular lattice [86], and a simulated 'Kondo' setup with a sharp peak and two side bands (right panel).

We proceed by back-continuing the known function $A(\omega)$ to the Matsubara axis,

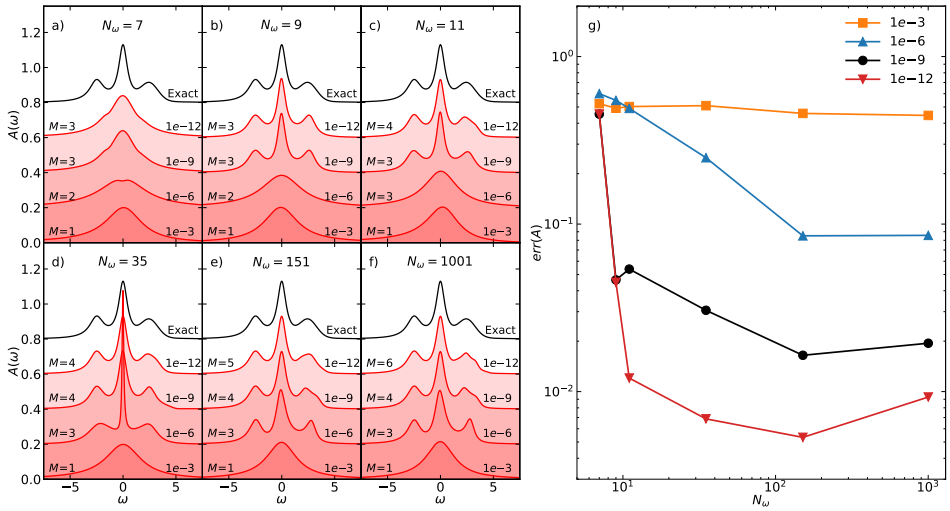

Figure 6.5: Recovered spectral functions at different numbers of data points $N_\omega \in \{7, 9, 11, 35, 151, 1001\}$ and various precision levels $\{10^{-3}, 10^{-6}, 10^{-9}, 10^{-12}\}$. Left panel: evolution of spectrum shapes. Right panel: convergence of $err(A)$ as a function of N_ω at different precision levels. System temperature is fixed at $\beta = 30$.

approximating it with ε chosen close to machine precision (resulting in $M = 10$, 9, 9, and 9), and plotting both $A(\omega)$ and $A_{\text{cont}}(\omega)$ as a function of frequency ω together with results from Maximum Entropy [13, 22] and the stochastic optimization method (SOM) [26, 31].

All four functions are difficult to analytically continue with standard methods, since they contain both broad and sharp features. The standard methodology of finding the 'smoothest' function consistent with input data within some error is not appropriate and introduces artificial 'ringing'. While precise knowledge of the location of the band edges and singularities could be used in a Nevanlinna function interpolation [17] followed by a Hardy function optimization [17] to pick the 'correct' function out of a Hardy function space, this knowledge is often not available.

The low-rank representation of the Green's function produced by the Prony method provides an unbiased alternative selection criterion that, in this case, is substantially more precise than a smoothness criterion.

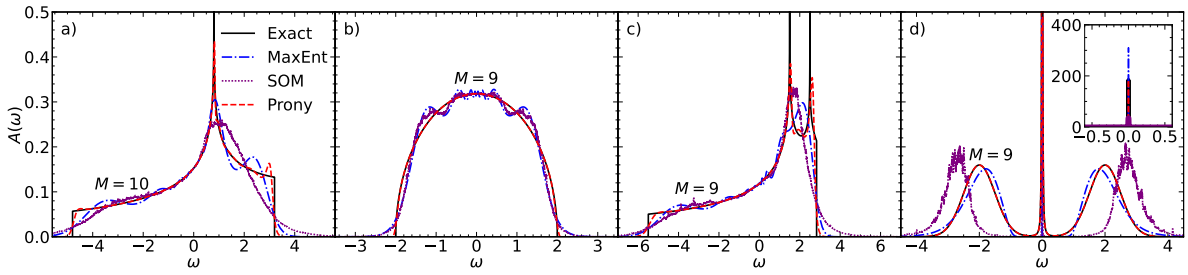

Figure 6.6: Continuation of continuous spectral functions. From left to right: tight binding density of states of 2d square lattice with nearest- and next-nearest neighbor hopping. Semicircular density of states. Tight-binding density of states of the anisotropic triangular lattice. 'Kondo'-like spectral function. Shown are the exact input $A(\omega)$ in black, a continuation with Maximum Entropy (blue), SOM (purple), and a Prony fit (this method) in red. Maximum Entropy parameters fine-tuned to yield best spectra possible.

6.3.5. Versatility

While a fermion Green's function of an operator and its corresponding adjoint corresponds to a positive spectral function [87] whose poles lie in the lower half of the complex plane [17], response functions of interest also include bosonic, anomalous, and off-diagonal cases which have different analytical properties. Importantly, they may not correspond to a probability distribution, ruling out the straightforward application of Maximum Entropy and related methods. While the issue can be circumvented by continuing related quantities [88–92], the procedure often amplifies errors [13].

The method presented here does not explicitly enforce an analytic structure. It can therefore be applied directly to bosonic, off-diagonal, and anomalous Green's functions as well as to self-energies. As an example we show the off-diagonal part of a continuous fermion spectral function in Fig. 6.7a; a discrete off-diagonal fermion system in Fig. 6.7b; a continuous diagonal boson system in Fig. 6.7c; and a discrete off-diagonal boson system in Fig. 6.7d. Note that the method for continuous and discrete systems is identical; it is the low-rank representation that places a minimum number of poles very close to the real axis to distinguish sharp (discrete) features from smooth (continuous) ones.

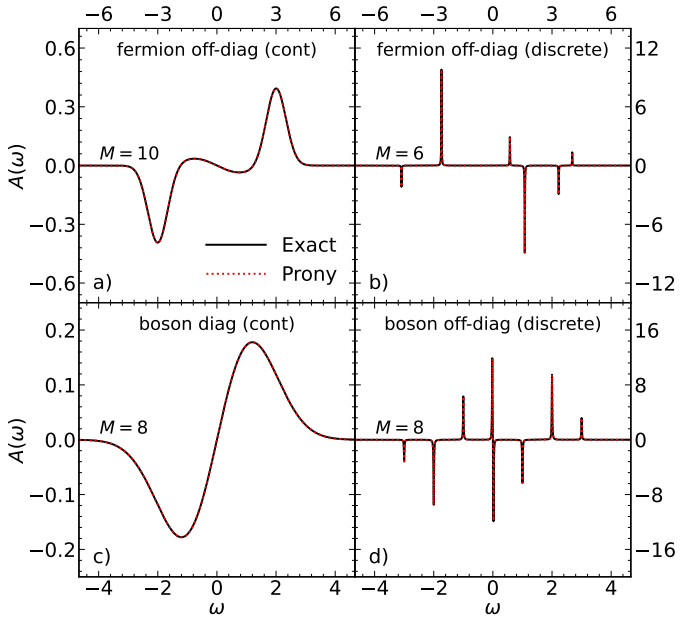

Figure 6.7: Analytic continuation of non-positive spectral functions in noiseless scenarios. Black: Exact input. Red: Continuation. Panel (a/b): off-diagonal continuous (a) and discrete (b) fermion case. (c) diagonal boson case. (d) discrete off-diagonal boson case.

6.3.6. Noisy data

Analytic continuation is commonly used on noisy Monte Carlo data, where a response function is known only within a given precision. The precision achievable depends very much on the Monte Carlo algorithm and the estimator used but is rarely better than 10^{-5}, and errors are often (but not always [93]) Gaussian distributed. In that case, we substitute ε as a proxy for the Monte Carlo error bar.

For a discrete and a continuous scenario, the left panels of Fig. 6.8 shows the convergence of the spectral function in our method, Maximum Entropy [13] and SOM [26] for simulated Gaussian errors with varying magnitude. The right panel shows the integrated error $err(A)$. It is evident that already very loose error tolerance reproduces the main features of the spectrum. As the simulated Monte Carlo errors are decreased, our method rapidly converges to the exact result whereas the spectrum is not recovered in Maximum Entropy and SOM.

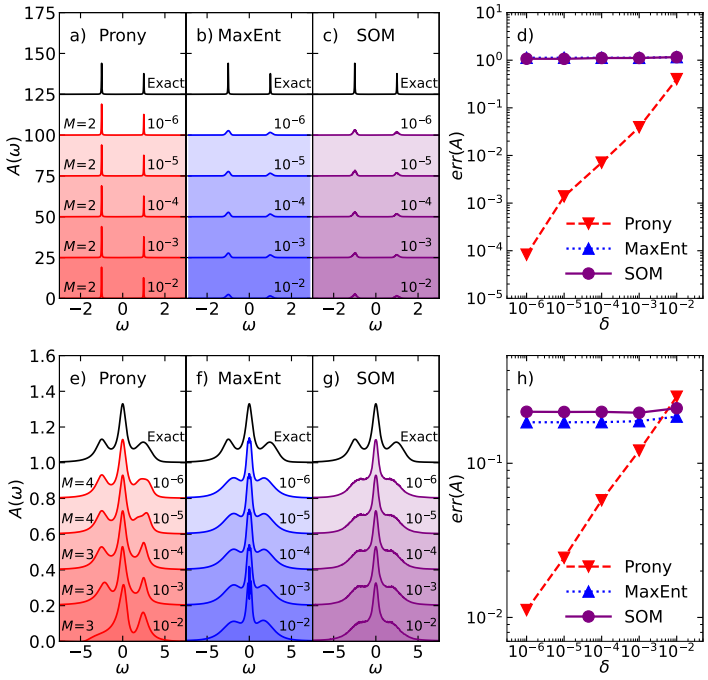

Figure 6.8: Spectral functions for different levels of relative Gaussian noise δ on the imaginary axis. Upper panel: discrete case. Lower panel: continuous case. Also indicated is the number of poles M.

6.3.7. Imaginary-time data

Our method can be readily extended to the analytic continuation of imaginary-time data. This extension involves two steps: (i) transforming the imaginary-time data to the Matsubara axis and (ii) performing analytic continuation from the resulting Matsubara data. Directly applying a discrete Fourier transform to imaginary-time data can distort the high-frequency Matsubara components, as expected from Nyquist's theorem. To overcome this limitation, several techniques have been developed in the community, including high-frequency tail expansions [94], Chebyshev polynomial representations [78], the intermediate representation [79], sparse sampling approaches [80], and the discrete Lehmann representation (DLR) [82].

Here, to demonstrate the feasibility of converging the spectral function as the precision of imaginary-time data increases, we consider two examples: a semicircular spectrum and the three-peak spectrum shown in Fig. 6.2(b). Starting from imaginary-time data $G(\tau)$ at a moderate temperature ($\beta = 30$), sampled on a uniform grid with a moderate

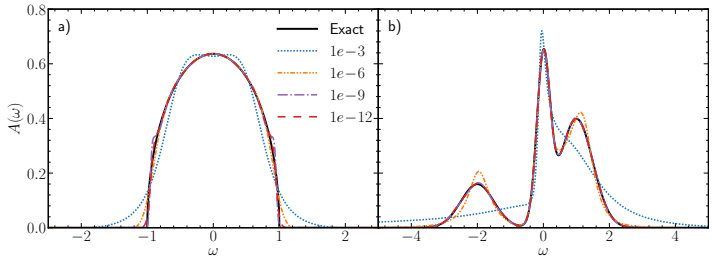

Figure 6.9: Evolution of the recovered spectrum shapes from the imaginary-time data $G(\tau)$ at various precision levels $\{10^{-3}, 10^{-6}, 10^{-9}, 10^{-12}\}$ for: (a) a semicircular spectrum; and (b) a multiple-featured spectrum.

number of data points ($N_\tau = 201$), we use least-squares fits to extract the DLR coefficients [82] of $G(\tau)$. We then analytically transform the fitted representation to the Matsubara axis and perform analytic continuation. As shown in Fig. 6.9, our simulations indicate that the recovered spectra continue to exhibit convergent behavior as the precision of $G(\tau)$ increases. A detailed comparison of different imaginary-time-to-frequency transformation techniques is beyond the scope of this work.

6.4. Conclusions

In conclusion, we have shown a method to systematically construct low-rank pole approximations to Matsubara response functions of quantum systems and used it to analytically continue spectral functions. We have demonstrated the control of the method in the sense that the error in the real-frequency response functions can be systematically reduced by improving the corresponding Matsubara fit.

We have also demonstrated the wide applicability of the method, including its suitability for diagonal, off-diagonal, fermionic, bosonic, continuous, and discrete response functions and we have examined the convergence in the presence of noise. We note that the same approximation scheme can also be used to model real-frequency response functions a short distance above the real axis, which may be useful in cases where a Matsubara representation is to be avoided entirely.

Apart from NAC, the compact representations introduced here offer a path towards faster numerical and analytical manipulation of response functions, and they offer physical insight by revealing the locations of poles and zeros in the complex plane.

Chapter 7

Matrix-Valued Extension and Systematic Improvements

This chapter is based on *Lei Zhang, Yang Yu and Emanuel Gull, Minimal pole representation and analytic continuation of matrix-valued correlation functions, Phys. Rev. B 110, 235131 (2024)* [39].

7.1. Introduction

Numerical simulations of finite-temperature field theories typically obtain response functions, such as single-particle Green's functions and susceptibilities, on the Matsubara axis. Such methods include Monte Carlo simulations of lattice [10, 11] and quantum impurity [12] problems in condensed matter, quantum chromodynamics simulations in nuclear and high-energy physics [58–60], and the self-consistent solution of low-order diagrammatic equations [7–9, 95] and embedding problems [96–99] of solids and molecules. Analysis, interpretation, and comparison to experiment of these correlation functions require an additional step of analytic continuation to the real axis.

Numerical analytic continuation (NAC) of imaginary axis data is difficult due to the ill-posed nature of the direct inversion [13]. Numerous methods for its regularization have been proposed, including interpolation with Padé [14, 15] and Nevanlinna functions [16, 17], the maximum entropy method [13, 18–23], approximation with Bayesian approaches [72, 100], stochastic fitting methods [24–26, 28, 29, 31], and sparse modeling [30] of the inverse problem.

Recent work [38] showed that compact pole representations with complex poles are

particularly powerful in this context. Pole expansions are in wide use in materials simulations [101] and have been used for analytic continuation in Refs. [35–37, 102, 103]. The methods in Ref. [38] now make it possible to systematically converge analytic continuations for typical response functions as data quality is increased.

Moreover, compact pole representations also facilitate the interpretation of Green's functions since they provide a decomposition into damped or undamped 'quasiparticles'. The representation is both efficient for storing Green's functions, requiring a small list of poles and weights rather than a discretization on a frequency grid, and for performing many-body calculations, enabling the use of residue calculus for diagram integration [104].

Pole representations are closely related to the decomposition of real-time Green's functions into sums of decaying complex exponentials which, when Fourier transformed, correspond to sums of poles. In the case of signal-processing, the recovery of poles from data was already discussed in the eighteenth century by Gaspard de Prony [47]. In the context of real-time propagation, extensions and approximations of this type have recently been analyzed with a wide variety of methods [56, 87, 105–108].

Most physical problems, especially in the context of simulations of real materials and ordered phases such as superconductivity, require the analysis of multi-orbital Green's functions with off-diagonal elements [23, 88–90, 92] that do not correspond to probability densities. While methods such as those presented in Ref. [38] can in principle be applied to diagonal and off-diagonal elements of Green's function matrices independently, such a procedure may not respect the Carathéodory [17] structure of these objects, which (for fermion Green's functions) requires the matrix-valued response function to be a positive semidefinite function anywhere in the upper half of the complex plane. Knowledge of this matrix-valued structure is a prerequisite for performing certain post-processing calculations such as the extraction of two-particle correlation functions, including the optical and magnetic susceptibility, and the calculation of the Green's function on the real axis via the Dyson equation from an analytically continued self-energy.

A pole expansion in terms of shared real or complex poles with orbital-dependent matrix-valued weights, given by the form

$$\boldsymbol{G}(z) = \sum_{l=1}^{M} \frac{\boldsymbol{A}_l}{z - \xi_l}, \ \operatorname{Im} \xi_l \leq 0, \tag{7.1}$$

which is valid throughout the upper half of the complex plane, is therefore desirable.

This chapter presents a minimal pole method (MPM) for generating such a representation in terms of a minimal number of poles M on or below the real axis using two approximations with damped exponential functions and an intermediate holomorphic mapping. We show results for fermionic and bosonic systems with discrete and continuous spectra, with and without noise. We also show applications to the self-energy and Green's function of self-consistent real-materials calculations.

The remainder of this chapter proceeds as follows. In Sec. 7.2, we revisit the ill-posed nature of NAC and discuss the main sources of ambiguity. In Sec. 7.3, we introduce the three algorithmic steps of the pole approximation, the optional restriction to poles on the real axis, the optional combination with the discrete Lehmann representation, the optional incorporation of prior knowledge, and applications beyond analytic continuation. In Secs. 7.4 and 7.5, we show numerical results and conclusions, respectively.

7.2. Sources of Ambiguity in NAC

In this section, we revisit the origin of the ill-posed nature of NAC. Reducing these ambiguities as much as possible will be the primary goal of the method introduced in the next section.

We begin with the inverse process of NAC, namely the construction of the Green's function from a given spectral function,

$$\boldsymbol{G}(iy) = \int_{-\infty}^{+\infty} d\omega \frac{\boldsymbol{A}(\omega)}{iy - \omega}. \tag{7.2}$$

This process can be carried out by numerical integration and is well-conditioned. If the approximate spectrum $\boldsymbol{A}_{\mathrm{approx}}(\omega)$ is controlled, i.e.,

$$\|\boldsymbol{A}_{\mathrm{approx}}(\omega) - \boldsymbol{A}_{\mathrm{exact}}(\omega)\| \leq \varepsilon \|\boldsymbol{A}_{\mathrm{exact}}(\omega)\|, \tag{7.3}$$

where ε is a small constant, then the corresponding Green's function is also controlled on any continuous interval of the imaginary axis away from singularities:

$$\|\boldsymbol{G}_{\mathrm{approx}}(iy) - \boldsymbol{G}_{\mathrm{exact}}(iy)\| \leq c \cdot \varepsilon, \tag{7.4}$$

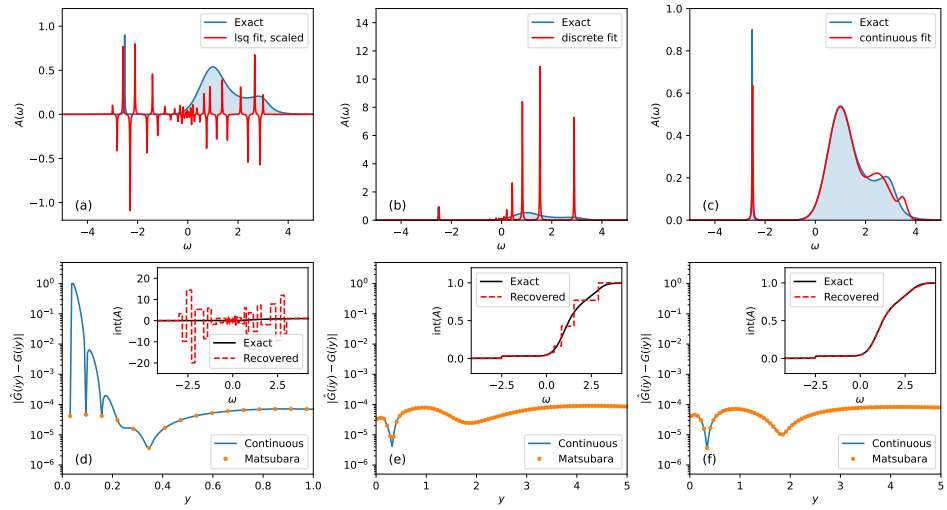

Figure 7.1: Illustration of the ill-posed nature of NAC. Left panels: unrestricted least-squares fits. Middle panels: fits using a discrete system. Right panels: fits using a continuous system. The top panels compare the exact spectrum (blue shaded region) with the recovered spectrum (red). The bottom panels show the difference between the recovered and exact Green's functions on the imaginary axis, with Matsubara points marked by orange dots. Insets: comparison between the exact and recovered integrated spectral functions, $\mathbf{int}(\boldsymbol{A})(\omega) = \int_{-\infty}^{\omega} d\omega' \boldsymbol{A}(\omega')$.

where c is a finite constant independent of ε. Conversely, in NAC, if the approximation (7.4) is not controlled on a continuous interval away from singularities, the recovered spectral function cannot be controlled.

However, the discrete nature of the input data imposes a fundamental limitation. At finite inverse temperature β, the Green's function is sampled only at Matsubara frequencies, whose spacing is at least $2\pi/\beta$. As a result, minimizing the difference at Matsubara points, for example through the Frobenius norm $\sum_n \|\boldsymbol{G}_{\text{approx}}(i\omega_n) - \boldsymbol{G}_{\text{input}}(i\omega_n)\|_F^2$, does not guarantee that Eq. (7.4) is controlled between adjacent Matsubara points, i.e., at points $i\omega_n < iy < i\omega_{n+1}$. This is the first ambiguity in NAC.

As an illustration, we consider a spectrum containing both sharp and broad features and perform unrestricted least-squares fits. The recovered spectrum is shown in Fig. 7.1(a), and it bears little resemblance to the exact spectrum. The reason is shown in Fig. 7.1(d): although the Matsubara data are fitted within the target precision $\varepsilon = 10^{-4}$, the recovered Green's function is poorly approximated on the continuous imaginary-axis

interval between the Matsubara points. The inset compares the integrated spectral function $\mathrm{int}(\boldsymbol{A})(\omega) = \int_{-\infty}^{\omega} d\omega' \boldsymbol{A}(\omega')$, further confirming the discrepancy.

A second ambiguity remains even if Eq. (7.4) is controlled on a continuous interval: under any finite precision, the recovered spectrum is still highly non-unique. For example, one can always find a discrete system with real poles that approximates the Green's function within ε on a continuous interval, as shown in Fig. 7.1(e). However, the corresponding spectrum, shown in Fig. 7.1(b), contains extra spiky features and deviates from the exact spectrum. The ambiguity also persists when the input data are fitted with a continuous model. As shown in Fig. 7.1(f), the continuous model can approximate the input data well. Nevertheless, the recovered spectrum in Fig. 7.1(c) contains extra oscillatory features compared with the broad structure of the exact spectrum.

Even so, enforcing Eq. (7.4) still improves the recovered result. Although the detailed spectral shape remains ambiguous, the recovered spectra show coarse-grained similarity to the exact spectrum, as illustrated by the integrated spectral functions in the insets of Figs. 7.1(e) and (f). Finally, we emphasize that simply improving the fitting precision may reduce, but cannot completely eliminate, this ambiguity. At any finite precision, one can still construct either a discrete or a continuous spectrum with additional features that fits the input data within the prescribed tolerance. Therefore, a criterion that suppresses such unphysical extra features without relying on prior knowledge is highly desirable.

7.3. Minimal Pole Method

We aim to construct a pole approximation of the form given in Eq. (7.1) in the upper half of the complex plane, based on data provided along the Matsubara axis. The evaluation of this function just above the real axis will then yield the analytically continued spectral function, see Eq. (3.46).

In analogy to the scalar-valued version of the method [38], the construction of the matrix-valued approximation consists of three steps. First, we approximate the Matsubara data on a finite interval of the imaginary axis using the scalar-valued ESPRIT method for each matrix element. Second, we map this interval onto the unit circle using the holomorphic mapping presented in Ref. [38] and evaluate the moments of the approximated function numerically. Finally, we apply matrix-valued ESPRIT to extract a

minimal number of shared complex poles, map the poles back using the corresponding inverse holomorphic mapping, and evaluate the spectral function. Note that due to the robustness to noise and its computational efficiency, a variant of ESPRIT [53, 54] rather than the Prony approximation [38] is employed in this chapter.

We discuss these three steps in the following.

7.3.1. Approximation on the Matsubara axis

Given input data $\boldsymbol{G}_{\text{input}}(i\omega_n)$ on a uniform grid along the non-negative Matsubara axis, we aim to find an approximation to the Green's function that is valid both at the input Matsubara grid points and in the intervals between them.

While many interpolation or fitting methods could be used to approximate this function, including a least squares fit to compact basis functions [76, 78, 79, 82] or an interpolation with cubic splines [77], a fit with the minimum number of decaying exponential functions results in a representation that is accurate both at the interpolation nodes and in between, since it minimizes spurious oscillations but does not overfit the data. For further details, refer to Fig. 7.4 as an example, discussed later.

Although the matrix-valued ESPRIT will become necessary in Sec. 7.3.3, sharing the same set of poles in Eq. (7.1) for each matrix element does not necessarily imply that they can be well-approximated by shared exponentials through Eq. (4.12). Therefore, we independently apply the scalar-valued exponential fit to each matrix element of the input data, yielding an approximation $\boldsymbol{G}^{(L)}_{\text{approx}}(iy)$ for $\omega_0 \leq y \leq \omega_{n_\omega - 1}$ with a fixed parameter L in ESPRIT. This ensures that

$$\left\| \boldsymbol{G}^{(L)}_{\text{approx}}(i\omega_n) - \boldsymbol{G}_{\text{input}}(i\omega_n) \right\| \leq \varepsilon \tag{7.5}$$

at the fitted Matsubara points.

Additionally varying L provides a convenient way to explore the sensitivity of the approximant between the interpolation points. If

$$\left\| \boldsymbol{G}^{(L')}_{\text{approx}}(iy) - \boldsymbol{G}^{(L)}_{\text{approx}}(iy) \right\| \leq \varepsilon \tag{7.6}$$

holds for $\omega_n < y < \omega_{n+1}$ for different values of L and L', the points are likely oversampled and the approximation on $[\omega_n, \omega_{n+1}]$ is assumed to be accurate; otherwise the

approximation on this interval may not be reliable. Typically only the low frequencies are not approximated reliably.

We may use this property to establish a lower threshold Matsubara frequency ω_{n_0}. Above ω_{n_0} the approximation is assumed to be accurate in the continuous interval $y \in [\omega_{n_0}, \omega_{n_\omega - 1}]$. In most of the continuous cases we tested, this lower threshold coincides with the zero Matsubara frequency, i.e., $\omega_{n_0} = \omega_0$. In practice, we use $L = 2N/5$ and $L' = N/2$ in all our implementations.

Before concluding this section, we summarize the algorithm used in this work for the reader's convenience:

Algorithm 2 Approximation Algorithm

1: **Input:** $\boldsymbol{G}_{\text{input}}(i\omega_n)$ on the grid $\{\omega_n\}, 0 \leq n \leq n_\omega - 1$.
2: **Output:** n_0 and $\boldsymbol{G}_{\text{approx}}(iy)$.
3: **for** $i = 1$ to n_{orb} **do**
4: **for** $j = 1$ to n_{orb} **do**
5: Obtain $\boldsymbol{G}_{ij}^{(L)}(iy)$ from ESPRIT for $[\boldsymbol{G}_{\text{input}}]_{ij}$.
6: Obtain another approximation $\boldsymbol{G}_{ij}^{(L')}(iy)$ at L'.
7: Estimate ε_{ij} from Eq. (7.7).
8: Estimate $(n_0)_{ij}$ from Eq. (7.8).
9: Obtain $n_0 = \max_{1 \leq i,j \leq n_{\text{orb}}} \{(n_0)_{ij}\}$.
10: Set the approximation to be $[\boldsymbol{G}_{\text{approx}}(iy)]_{ij} = \boldsymbol{G}_{ij}^{(L)}(iy)$.

In the algorithm, the following equations are utilized:

$$\varepsilon_{ij} = \max \left\{ \max_{0 \leq n \leq n_\omega - 1} \left| \boldsymbol{G}_{ij}^{(L)}(i\omega_n) - [\boldsymbol{G}_{\text{input}}(i\omega_n)]_{ij} \right|, \right. \\ \left. \max_{0 \leq n \leq n_\omega - 1} \left| \boldsymbol{G}_{ij}^{(L')}(i\omega_n) - [\boldsymbol{G}_{\text{input}}(i\omega_n)]_{ij} \right| \right\}, \tag{7.7}$$

and

$$(n_0)_{ij} = \min \left\{ n \left| \max_{\omega_n \leq y < \omega_{n+1}} \{ |\boldsymbol{G}_{ij}^{(L')}(iy) - \boldsymbol{G}_{ij}^{(L)}(iy)| \} \leq \varepsilon_{ij} \text{ and } \right. \right. \\ \left. \left. \frac{\max_{\omega_n \leq y < \omega_{n+1}} \{ |\boldsymbol{G}_{ij}^{(L')}(iy) - \boldsymbol{G}_{ij}^{(L)}(iy)| \}}{\max_{\omega_{n+1} \leq y < \omega_{n+2}} \{ |\boldsymbol{G}_{ij}^{(L')}(iy) - \boldsymbol{G}_{ij}^{(L)}(iy)| \}} < \text{tol} \right\}. \tag{7.8}$$

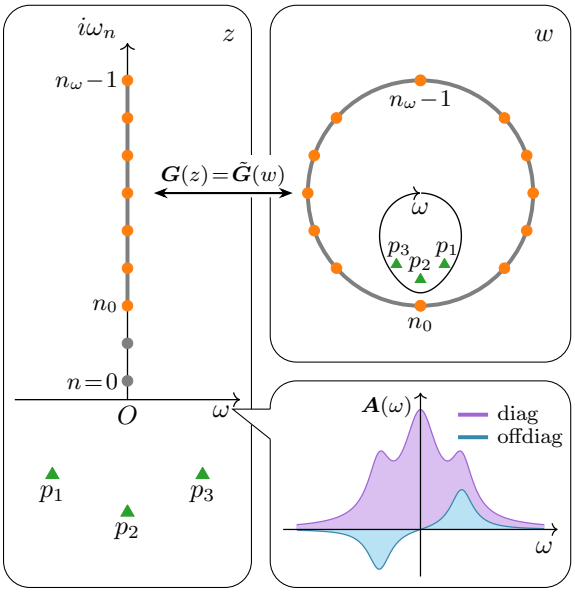

Figure 7.2: Left: Complex plane with real and imaginary axis (black lines), Matsubara frequencies in the interval $[i\omega_{n_0}, i\omega_{n_\omega-1}]$ (orange dots), remaining frequencies (grey dots), and pole locations of a spectral function (green triangles). Right top: Complex plane after holomorphic mapping (7.9) transforming the interval onto the unit circle, mapping the remainder of the complex plane to the interior of the unit disk, infinity to zero, and the poles to the interior of the egg-shaped image of the real axis. Bottom right: Example of diagonal and off-diagonal spectral functions belonging to shared complex poles (green triangles in other panels).

7.3.2. Holomorphic mapping

The procedure of Sec. 7.3.1 results in an accurate description of the Green's function on an interval on the Matsubara axis, though not in the form of a pole representation. The problem now reduces to recovering the complex poles of such a function. As pointed out in Refs. [36, 37], applying a holomorphic transform to map the interval to the unit circle reveals this pole information.

We choose the transform presented in Ref. [38] to map the entire complex plane into the closed unit disk $\bar{D}$:

$$\begin{cases} w &= g(z) &= z_\mathrm{s} - \sqrt{z_\mathrm{s}^2+1} \text{ with } z_\mathrm{s} = \frac{z-i\omega_\mathrm{m}}{\Delta\omega_\mathrm{h}} \\ z &= g^{-1}(w) &= \frac{\Delta\omega_\mathrm{h}}{2}(w - \frac{1}{w}) + i\omega_\mathrm{m} \end{cases} , \tag{7.9}$$

where $\omega_{\mathrm{m}} = (\omega_{n_0} + \omega_{n_\omega - 1})/2$, $\Delta\omega_{\mathrm{h}} = (\omega_{n_\omega - 1} - \omega_{n_0})/2$, and the branch of the square root is chosen such that $|w| \leq 1$ (cf. Fig. 7.2). This transformation offers several benefits: First, the correspondence is straightforward. Each point on the continuous interval $[i\omega_{n_0}, i\omega_{n_\omega - 1}]$ is mapped to two points with identical vertical coordinates, except for the endpoints $i\omega_{n_0}$ and $i\omega_{n_\omega - 1}$, which are mapped to the unique points $-i$ and i on the unit circle, respectively. Second, it preserves the 'left-right' symmetry $\operatorname{Re} g(x + iy) = -\operatorname{Re} g(-x + iy)$ of the plane, which simplifies the contour integrals that will be defined in Eq. (7.11). Third, every line segment dy on the continuous interval in the z-plane is mapped to the arc (dx', dy') in the w-plane with $dy' = \frac{1}{\Delta\omega_{\mathrm{h}}} dy$, where $\Delta\omega_{\mathrm{h}}$, as defined above, is a constant. This avoids distortions of the interval along the vertical direction, which enhances robustness when some Matsubara points exhibit large fluctuations while others do not.

Eq. (7.9) transforms the complex pole representation (7.1) to:

$$\tilde{\boldsymbol{G}}(w) = \sum_{l=1}^{M} \frac{\tilde{\boldsymbol{A}}_l}{w - \tilde{\xi}_l} + \text{analytic part} \,. \tag{7.10}$$

With the matrix-valued integrals $\boldsymbol{h}_k$ over the unit circle defined as

$$\boldsymbol{h}_k := \frac{1}{2\pi i} \int_{\partial \bar{D}} dw \tilde{\boldsymbol{G}}(w) w^k \tag{7.11}$$

the residue theorem [36, 37] implies that:

$$\boldsymbol{h}_k = \sum_{l=1}^{M} \tilde{\boldsymbol{A}}_l \tilde{\xi}_l^k, \;\; k \geq 0. \tag{7.12}$$

The preservation of left-right symmetry in Eq. (7.9) additionally simplifies the contour integral:

$$\boldsymbol{h}_k = \begin{cases} \frac{i}{\pi} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \boldsymbol{G}(i(\omega_{\mathrm{m}} + \Delta\omega_{\mathrm{h}} \sin\theta)) \sin(k+1)\theta d\theta, \;\; k \text{ even} \\ \frac{1}{\pi} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \boldsymbol{G}(i(\omega_{\mathrm{m}} + \Delta\omega_{\mathrm{h}} \sin\theta)) \cos(k+1)\theta d\theta, \;\; k \text{ odd} \end{cases} . \tag{7.13}$$

Replacing $\boldsymbol{G}(iy)$ with $\boldsymbol{G}^{(L)}_{\mathrm{approx}}(iy)$ in Eq. (7.13) introduces uncertainty in $\boldsymbol{h}_k$ on the order of the tolerance parameter ε. In practice, each $\boldsymbol{h}_k$ is calculated using an adaptive

quadrature rule with error tolerance set much lower than ε, and the number of integrals is truncated when $\|\boldsymbol{h}_k\| < \varepsilon$. Since the holomorphic mapping maps poles into the interior of the unit disk, Eq. (7.12) implies that the number of moments quickly decays as a function of k.

7.3.3. Pole extraction

Eq. (7.12) forms a second Prony problem (4.12). Applying the method of matrix-valued ESPRIT to $\boldsymbol{h}_k$ at precision ε, the mapped pole weights $\tilde{\boldsymbol{A}}_l$ and locations $\tilde{\xi}_l$ are recovered. The exponentially decaying nature of the singular values of the Hankel matrix ensures that a minimal number of complex poles, denoted by M, are recovered in the current framework.

The one-to-one correspondence between $\tilde{\xi}_l$ and ξ_l [36, 37] via

$$\xi_l = g^{-1}(\tilde{\xi}_l) \,, \tag{7.14}$$

is then used to recover the pole locations.

To recover the pole weights $\boldsymbol{A}_l$, there are two main options, each with its own advantages. Given that the weights have a one-to-one correspondence via the residue theorem [38]:

$$\begin{aligned} \boldsymbol{A}_l &= \mathrm{Res}[\boldsymbol{G}(z), \xi_l] \\ &= \left.\frac{dz}{dw}\right|_{\tilde{\xi}_l} \times \mathrm{Res}[\tilde{\boldsymbol{G}}(w), \tilde{\xi}_l] = \left.\frac{dz}{dw}\right|_{\tilde{\xi}_l} \times \tilde{\boldsymbol{A}}_l, \end{aligned} \tag{7.15}$$

one approach is to recover $\boldsymbol{A}_l$ directly from $\tilde{\boldsymbol{A}}_l$. An advantage of this method is its effectiveness even when the input function contains a constant term, as seen in self-energy-like quantities, since the constant term can be automatically filtered out by Eq. (7.11). Additionally, it is highly stable, as the noise has already been filtered in the first step. However, the transformation may introduce slight distortions, though these did not affect the overall shape of the spectrum in our tests (not shown).

Alternatively, one may obtain the weights by solving the following system of equations:

$$\begin{pmatrix} \vec{G}_0^T \\ \vec{G}_1^T \\ \vdots \\ \vec{G}_{n_\omega-1}^T \end{pmatrix} = \begin{pmatrix} \frac{1}{i\omega_0-\xi_1} & \frac{1}{i\omega_0-\xi_2} & \cdots & \frac{1}{i\omega_0-\xi_M} \\ \frac{1}{i\omega_1-\xi_1} & \frac{1}{i\omega_1-\xi_2} & \cdots & \frac{1}{i\omega_1-\xi_M} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{i\omega_{n_\omega-1}-\xi_1} & \frac{1}{i\omega_{n_\omega-1}-\xi_2} & \cdots & \frac{1}{i\omega_{n_\omega-1}-\xi_M} \end{pmatrix} \begin{pmatrix} \vec{A}_1^T \\ \vec{A}_2^T \\ \vdots \\ \vec{A}_M^T \end{pmatrix}, \tag{7.16}$$

where $\vec{G}_n$ and $\vec{A}_l$ are the vectorized forms of $\boldsymbol{G}_{\text{input}}(i\omega_n)$ and $\boldsymbol{A}_l$, as described in Sec. 4.3. In this approach, no distortion occurs, and if some of the Matsubara points were ignored in the previous step ($n_0 > 0$) they can now be utilized. However, the method fails in the presence of a constant term. An optimization condition can also be incorporated into both methods to explicitly satisfy the analytic properties:

$$\begin{array}{l} \boldsymbol{A}_l \text{ (for fermions) or } \operatorname{sign}(\xi_l)\boldsymbol{A}_l \text{ (for bosons) is} \\ \text{positive semidefinite with } (\sum_l \boldsymbol{A}_l)_{ij} = [d_i, d_j^\dagger]_\pm \end{array}, \tag{7.17}$$

for discrete systems or

$$\begin{array}{l} -\operatorname{Im}(\sum_l \frac{\boldsymbol{A}_l}{\omega-\xi_l}) \text{ (for fermions) or } -\operatorname{sign}(\omega)\times \\ \operatorname{Im}(\sum_l \frac{\boldsymbol{A}_l}{\omega-\xi_l}) \text{ (for bosons) is positive semide-} \\ \text{finite for any } \omega \in \mathbb{R}, \text{with } (\sum_l \boldsymbol{A}_l)_{ij} = [d_i, d_j^\dagger]_\pm \end{array} \tag{7.18}$$

for continuous systems, albeit at the cost of additional runtime due to the optimization process. As demonstrated in Ref. [38], even in the presence of noise, the non-causal parts of the unrestricted solutions are typically not observable. However, when the data is too noisy to reliably contain causal information, or when the spectrum exhibits singularities, imposing restrictions becomes crucial.

In practice, we find that the second method yields better results for computing pole weights, so we always use Eq. (7.16). To address the issue of a non-zero constant term, we first calculate tentative weights $\boldsymbol{A}_l^{\text{t}}$ from the unrestricted solution of Eq. (7.15). The constant term is then estimated as:

$$\text{const} = \frac{1}{n_\omega - n_0} \sum_{n=n_0}^{n_\omega-1} \left(\boldsymbol{G}_{\text{input}}(i\omega_n) - \sum_{l=1}^{M} \frac{\boldsymbol{A}_l^{\text{t}}}{i\omega_n - \xi_l} \right). \tag{7.19}$$

We then substitute $\boldsymbol{G}_{\text{input}}(i\omega_n) - \text{const}$ into the left-hand side of Eq. (7.16) with optional restrictions (7.17) or (7.18) to obtain the final weights $\boldsymbol{A}_l$. After obtaining $\boldsymbol{A}_l$ and ξ_l, the spectral function is finally recovered using:

$$\boldsymbol{A}_{\text{rec}}(\omega) = -\frac{1}{\pi} \operatorname{Im} \left(\sum_{l=1}^{M} \frac{\boldsymbol{A}_l}{\omega + i0^+ - \xi_l} \right) . \tag{7.20}$$

7.3.4. Restriction to real poles

It is sometimes useful to restrict the locations of poles to the real axis, e.g., when it is known that the system consists of discrete excitations, or when a representation of the Green's function is desired that is valid both in the upper and in the lower half of the complex plane [104]. This restriction typically comes at the cost of a larger set of poles.

Instead of using Eq. (7.9), now we consider the holomorphic mapping

$$\begin{cases} w &= g(z) &= \omega_{n_0} \left(\sqrt{\frac{1}{z^2} + \frac{1}{\omega_{n_0}^2}} - \frac{1}{z} \right) \\ z &= g^{-1}(w) &= 2\omega_{n_0} \frac{w}{1-w^2} \end{cases} , \tag{7.21}$$

which reduces to the holomorphic mapping presented in Ref. [37] for fermionic systems when $n_0 = 0$. This maps the continuous imaginary interval $(-\infty, -i\omega_{n_0}] \cup [i\omega_{n_0}, +\infty)$ to the unit circle, see Fig. 7.3 for more details. The method of Sec. 7.3.1 provides a controlled approximation over the interval $[i\omega_{n_0}, i\omega_{n_\omega - 1}]$, while the tail behavior in $[i\omega_{n_\omega - 1}, +\infty)$ can be well approximated using other techniques, such as the complex pole representation introduced above. As for the approximation on $(-\infty, -i\omega_{n_0}]$, it can be directly derived from the symmetry $\boldsymbol{G}(z^*) = (\boldsymbol{G}(z))^\dagger$.

The contour integral in Eq. (7.11) can be simplified in this case to:

$$\boldsymbol{h}_k = \begin{cases} \frac{i}{\pi} \int_0^{\frac{\pi}{2}} [\boldsymbol{G}(\frac{i\omega_{n_0}}{\sin\theta}) - (\boldsymbol{G}(\frac{i\omega_{n_0}}{\sin\theta}))^\dagger] \sin(k+1)\theta d\theta, k \text{ even} \\ \frac{1}{\pi} \int_0^{\frac{\pi}{2}} [\boldsymbol{G}(\frac{i\omega_{n_0}}{\sin\theta}) + (\boldsymbol{G}(\frac{i\omega_{n_0}}{\sin\theta}))^\dagger] \cos(k+1)\theta d\theta, k \text{ odd} \end{cases} . \tag{7.22}$$

In many cases, the Hamiltonian is real-valued, so that the Green's function exhibits the

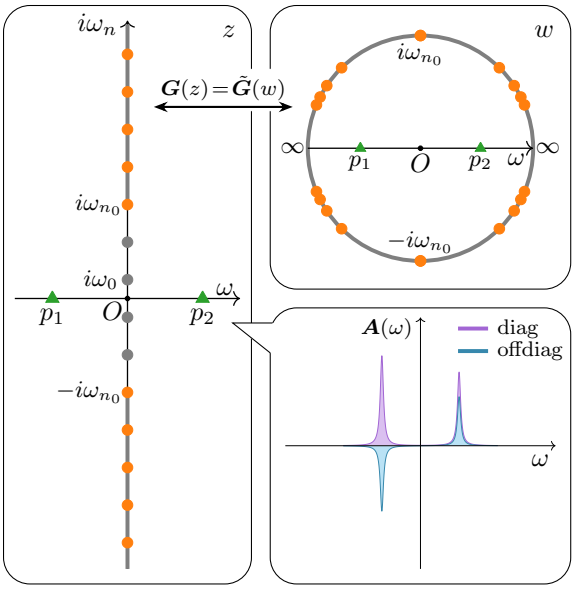

Figure 7.3: Illustration of the holomorphic mapping (7.21). Left: The original plane, showing Matsubara frequencies in the interval $(-\infty, -i\omega_{n_0}] \cup [i\omega_{n_0}, +\infty)$ **(orange dots), remaining frequencies (grey dots), and pole locations of a spectral function (green triangles). Top right: The complex plane after the holomorphic mapping (7.21), which transforms the interval onto the unit circle, maps the remainder of the complex plane inside the unit disk, and maps the real axis onto the line segment** $(-1, 0)$ **to** $(1, 0)$**, with the origin mapped to the origin and infinity mapped to** $(\pm 1, 0)$**. Bottom right: Examples of diagonal and off-diagonal spectral functions corresponding to the shared real poles. Delta peaks are broadened for better visualization.**

symmetry $[\boldsymbol{G}(z^*)]_{ij} = [\boldsymbol{G}^*(z)]_{ij}$. This symmetry further simplifies Eq. (7.22) to:

$$\boldsymbol{h}_k = \begin{cases} -\frac{2}{\pi} \int_0^{\frac{\pi}{2}} \mathrm{Im}[\boldsymbol{G}(\frac{i\omega_{n_0}}{\sin\theta})] \sin(k+1)\theta d\theta, & k \text{ even} \\ +\frac{2}{\pi} \int_0^{\frac{\pi}{2}} \mathrm{Re}[\boldsymbol{G}(\frac{i\omega_{n_0}}{\sin\theta})] \cos(k+1)\theta d\theta, & k \text{ odd} \end{cases} . \tag{7.23}$$

Since Eqs. (7.10) to (7.12) still hold for the new mapping, all other steps presented in Sec. 7.3 remain applicable, resulting in poles that lie on the real axis. In practice, unlike the generic method presented in the previous section, we found that obtaining pole weights from the transformed plane is more stable than from the original plane. Therefore, we use Eq. (7.15) rather than Eq. (7.16) in this case.

7.3.5. Combination with the Discrete Lehmann Representation

In situations where computational efficiency, rather than accuracy, is important, it is possible to utilize the discrete Lehmann representation (DLR) [82, 109] directly for approximating Matsubara data. By specifying an error tolerance ε and a sufficient cutoff $\Lambda = \beta\omega_{\max}$, where $\omega_{\max}$ represents the energy support of the system and can be chosen to be sufficiently large, DLR generates r fixed real poles $\omega_i^{(\mathrm{dlr})}$, where $r = \mathcal{O}(\log(\Lambda)\log(\frac{1}{\varepsilon}))$. The pole weights $\boldsymbol{g}_i^{(\mathrm{dlr})}$ are then determined based on the input data $\boldsymbol{G}_{\mathrm{input}}(i\omega_n)$. The resulting function, $\boldsymbol{G}_{\mathrm{approx}}(i\omega_n) = \sum_{i=1}^{r} \frac{\boldsymbol{g}_i^{(\mathrm{dlr})}}{i\omega_n - \omega_i^{(\mathrm{dlr})}}$, provides an approximation of the Matsubara data with an accuracy within ε.

Although the DLR method does not provide a precise approximation between Matsubara points (see Sec. 7.4.1.1), there typically is a threshold n_0 such that it offers a precise approximation in the intervals $(-\infty, -i\omega_{n_0}] \cup [i\omega_{n_0}, +\infty)$. In practice, and especially at low temperatures, ω_{n_0} is very close to zero, making the impact of disregarding the first n_0 points negligible. In that case, the calculation of the contour integral in Eq. (7.11), directly using the DLR expansion, simplifies to:

$$\boldsymbol{h}_k = \sum_{i=1}^{r} \tilde{\boldsymbol{g}}_i^{(\mathrm{dlr})} \tilde{\omega}_i^{(\mathrm{dlr})k}, \; k \geq 0, \tag{7.24}$$

where $\tilde{\boldsymbol{g}}_i^{(\mathrm{dlr})}$ and $\tilde{\omega}_i^{(\mathrm{dlr})}$ can be derived from $\boldsymbol{g}_i^{(\mathrm{dlr})}$ and $\omega_i^{(\mathrm{dlr})}$ using the transformations in Eqs. (7.15) and (7.14), respectively. This eliminates the need for time-consuming numerical quadrature. Additionally, there is no need to truncate $\boldsymbol{h}_k$ at $\|\boldsymbol{h}_k\| < \varepsilon$, since the number of recovered poles is strictly bounded by r. The method described in Sec. 7.3.3 may then be used to recover the pole information.

The use of DLR rather than the first exponential approximation of Sec. 7.3.1 results in a loss of precision, especially at high temperature, and requires at least r input points. Results for this variant and an estimate of the loss of precision are shown in Sec. 7.4.7.

7.3.6. Incorporation of prior knowledge

For some spectra, the corresponding complex poles may be very close to the real axis, close to one another, and far from the available sampling points. In such situations, recovering the exact spectral shape at finite precision may be difficult. Nevertheless,

useful information can still be extracted from the moments of the recovered spectrum,

$$\hat{\boldsymbol{A}}_k = \sum_l \boldsymbol{A}_l \xi_l^k \tag{7.25}$$

which can then be combined with prior knowledge to fit the spectrum to an appropriate model. This point will be illustrated in Sec. 7.4.5.

7.3.7. Application to heating and cooling of self-consistent many-body simulations

Apart from obtaining the spectral function, an additional application of this methodology consists in the temperature extrapolation of self-consistent many-body simulations [86]. In these methods, results that are obtained for one value of temperature are then used to initialize the convergence of self-consistent iterations at a different temperature. In the current context, this implies evaluating Eq. (7.1) on the Matsubara grid for a different temperature and using those results to initialize self-consistent iterations. Such a procedure may be needed if multiple stable fixed-points exist, such as in the coexistence regime of a first-order transition. It may also be needed in situations where self-consistent iterations (in a damped, undamped, or accelerated [110] convergence scheme) do not converge to a physical solution.

In these situations, the temperature extrapolation of the matrix-valued propagator results in a 'physical' starting solution which inherits the spectra from a nearby temperature. This procedure may substantially reduce the number of iterations needed for convergence, similar to the interpolation with Carathéodory functions used in Ref. [86].

7.4. Results

7.4.1. Ambiguity issues revisited

As discussed in Sec. 7.2, NAC of Matsubara data involves two primary sources of ambiguity. First, to ensure that the recovered spectrum is controlled to within a prescribed precision ε, the approximated Green's function must be controlled on a continuous interval of the imaginary axis, not only at the discrete Matsubara points. Second, even

if this condition is satisfied, the recovered spectrum remains highly non-unique at any finite precision and may exhibit only coarse-grained similarity to the exact spectrum. In the following, we examine how these two ambiguity issues are addressed in Secs. 7.4.1.1 and 7.4.1.2, respectively.

7.4.1.1. Approximation on the imaginary-frequency interval

We begin by examining whether the first interpolation step described in Sec. 7.3.1 can approximate the Green's function accurately on a continuous interval of the imaginary axis. This is a crucial requirement because the subsequent pole-extraction step involves contour integrals over the interpolated function [Eq. (7.13) or Eq. (7.22)]. Therefore, the response function must be approximated accurately not only at the Matsubara points, but also between them. In our method, this first approximation is obtained from ESPRIT. Although the resulting approximant is not itself a pole representation of the Green's function, it provides a smooth function that can be evaluated anywhere on the Matsubara axis.

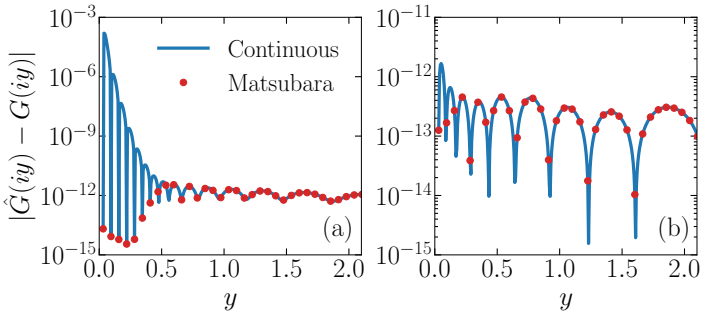

Figure 7.4: Left panel: interpolation of an example Green's function at and between interpolation points, obtained from a least-squares fit to DLR coefficients. Right panel: same quantity obtained using the method of Sec. 7.3.1.

Figure 7.4 compares two different approximations of the same Green's function. The left panel shows a standard least-squares fit to DLR coefficients, while the right panel shows the result obtained using the method described in Sec. 7.3.1. The model considered here is a tight-binding model on the Bethe lattice with hopping t, whose spectral function is semicircular:

$$A(\omega) = \frac{1}{2\pi t^2}\sqrt{4t^2 - \omega^2}\,, \tag{7.26}$$

and whose Green's function has the analytical form

$$G(iy) = \frac{i}{2t^2}(y - \sqrt{y^2 + 4t^2}) \quad \text{for } y > 0\,. \tag{7.27}$$

The comparison is performed at $t = 1$ and $\beta = 100$. The ESPRIT-based approximation shown in the right panel is accurate both at the Matsubara points and between them. In contrast, the standard least-squares fit to DLR coefficients, shown in the left panel, produces large errors at small y despite fitting the data on the Matsubara grid.

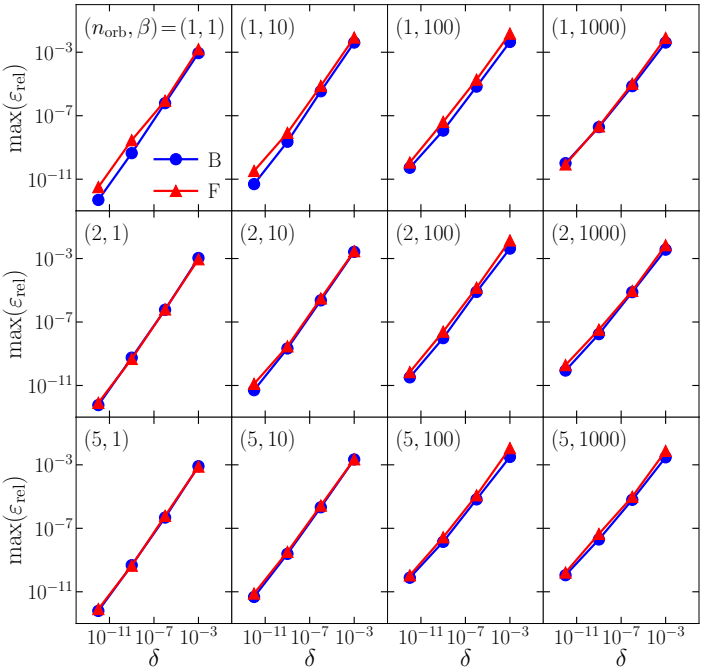

Figure 7.5: Maximum relative deviation $\varepsilon_{\rm rel}$ [Eq. (7.29)] on the imaginary axis, evaluated both at and between Matsubara points, among all random configurations. Results are shown for inverse temperatures $\beta \in \{1, 10, 100, 1000\}$, orbital numbers $n_{\rm orb} \in \{1, 2, 5\}$, and noise levels $\delta \in \{10^{-3}, 10^{-6}, 10^{-9}, 10^{-12}\}$. Blue circles denote bosonic problems, and red triangles denote fermionic problems.

We further test the robustness of this approximation in Fig. 7.5 using random test functions with fermionic and bosonic analytic properties. These functions are generated as follows. First, we randomly choose the number of poles n_p between 1 and $(n_p)_{\rm max}$. Second, we randomly generate n_p real poles ξ_l with $1 \leq l \leq n_p$ within the energy

window $[-\omega_{\max}, \omega_{\max}]$. Third, we randomly assign a matrix-valued weight $\boldsymbol{A}_l$ to each pole ξ_l, requiring $\boldsymbol{A}_l$ to be positive semidefinite for fermions and $\mathrm{sign}(\xi_l)\boldsymbol{A}_l$ to be positive semidefinite for bosons. Finally, we normalize the weights such that $\sum_l \boldsymbol{A}_l = [d_i, d_j^\dagger]_\pm$. This procedure produces random Green's functions with the correct analytic properties.

The exact Matsubara data are then evaluated as $\boldsymbol{G}_{\mathrm{exact}}(i\omega_n) = \sum_l \boldsymbol{A}_l/(i\omega_n - \xi_l)$. To model noisy input data, we add a Gaussian-distributed perturbation to each matrix element:

$$[\boldsymbol{G}_{\mathrm{input}}(i\omega_n)]_{ij} = [\boldsymbol{G}_{\mathrm{exact}}(i\omega_n)]_{ij}[1 + \delta \times N_{\mathbb{C}}(0,1)], \tag{7.28}$$

where $N_{\mathbb{C}}(0,1)$ denotes a complex-valued normal Gaussian random variable. We then obtain $\boldsymbol{G}_{\mathrm{approx}}(i\omega_n)$ from the noisy data $\boldsymbol{G}_{\mathrm{input}}(i\omega_n)$ and compute the maximal relative deviation on the continuous interpolation interval:

$$\varepsilon_{\mathrm{rel}} = \frac{\max\limits_{\omega_{n_0} \le y \le \omega_{n_\omega - 1}} \|\boldsymbol{G}_{\mathrm{approx}}(iy) - \boldsymbol{G}_{\mathrm{exact}}(iy)\|}{\max\limits_{0 \le n \le n_\omega - 1} \|\boldsymbol{G}_{\mathrm{exact}}(i\omega_n)\|} \,. \tag{7.29}$$

This procedure is repeated over many random configurations to obtain a statistical sample.

The results are shown in Fig. 7.5, with fermionic systems shown in red and bosonic systems shown in blue. We consider inverse temperatures $\beta \in \{1, 10, 100, 1000\}$, orbital numbers $n_{\mathrm{orb}} \in \{1, 2, 5\}$, and noise levels $\delta \in \{10^{-3}, 10^{-6}, 10^{-9}, 10^{-12}\}$. For each $(\beta, n_{\mathrm{orb}}, \delta)$, we generate 10^4 random configurations $\{\boldsymbol{A}_l, \xi_l\}$, with $(n_p)_{\max} = 100$ and $\omega_{\max} = 10$.

For all tested temperatures, orbital numbers, noise levels, and random configurations, the ESPRIT-based first approximation becomes systematically more accurate on the full continuous interval $[i\omega_{n_0}, i\omega_{n_\omega - 1}]$ as the noise level is reduced. Since the later stages of the analytic continuation procedure require accurate contour integrals over the Matsubara axis, as in Eq. (7.11), this behavior is a necessary prerequisite for systematically improvable continuations. These results show that the first ambiguity discussed in Sec. 7.2 can be controlled by constructing an approximation that remains accurate not only at Matsubara points, but also between them.

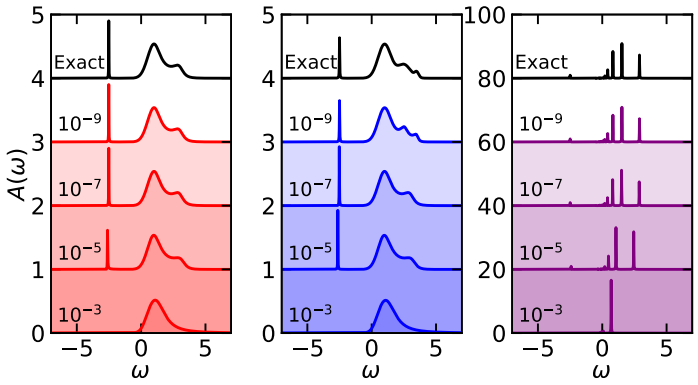

Figure 7.6: Three scenarios in which the exact Matsubara data are indistinguishable above the precision level 10^{-4}. MPM is performed at noise levels $\delta \in \{10^{-3}, 10^{-5}, 10^{-7}, 10^{-9}\}$. The inverse temperature is fixed at $\beta = 100$.

7.4.1.2. Convergence to the correct spectrum

We now examine whether the second ambiguity discussed in Sec. 7.2 can be overcome. To this end, we revisit the three examples introduced there. The corresponding Matsubara data for the three spectral functions are indistinguishable above the precision level 10^{-4}, but the spectra themselves are visibly different. We perform analytic continuation at noise levels $\delta \in \{10^{-3}, 10^{-5}, 10^{-7}, 10^{-9}\}$. For the two continuous spectra, we use the generic mapping in Eq. (7.9); for the discrete spectrum, we use the mapping in Eq. (7.21) to impose the up-down symmetry and accelerate convergence.

As shown in Fig. 7.6, all recovered spectra eventually converge to the exact spectra as the data precision increases. This occurs because the second ESPRIT step extracts the minimal number of complex poles needed to fit the data within the prescribed precision. As a result, features not supported by the input data are not introduced into the recovered spectrum. When the data precision is low, only coarse-grained spectral information is recovered. As the precision increases, finer spectral features become necessary to fit the input data and are therefore incorporated into the minimal pole representation.

Thus, within the minimal pole formalism, the second ambiguity is resolved in a systematic sense: although finite-precision data may still admit many possible spectra, increasing the input precision drives the recovered spectral function toward the correct one. This convergence does not rely on explicit prior knowledge of the spectral shape

and is observed regardless of temperature [38].

7.4.2. Matrix-valued continuations: fermion case

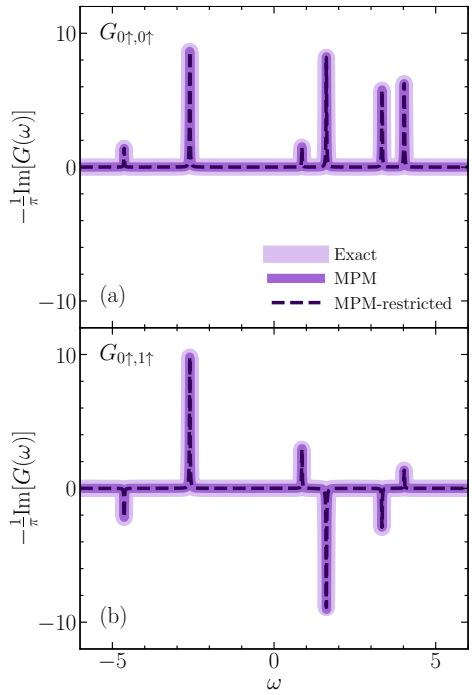

Figure 7.7: Exact solution (light purple, large) and analytic continuation of diagonal (top panel) and off-diagonal (bottom panel) fermionic Matsubara Green's functions of a Hubbard dimer. Continuation is performed using both the mapping (7.9) (MPM, purple) and the mapping (7.21) (MPM-restricted, dashed deep purple). The simulations are conducted at $\beta = 10$.

Next, we show an example of a matrix-valued continuation for a simple multi-orbital system. We analyze the (fermionic) Hubbard dimer given by the Hamiltonian

$$\hat{H} = \hat{H}_0 + \hat{H}_{\text{int}} + \hat{H}_{\text{ext}} + \hat{H}_{\text{sb}}, \tag{7.30}$$

where $\hat{H}_0 = -\sum_\sigma (t\hat{c}^\dagger_{0\sigma}\hat{c}_{1\sigma} + \text{h.c.}) - \sum_{i\sigma}\mu\hat{n}_{i\sigma}$, $\hat{H}_{\text{int}} = \sum_i U\hat{n}_{i\uparrow}\hat{n}_{i\downarrow} - \sum_{i\sigma}\frac{U}{2}\hat{n}_{i\sigma}$, with $\hat{n}_{i\sigma} = \hat{c}^\dagger_{i\sigma}\hat{c}_{i\sigma}$ being the occupation number operator and $i \in \{0, 1\}$. Additional terms $\hat{H}_{\text{ext}} = \sum_i h(\hat{n}_{i\uparrow} - \hat{n}_{i\downarrow})$ and $\hat{H}_{\text{sb}} = U_a(\hat{n}_{0\uparrow}\hat{n}_{0\downarrow} - \hat{n}_{1\uparrow}\hat{n}_{1\downarrow}) + \mu_a(\hat{n}_{0\uparrow} + \hat{n}_{0\downarrow} - \hat{n}_{1\uparrow} - \hat{n}_{1\downarrow}) + h_a(\hat{n}_{0\uparrow} -$

$\hat{n}_{0\downarrow} - \hat{n}_{1\uparrow} + \hat{n}_{1\downarrow}$) are added to break symmetries and generate additional excitations. This system has two orbitals (four spin-orbitals), and an analytic solution for the spectral function is available. The Hubbard dimer is a starting point for studying the extended Hubbard model which is a paradigmatic model of strongly correlated electron systems [111, 112].

Since the energy levels are discrete, sharp resonances, extracting accurate spectral functions using the maximum entropy method is known to be difficult [17].

To illustrate the matrix-valued continuation, we study the system in the absence of noise and evaluate all Green's function data on the Matsubara axis in double precision. The parameters are chosen to match those in Ref. [17], with $t = 1$, $U = 5$, $\mu = 0.7$, $h = 0.3$, $U_a = 0.5$, $\mu_a = 0.2$, and $h_a = 0.03$. We perform exact diagonalization at $\beta = 10$ to obtain both the exact spectral function and the exact Matsubara Green's function $\boldsymbol{G}(i\omega_n)$.

The Matsubara data is then processed using the methodology of Sec. 7.3, with both the generic holomorphic mapping (7.9), which allows poles anywhere on the complex plane, and the restricted mapping (7.21), which restricts them to the real axis.

Fig. 7.7 shows that both methods successfully recover the correct diagonal (top panel) and off-diagonal (bottom panel) results for the Hubbard dimer. The broadening parameter, η, is set to 0.01 for visualization purposes. The recovered spectral functions share the same set of poles, with matrix-valued weights being positive semidefinite.

7.4.3. Matrix-valued continuations: bosonic case

In Fig. 7.8, we obtain similar results for a discrete bosonic response function, using the Hamiltonian of Eq. (7.30), but replacing the fermionic operators d and $d^\dagger$ in Eq. (3.28) with bosonic density operators $\hat{n}_{i\sigma} = \hat{n}_{i\sigma}^\dagger = \hat{c}_{i\sigma}^\dagger \hat{c}_{i\sigma}$. All other parameters remain the same, except for U, which is now set to 2.5 for better visualization. As shown in Fig. 7.8, both restricted and unrestricted methods successfully recover the correct spectra.

Note that while it is straightforward to obtain bosonic continuations with the method of Sec. 7.3, maximum entropy-related methods struggle for bosonic functions due to the conditioning of the bosonic kernel and the fast decay of bosonic response functions [13]. For noise-free data, a generalization of the bosonic scalar Nevanlinna method [91] to matrix-valued Carathéodory functions may also be used.

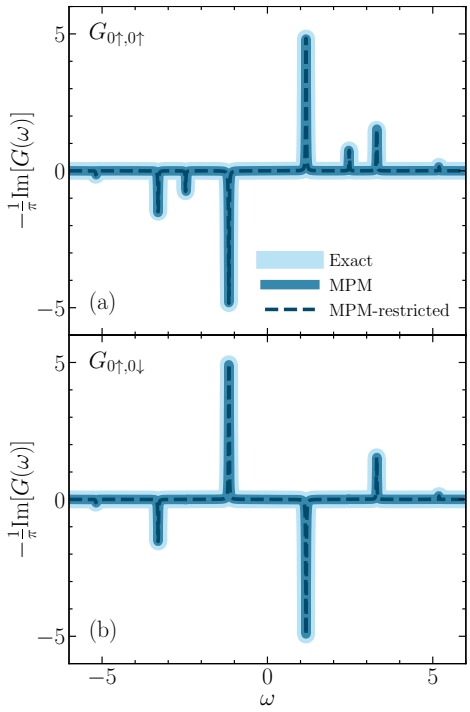

Figure 7.8: Analytic continuation of bosonic Matsubara Green's functions obtained from the exact diagonalization of the same discrete Hubbard dimer system as in Fig. 7.7.

7.4.4. Matrix-valued continuations: continuous system

In Fig. 7.9, we examine the method for the case of a continuous spectral function. Poles of this approximation have to be in the lower half of the complex plane, since poles on the real axis would result in a discrete spectral function.

For the fermion case, we model the diagonal and off-diagonal spectra as $A_{\mathrm{diag}}(\omega) = 0.25g(\omega, -1.5, 0.5)+0.5g(\omega, 0, 0.5)+0.25g(\omega, 1.5, 0.5)$ and $A_{\mathrm{off-diag}}(\omega) = -0.2g(\omega, -1, 0.6) +0.2g(\omega, 1, 0.6)$, where $g(\omega, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(\omega-\mu)^2}{2\sigma^2}\right)$ is the Gaussian function. For the bosonic case, we use $A_{\mathrm{diag}}(\omega) = -0.6g(\omega, -1.2, 0.8)+0.6g(\omega, 1.2, 0.8)$ and $A_{\mathrm{off-diag}}(\omega) = -0.13g(\omega, -2, 0.5) + 0.1g(\omega, -1, 1) - 0.1g(\omega, 1, 1) + 0.13g(\omega, 2, 0.5)$. The Matsubara Green's functions are calculated from Eq. (3.40) at $\beta = 20$. The matrix-valued Matsubara data are then processed with the mapping (7.9).

These superpositions of Gaussians are not easily approximated by sums of poles, as the pole spectra correspond to Lorentzians. Nevertheless, both fermionic and bosonic

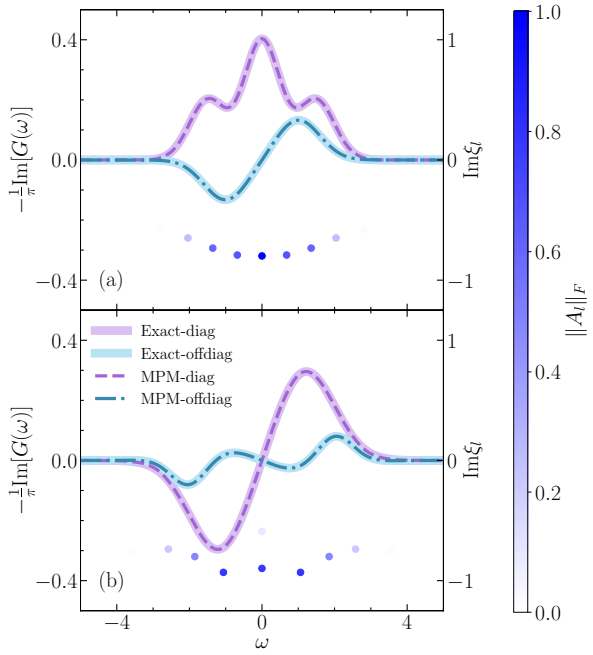

Figure 7.9: Analytic continuation of a continuous spectral function. Top panel: fermionic case; bottom panel: bosonic case. Blue circles represent the locations of the shared poles, with their weights indicated by the color bar. The simulations are carried out at $\beta = 20$.

spectra are recovered well with shared poles. Fig. 7.9 shows the diagonal (thick purple lines) and off-diagonal (thick blue lines) spectra along with the recovered analytic continuation of the matrix-valued problem (dashed purple and dot-dashed blue lines). The fermion problem is shown in the top graph, the bosonic problem in the bottom graph, both evaluated at $\eta = 0^+$. Also shown are the pole locations ξ_l and the Frobenius norm of the pole strength, as indicated in the color bar on the right. Note that poles are shared between diagonal and off-diagonal Green's function components.

Fig. 7.10 analyzes the sensitivity of the analytic continuation to random noise. We evaluate the robustness to noise by adding Gaussian random noise $\delta \in \{10^{-2}, 10^{-4}, 10^{-6}\}$ to the input Matsubara, as described in Eq. (7.28). In agreement with the scalar-valued case [38], the main features of the spectra are well reproduced even with relatively high noise levels. As the noise level decreases, both the diagonal and off-diagonal spectra rapidly converge to the exact result.

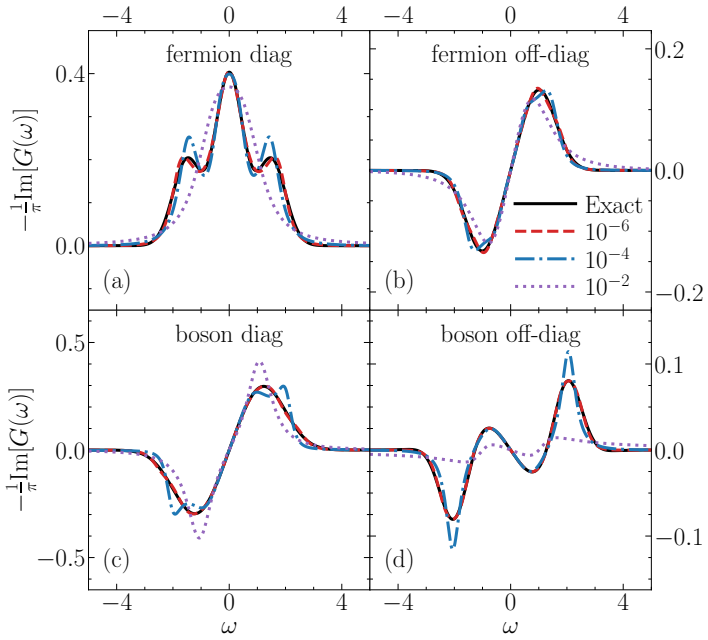

Figure 7.10: Noise robustness analysis of the MPM algorithm, performed by adding Gaussian random noise with amplitudes $\delta \in \{10^{-2}, 10^{-4}, 10^{-6}\}$. (a) Diagonal and (b) off-diagonal fermionic case. (c) Diagonal and (d) off-diagonal bosonic case.

7.4.5. Recovering extreme spectra with prior knowledge

Although MPM can recover the correct spectrum as the data precision increases, some spectra may be so extreme that the precision required to observe convergence becomes prohibitively high. In this section, we demonstrate how prior knowledge can be incorporated in such situations.

We consider the example examined in Ref. [29]. The spectral function is given by $\boldsymbol{A}(\omega) = g(\omega, 2.93, 0.064)$, which is sharp, non-Lorentzian, and far from the imaginary axis. Fig. 7.11(a) compares the exact and recovered spectra and shows a significant discrepancy. The origin of this discrepancy is illustrated in the inset: the corresponding Green's functions on the imaginary axis $(-i\infty, i\infty)$ are indistinguishable up to 12-digit precision, indicating that the spectra in this case are highly non-unique at double precision. Therefore, without additional prior knowledge, 64-bit floating-point precision is insufficient to recover the correct spectrum.

If prior knowledge is available, however, the convergence can be significantly ac-

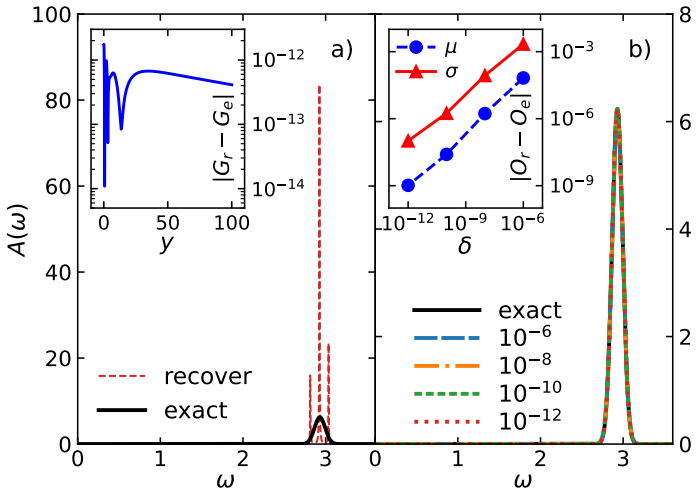

Figure 7.11: An extreme example of a spectral function that is sharp, non-Lorentzian, and far from the imaginary axis. (a) Continuation performed at 64-bit floating-point precision without introducing prior knowledge. The inset in (a) shows the difference between the corresponding Green's functions on the imaginary axis $[0, i\infty)$. (b) Continuation after incorporating prior knowledge of the Gaussian peak, performed at different noise levels $\delta \in \{10^{-6}, 10^{-8}, 10^{-10}, 10^{-12}\}$. The inset in (b) shows the convergence of the recovered mean value μ and standard deviation σ of the Gaussian peak as the noise level δ decreases.

celerated. Fig. 7.11(b) shows the results obtained after incorporating the knowledge that the spectrum consists of a Gaussian peak. The continuation is performed at noise levels $\delta \in \{10^{-6}, 10^{-8}, 10^{-10}, 10^{-12}\}$. The recovered pole information $\{\boldsymbol{A}_l, \xi_l\}$ is first used to compute the spectral moments $\hat{\boldsymbol{A}}_k$ via Eq. (7.25). The mean value and standard deviation of the Gaussian peak are then estimated as $\mu \approx \hat{\boldsymbol{A}}_1/\hat{\boldsymbol{A}}_0$ and $\sigma \approx \sqrt{\hat{\boldsymbol{A}}_2/\hat{\boldsymbol{A}}_0 - (\hat{\boldsymbol{A}}_1/\hat{\boldsymbol{A}}_0)^2}$, respectively. As shown in Fig. 7.11(b), already at a noise level of 10^{-6}, the width of the Gaussian peak, i.e., its standard deviation, can be reliably recovered. As the noise level decreases further, both μ and σ converge rapidly. For $\delta \geq 10^{-5}$, the Gaussian width cannot be reliably recovered, consistent with the conclusion of Ref. [29].

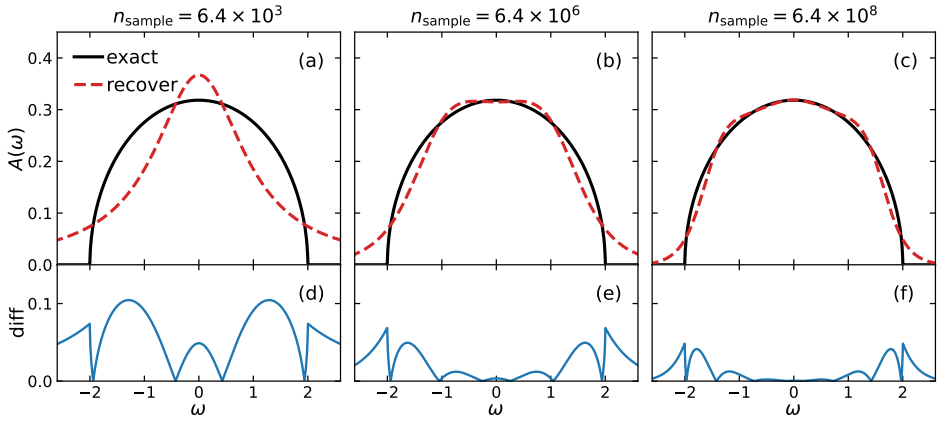

Figure 7.12: MPM continuation of Quantum Monte Carlo (CT-HYB) data obtained from the single impurity Anderson model embedded in an infinite-dimensional Bethe lattice. The upper panels compare the recovered and exact spectra for different numbers of Monte Carlo samples, while the lower panels show the corresponding differences. Simulations are performed at $\beta = 20$.

7.4.6. Application to quantum Monte Carlo data

We further test the method on noisy Quantum Monte Carlo data obtained from a continuous-time hybridization expansion (CT-HYB) simulation [113]. The system considered here is the single impurity Anderson model embedded in an infinite-dimensional Bethe lattice. The simulations are performed at inverse temperature $\beta = 20$, and analytic continuation is carried out for different numbers of Monte Carlo samples, n_{sample}.

The upper panels of Fig. 7.12 show that the quality of the recovered spectrum improves systematically as the number of samples increases, or equivalently, as the precision of the input data improves. The lower panels show the corresponding differences between the exact and recovered spectra. The error decreases with increasing n_{sample}, confirming that the continuation improves as the statistical uncertainty of the CT-HYB data is reduced. This example demonstrates that the method can be applied to noisy Monte Carlo data and that the recovered real-frequency spectrum becomes more reliable as the input data quality improves.

7.4.7. Application to scalar and matrix-valued real-materials data

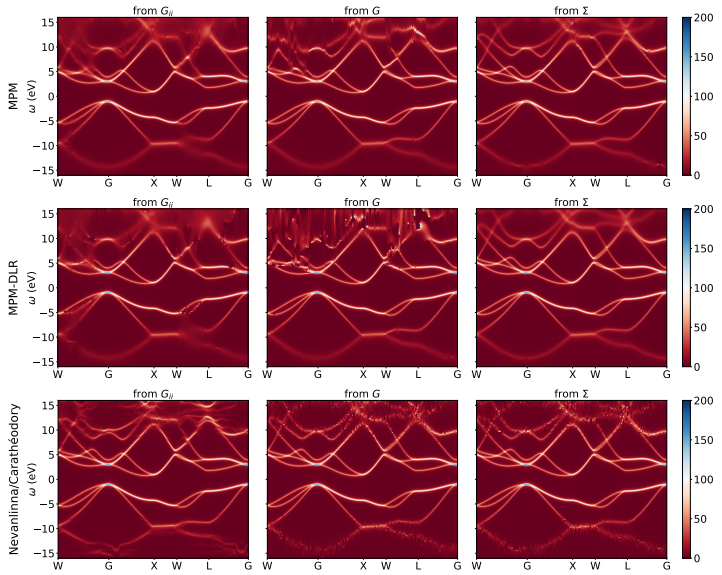

Figure 7.13: Spectral function (band structure) of Si obtained by analytic continuation of self-consistent GW data [17, 95] at inverse temperature $\beta = 700\ \mathrm{Ha}^{-1}$. We use the same input Matsubara data for all plots and, for clearer visualization, evaluate the spectral function at $\eta = 0.005$ above the real axis, introducing an artificial broadening. Left panel: scalar continuation of the diagonal elements of the Green's function. Middle panel: matrix-valued continuation of the Green's function. Right panel: matrix-valued continuation of the self-energy followed by a Dyson equation on the real axis to obtain the real-frequency Green's function. Top set of plots: continuation with the method presented in Sec. 7.3. Middle set of plots: evaluation with DLR coefficients as presented in Sec. 7.3.5. Bottom panel: evaluation with Nevanlinna [16] and Carathéodory [17] methods.

As a final example, we present an application of the method to a real materials simulation. Response functions in such simulations typically spread a large range of energies and are therefore very sensitive to inaccuracies in the real- or imaginary-frequency grids and the continuation method.

We choose crystalline Si as an example system, simulated in the fully self-consistent GW approximation with the GREEN open-source software package [83]. Simulations are carried out in Gaussian orbitals (basis set gth-dzvp-molopt-sr [114] with gth-pbe

pseudopotential [115]), with integrals generated by the PySCF package [116]. Calculations are performed on a $6 \times 6 \times 6$ k-point grid with 26 orbitals per unit cell. After achieving convergence, the orbitals are interpolated on a grid with 200 k points along a high-symmetry path in reciprocal space and then transformed into an orthogonal orbital basis. The simulations are conducted at $\beta = 700$ Ha^{-1}, where Ha denotes the Hartree energy. To perform the continuation, the Matsubara data are evaluated [80] on the first 1000 positive Matsubara frequencies from 104 IR frequencies [79].

We present results of the continuation presented in this chapter in the top panel of Fig. 7.13. The top left panel displays the sum of the scalar analytic continuation of the diagonal elements. In this approach, each continuation is carried out as described in Ref. [38], with n_0 automatically determined using the method outlined in Sec. 7.3.1. The middle panel presents results from a matrix-valued continuation where the Green's functions share common poles, and all diagonal and off-diagonal entries are simultaneously continued, respecting the positive semidefinite pole structure. Finally, the right panel shows results from the matrix-valued continuation of the self-energy, followed by the evaluation of the Dyson equation on the real axis to obtain the real-frequency Green's function. This procedure is analogous to the one described in Ref. [17]. The poles are allowed to lie anywhere in the lower half of the complex plane.

Results from the accelerated algorithm using DLR coefficients for diagonal, matrix, and self-energy continuation are shown in the middle row of Fig. 7.13.

In the lower panel of Fig. 7.13, we present results from state-of-the-art methods: the Nevanlinna continuation [16] for scalar-valued cases and the Carathéodory continuation [17] for matrix-valued cases, both performed directly on the IR grid. The broadening parameter η is set to 0.005 for all plots to ensure a consistent comparison.

All continuation methods result in sharp bands and consistent band gaps; the degeneracies of the bands at high-symmetry points and along high-symmetry directions are accurately recovered. Green's function and self-energy continuations agree on the value of band gaps and the main spectral features. High-lying bands are generally sharper in the continuations of the self-energy. The DLR based algorithm presented in Sec. 7.3.5 typically generates a less clear band structure than the algorithm employing the method of Sec. 7.3.1 for both the scalar and the matrix continuation of the Green's function. In contrast, when continuing from the self-energy, the two methods exhibit nearly identical resolution, which is observed to be superior to that of the Green's function continuation

and comparable to results from the Nevanlinna or Carathéodory continuation.

We attribute the above observations to the following reasons. First, since different elements of the Green's function have similar accuracy, including off-diagonal information in the matrix-valued continuation shown in the middle panels provides little, if any, improvement over the scalar continuation shown in the left panels. Second, the DLR variant can be viewed as a faster version of MPM with a larger n_0, meaning that more low-frequency Matsubara points are discarded. It therefore usually runs faster, but produces less accurate results than the original MPM for both the scalar and matrix-valued continuations of the Green's function. On the other hand, the spectral density of the self-energy in this example is much smoother and contains fewer sharp features than that of the Green's function. Its convergence with respect to the number of complex poles is therefore much faster, so the precision of the input data is more than sufficient. As a result, the DLR variant yields a band structure of comparable quality to that obtained from the original MPM, while requiring substantially less computational time. Note that, to eventually obtain the Green's function, one must continue the full self-energy matrix, since the Dyson equation involves a matrix inverse. This justifies the matrix-valued continuation developed in this work.

All results from the minimal pole method and its DLR variant are obtained using the mapping in Eq. (7.9). Due to the sharp features present in the band structure, one can alternatively use the mapping in Eq. (7.21) to restrict the poles to the real axis when continuing scalar and matrix-valued Green's functions.

Compared to the Carathéodory continuation method discussed in Ref. [17], which interpolates a causal matrix-valued solution, the fitting approach presented here operates in double precision and is robust to noise. Due to the need for multi-precision arithmetic, results for the Carathéodory continuation as implemented by the TRIQS Nevanlinna package [117] [bottom panel of Fig. 7.13] required ten times more CPU time than the results based on the method introduced in Sec. 7.3.1 [top panel of Fig. 7.13]. Furthermore, the continuation based on the DLR expansion, presented in Sec. 7.3.5 [middle panel of Fig. 7.13], completed in minutes on a single core, resulting in an additional computational speedup of three orders of magnitude.

7.5. Conclusions

The analytic continuation of matrix-valued correlation functions of data from finite-temperature field theories is essential for interpreting spectral functions in systems with off-diagonal correlations, for evaluating derived quantities such as optical response functions on the real axis, and for interpreting anomalous order in superconducting systems.

This chapter presents a theoretical framework and numerical method based on the construction of an approximation with shared poles for analyzing and continuing such correlation functions. We show that the method works well in a variety of setups, including fermionic and bosonic, discrete and continuous, as well as model and real materials systems. Unlike methods based on interpolation with causal functions [17], this method is robust to noise and eliminates the need for multi-precision arithmetic. In contrast to methods based on smoothness principles [13], it is capable of resolving both sharp and smooth features.

Numerical analytic continuation remains an ill-posed problem. In practice, whether the minimum information principle imposed by the pole approximation presented here leads to a suitable spectral function for a system at hand needs to be evaluated on a case-by-case basis.

The representation and compression of spectral functions on the real axis with a minimum number of complex poles using the methods presented here, and the solution of dissipative systems based on these pole representations are closely related problems that will be examined in future work.

Our work is accompanied by an open-source software implementation of all methods presented here, written in the programming language Python [118, 119].

Chapter 8

Real-Frequency Bath Fitting

This chapter is based on *Lei Zhang, André Erpenbeck, Yang Yu and Emanuel Gull, Minimal pole representation for spectral functions, J. Chem. Phys. 162, 214111 (2025)* [40].

8.1. Introduction

The accurate representation of a continuous quantum system by a few discrete degrees of freedom is a fundamental problem in condensed matter science. Applications of this problem appear whenever the solution of quantum impurity problems requires a Hamiltonian representation [120–123] with a few degrees of freedom [120, 124–127]; whenever transport in open systems is modeled by coupling dissipative bath states to a correlated subsystem [128–130], and wherever fast quasi-analytic calculations in a Green's function language are needed [104].

Compact and accurate representations of Green's functions are essential in computational many-body simulations [76, 77, 79, 80, 82, 131–133]. A compact and accurate representation respecting the analytic properties of equilibrium Green's function has recently been introduced [38, 39] in the context of analytic continuation [13, 16, 17, 35–37, 100] of imaginary-frequency Matsubara Green's functions and is based on the realization that most objects of interest in many-body theory, such as retarded or advanced Green's functions, susceptibilities, and spectral functions are related to Nevanlinna functions [16, 17]. The retarded Green's function in particular is analytic in the upper half of the complex plane, its imaginary part coincides (up to a constant prefactor) with the spectral function just above the real axis, and all of its poles lie in the lower half of

the complex plane. With the help of suitable holomorphic mappings, the problem of accurately approximating the Green's function in the upper half plane simplifies to an exponential approximation problem [36–39], for which well-established signal processing methodology exists [47, 51, 54].

Although functions known directly on the real axis do not suffer from the ill-conditioning of the analytic continuation problem, in practical applications they, too, benefit from systematically improvable compact representations in terms of poles in the complex plane. Such representations enable fast calculations of diagrams in perturbation theory [104, 134], fast propagation of real-time quantities in hierarchical equation of motion calculations [128–130], and a systematic analysis of the features contained in spectral data [38, 39].

This chapter will present systematic methods for constructing approximations of spectral functions from real-frequency spectral data and show applications to real-frequency spectra, as well as their real-time Fourier transforms. The remainder of the chapter is organized as follows. In Sec. 8.2, we introduce the minimal pole method (MPM) for constructing compact approximations of spectral functions. Section 8.3 presents numerical results that demonstrate the performance of the method on representative spectral functions, Green's functions in the complex plane, and bath correlation functions for both synthetic and structured spectral data. Finally, Sec. 8.4 provides a summary and concluding remarks.

8.2. Method

The main object of interest in this work is a real-valued spectral function $A^{\mathrm{ex}}(\omega)$ known on the real axis, $\omega \in \mathbb{R}$. We aim to find an approximation to this function consisting of $2M$ poles in the complex plane of the form

$$A(z) = \sum_{l=1}^{M} \left(\frac{A_l^{(\mathrm{dn})}}{z - \xi_l^{(\mathrm{dn})}} + \frac{A_l^{(\mathrm{up})}}{z - \xi_l^{(\mathrm{up})}} \right) , \tag{8.1}$$

with $z \in \mathbb{C}$ and $A(z = \omega) \approx A^{\mathrm{ex}}(\omega)$, where $A_l^{(\mathrm{dn})}$ and $\xi_l^{(\mathrm{dn})}$ are complex weights and pole locations in the lower half of the complex plane, while $A_l^{(\mathrm{up})}$ and $\xi_l^{(\mathrm{up})}$ are their counterparts in the upper half-plane.

Since the spectral function is real, the poles satisfy the following symmetry:

$$\xi_l^{(\mathrm{up})} = (\xi_l^{(\mathrm{dn})})^* \quad \text{and} \quad A_l^{(\mathrm{up})} = (A_l^{(\mathrm{dn})})^* \,, \tag{8.2}$$

and we restrict most of the following discussion to recovering the M poles in the lower half of the complex plane. These poles are related to those of the corresponding retarded Green's function.

8.2.1. Minimal Pole Method

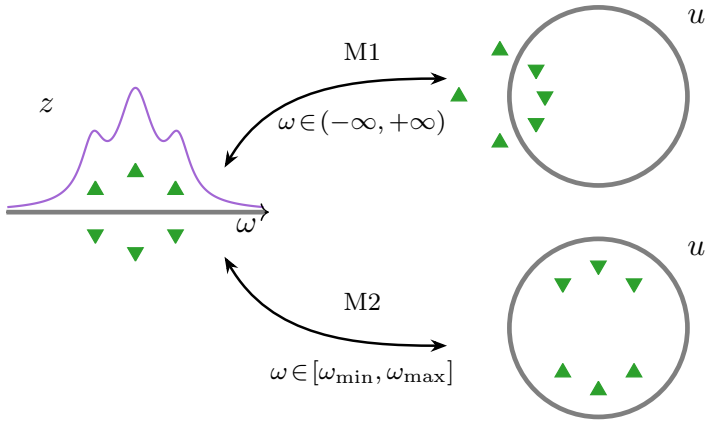

Figure 8.1: Left: The original complex plane z with the spectral function on the real-frequency axis (purple) and pole locations (green triangles). Right: Representations using two different mappings (M1 and M2). Top right: The complex plane u after applying the holomorphic mapping (8.3), which transforms the entire real-frequency range $(-\infty, +\infty)$ onto the unit circle. This mapping places poles from the lower (upper) half-plane into the interior (exterior) of the unit circle. Bottom right: Similar to M1, but employing the holomorphic mapping (8.4). This transformation maps the finite real-frequency interval $[\omega_{\mathrm{min}}, \omega_{\mathrm{max}}]$ onto the unit circle, positioning all poles inside the unit circle.

As first discussed in the context of analytic continuation in Refs. [36, 37], applying a holomorphic mapping to Matsubara data followed by the application of the residue theorem reformulates the continuation problem into an exponential approximation or 'Prony' problem. More recently, Refs. [38, 39] demonstrated that once a continuous imaginary interval is adequately approximated from discrete Matsubara data, this methodology

ensures systematic convergence of the spectral function with increasing data precision. The spectral approximation problem on the real axis can be formulated in much the same way, with the main difference between the analytic continuation and the real-frequency problem being the holomorphic mappings. In this chapter, we abbreviate the minimal pole method as MPM.

Among the infinitely many possible choices, we find the following two holomorphic mappings particularly useful for the real-frequency problem: one is used to construct a controlled approximation over the entire real-frequency axis $\omega \in (-\infty, +\infty)$, while the other is designed for controlled approximation over a finite interval $\omega \in [\omega_{\min}, \omega_{\max}]$ of interest. We note that other mappings are possible.

For the former case, which takes the spectral weight of $A(\omega)$ over the entire frequency range as input, we propose applying the Möbius transform

$$\begin{cases} u &= f(z) &= \frac{z+i\omega_p}{z-i\omega_p} \\ z &= f^{-1}(u) &= i\omega_p \frac{u+1}{u-1} \end{cases}, \tag{8.3}$$

where $\omega_p \in \mathbb{R}^+$ is a free parameter, chosen in practice such that the mapped poles are well separated—typically on the same order as the width of the energy window (see Sec. 8.3 for concrete examples). We denote this mapping as M1.

For the latter case, where $A(\omega)$ has compact support in the interval $[\omega_{\min}, \omega_{\max}]$, we propose applying the transform

$$\begin{cases} u &= f(z) &= z_{\mathrm{s}} + \sqrt{z_{\mathrm{s}}^2 - 1} \text{ with } z_{\mathrm{s}} = \frac{z-\omega_{\mathrm{m}}}{\Delta\omega_{\mathrm{h}}} \\ z &= f^{-1}(u) &= \frac{\Delta\omega_{\mathrm{h}}}{2}(u + \frac{1}{u}) + \omega_{\mathrm{m}} \end{cases}, \tag{8.4}$$

where $\omega_{\mathrm{m}} = (\omega_{\min} + \omega_{\max})/2$, $\Delta\omega_{\mathrm{h}} = (\omega_{\max} - \omega_{\min})/2$, and the branch of the square root is chosen such that $|u| \leq 1$. We denote this mapping as M2. Illustrations of both M1 and M2 are shown in Fig. 8.1, with a detailed performance comparison provided in Sec. 8.3.

By applying these mappings $u = f(z)$, the real axis or the finite interval are mapped onto the unit circle $\partial\bar{D}$ in the complex u-plane and Eq. (8.1) transforms into

$$A'(u) = \sum_{l=1}^{M} \left(\frac{A'_l(dn)}{u - \xi'_l(dn)} + \frac{A'_l(up)}{u - \xi'_l(up)} \right) + \text{analytic part}, \tag{8.5}$$

where $A(z) \equiv A'(u)$. Since the function is known on the unit circle of the mapped plane, one may apply the residue theorem to extract the location of the mapped poles via contour integrals along the unit circle. As shown in Refs. [36, 37], after defining the contour integral h_k as

$$h_k := \frac{1}{2\pi i} \int_{\partial \bar{D}} du\, A'(u) u^k, \quad k \geq 0, \tag{8.6}$$

the residue theorem implies

$$h_k = \sum_l A'_l \xi'^k_l \,, \tag{8.7}$$

which is a Prony-type problem (see Ch. 4 for a discussion of Prony approximation problems and their solution). Here, $\{A'_l, \xi'_l\}$ represent the poles mapped inside the unit circle. As illustrated in Fig. 8.1, the poles inside the unit circle correspond to those mapped from the lower half-plane in M1, and to all poles in M2.

For the mapping M1 of Eq. (8.3), the contour integral Eq. (8.6) simplifies to

$$h_k = \frac{1}{2\pi} \int_0^{2\pi} d\theta\, A(\omega_p \cot(\theta/2)) e^{i(k+1)\theta}. \tag{8.8}$$

Similarly, the contour integral Eq. (8.6) for M2 of Eq. (8.4) simplifies to

$$h_k = \frac{1}{\pi} \int_0^{\pi} d\theta A(\omega_\mathrm{m} + \Delta\omega_\mathrm{h} \cos\theta) \cos(k+1)\theta. \tag{8.9}$$

Given spectral functions on the real axis, the integrals h_k ($k = 0, 1, 2, \ldots$) can be numerically evaluated to high precision using standard quadrature rules. Since the poles $\{A'_l, \xi'_l\}$ lie strictly inside the unit circle, h_k decays with increasing k according to Eq. (8.7). Therefore, the computation of h_k can be terminated once the target precision ε is reached—i.e., when $|h_k| \leq \varepsilon$—or when a maximum number of terms k_max is reached if h_k decays slowly, whichever occurs first.

Knowing h_k, the Prony-type approximation problem of Eq. (8.7) can then be solved with standard solution methods for the Prony problem, resulting in M values $A'_l(dn)$ and $x'_l(dn)$. M is either determined a priori as a fixed number of poles, or adjusted dynamically such that a target precision is reached. Ch. 4 contains more details for the Prony problem and its solution with the ESPRIT method.

Finally, the values $x_l^{(\mathrm{dn})}$ are recovered using the inverse transform [36, 37]

$$\begin{aligned}\xi_l^{(\mathrm{dn})} &= f^{-1}(\xi_l'(dn)) \\ &= \begin{cases} i\omega_p \frac{\xi_l'(dn)+1}{\xi_l'(dn)-1} & \text{, for M1} \\ \frac{\Delta\omega_\mathrm{h}}{2}(\xi_l'(dn) + \frac{1}{\xi_l'(dn)}) + \omega_\mathrm{m} & \text{, for M2} \end{cases} ,\end{aligned} \tag{8.10}$$

and $A_l^{(\mathrm{dn})}$ are obtained using the relation [38, 39]

$$\begin{aligned}A_l^{(\mathrm{dn})} &= \left.\frac{dz}{du}\right|_{\xi_l'(dn)} \times A_l'(dn) \\ &= \begin{cases} -\frac{2i\omega_p}{(\xi_l'(dn)-1)^2} A_l'(dn) & \text{, for M1} \\ \frac{\Delta\omega_\mathrm{h}}{2}\left(1 - \frac{1}{\xi_l'(dn)^2}\right) A_l'(dn) & \text{, for M2} \end{cases} .\end{aligned} \tag{8.11}$$

This procedure yields the poles in the lower half of the complex plane. The poles in the upper half-plane are determined using the symmetry relation of Eq. (8.2).

In the case of multi-orbital systems and matrix-valued retarded Green's functions, it is often physically insightful to approximate the system using shared complex poles with matrix-valued weights. In this case, a matrix-valued generalization of the ESPRIT algorithm [39] can be employed.

8.2.2. Applications

8.2.2.1. From the spectral function to the Green's function on the complex plane

The Green's function $G(z)$ in the frequency domain is related to the spectral function $A(\omega)$ by [1, 3, 5, 6, 135]

$$\begin{cases} G(z) = \int_{-\infty}^{+\infty} d\omega \frac{A(\omega)}{z-\omega}, \\ A(\omega) = -\frac{1}{\pi}\mathrm{Im}[G(\omega + i0^+)]. \end{cases} \tag{8.12}$$

The retarded and the advanced Green's functions on the real axis are related to $G(z)$ as

$$G^{\mathrm{ret/adv}}(\omega) = G(\omega \pm i0^+) \, . \tag{8.13}$$

When evaluated on the Matsubara frequencies $i\omega_n$ with $\omega_n = (2n+1)\pi/\beta$ for fermionic and $\omega_n = 2n\pi/\beta$ for bosonic Green's functions, $n \in \mathbb{Z}$, the function $G(z)$ corresponds to the Matsubara Green's function:

$$G^{\mathrm{Mat}}(i\omega_n) = G(i\omega_n) \; . \tag{8.14}$$

Below, we discuss the relationship between the poles in Eq. (8.1) and $G(z)$.

The real and imaginary parts of $G(z)$ are closely related to each other. We divide the discussion here into two cases: First, the case close to the real axis, $z = x \pm i0^+$ and second the general case where $\mathrm{Im}z \neq 0$ is finite.

The Hilbert transform along the real axis is defined as $\mathcal{H}f(x+iy) = \frac{1}{\pi}\mathcal{P}\int_{-\infty}^{+\infty} dx' \frac{f(x'+iy)}{x-x'}$, where $\mathcal{P}$ denotes the Cauchy principal value. For $z = x \pm i0^+$, using the Sokhotski–Plemelj theorem $\frac{1}{x\pm i0^+} = \mp i\pi\delta(x) + \mathcal{P}(\frac{1}{x})$ and the properties of the Hilbert transform: $\mathcal{H}\delta(x) = \frac{1}{\pi x}$ and $\mathcal{H}\mathcal{H}f \equiv -f$, the real and imaginary parts of the retarded/advanced Green's function satisfy $\mathrm{Im}G(\omega \pm i0^+) = \pm\mathcal{H}\mathrm{Re}G(\omega \pm i0^+)$, which are the Kramers-Kronig relations [44–46]. For $\mathrm{Im}z$ away from the real axis, since $\mathcal{H}\frac{|y|}{x^2+y^2} = \frac{x}{x^2+y^2}$, it can also be shown that $\mathrm{Im}G(x \pm iy) = \pm\mathcal{H}\mathrm{Re}G(x \pm iy)$ for $y > 0$. In summary, the real and imaginary parts of the Green's function exhibit different parities in the upper- and lower-half of the complex plane:

$$\mathrm{Im}G(z) = \mathrm{sign}(\mathrm{Im}z)\; \mathcal{H}\mathrm{Re}G(z) \; . \tag{8.15}$$

In addition, due to the symmetry

$$G(z^*) = G(z)^* \; , \tag{8.16}$$

it suffices to focus on recovering $G(z)$ in the upper half-plane. Because of the symmetry in Eq. (8.2), $A(\omega)$ can be reconstructed solely from poles in the lower half-plane.

Furthermore,

$$
\begin{aligned}
2\sum_{l=1}^{M}\frac{A_l^{(\mathrm{dn})}}{\omega+i0^+-\xi_l^{(\mathrm{dn})}} &= A(\omega)+i\mathcal{H}A(\omega)\\
&= -\frac{1}{\pi}\mathrm{Im}G(\omega+i0^+)+\frac{i}{\pi}\mathrm{Re}G(\omega+i0^+)\\
&= \frac{i}{\pi}G(\omega+i0^+)\ .
\end{aligned}
\tag{8.17}
$$

Therefore, poles in the lower half-plane provide a natural way to perform the Hilbert transform, and the retarded Green's function can be constructed as

$$
G(\omega+i0^+) = -2\pi i\sum_{l=1}^{M}\frac{A_l^{(\mathrm{dn})}}{\omega+i0^+-\xi_l^{(\mathrm{dn})}}. \tag{8.18}
$$

Replacing $\omega+i0^+$ with z for $\mathrm{Im}z>0$ in Eq. (8.18) gives an approximation of $G(z)$ in the upper half-plane

$$
G(z) = -2\pi i\sum_{l=1}^{M}\frac{A_l^{(\mathrm{dn})}}{z-\xi_l^{(\mathrm{dn})}}, \quad \mathrm{Im}z>0. \tag{8.19}
$$

As we will show, $A(\omega)$ can be approximated by complex poles to arbitrary precision over $\omega\in(-\infty,+\infty)$ by utilizing the M1 mapping. By Eq. (8.18), the approximation of $G(z)$ at $z=\omega+i0^+$ ($\omega\in(-\infty,+\infty)$) can also be achieved to arbitrary precision. Since both the exact Green's function in Eq. (8.12) and the approximation in Eq. (8.19) are analytic for $\mathrm{Im}\,z>0$ and their differences on the boundary can be made arbitrarily small, the Green's function throughout the upper half-plane can be fully described by $A_l^{(\mathrm{dn})}$ and $\xi_l^{(\mathrm{dn})}$ with arbitrary precision according to the maximum modulus principle. This confirms that the complex pole representation is numerically sufficient to capture all features of $G(z)$, which forms the theoretical foundation for works based on the complex pole representation, such as Refs. [36–39, 136, 137].

8.2.2.2. Decomposition of exponentials for real-time evolution

In the simulation of open quantum systems, the bath correlation function (BCF), denoted as $C(t)$, plays a central role as it fully characterizes the influence of the environ-

ment on the system of interest. Simulation methods, such as the hierarchical equations of motion (HEOM) [56, 128, 130, 138–146], the hierarchy of pure states [147], the pseudomodes approach [148], auxiliary master equation methods (AMEA) [129, 149, 150], and the recently developed quasi-Lindblad theory [108], rely on its efficient expansion of exponentials,

$$\begin{aligned} C(t) &= \frac{1}{2\pi}\int_{-\infty}^{+\infty} d\omega J(\omega)\left[\coth\left(\frac{\beta\omega}{2}\right)+1\right]e^{-i\omega t} \\ &\approx \sum_{l=1}^{M}\eta_l e^{-\gamma_l t}\ . \end{aligned} \tag{8.20}$$

The first identity follows from the fluctuation–dissipation theorem for a system coupled to a bosonic reservoir, where $J(\omega)$ represents the bath spectral density. The second line expresses $C(t)$ as a finite sum of decaying complex exponentials, with $\gamma_l \in \mathbb{C}$ ($\mathrm{Re}[\gamma_l] > 0$) and complex weights $\eta_l \in \mathbb{C}$.

Defining the effective spectral function as

$$A(\omega) = J(\omega)\left[\coth\left(\frac{\beta\omega}{2}\right)+1\right]\ , \tag{8.21}$$

and decomposing it into a sum of poles as in Eq. (8.1), the residue theorem ensures that the approximation in Eq. (8.20) can be recovered as

$$\eta_l = -iA_l^{(\mathrm{dn})}, \quad \gamma_l = i\xi_l^{(\mathrm{dn})}\ . \tag{8.22}$$

A similar procedure can be readily applied in the zero-temperature limit. In this case, Eq. (8.20) simplifies to

$$C(t) = \frac{1}{\pi}\int_0^{+\infty} d\omega\, J(\omega)e^{-i\omega t}\ . \tag{8.23}$$

One can then define the effective spectral function as $A(\omega) = 2J(\omega)$ for $\omega > 0$, and $A(\omega) = 0$ otherwise. The relation in Eq. (8.22) remains valid in this limit.

In practice, when using the mapping MPM (M1), $A(\omega)$ is accurate across the entire real-frequency axis, ensuring that $C(t)$ remains accurate across the entire real-time axis. In contrast, when using MPM (M2), since the approximation is only accurate over a

finite interval, there exists a time cutoff t_c beyond which the approximation of $C(t)$ becomes less accurate. The cutoff t_c can be identified by adjusting the L parameter in the numerical solution of the problem Eq. (8.7) with ESPRIT (see Ch. 4). Let $\hat{C}^{(L_1)}(t)$ and $\hat{C}^{(L_2)}(t)$ denote two approximations of $C(t)$ obtained using different L values. If

$$|\hat{C}^{(L_1)}(t) - \hat{C}^{(L_2)}(t)| \leq \varepsilon \ , \tag{8.24}$$

the approximation is assumed to be accurate at time t. Conversely, if the difference is uncontrolled, the approximation is considered unreliable. In practice, we use $L_1 = 2N/5$ and $L_2 = N/2$ in all our implementations.

8.3. Results

8.3.1. Approximation of typical spectral functions

We evaluate the pole approximation method by examining three paradigmatic spectral functions that are analytically known on the real axis: a Kondo-like structure, a bosonic multi-orbital spectral function, and a semicircular density of states for MPM (M1). For convenience, we adopt atomic units in Secs. 8.3.1 and 8.3.2, as is customary in studies of model Hamiltonians within the condensed matter community. The free parameter ω_p in the mapping is set to $\omega_p = 2$. For the calculation of the contour integral, we use $k_{\max} = 3000$ and a precision of $\varepsilon = 10^{-12}$, as defined in Sec. 8.2.1. MPM (M2) yields very similar results, except that the approximation is confined to the finite interval $[\omega_{\min}, \omega_{\max}]$ (not shown).

We first examine the performance for a "Kondo-like" spectral function of the form

$$A(\omega) = 0.2g(\omega, -2, 0.5) + 0.6l(\omega, 0, 0.01) + 0.2g(\omega, 2, 0.5) \ , \tag{8.25}$$

where $g(\omega, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\}$ is a Gaussian function, and $l(\omega, \mu, \gamma) = \frac{1}{\pi}\frac{\gamma}{(\omega-\mu)^2+\gamma^2}$ is a Lorentzian. As shown in Fig. 8.2(a), the spectrum exhibits both sharp and smooth features but is analytic on the real axis. Fig. 8.2(b) presents the deviation between the approximated and exact functions as a function of M, evaluated on a uniform real-frequency grid spanning $[-10, 10]$ with 10^5 points for concreteness. The approximation remains well controlled even at higher frequencies. As seen in Fig. 8.2(c), increasing the

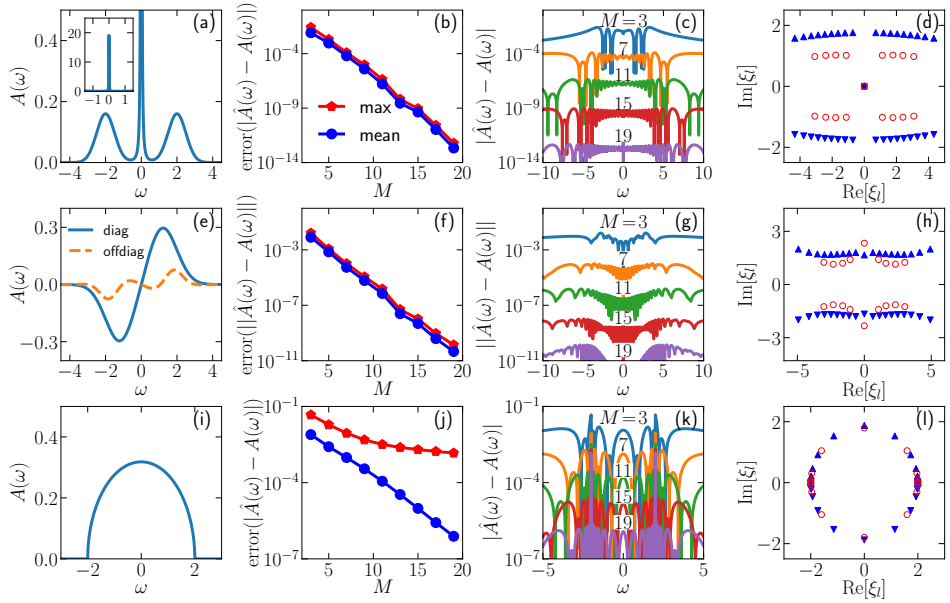

Figure 8.2: Spectral recovery using MPM (M1) for different density of states: Kondo-like (top row), bosonic multi-orbital (middle row), and semi-circular (bottom row). First column: Exact spectral function $A(\omega)$. Second column: Maximum (red pentagons) and mean (blue circles) of the deviation as a function of the number of complex poles M in the lower half-plane. Third column: Distributions of the absolute difference as a function of real frequency ω. Fourth column: Distribution of complex poles in the complex plane z for $M = 9$ (red circles) and $M = 19$ (blue triangles).

number of poles leads to exponential convergence of the spectral function to the known solution. Moreover, the sharp peak at the origin is approximated just as accurately as the broader peaks around $\omega = \pm 2$. Finally, Fig. 8.2(d) shows the distribution of poles, with those corresponding to sharp features located close to the real axis, while those representing smooth features are positioned farther away.

We further assess the quality of the approximation for a multi-orbital bosonic system, where the spectral functions are given by $A_{\text{diag}}(\omega) = -0.6g(\omega, -1.2, 0.8) + 0.6g(\omega, 1.2, 0.8)$ and $A_{\text{off-diag}}(\omega) = -0.13g(\omega, -1.8, 0.5) + 0.1g(\omega, -1, 1) - 0.1g(\omega, 1, 1) + 0.13g(\omega, 1.8, 0.5)$. In this case, the absolute difference $|\cdot|$ for scalars is replaced by the matrix norm $||\cdot||$, defined as the maximum absolute difference among all matrix elements. As shown in Figs. 8.2(f) and (g), the convergence behavior remains similar to the scalar case, with no noticeable slowdown due to the inclusion of off-diagonal elements. The distribution

of shared poles, shown in Fig. 8.2(h), reveals that all poles lie far from the real axis. This is a direct consequence of the absence of sharp spectral features.

We remark that the above conclusions hold for all of the smooth spectral functions we tested, regardless of their precise form. Given an error tolerance ε, MPM (M1) could always determine a finite set of poles such that $|\hat{A}(\omega) - A(\omega)| \leq \varepsilon$ for any $\omega \in (-\infty, +\infty)$. Furthermore, the number of poles is minimal for a given precision, as ensured by the exponential decay of singular values in the singular value decomposition (SVD) of ESPRIT (see Ch. 4).

Finally, we evaluate the approximation for a spectral function featuring singularities: a semicircular density of states,

$$A(\omega) = \begin{cases} \frac{1}{2\pi t^2}\sqrt{4t^2 - \omega^2} & , |\omega| \leq 2t \\ 0 & , |\omega| > 2t \end{cases} , \tag{8.26}$$

which corresponds to the local density of states for an infinite-dimensional Bethe lattice [151]. The hopping parameter is set to $t = 1$, resulting in square-root singularities at $\omega = \pm 2$. As shown in Fig. 8.2(k), the approximation converges exponentially away from the singularities as M increases. However, as depicted in Fig. 8.2(j), the presence of singularities slows down convergence. Despite the challenges posed by these singularities, their impact on practical calculations based on complex poles is typically small, as will be demonstrated in the next section. Fig. 8.2(l) shows the distribution of poles which cluster around the singular points.

8.3.2. Reconstructing the Green's function in the complex plane

In this section, we assess the accuracy of the Hilbert transform to recover the Green's function in the entirety of the complex plane for the examples discussed in the previous section.

The exact expression for the real part of the retarded Green's function corresponding to the Kondo-like spectral function in Eq. (8.25) is given by

$$\mathrm{Re}[G^{\mathrm{ret}}(\omega)] = 0.2\pi\tilde{g}(\omega, -2, 0.5) + 0.6\pi\tilde{l}(\omega, 0, 0.01) + 0.2\pi\tilde{g}(\omega, 2, 0.5) \, , \tag{8.27}$$

where $\tilde{g}(\omega,\mu,\sigma) = \frac{\sqrt{2}}{\pi\sigma} D\left(\frac{\omega-\mu}{\sqrt{2}\sigma}\right)$ and $\tilde{l}(\omega,\mu,\gamma) = \frac{1}{\pi}\frac{\omega-\mu}{(\omega-\mu)^2+\gamma^2}$ are the Hilbert transforms of the Gaussian and Lorentzian functions, respectively, with $D(x)$ denoting the Dawson function. Similar to the spectral function (imaginary part), the real part exhibits sharp features at the origin and smooth variations elsewhere, while remaining analytic on the entire real axis, as shown in Fig. 8.3(a). The approximation is accurate on all of the real axis, as demonstrated in Figs. 8.3(b) and (c). Additionally, Fig. 8.3(d) illustrates the performance of the approximation along the entire positive imaginary axis, which is consistently at least as accurate as on the real axis.

The exact expression for the matrix-valued bosonic spectrum is given by $\mathrm{Re}[G^{\mathrm{ret}}_{\mathrm{diag}}(\omega)] = -0.6\pi\tilde{g}(\omega,-1.2,0.8) + 0.6\pi\tilde{g}(\omega,1.2,0.8)$ and $\mathrm{Re}[G^{\mathrm{ret}}_{\mathrm{off-diag}}(\omega)] = -0.13\pi\tilde{g}(\omega,-1.8,0.5) + 0.1\pi\tilde{g}(\omega,-1,1) - 0.1\pi\tilde{g}(\omega,1,1) + 0.13\pi\tilde{g}(\omega,1.8,0.5)$. As shown in the middle panel of Fig. 8.3, the convergence behavior closely follows that of the spectrum presented in Fig. 8.2.

For any smooth spectral function, MPM (M1) determines a minimal number of poles such that, for a given precision ε, the approximation satisfies

$$|\hat{G}^{\mathrm{ret}}(\omega) - G^{\mathrm{ret}}(\omega)| \leq \varepsilon \text{ for any } \omega \in (-\infty,+\infty). \tag{8.28}$$

Since both $G^{\mathrm{ret}}(z)$ and $\hat{G}^{\mathrm{ret}}(z)$ are analytic in the upper half-plane, the maximum modulus principle ensures that the approximation remains strictly bounded by ε throughout the entire upper half-plane.

The analysis continues with the remaining example of semicircular density of states given in Eq. (8.27), which has the following exact expression for the retarded Green's function:

$$\mathrm{Re}[G^{\mathrm{ret}}(\omega)] = \begin{cases} \frac{\omega}{2t^2} & , |\omega| \leq 2t \\ \frac{\omega - \mathrm{sign}(\omega)\sqrt{\omega^2-4t^2}}{2t^2} & , |\omega| > 2t \end{cases}. \tag{8.29}$$

As shown in Fig. 8.3, the approximation quality closely follows the trend observed in Fig. 8.2: points far from singularities converge significantly faster than those near them. No computational issues arise along the imaginary axis, as illustrated in Fig. 8.3(l).

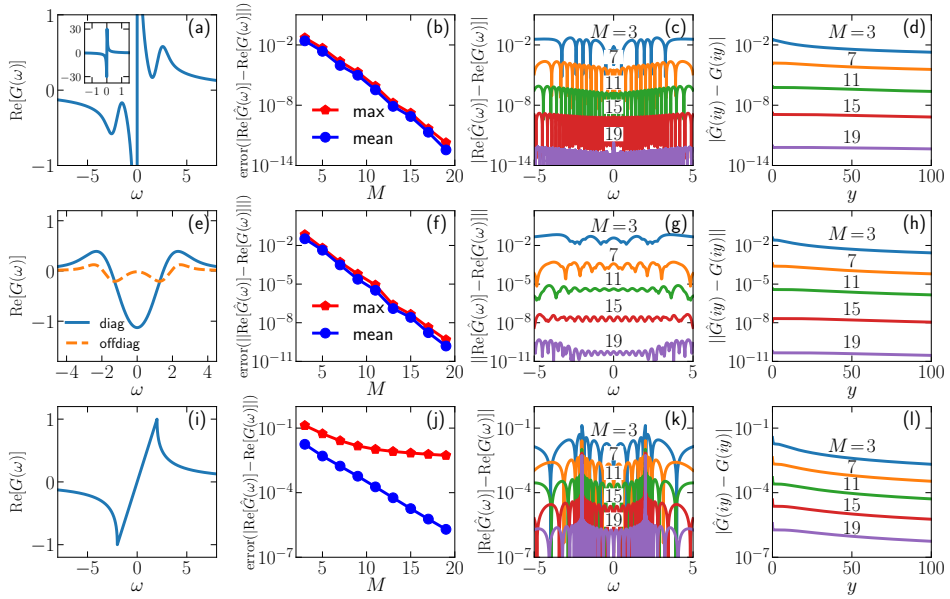

Figure 8.3: Same setup as Fig. 8.2, but now for the real part of the retarded Green's function and the Matsubara Green's function. First column: Exact real part of the retarded Green's function, corresponding to the Hilbert transform of $\pi A(\omega)$. Second column: Error of $\mathrm{Re}[G(\omega)]$ as a function of M. Third column: Distribution of the absolute difference $|\mathrm{Re}[\hat{G}(\omega)] - \mathrm{Re}G(\omega)|$ as a function of ω. Fourth column: Distribution of the absolute difference $|\hat{G}(iy) - G(iy)|$ on the positive imaginary axis.

8.3.3. Decomposing model spectral densities for real-time evolution

In this section, we evaluate the performance of the methods on the BCF derived from a power-law spectral density with an exponential cutoff. To better serve the open quantum systems community, we explicitly provide the units of physical quantities in Secs. 8.3.3 and 8.3.4. We compare our results with those obtained using the adaptive Antoulas–Anderson (AAA) algorithm [152, 153], which operates on the real-frequency domain, as well as with those obtained from the direct decomposition of the BCF using ESPRIT [53, 54].

The spectral density is defined as

$$J(\omega) = \alpha \omega_c^{1-s} \omega^s e^{-\omega/\omega_c} \quad \text{for } \omega \geq 0, \tag{8.30}$$

and extended to negative frequencies $\omega < 0$ by $J(\omega) = -J(-\omega)$. This form is classified as Ohmic, sub-Ohmic, or super-Ohmic for $s = 1$, $0 < s < 1$, and $s > 1$, respectively. The corresponding BCF has the analytic form [154, 155]

$$C(t) = \frac{1}{\pi}\alpha\omega_c^{1-s}\beta^{-(s+1)}\Gamma(s+1) \times \left[\zeta\left(s+1, \frac{1+\beta\omega_c - i\omega_c t}{\beta\omega_c}\right) + \zeta\left(s+1, \frac{1+i\omega_c t}{\beta\omega_c}\right)\right], \tag{8.31}$$

where $\Gamma(z)$ is the gamma function, and $\zeta(z, z')$ is the Hurwitz zeta function.

In this section, we fix the parameters to $\alpha = 1$, $\omega_c = 50$ cm^{-1} $\approx 9.4 \times 10^{-3}$ fs^{-1}, and $t_c = 2000$ fs, following Ref. [56]. The sampling points for ESPRIT are chosen on a uniform grid,

$$t_j = \frac{j}{N_t} t_c, \quad (0 \le j \le N_t), \tag{8.32}$$

whereas for AAA, a logarithmic discretization [56, 156] over $[-\omega_{\max}, \omega_{\max}]$ is employed to ensure efficient sampling in the low-frequency domain:

$$\omega_j = \mathrm{sign}(j - K - \frac{1}{2})\omega_{\max}\frac{\ln\left(\frac{K}{K-|j-K-\frac{1}{2}|}\right)}{\ln N_\omega}, \tag{8.33}$$

where $N_\omega = 2K$ is an even integer, and j runs from 1 to N_ω.

In our simulations, both N_t and N_ω are fixed at 5000. We consistently use $\omega_p = 0.1$ fs^{-1} $\approx 10.6\ \omega_c$ for MPM (M1), and $\omega_{\max} = -\omega_{\min} = 0.5$ fs^{-1} $= 1000\ t_c^{-1} \approx 53.1\ \omega_c$ for MPM (M2), which matches the value used in AAA. Additionally, $k_{\max}$ is fixed at 1000 for both MPM (M1) and MPM (M2) throughout this section. We note that the choice of ω_p in MPM (M1) is independent of t_c; instead, it is determined by the shape of $A(\omega)$. In contrast, we observe that for such systems, $\omega_{\min}$ and $\omega_{\max}$ in MPM (M2) are inversely proportional to the time cutoff.

To quantify the performance, we compute the relative errors of the approximations $\hat{A}(\omega)$ and $\hat{C}(t)$ with respect to their exact counterparts:

$$\delta A(\omega) = \frac{|\hat{A}(\omega) - A(\omega)|}{|A(0)|} \tag{8.34}$$

and

$$\delta C(t) = \frac{|\hat{C}(t) - C(t)|}{|C(0)|}. \tag{8.35}$$

To analyze convergence as a function of the number of poles M, we evaluate the mean error over the approximation interval in the time domain:

$$\text{error}_{\text{inside}} = \frac{1}{N_t + 1} \sum_{j=0}^{N_t} \delta C(t_j). \tag{8.36}$$

To assess the performance on the long-time tail, we compute the mean error over the extended interval $t \in [t_c, nt_c]$:

$$\text{error}_{\text{outside}} = \frac{1}{(n-1)N_t + 1} \sum_{j=N_t}^{nN_t} \delta C(t_j), \tag{8.37}$$

where t_j is defined from Eq. (8.32). For concreteness, we set $n = 10$.

8.3.3.1. Ohmic bath

We begin our discussion with the Ohmic bath ($s = 1$) at a temperature of $T = 300$ K, corresponding to an inverse temperature of $\beta \approx 25.5$ fs. As a reference, the left column of Fig. 8.4 displays the exact input function. In the middle panel, we fix $M = 8$ for all methods and compare their performance in both the frequency and time domains.

In the frequency domain, see Fig. 8.4(b), the approximation is more challenging at the origin due to the discontinuity in $dA(\omega)/d\omega$. MPM (M2) and ESPRIT exhibit similar performance, whereas MPM (M1) provides a slightly better approximation at the origin but performs slightly worse away from it. In contrast, AAA approximates the entire interval more uniformly.

In the time domain, see Fig. 8.4(e), within the approximation interval, MPM (M2) and ESPRIT remain nearly identical and provide more accurate results than MPM (M1). However, beyond the interval ($t > t_c$), MPM (M1) retains control over the error, while the errors in MPM (M2) and ESPRIT increase rapidly. AAA is less accurate than MPM (M1) both inside and outside the interval. Here, the frequency cutoff $\omega_{\max}$ for MPM (M2) is set to 0.5 fs^{-1}, determined automatically from Eq. (8.24), as indicated by the thick gray line in Fig. 8.4(e).

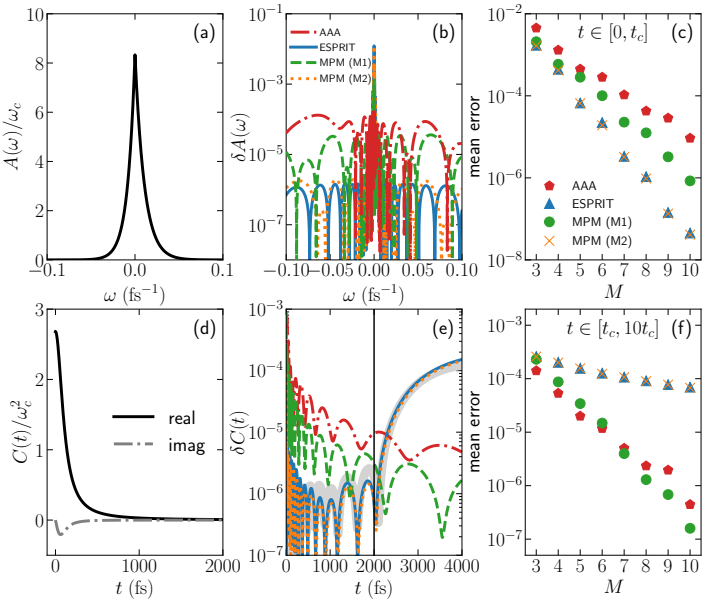

Figure 8.4: Comparison of AAA (red), ESPRIT (blue), MPM (M1) (green), and MPM (M2) (orange) for an Ohmic bath at $T = 300\,\mathrm{K}$ ($\beta\omega_c \approx 0.240$). Left panel: (a) Exact effective spectral function $A(\omega)$ and (d) exact bath correlation function $C(\omega)$. Middle panel: Relative error of (b) $A(\omega)$ as defined in Eq. (8.34) and (e) $C(t)$ as defined in Eq. (8.35). All approximations use $M = 8$ terms. The vertical line separates the inside and outside of the approximated real-time interval. The relative difference $|\hat{C}^{(L_1)}(t) - \hat{C}^{(L_2)}(t)|/|\hat{C}^{(L_1)}(0)|$ is shown by the thick gray curve. Right panel: Mean error (Eqs. (8.36) and (8.37)) as a function of M, evaluated (c) within the interval $[0, t_c]$ and (f) over the long tail $[t_c, 10t_c]$.

Finally, the right column of Fig. 8.4 shows the convergence of the mean error as a function of the number of poles M. Within the interval, MPM (M2) and ESPRIT overlap and provide the best error control, while MPM (M1) converges slightly more slowly, and AAA converges the slowest. However, for the long tail, MPM (M2) and ESPRIT exhibit the least control, whereas MPM (M1) and AAA converge significantly faster.

We now examine the same system at a much lower temperature, $T = 0.001$ K, corresponding to an inverse temperature of $\beta \approx 7.64 \times 10^6$ fs. In this regime, the effective spectrum varies rapidly near the origin. As shown in Fig. 8.5, no convergence slowdown with respect to M is observed at low temperature for the methods discussed in this chapter. Their performance trends remain consistent with those observed at higher

temperatures. Although AAA occasionally achieves an accuracy in the long tail similar to MPM (M1), its convergence behavior is less robust and its accuracy within the finite interval is consistently lower than that of MPM (M1).

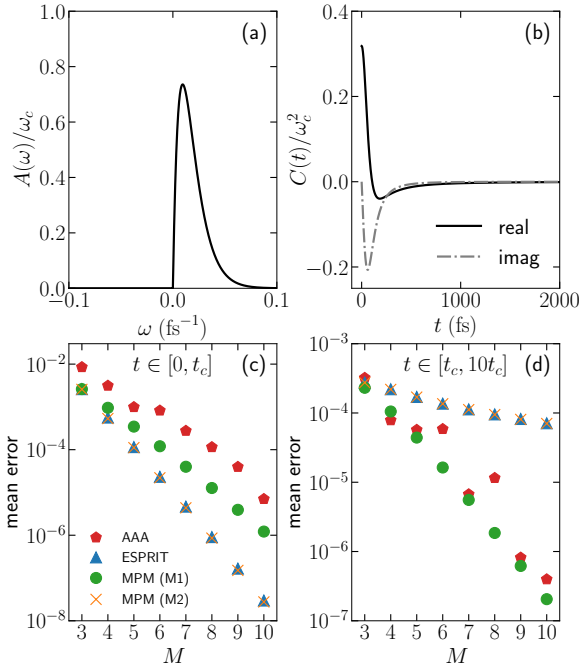

Figure 8.5: Same setup as Fig. 8.4, except that temperature is now set to $T = 0.001\,\mathrm{K}$ ($\beta\omega_c \approx 7.19 \times 10^4$). Top panel: (a) Exact spectral function. (b) Exact bath correlation function. Bottom panel: Mean error for (c) the approximated interval and (d) the long-tail.

8.3.3.2. Sub-Ohmic bath

In this section, we examine a sub-Ohmic bath ($s = 0.5$) at $T = 50$ K ($\beta \approx 153$ fs). The spectral function, which is shown in Fig. 8.6(a), diverges at $\omega = 0$, but its integral remains finite, ensuring the existence of the BCF. However, the real part of $C(t)$, which is shown in Fig. 8.6(b), exhibits an extremely slowly decaying tail.

Both AAA and ESPRIT encounter significant issues in this case. As shown in Fig. 8.6(c) and (d), AAA completely fails. This failure arises because AAA attempts to approximate the spectral shape on the given grid, causing it to consistently underestimate the exact spectrum near the divergence point, as illustrated in the inset of

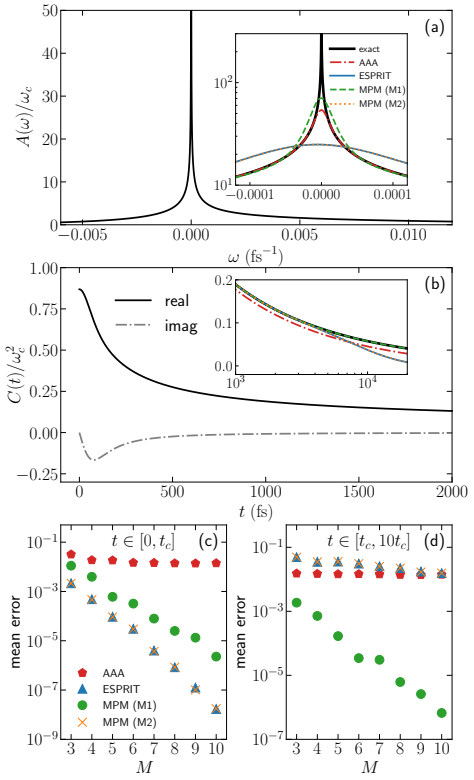

Figure 8.6: Comparison for a sub-Ohmic bath at $T = 50\,\mathrm{K}$ ($\beta\omega_c \approx 1.44$). Eight terms are used for all approximations. (a) Exact spectral function, which has a divergence at $\omega = 0$. (b) Exact bath correlation function, which exhibits an extremely slowly decaying tail for the real part. The insets show the performance of different methods in a zoomed-in region. Mean error for the approximated interval and the long tail are shown in subfigures (c) and (d), respectively.

Fig. 8.6(a).

A similar issue arises in the real-time domain for ESPRIT. Due to the extremely slow decay of $C(t)$, ESPRIT loses error control immediately beyond the finite interval. This presents a fundamental challenge: to maintain error control within 10^{-3} over the long tail, for example, t_c would need to be increased beyond 10^7 fs. However, since computational constraints limit the number of sampling points to a few thousand, the resulting coarse time grid spacing prevents accurate resolution of the peak in the imaginary part of $C(t)$.

In contrast, MPM does not suffer from these issues. On one hand, MPM (M2) achieves

an accuracy comparable to ESPRIT, both within and beyond the approximation interval. On the other hand, although MPM (M1) converges slightly more slowly than MPM (M2) and ESPRIT within the finite interval, it maintains control over the long tail. In particular, MPM (M1) is the only method that effectively handles the long tail. The success of MPM stems from its approach of approximating spectral moments rather than the spectral shape. Although the spectral function diverges at the origin, its moments remain finite.

8.3.4. Decomposing structured spectra for real-time evolution

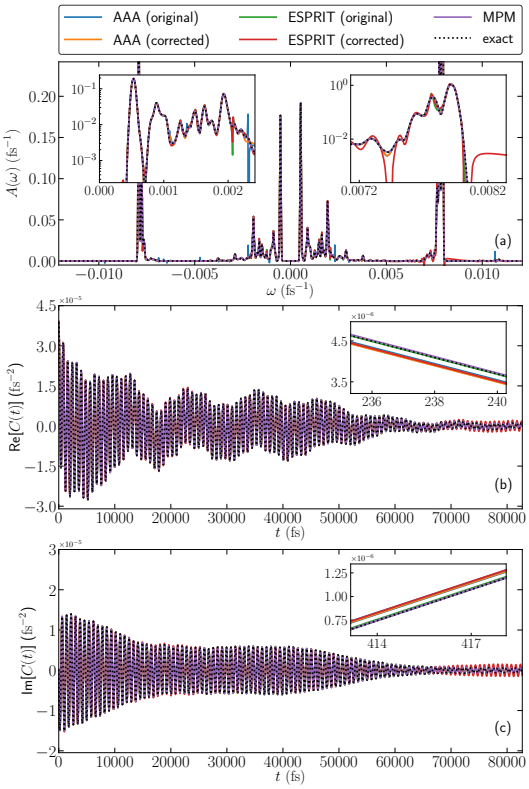

Figure 8.7: Comparison of different approximation methods (solid lines) with the exact results (dashed lines) for a structured spectral function at $T = 300\,\mathrm{K}$. Results are obtained for $M = 100$. (a) Structured spectrum, (b) real part, and (c) imaginary part of the corresponding bath correlation function. The insets show the performance of different methods in magnified regions.

Finally, we evaluate the performance of the methods on a structured spectrum derived from exciton–phonon couplings in quantum dots [157, 158], which plays a key role in understanding nonradiative exciton dynamics in these systems.

The input data consists of the spectral density $J(\omega)$ sampled on a dense uniform grid. To enable flexible sampling at arbitrary frequency points, the AAA algorithm is applied as a preprocessing step to the effective spectrum at $T = 300$ K, with accuracy prioritized over the number of poles. The error tolerance is set to 10^{-8}, resulting in 704 poles $\{A_l^{(\mathrm{AAA})}, \xi_l^{(\mathrm{AAA})}\}$, evenly distributed between the upper and lower half-planes. These poles are treated as the exact solution and serve as the reference for all methods compared in this section. In this case, the contour integral in Eq. (8.6) for MPM can be evaluated analytically using $h_k = \sum_l A_l^{\prime(\mathrm{AAA})} (\xi_l^{\prime(\mathrm{AAA})})^k$ [Eq. (8.7)] instead of a numerical integral, thereby improving computational efficiency.

We present the results in Fig. 8.7, where M is fixed at 100. For AAA, the number of poles to be recovered is set to $2M$, as it captures poles across the entire complex plane. For ESPRIT, the parameters $t_c = 20000\ \mathrm{eV}^{-1} \approx 8.27 \times 10^4$ fs and $N_t = 5000$ are used. Issues arise for both the AAA and ESPRIT algorithms. On one hand, the original AAA method consistently recovers some poles on the real axis, leading to non-causal jumps in the spectrum. Denoting these poles as $\{A_l^{(\mathrm{real})}, \xi_l^{(\mathrm{real})}\}$ and their count as M_{real}, these spurious poles introduce a constant term in $C(t)$, degrading accuracy in both the real-frequency and real-time domains. To ensure a reliable real-time evolution, the method should be modified to remove these poles. However, as shown in Fig. 8.7, even after correction, both the approximated spectrum and the BCF exhibit some deviations from the exact results. On the other hand, ESPRIT becomes unstable for some values of the parameter L. Although this issue can be mitigated by carefully tuning L, allowing ESPRIT to approximate the BCF similarly to MPM, we note that it consistently recovers some poles in the upper half plane. This leads to a divergent amplitude of the BCF at long times, even though the exact BCF exhibits decay. To eliminate the asymptotic instability, these non-physical poles must be removed [159]. In practice, this is done by first obtaining M poles from ESPRIT, discarding the unphysical ones, and then performing a least-squares fit to determine the weights of the remaining poles. Unsurprisingly, we find that once some poles are disregarded, ESPRIT immediately loses error control, as evidenced by its deviation from the exact results in Figs. 8.7(b) and (c).

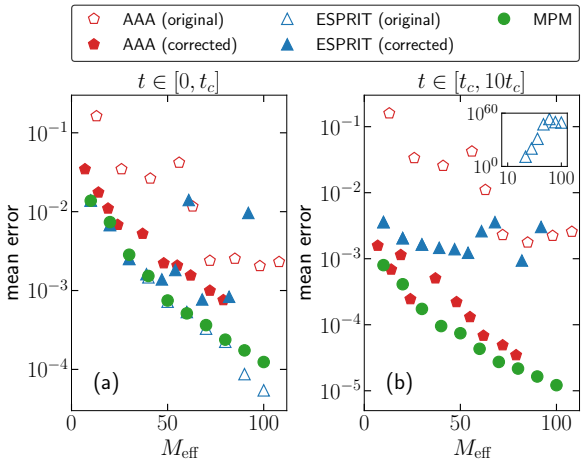

Figure 8.8: Mean error from Eqs. (8.36) and (8.37) as a function of the effective number of poles used in the computation of $C(t)$. Unfilled markers represent the original results that should not be used, while filled markers represent the results after correction. The inset shows the performance of ESPRIT (original) on a larger scale.

In contrast, MPM does not suffer from these issues. Its poles are automatically placed in the lower half-plane, ensuring that the approximation remains stable and well-controlled. As a result, MPM provides the most accurate approximation in both the time and frequency domains. Furthermore, in this case, MPM (M1) and MPM (M2) yield nearly identical results due to the rapid decay of the long tails in $A(\omega)$. Consequently, we present only MPM (M1) results throughout this section, with ω_p fixed at 0.04 eV $\approx 9.67 \times 10^{-3}$ fs^{-1} and $k_{\max}$ set to 3000.

Fig. 8.8 presents the results for M ranging from 10 to 100. To ensure a fair comparison, we use the effective number of poles, M_{eff}, defined as the number of poles contributing to the BCF calculation. Specifically, M_{eff} is given by $M + \frac{M_{\text{real}}}{2}$ for the original AAA, $M - \frac{M_{\text{real}}}{2}$ for the corrected AAA, M for MPM and the original ESPRIT, and potentially less than M for the corrected ESPRIT if non-physical poles appear. As discussed previously, while the original ESPRIT can sometimes match or even slightly outperform MPM, it frequently produces non-physical poles. Removing these poles leads to a complete loss of error control. For AAA, eliminating real-axis poles consistently improves accuracy both within and beyond the approximation interval. However, AAA remains less robust and less accurate than MPM, reinforcing MPM as the most reliable method among those

compared.

8.4. Conclusions

In conclusion, we have presented a method to approximate real-frequency spectral functions as a sum of poles in the complex plane. The method is systematically improvable in the sense that, for a given set of real-frequency data and a given approximation precision, a minimal number of poles yielding the most accurate representation possible is generated.

We have shown the behavior of the method for synthetic spectral functions as well as for spectral functions with a definite structure.

We expect that the methodology will find applications whenever a Green's function of a continuous quantum system needs to be represented by a finite (possibly very small) number of discrete degrees of freedom, and we have provided examples that are typical applications of the hierarchical equations of motion method.

Our work is accompanied by an open-source software implementation of the proposed method, written in the programming language PYTHON [119, 160].

Chapter 9

Quantum Impurity Solvers

In the previous chapters, we introduced applications of the complex pole representation to pre- and post-processing steps, including hybridization fitting, compact basis construction, and numerical analytic continuation. In this chapter, we explore direct applications of the complex pole representation to quantum impurity models in both real- and imaginary-axis simulations. For simplicity, we focus solely on fermionic systems and illustrate the methods using the single-impurity Anderson model (SIAM). Generalizations to multi-orbital cases and bosonic systems are straightforward and involve no conceptual difficulties.

9.1. Introduction

For a general quantum impurity model, the Hamiltonian $\hat{H}_{\mathrm{phy}}$ consists of three terms: $\hat{H}_{\mathrm{imp}}$, which describes the small impurity; $\hat{H}_{\mathrm{bath}}$, which describes the infinite bath into which the impurity is embedded; and $\hat{H}_{\mathrm{hyb}}$, which describes the coupling between the impurity and the bath. Thus, the full system takes the form

$$\hat{H}_{\mathrm{phy}} = \hat{H}_{\mathrm{imp}} + \hat{H}_{\mathrm{bath}} + \hat{H}_{\mathrm{hyb}} \ , \tag{9.1}$$

where

$$\hat{H}_{\text{imp}} = \sum_{ij} t_{ij} \hat{d}_i^\dagger \hat{d}_j + \frac{1}{2} \sum_{ijkl} v_{ijkl} \hat{d}_i^\dagger \hat{d}_j^\dagger \hat{d}_k \hat{d}_l \ , \tag{9.2}$$

$$\hat{H}_{\text{bath}} = \int_{-\infty}^{+\infty} d\omega \ \omega \hat{c}_\omega^\dagger \hat{c}_\omega \ , \tag{9.3}$$

$$\hat{H}_{\text{hyb}} = \sum_j \int_{-\infty}^{+\infty} d\omega [g_j(\omega) \hat{d}_j^\dagger \hat{c}_\omega + g_j^*(\omega) \hat{c}_\omega^\dagger \hat{d}_j] \ , \tag{9.4}$$

where $\hat{d}_j/\hat{d}_j^\dagger$ and $\hat{c}_\omega/\hat{c}_\omega^\dagger$ are fermionic operators in the impurity and bath, respectively. We assume that there may be infinitely many bath states, so that the bath spectral function is broadened.

Simulations of these systems can be roughly classified into real-axis and imaginary-axis simulations. Each category can be further divided into diagrammatic approaches and effective-model-based methods.

Effective-model-based methods, such as Hamiltonian-based or Liouvillian-based methods, are based on the observation that the full information of the bath is contained in the hybridization function or bath correlation function (BCF). For imaginary-axis calculations, the bath enters the calculation only through

$$\mathbf{\Delta}_{jj'}(i\omega_n) = \int_{-\infty}^{+\infty} d\omega \frac{g_j(\omega) g_{j'}^*(\omega)}{i\omega_n - \omega} \ . \tag{9.5}$$

For real-axis calculations, when the initial density matrix of the system at $t = 0$ can be separated as $\hat{\rho}_{\text{phy}}(0) = \hat{\rho}_{\text{imp}}(0) \otimes \hat{\rho}_{\text{bath}}(0)$ and the bath is in thermal equilibrium, i.e., $\hat{\rho}_{\text{bath}}(0) = \frac{e^{-\beta \hat{H}_{\text{bath}}}}{\text{Tr}[e^{-\beta \hat{H}_{\text{bath}}}]}$, the bath enters the calculation only through

$$\mathbf{\Delta}_{jj'}^{>}(t) = \int_{-\infty}^{+\infty} d\omega g_j(\omega) g_{j'}^*(\omega)(1 - n_F(\omega)) e^{-i\omega t} \ , \tag{9.6}$$

$$\mathbf{\Delta}_{jj'}^{<}(t) = \int_{-\infty}^{+\infty} d\omega g_j^*(\omega) g_{j'}(\omega) n_F(\omega) e^{+i\omega t} \ , \tag{9.7}$$

where $n_F(\omega) = \frac{1}{e^{\beta\omega}+1}$ is the fermionic distribution function. Therefore, one can construct an effective finite-size model whose hybridization function or bath correlation function matches that of the original system to within a prescribed precision. The re-

sulting effective model can then be solved using methods such as exact diagonalization (ED). Note that the definition used here is slightly different from the definition of the greater/lesser Green's functions in Ch. 3. This choice is made for compatibility with the quasi-Lindblad theory [108] used in Sec. 9.3.1.

On the other hand, diagrammatic approaches formulate the quantum impurity problem in terms of Green's functions and self-energies, and evaluate the latter through a selected class of Feynman diagrams. In these methods, the effect of interactions is incorporated perturbatively or self-consistently by constructing the impurity self-energy from products and convolutions of interacting or noninteracting Green's functions. Compared with effective-model-based methods, diagrammatic methods usually avoid the explicit construction of the many-body Hilbert space. Their accuracy depends on the diagram classes retained and the parameter regimes being studied.

9.2. Imaginary-Axis Impurity Solvers

In this section, we explore the application of the complex pole representation to imaginary-axis simulations, including both Hamiltonian-based and diagrammatic methods.

9.2.1. Hamiltonian-Based Impurity Solver

The Hamiltonian-based impurity solver is constructed by introducing the following finite-size auxiliary system:

$$\hat{H}_{\mathrm{aux}} = \hat{H}_{\mathrm{imp}} + \hat{H}^{\mathrm{aux}}_{\mathrm{bath}} + \hat{H}^{\mathrm{aux}}_{\mathrm{hyb}} \,, \tag{9.8}$$

where

$$\hat{H}^{\mathrm{aux}}_{\mathrm{bath}} = \sum_{k} \varepsilon_k \hat{c}^\dagger_k \hat{c}_k \,, \quad \hat{H}^{\mathrm{aux}}_{\mathrm{hyb}} = \sum_{jk} (V_{jk} \hat{d}^\dagger_j \hat{c}_k + V'_{jk} \hat{c}^\dagger_k \hat{d}_j) \,. \tag{9.9}$$

The corresponding hybridization function has the form

$$\boldsymbol{\Delta}_{jj'}(i\omega_n) = \sum_k \frac{V_{jk} V'_{j'k}}{i\omega_n - \varepsilon_k} \,. \tag{9.10}$$

As discussed above, as long as Eq. (9.10) matches Eq. (9.5), the impurity Green's functions and self-energies will match as well.

In the conventional hybridization fitting procedure [127], the poles in Eq. (9.10) are restricted to the real axis and the pole weights are positive-semidefinite matrices. Thus, $\varepsilon_k \in \mathbb{R}$ and $V'_{jk} = (V_{jk})^*$, so that the auxiliary Hamiltonian $\hat{H}_{\text{aux}}$ is Hermitian and can be diagonalized using techniques such as the Hermitian Lanczos algorithm [161, 162]. However, a major issue is that the convergence rate of the approximation to $\Delta(i\omega_n)$ is usually slow with real-valued poles, which limits this Hamiltonian-based method to relatively small impurity sizes. Another issue is that this formalism in principle works only for Matsubara points. To obtain the real-axis spectral function, one still needs to perform numerical analytic continuation. Although one can in principle avoid this step by directly evaluating the spectral function from the obtained real poles, this usually leads to oscillations and unreliable artificial broadening [104]. Therefore, the main goal of this section is to explore the direct use of the complex pole representation.

As discussed in Sec. 3.3.2, when we consider real-valued hopping and interaction terms, the Hamiltonian $\hat{H}_{\text{phy}}$ can be chosen to be real symmetric. In this case, the matrix-valued hybridization function can be represented by complex poles as

$$\boldsymbol{\Delta}(i\omega_n) = \sum_k \frac{\boldsymbol{A}_k}{i\omega_n - \xi_k} , \tag{9.11}$$

which is valid in the upper-half plane. Here, the poles $\text{Im}\xi_k \leq 0$ are located in the lower-half plane, and the matrix-valued weights $\boldsymbol{A}_k$ are in general complex symmetric. For a complex-valued symmetric matrix, the left eigenvector is the transpose of the right eigenvector rather than the conjugate transpose. This implies that the coupling strengths V_{jk} and V'_{jk} in Eq. (9.10) are equal. Therefore, in the complex-pole framework, we have $\varepsilon_k = \xi_k \in \mathbb{C}$ ($\text{Im}\varepsilon_k \leq 0$) and $V_{jk} = V'_{jk} \in \mathbb{C}$. The resulting auxiliary Hamiltonian $\hat{H}_{\text{aux}}$ is therefore complex symmetric.

For such a Hamiltonian, denote its eigenvalues and right eigenvectors by $E_i \in \mathbb{C}$ and v_i, i.e., $\hat{H}_{\text{aux}} v_i = E_i v_i$. One can perform Alg. 3 to orthonormalize its eigenvectors, so that $v_i^T v_j = \delta_{ij}$ and $\sum_i v_i^T v_i = \mathbb{1}$. As a result, the generic single-particle Green's function

can be written as

$$\boldsymbol{G}_{ij}(z) = \frac{1}{\mathcal{Z}} \sum_{mn} \frac{e^{-\beta E_m} \mp e^{-\beta E_n}}{z - (E_n - E_m)} (v_m^T \hat{d}_i v_n)(v_n^T \hat{d}_j^\dagger v_m) \ , \tag{9.12}$$

where both the eigenenergy E_m and the partition function $\mathcal{Z}$ are now complex-valued.

Algorithm 3 Orthonormalization Algorithm

```
for i = 1 to N do
    v_i ← v_i / sqrt(v_i^T v_i).
    for j = i + 1 to N do
        if v_j^T v_i ≠ 0 then
            v_j ← v_j − (v_i^T v_j) v_i.
```

Algorithm 4 Non-interacting Bethe-lattice DMFT self-consistent loop

1: Initialize M, r and $\{A_k, \xi_k\}$ so that $\Delta^{\mathrm{Ret}}(\omega) = t^2 G^{\mathrm{Ret}}(\omega) = \sum_{k=1}^{M} \frac{A_k}{\omega - \xi_k}$.
2: Set diff $\leftarrow \infty$.
3: **while** diff $> \varepsilon$ **do**
4: Construct $\hat{H}_{\mathrm{aux}}$ via Eq. (9.14) with $\varepsilon_k = \xi_k$ and $V_k = \sqrt{A_k}$.
5: Compute poles of $G^{\mathrm{Ret}}(\omega)$ via Eq. (9.12), denoted by $G^{\mathrm{Ret}}(\omega) = \sum_{k=1}^{M'} \frac{A'_k}{\omega - \xi'_k}$.
6: Compute the spectral function $A(\omega) = -\frac{1}{\pi} \operatorname{Im} G^{\mathrm{Ret}}(\omega)$.
7: Update $\Delta^{\mathrm{Ret}}(\omega)$ using M poles obtained from performing MPM [40] to $(1-r) \sum_{k=1}^{M} \frac{A_k}{\omega - \xi_k} + rt^2 \sum_{k=1}^{M'} \frac{A'_k}{\omega - \xi'_k}$.
8: Compute the convergence error diff $= \int d\omega |G^{\mathrm{Ret}}_{\mathrm{new}}(\omega) - G^{\mathrm{Ret}}(\omega)|$.
9: **return** $G^{\mathrm{Ret}}(\omega)$.

As an illustration, we choose the non-interacting spinless single-impurity case as an example, where the original and auxiliary Hamiltonians read

$$\hat{H}_{\mathrm{phy}} = \varepsilon_d \hat{d}^\dagger \hat{d} + \int_{-\infty}^{+\infty} d\omega \ \omega \hat{c}_\omega^\dagger \hat{c}_\omega + \int_{-\infty}^{+\infty} d\omega g(\omega) (\hat{d}^\dagger \hat{c}_\omega + \hat{c}_\omega^\dagger \hat{d}) \ , \tag{9.13}$$

and

$$\hat{H}_{\mathrm{aux}} = \varepsilon_d \hat{d}^\dagger \hat{d} + \sum_k \varepsilon_k \hat{c}_k^\dagger \hat{c}_k + \sum_k V_k (\hat{d}^\dagger \hat{c}_k + \hat{c}_k^\dagger \hat{d}) \ . \tag{9.14}$$

We take the Bethe-lattice DMFT self-consistency loop as an example, where the exact spectral function is known to be semicircular. The implementation details are

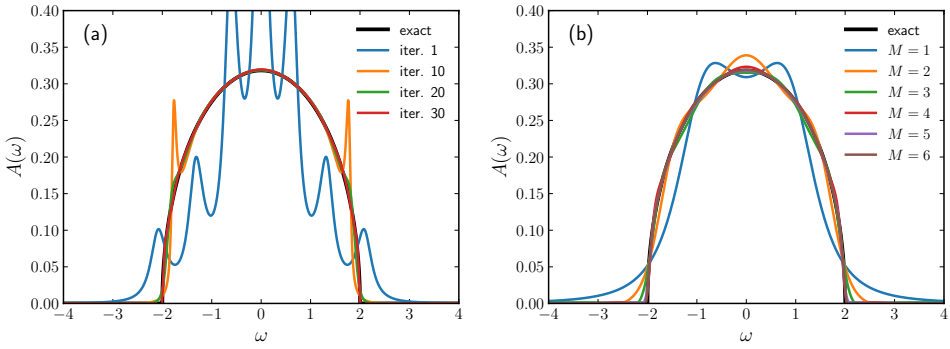

Figure 9.1: Simulated spectral function $A(\omega)$ obtained from the DMFT self-consistency loop. (a) Evolution with the number of iterations for $M = 6$. (b) Converged results for different numbers of poles $M \in \{1, 2, 3, 4, 5, 6\}$.

outlined in Alg. 4, and the simulation parameters are chosen as $\varepsilon_d = 0$ and $r = 0.5$. As shown in Fig. 9.1, as the number of iterations increases, the simulated spectral function quickly converges, independent of the initial state. As the number of poles M increases, the converged spectral function $A(\omega)$ quickly approaches the exact solution. However, the reader should be aware that this formalism is exact only for non-interacting cases. The reason is that the complex pole representation is valid only in a half-plane [cf. Eq. (5.8)]. As the interactions are turned on, the self-energy will mix the upper- and lower-half planes, which invalidates this formalism. More details will be discussed in the next section.

9.2.2. Real-Pole-Based Diagrammatic Impurity Solver

In this section, we explore the application of the complex pole representation to imaginary-axis diagrammatic calculations.

For the spin-symmetric single-impurity Anderson model, the second-order self-energy can be obtained as

$$\Sigma_2(i\omega_n) = -\frac{U^2}{\beta^2} \sum_{m,p} G(i\omega_m) G(i\omega_p) G(i\omega_m + i\omega_p - i\omega_n) \; . \tag{9.15}$$

Although at first glance one might attempt to split each $G(i\omega_m)$ and $G(i\omega_p)$ into two parts, each valid in one half of the complex plane, difficulties arise from the third term,

since the choice of poles depends on the specific combination of (m, n, p). Therefore, no generic expression can be obtained, making it difficult to apply techniques such as the residue theorem. One way to circumvent this issue is to restrict the poles to the real axis, as described in Ch. 7. As shown in Ref. [104], although MPM can lead to a more compact representation than the popular discrete Lehmann representation (DLR) [82], the oscillations caused by real poles still persist, and the convergence of the calculations is usually slower than that obtained using complex poles. Another way to fully resolve this issue and directly utilize complex poles is to use the Keldysh formalism on the real axis, as will be described in Sec. 9.3.2.

9.3. Real-Axis Impurity Solvers

In this section, we explore the application of the complex pole representation to real-axis simulations, including both Liouvillian-based and diagrammatic methods.

9.3.1. Liouvillian-Based Impurity Solver for Steady States

9.3.1.1. Theory

Effective Model In this section, we aim to represent the quantum impurity model in steady states by an auxiliary open quantum system. By using the Minimal Pole Method (MPM) [40], the effective spectral function $g_j(\omega)g^*_{j'}(\omega)(1 - n_F(\omega))$ or $g^*_j(\omega)g_{j'}(\omega)n_F(\omega)$ in Eqs. (9.6) and (9.7) can be approximated by a sum of complex poles $\{\boldsymbol{A}_l, \xi_l\}$, as described in Ch. 8, with $\boldsymbol{A}_l$ being matrix-valued pole weights and ξ_l being shared pole locations. The question then reduces to constructing an effective model that can reproduce the same result within a prescribed precision, while reducing the number of degrees of freedom in the original system.

If $\boldsymbol{A}_l$ were diagonal and purely real, we could use the standard pseudomode theory [148] to represent the physics of the system with Liouvillian dynamics. However, this corresponds to approximating spectral functions as a sum of Lorentzians, which is known to converge very slowly [163], as also explained in Ch. 5. To achieve rapid convergence as the number of poles increases, one has to allow $\boldsymbol{A}_l$ to have complex entries. This generalizes the original Lindblad dynamics [164, 165] to the so-called quasi-Lindblad dynamics [108]. In this case, we can proceed by using the following

auxiliary system [108]

$$\hat{H}_{\rm aux} = \hat{H}_{\rm imp} + \hat{H}^{\rm aux}_{\rm bath} + \hat{H}^{\rm aux}_{\rm hyb} \,, \tag{9.16}$$

where

$$\hat{H}^{\rm aux}_{\rm bath} = \sum_{k_1} \varepsilon_{k_1} \hat{c}^\dagger_{k_1} \hat{c}_{k_1} + \sum_{k_2} \varepsilon_{k_2} \hat{c}^\dagger_{k_2} \hat{c}_{k_2} \,, \tag{9.17}$$

$$\hat{H}^{\rm aux}_{\rm hyb} = \sum_j (\hat{d}^\dagger_j \hat{A}_j + \hat{A}^\dagger_j \hat{d}_j) \quad \text{with} \quad \hat{A}_j = \sum_{k_1} (\boldsymbol{V}_1)_{jk_1} \hat{c}_{k_1} + \sum_{k_2} (\boldsymbol{V}_2)_{jk_2} \hat{c}_{k_2}, \tag{9.18}$$

with the initial state $\hat{\rho}_{\rm bath}(0) = \hat{\rho}_{A_1}(0) \otimes \hat{\rho}_{A_2}(0) = |\mathbf{0}\rangle\langle\mathbf{0}| \otimes |\mathbf{1}\rangle\langle\mathbf{1}|$. Here, we denote the auxiliary baths associated with the indices k_1 and k_2 by A_1 and A_2, respectively. For convenience, we further denote the impurity by S in the following. In addition to the unitary part in Eq. (9.16), the system is also coupled to the dissipators

$$\mathcal{D}_{A_1} \bullet = \sum_{k_1} 2\gamma_{k_1} (\hat{c}_{k_1} \bullet \hat{c}^\dagger_{k_1} - \frac{1}{2} \{\hat{c}^\dagger_{k_1} \hat{c}_{k_1}, \bullet\}) \,, \tag{9.19}$$

$$\mathcal{D}_{A_2} \bullet = \sum_{k_2} 2\gamma_{k_2} (\hat{c}^\dagger_{k_2} \bullet \hat{c}_{k_2} - \frac{1}{2} \{\hat{c}_{k_2} \hat{c}^\dagger_{k_2}, \bullet\}) \,, \tag{9.20}$$

$$\mathcal{D}_{SA_1} \bullet = \sum_j (\hat{d}_j \bullet \hat{L}^\dagger_{1j} + \hat{L}_{1j} \bullet \hat{d}^\dagger_j - \frac{1}{2} \{\hat{L}^\dagger_{1j} \hat{d}_j + \hat{d}^\dagger_j \hat{L}_{1j}, \bullet\}) \,, \tag{9.21}$$

$$\mathcal{D}_{SA_2} \bullet = \sum_j (\hat{d}^\dagger_j \bullet \hat{L}_{2j} + \hat{L}^\dagger_{2j} \bullet \hat{d}_j - \frac{1}{2} \{\hat{L}_{2j} \hat{d}^\dagger_j + \hat{d}_j \hat{L}^\dagger_{2j}, \bullet\}) \,, \tag{9.22}$$

where $\hat{L}_{1j} = \sum_{k_1} 2(\boldsymbol{M}_1)_{jk_1} \hat{c}_{k_1}$ and $\hat{L}_{2j} = \sum_{k_2} 2(\boldsymbol{M}_2)_{jk_2} \hat{c}_{k_2}$. The time-evolution equation for the total density matrix $\rho_{\rm aux}(t)$ can then be written as

$$\frac{d\hat{\rho}_{\rm aux}(t)}{dt} = -i[\hat{H}_{\rm aux}, \hat{\rho}_{\rm aux}(t)] + (\mathcal{D}_{A_1} + \mathcal{D}_{A_2} + \mathcal{D}_{SA_1} + \mathcal{D}_{SA_2}) \hat{\rho}_{\rm aux}(t) \,. \tag{9.23}$$

The BCF in this system is written as

$$\boldsymbol{\Delta}^{>}(t) = (\boldsymbol{V}_1 - i\boldsymbol{M}_1)e^{-\boldsymbol{Z}_1 t}(\boldsymbol{V}_1 + i\boldsymbol{M}_1)^{\dagger} , \tag{9.24}$$

$$\boldsymbol{\Delta}^{<}(t) = (\boldsymbol{V}_2 - i\boldsymbol{M}_2)^{*}e^{-\boldsymbol{Z}_2 t}(\boldsymbol{V}_2 + i\boldsymbol{M}_2)^{T} . \tag{9.25}$$

According to Ref. [108], when Eqs. (9.24) and (9.25) exactly match Eqs. (9.6) and (9.7) for any $t \geq 0$, the impurity reduced density matrices are equivalent, i.e., $\mathrm{Tr}_{\mathrm{bath}}[\hat{\rho}_{\mathrm{phy}}(t)] = \mathrm{Tr}_{\mathrm{bath}}[\hat{\rho}_{\mathrm{aux}}(t)]$. Meanwhile, the auxiliary system offers significantly greater computational efficiency by reducing the infinitely large bath to a finite set of auxiliary states with effective dissipation.

Vectorization The right-hand side of the master equation (9.23) maps one operator $\hat{\rho}_{\mathrm{aux}}(t)$ in the original Hilbert space to another operator. It is therefore an operator acting on operators, namely, a superoperator. Thus, it can be written abstractly as

$$\frac{d\hat{\rho}_{\mathrm{aux}}(t)}{dt} = \hat{\hat{\mathcal{L}}}\hat{\rho}_{\mathrm{aux}}(t) . \tag{9.26}$$

Suppose the basis of the original Hilbert space is $\{|n\rangle\}$. According to Ref. [166], one can construct the so-called *super-fermion space*, whose basis is $\{|m\rangle \otimes |\tilde{n}\rangle\}$, where the tilde $\tilde{}$ is used to distinguish the duplicated space from the original space. For a generic operator $\hat{A} = \sum_{mn} A_{mn}|m\rangle\langle n|$, this amounts to vectorizing it as a vector $|A\rangle = \sum_{mn} A_{mn}|m\rangle \otimes |\tilde{n}\rangle$ in the super-fermion space. This is achieved by $|A\rangle = (\hat{A} \otimes \hat{\mathbb{1}})|I\rangle$, where $\mathbb{1}$ is the identity operator and $|I\rangle = \sum_n |n\rangle \otimes |\tilde{n}\rangle$ is the so-called *left-vacuum vector*. By using the conjugation relations explained in Sec. B.2.2, one can represent $\hat{\hat{\mathcal{L}}}$ and $\hat{\rho}_{\mathrm{aux}}(t)$ as an operator $\hat{\mathcal{L}}$, called the *Liouvillian*, and a state vector $|\rho_{\mathrm{aux}}(t)\rangle$ in the super-fermion space, respectively. As a result, the master equation is formulated as

$$\frac{d|\rho_{\mathrm{aux}}(t)\rangle}{dt} = \hat{\mathcal{L}}|\rho_{\mathrm{aux}}(t)\rangle . \tag{9.27}$$

Therefore, the density matrix at any time t can be expressed as $|\rho_{\mathrm{aux}}(t)\rangle = e^{\hat{\mathcal{L}}t}|\rho_{\mathrm{aux}}(0)\rangle$. The expectation value of an operator $\hat{A}$ in the original space can also be reformulated

in the super-fermion space as

$$\langle A(t)\rangle = \mathrm{Tr}[\hat{A}\rho_{\mathrm{aux}}(t)] = \langle I|\hat{A}\otimes\mathbb{1}|\rho_{\mathrm{aux}}(t)\rangle\ . \tag{9.28}$$

For simplicity, we abbreviate the right-hand side as $\langle I|\hat{A}|\rho_{\mathrm{aux}}(t)\rangle$ without causing confusion.

As a result, since both Eq. (9.27) and Eq. (9.28) have been rewritten in terms of regular operators and state vectors, we can study them using diagonalization-based techniques, as explained in the following.

Green's Function Calculation Our primary aim is to study the steady states. In this case, the time derivative of the density matrix is zero, i.e., $\hat{\mathcal{L}}|\rho_{\mathrm{aux}}(\infty)\rangle = 0$. This means that the steady state corresponds to the eigenvector(s) of $\hat{\mathcal{L}}$ with zero eigenvalues. In practice, we usually find that $\hat{\mathcal{L}}$ has only one zero eigenvalue and one corresponding eigenvector.

Once the steady-state density matrix is obtained, one can readily calculate the single-particle Green's functions and extract the corresponding spectral information. In particular, we are interested in the steady-state retarded Green's function; other Green's functions can be written down similarly. It is given by

$$\begin{aligned} i\boldsymbol{G}^{\mathrm{Ret}}_{ij}(t>0) &= \langle \hat{d}_i(t)\hat{d}_j^\dagger\rangle + \langle \hat{d}_j^\dagger\hat{d}_i(t)\rangle \\ &= \langle \hat{d}_i(t)\hat{d}_j^\dagger\rangle + (\langle \hat{d}_i^\dagger(t)\hat{d}_j\rangle)^* \\ &= \langle I|\hat{d}_i e^{\hat{\mathcal{L}}t}\hat{d}_j^\dagger|\rho_{\mathrm{aux}}(\infty)\rangle + (\langle I|\hat{d}_i^\dagger e^{\hat{\mathcal{L}}t}\hat{d}_j|\rho_{\mathrm{aux}}(\infty)\rangle)^*\ , \end{aligned} \tag{9.29}$$

In practice, the calculation of $e^{\hat{\mathcal{L}}t}$ can be achieved by diagonalizing $\hat{\mathcal{L}}$, the details of which will be explained later.

Before moving on to the implementation details, we point out that a similar framework has already been presented in the auxiliary master equation approach (AMEA) [129, 149]. The main difference is that we use complex poles with complex-valued pole weights in the quasi-Lindblad framework, whereas AMEA is restricted to Lorentzians in the Lindblad framework, which are known to converge slowly.

This method offers several advantages over existing approaches. First, by allowing the pole weights to have imaginary parts in the quasi-Lindblad framework, the conver-

gence with respect to the number of poles is significantly faster than in the traditional Lorentzian pseudomode method [163]. Second, by employing the Minimal Pole Method (MPM) in the real-frequency domain, the long-tail behavior of the bath correlation function (BCF) in the time domain is effectively controlled [40], unlike ESPRIT, which operates on a finite real-time interval. Finally, by using exact diagonalization (ED), the computation of the steady state avoids the numerical instability typically encountered in methods that rely on direct integration of the master equation. Since every step of the procedure is systematically controlled, the resulting steady-state Green's function is systematically improvable as the number of quasi-Lindblad pseudomodes increases.

9.3.1.2. Implementation

In practice, the total Hilbert space grows rapidly as the number of pseudomodes increases. As a result, the current implementation is restricted to the single-orbital case. Specifically, we focus on the single-impurity Anderson model (SIAM):

$$\hat{H}_{\mathrm{imp}} = \sum_{\sigma} \varepsilon_{\sigma} \hat{d}_{\sigma}^{\dagger} \hat{d}_{\sigma} + U \hat{d}_{\uparrow}^{\dagger} \hat{d}_{\uparrow} \hat{d}_{\downarrow}^{\dagger} \hat{d}_{\downarrow} \, . \tag{9.30}$$

In this case, $\hat{H}_{\mathrm{bath}}^{\mathrm{aux}}$ and $\hat{H}_{\mathrm{hyb}}^{\mathrm{aux}}$ take the form

$$\hat{H}_{\mathrm{bath}}^{\mathrm{aux}} = \sum_{\sigma} \sum_{k_1=1}^{M_1} (\varepsilon_1)_{k_1\sigma} \hat{c}_{k_1\sigma}^{\dagger} \hat{c}_{k_1\sigma} + \sum_{\sigma} \sum_{k_2=1}^{M_2} (\varepsilon_2)_{k_2\sigma} \hat{c}_{k_2\sigma}^{\dagger} \hat{c}_{k_2\sigma} \, , \tag{9.31}$$

and

$$\hat{H}_{\mathrm{hyb}}^{\mathrm{aux}} = \sum_{\sigma} (\hat{d}_{\sigma}^{\dagger} \hat{A}_{\sigma} + \hat{A}_{\sigma}^{\dagger} \hat{d}_{\sigma}) \quad \text{with} \quad \hat{A}_{\sigma} = \sum_{k_1=1}^{M_1} (\boldsymbol{V}_1)_{k_1\sigma} \hat{c}_{k_1\sigma} + \sum_{k_2=1}^{M_2} (\boldsymbol{V}_2)_{k_2\sigma} \hat{c}_{k_2\sigma}, \tag{9.32}$$

and the Liouvillian $\hat{\mathcal{L}}$ in the super-fermion space can be written accordingly. We provide its explicit expression in Sec. B.3. One can calculate the eigenvalues $\{\lambda_i\}$ with biorthonormal left and right eigenvectors $\{|i\rangle_L, |i\rangle_R\}$, which satisfy $\langle i|_L |j\rangle_R = \delta_{ij}$ and $\mathbb{1} = \sum_{i=1} |i\rangle_R \langle i|_L$. Therefore, $\hat{\mathcal{L}}$ can be decomposed as $\hat{\mathcal{L}} = \sum_{i=1} \lambda_i |i\rangle_R \langle i|_L$. Defining $\Lambda = \mathrm{diag}\{\lambda_1, \lambda_2, \cdots\}$, $L = (|1\rangle_L, |2\rangle_L, \cdots)$, and $R = (|1\rangle_R, |2\rangle_R, \cdots)$, this decomposition

can be rewritten as $\hat{\mathcal{L}} = R\Lambda L^\dagger$, and Eq. (9.29) can be rewritten as

$$G^{\rm Ret}_{\sigma\sigma}(t>0) = -iv^\dagger_{1L}\begin{pmatrix} e^{\lambda_1 t} & & \\ & e^{\lambda_2 t} & \\ & & \ddots \end{pmatrix} v_{1R} - iv^\dagger_{2R}\begin{pmatrix} e^{\lambda_1^* t} & & \\ & e^{\lambda_2^* t} & \\ & & \ddots \end{pmatrix} v_{2L} \ , \tag{9.33}$$

where

$$v_{1L} = R^\dagger \hat{d}^\dagger_\sigma |I\rangle \ , \quad v_{1R} = L^\dagger \hat{d}^\dagger_\sigma |\rho_{\rm aux}(\infty)\rangle \ , \tag{9.34}$$

$$v_{2L} = R^\dagger \hat{d}_\sigma |I\rangle \ , \quad v_{2R} = L^\dagger \hat{d}_\sigma |\rho_{\rm aux}(\infty)\rangle \ . \tag{9.35}$$

After performing the Fourier transform, one obtains

$$\begin{aligned} G^{\rm Ret}_{\sigma\sigma}(\omega) &= v^\dagger_{1L}\begin{pmatrix} \frac{1}{\omega - i\lambda_1} & & \\ & \frac{1}{\omega - i\lambda_2} & \\ & & \ddots \end{pmatrix} v_{1R} + v^\dagger_{2R}\begin{pmatrix} \frac{1}{\omega - i\lambda_1^*} & & \\ & \frac{1}{\omega - i\lambda_2^*} & \\ & & \ddots \end{pmatrix} v_{2L} \\ &= \sum_j \frac{(v_{1L})^*_j (v_{1R})_j}{\omega - i\lambda_j} + \sum_j \frac{(v_{2R})^*_j (v_{2L})_j}{\omega - i\lambda_j^*} \ . \end{aligned} \tag{9.36}$$

Denoting by N_σ the total number of particles with spin σ in the original space, one can verify that the total particle number $N = N_\uparrow + N_\downarrow$ and the total spin along the z direction $S_z = N_\uparrow - N_\downarrow$ are conserved quantities. Then $(N - \tilde{N}, S_z - \tilde{S}_z)$ are conserved quantities in the super-fermion space [129, 149], which can be verified from the explicit expression in Eq. (B.26). One can further verify that the steady state $|\rho_{\rm aux}(t)\rangle$ exists only in the $(0,0)$ block, and that only eigenvalues in the $(\pm 1, \pm 1)$ blocks contribute to the poles in Eq. (9.36). Therefore, the above equation is simplified to

$$G^{\rm Ret}_{\sigma\sigma}(\omega) = \sum_j \frac{(v^{(+1,\pm1)}_{1L})^*_j (v^{(+1,\pm1)}_{1R})_j}{\omega - i\lambda^{(+1,\pm1)}_j} + \sum_j \frac{(v^{(-1,\mp1)}_{2R})^*_j (v^{(-1,\mp1)}_{2L})_j}{\omega - i(\lambda^{(-1,\mp1)}_j)^*} \ , \tag{9.37}$$

where

$$v^{(+1,\pm1)}_{1L} = R^{(+1,\pm1)\dagger} \hat{d}^\dagger_\sigma |I_0\rangle \ , \quad v^{(+1,\pm1)}_{1R} = L^{(+1,\pm1)\dagger} \hat{d}^\dagger_\sigma |\rho_{\rm aux}(\infty)\rangle \ , \tag{9.38}$$

$$v^{(-1,\mp1)}_{2L} = R^{(-1,\mp1)\dagger} \hat{d}_\sigma |I_0\rangle \ , \quad v^{(-1,\mp1)}_{2R} = L^{(-1,\mp1)\dagger} \hat{d}_\sigma |\rho_{\rm aux}(\infty)\rangle \ , \tag{9.39}$$

and $|I_0\rangle$ is the projection of $|I\rangle$ onto the $(0,0)$ block, so that both $|I_0\rangle$ and $|\rho_{\mathrm{aux}}(\infty)\rangle$ are located only in the $(0,0)$ block. The superscripts $(+1,\pm1)$ and $(-1,\mp1)$ refer to the blocks in which the diagonalization is performed, with the upper and lower signs corresponding to $\sigma=\uparrow$ and $\sigma=\downarrow$, respectively.

Therefore, this leads to the following steps:

- Given the effective bath spectral densities $|g_\sigma(\omega)|^2(1-n_F(\omega))$ and $|g_\sigma(\omega)|^2 n_F(\omega)$ in the original system, approximate them using MPM via Eq. (8.1) and obtain a set of complex poles $\{A_{k_1\sigma},\xi_{k_1\sigma}\}$ and $\{A_{k_2\sigma},\xi_{k_2\sigma}\}$, respectively.

- Recover the parameters $\boldsymbol{V}_1$, $\boldsymbol{M}_1$, $\boldsymbol{Z}_1$, $\boldsymbol{V}_2$, $\boldsymbol{M}_2$, and $\boldsymbol{Z}_2$ according to Eq. (B.28), so that the unitary system is mapped to a quasi-Lindblad system and the Liouvillian $\hat{\mathcal{L}}$ can then be obtained from Eq. (B.26).

- Perform exact diagonalization of $\hat{\mathcal{L}}$ in the $(0,0)$ block and obtain the eigenvalues λ_i together with the corresponding left and right eigenvectors $\langle i|_L$ and $|i\rangle_R$. There should exist a non-degenerate eigenvalue $\lambda_{i_0}=0$, with all other eigenvalues satisfying $\mathrm{Re}[\lambda_i]<0$ for $i\neq i_0$. The steady state is obtained as $|\rho(\infty\rangle = |i_0\rangle_R/\langle I_0|i_0\rangle_R$.

- Perform exact diagonalization of the blocks $(+1,\pm1)$ and $(-1,\mp1)$ to obtain $\lambda^{(+1,\pm1)}$, $L^{(+1,\pm1)}$, $R^{(+1,\pm1)}$, $\lambda^{(-1,\mp1)}$, $L^{(-1,\mp1)}$, and $R^{(-1,\mp1)}$, and obtain the steady-state retarded Green's function via Eq. (9.37).

9.3.1.3. Results

In our simulations, we set $\varepsilon_\uparrow=\varepsilon_\downarrow$ and $g_\uparrow(\omega)=g_\downarrow(\omega)$, so that the SIAM is spin symmetric. We can therefore omit the spin index in the retarded Green's function and denote it simply by $G^{\mathrm{Ret}}(\omega)$.

We focus on the particle-hole symmetric case, i.e., $\varepsilon_\sigma=-\frac{U}{2}$, and set the numbers of pseudomodes for $\Delta^>$ and $\Delta^<$ to be equal, i.e., $M_1=M_2=M$. We perform simulations for $M=1$, 2, and 3. For $M=1$, full diagonalization is feasible. However, for $M>1$, the rapid growth of the Hilbert space requires the use of sparse matrix diagonalization. In practice, we use the non-Hermitian Lanczos algorithm [149, 161]; the details can be found in Sec. B.4.

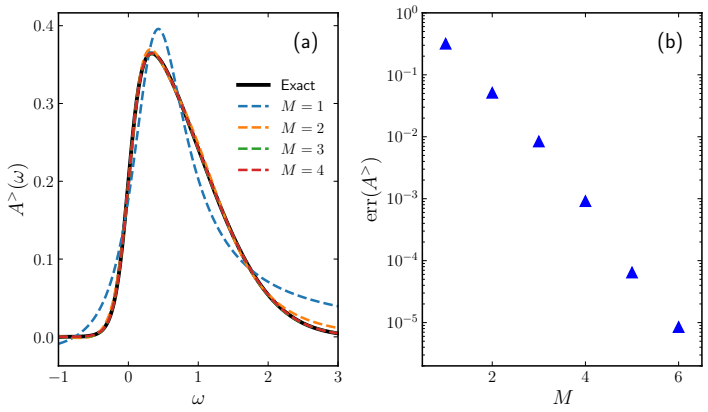

Figure 9.2: Convergence of the approximation to the effective greater spectral function $A^>(\omega)$. (a) Approximation with different numbers of poles M. (b) Convergence of the approximation error as a function of M.

As a benchmark, we choose a Gaussian spectral function for the bath,

$$A(\omega) = |g(\omega)|^2 = \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{(\omega-\mu)^2}{2\sigma^2}\right\} , \tag{9.40}$$

with $\mu = 0$ and $\sigma = 1$ and perform simulations at $\beta = 10$ and $U = 1$. We apply MPM [40] separately to the effective spectral functions $A^>(\omega) = A(\omega)(1 - n_F(\omega))$ and $A^<(\omega) = A(\omega)n_F(\omega)$. Fig. 9.2(a) shows the approximations obtained using $M \in \{1, 2, 3, 4\}$ poles. Since $A^>(\omega)$ and $A^<(\omega)$ are symmetric with respect to the origin, only the results for $A^>(\omega)$ are shown. As shown in the figure, the approximation error decreases rapidly as the number of poles M increases. Already at $M = 3$, the approximated effective spectral function is visually indistinguishable from the exact one. Fig. 9.2(b) shows the convergence as a function of M, where the error is calculated as $\mathrm{err}(A^>) = \int d\omega |A^>_{\mathrm{approx}}(\omega) - A^>_{\mathrm{exact}}(\omega)|$. The error exhibits exponential convergence.

Fig. 9.3 shows the convergence of the simulated spectral function as a function of the number of non-Hermitian Lanczos eigenvalues k for $M = 3$. The convergence is rapid, and already $k = 20$ eigenvalues yield converged results.

Fig. 9.4 shows the simulated steady-state spectral function for $M = 1$, 2, and 3. Since each step of this procedure is controlled, the error in the spectral function can be estimated from the quality of the initial bath fitting. Here, the error is indicated by the shaded region and is estimated as $\max(|A^>_{\mathrm{approx}}(\omega) - A^>_{\mathrm{exact}}(\omega)|)$. As shown in the

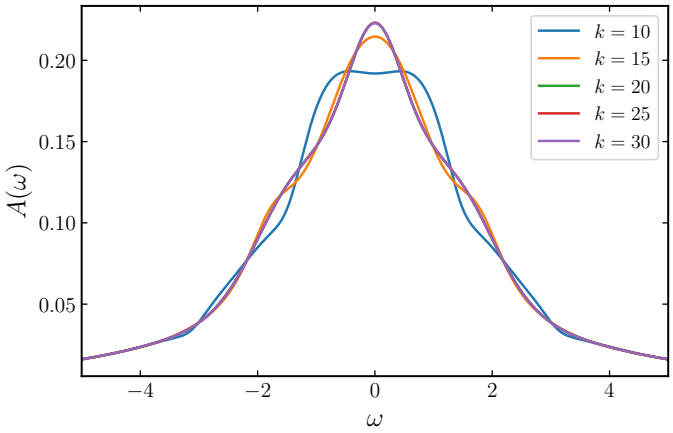

Figure 9.3: Convergence of the spectral function $A(\omega)$ for $M = 3$ as a function of the number of non-Hermitian Lanczos eigenvalues k.

figure, the results converge very quickly. For $M = 3$, the error is already invisible on the scale of the plot, demonstrating the reliability of the results. Simulations for $M \geq 4$ are beyond the current capacity of the non-Hermitian Lanczos approach. To make such simulations feasible, one needs to resort to other techniques, such as the Configuration Interaction (CI) method [121, 167].

As pointed out in Ref. [149], the method presented here can reproduce qualitatively correct results for equilibrium physics. It is therefore interesting to use this method to explore the differences between nonequilibrium steady states and equilibrium states, and to potentially develop it as an alternative to existing equilibrium impurity solvers.

Since the quasi-Lindblad formalism breaks the complete positivity of the density matrix, we find that it can sometimes lead to unstable solutions at small M, where the largest eigenvalue of the Liouvillian becomes greater than zero. However, as M increases, this issue is observed to become progressively less severe. With CI, the number of affordable poles is expected to increase significantly, and this instability should become less concerning. Another promising way to circumvent this issue is to incorporate the so-called coupled-Lindblad formalism [163], which uses complex poles while preserving the complete positivity of the density matrix.

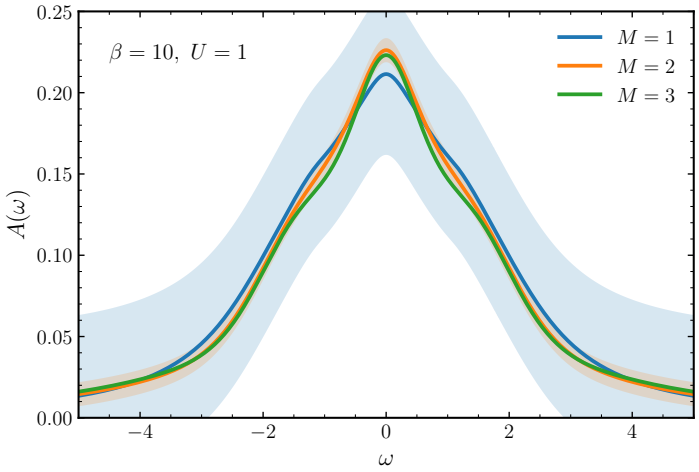

Figure 9.4: Simulated spectral function $A(\omega)$ for the Gaussian bath at $\beta = 10$ and $U = 1$, using different numbers of poles $M \in \{1, 2, 3\}$.

9.3.2. Complex-Pole-Based Diagrammatic Impurity Solver

9.3.2.1. Method

In this section, we use second-order Green's function perturbation theory (GF2) to illustrate the direct use of the complex pole representation in Feynman diagram evaluations within the Keldysh formalism [6]. We emphasize, however, that the underlying logic is not restricted to GF2 and can be readily applied to other diagrammatic evaluations. This is because the full information of the system enters the diagrams through the real-axis Keldysh components, $\boldsymbol{G}^{>}$, $\boldsymbol{G}^{<}$, $\boldsymbol{G}^{T}$, and $\boldsymbol{G}^{\tilde{T}}$, or equivalently through $\boldsymbol{G}^{\mathrm{Ret}}$, $\boldsymbol{G}^{\mathrm{Adv}}$, and $\boldsymbol{G}^{K}$, and all of these components can be accurately represented by complex poles, as shown in Ch. 5.

For the single-impurity Anderson model (SIAM) discussed in Sec. 9.3.1.2, the self-energy in GF2 has the form

$$\Sigma_2^{>}(\omega)/U^2 = \int \frac{d\omega_1}{2\pi}\frac{d\omega_2}{2\pi} G^{>}(\omega_1)G^{>}(\omega_2)G^{<}(\omega_1+\omega_2-\omega)\ , \tag{9.41}$$

$$\Sigma_2^{<}(\omega)/U^2 = \int \frac{d\omega_1}{2\pi}\frac{d\omega_2}{2\pi} G^{<}(\omega_1)G^{<}(\omega_2)G^{>}(\omega_1+\omega_2-\omega)\ . \tag{9.42}$$

As discussed in Ch. 5, all real-axis Green's functions can be written in complex pole

form. Specifically, the greater and lesser Green's functions have the form

$$G^{>}(\omega) = \sum_{l=1}^{M} \left(\frac{A_l^{>(\mathrm{dn})}}{\omega - \xi_l^{>(\mathrm{dn})}} + \frac{A_l^{>(\mathrm{up})}}{\omega - \xi_l^{>(\mathrm{up})}} \right) , \tag{9.43}$$

$$G^{<}(\omega) = \sum_{l=1}^{M} \left(\frac{A_l^{<(\mathrm{dn})}}{\omega - \xi_l^{<(\mathrm{dn})}} + \frac{A_l^{<(\mathrm{up})}}{\omega - \xi_l^{<(\mathrm{up})}} \right) , \tag{9.44}$$

which, for simplicity, can be abbreviated as

$$G^{>}(\omega) = \sum_{l=1}^{2M} \frac{A_l^{>}}{\omega - \xi_l^{>}} , \quad G^{<}(\omega) = \sum_{l=1}^{2M} \frac{A_l^{<}}{\omega - \xi_l^{<}} . \tag{9.45}$$

Substituting these expressions into Eq. (9.41) gives

$$\Sigma_2^{>}(\omega)/U^2 = \sum_i \sum_j \sum_k \int \frac{d\omega_1}{2\pi} \frac{d\omega_2}{2\pi} \frac{A_i^{>}}{\omega_1 - \xi_i^{>}} \frac{A_j^{>}}{\omega_2 - \xi_j^{>}} \frac{A_k^{<}}{\omega_1 + \omega_2 - \omega - \xi_k^{<}} . \tag{9.46}$$

Since the poles of the integrand are always away from the real axis and the integrand vanishes at infinity, one can close the real-axis contour with an infinitely large semicircle and apply the residue theorem. For concreteness, we close the contour with a semicircle in the upper-half plane. Choosing the lower-half plane gives the same result. This evaluation leads to the compact expression

$$\begin{aligned} \Sigma_2^{>}(\omega)/U^2 = & \left(\sum_k \sum_{j\in\mathrm{up}} \sum_{i\in\mathrm{up}} + \sum_j \sum_{i:(k-i)\in\mathrm{up}} \sum_{k\in\mathrm{up}} \right) \frac{A_i^{>} A_j^{>} A_k^{<}}{\omega - (\xi_i^{>} + \xi_j^{>} - \xi_k^{<})} \\ & - \left(\sum_i \sum_{j\in\mathrm{up}} \sum_{k\in\mathrm{up}} + \sum_j \sum_{i:(k-i)\in\mathrm{up}} \sum_{i\in\mathrm{up}} \right) \frac{A_i^{>} A_j^{>} A_k^{<}}{\omega - (\xi_i^{>} + \xi_j^{>} - \xi_k^{<})} , \end{aligned} \tag{9.47}$$

where $i \in \mathrm{up}$ denotes that the sum is restricted to poles in the upper-half plane, and $i : (k-i) \in \mathrm{up}$ denotes that the sum over i is restricted to those poles for which $\xi_k - \xi_i$

Algorithm 5 Bethe-lattice DMFT self-consistent loop

Initialize $\Delta^{\rm Ret}(\omega) = t^2 G^{\rm Ret}(\omega)$, $G^{>}(\omega)$ and $G^{<}(\omega)$ via $A(\omega)$.
Set diff $\leftarrow \infty$.
while diff $> \varepsilon$ **do**
    Compute $\Sigma_2^{>}(\omega)$ and $\Sigma_2^{<}(\omega)$ via Eqs. (9.47) and (9.48).
    Compute $\mathrm{Im}\Sigma_2^{\rm Ret}(\omega)$ via Eq. (9.49).
    Recover $\Sigma_2^{\rm Ret}(\omega)$ via MPM [40].
    Compute the impurity Green's function

$$G^{\rm Ret}(\omega) = \frac{1}{\omega + i0^+ - \Delta^{\rm Ret}(\omega) - \Sigma_2^{\rm Ret}(\omega)}.$$

    Compute the spectral function $A(\omega) = -\frac{1}{\pi}\,\mathrm{Im}\,G^{\rm Ret}(\omega)$.
    Update $\Delta^{\rm Ret}(\omega) = t^2 G^{\rm Ret}(\omega)$, $G^{>}(\omega)$ and $G^{<}(\omega)$.
    Compute the convergence error diff $= \int d\omega |G^{\rm Ret}_{\rm new}(\omega) - G^{\rm Ret}(\omega)|$.
return $G^{\rm Ret}(\omega)$, $\Sigma^{\rm Ret}(\omega)$.

lies in the upper-half plane. Similarly, the lesser self-energy can be written as

$$\begin{aligned}\Sigma_2^{<}(\omega)/U^2 = &\left(\sum_k \sum_{j\in\mathrm{up}} \sum_{i\in\mathrm{up}} + \sum_j \sum_{i:(k-i)\in\mathrm{up}} \sum_{k\in\mathrm{up}}\right) \frac{A_i^{<} A_j^{<} A_k^{>}}{\omega - (\xi_i^{<} + \xi_j^{<} - \xi_k^{>})} \\ &- \left(\sum_i \sum_{j\in\mathrm{up}} \sum_{k\in\mathrm{up}} + \sum_j \sum_{i:(k-i)\in\mathrm{up}} \sum_{i\in\mathrm{up}}\right) \frac{A_i^{<} A_j^{<} A_k^{>}}{\omega - (\xi_i^{<} + \xi_j^{<} - \xi_k^{>})}\,. \end{aligned} \tag{9.48}$$

The imaginary part of the retarded self-energy can then be obtained from

$$\mathrm{Im}\Sigma_2^{\rm Ret}(\omega) = \frac{1}{2i}[\Sigma_2^{>}(\omega) - \Sigma_2^{<}(\omega)]\,, \tag{9.49}$$

and its real part can be obtained in a controlled manner from MPM [40] via the Hilbert transform, as explained in Ch. 8.

This procedure can be readily incorporated into the dynamical mean-field theory (DMFT) self-consistency loop. For example, the self-consistency loop on the Bethe lattice can be carried out as described in Alg. 5.

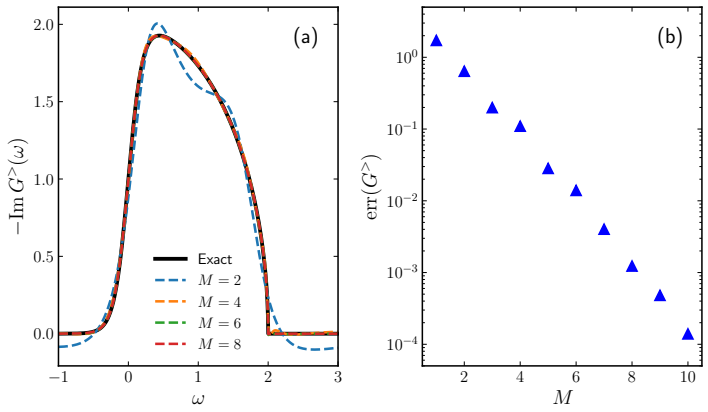

Figure 9.5: Convergence of the approximation to the input greater Green's function $G^{>}(\omega)$. (a) Approximations obtained using different numbers of poles M. (b) Convergence of the approximation error as a function of M.

9.3.2.2. Results

As a benchmark, we choose a semicircular spectral function

$$A(\omega) = \frac{1}{2\pi t^2}\sqrt{4t^2 - \omega^2}\ . \tag{9.50}$$

with $t = 1$, and set the inverse temperature to $\beta = 10$. The greater and lesser Green's functions are then obtained from

$$G^{>}(\omega) = -2\pi i A(\omega)(1 - n_F(\omega))\ , \tag{9.51}$$

$$G^{<}(\omega) = +2\pi i A(\omega) n_F(\omega)\ . \tag{9.52}$$

Fig. 9.5(a) shows the approximations obtained using different numbers of poles. Since $G^{>}(\omega)$ and $G^{<}(\omega)$ are symmetric with respect to the origin, only the results for $G^{>}(\omega)$ are shown. Similar to the results in the previous section, the approximation error decreases rapidly as the number of poles M increases. Already at $M = 6$, the approximated $G^{>}(\omega)$ is visually indistinguishable from the exact one. Fig. 9.5(b) shows the convergence as a function of M, where the error is calculated as $\mathrm{err}(G^{>}) = \int d\omega |G^{>}_{\mathrm{approx}}(\omega) - G^{>}_{\mathrm{exact}}(\omega)|$. The error exhibits exponential convergence.

Fig. 9.6 shows the simulated GF2 self-energy obtained from Eqs. (9.47), (9.48) and

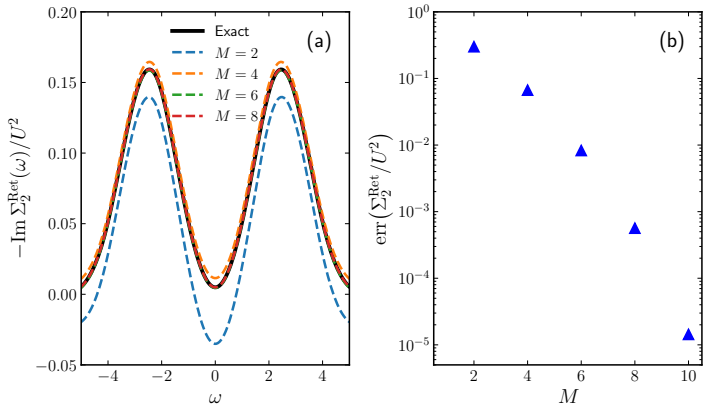

Figure 9.6: (a) Simulated GF2 self-energy obtained using different numbers of poles M. (b) Convergence of the self-energy error as a function of M.

(9.49). The exact result is computed from

$$\Sigma_2^{\mathrm{Ret}}(\omega)/U^2 = \int d\omega_1 d\omega_2 d\omega_3 A(\omega_1)A(\omega_2)A(\omega_3)\frac{n_F(\omega_2)(1-n_F(\omega_1))(1-n_F(\omega_3))}{\omega+\omega_2-\omega_1-\omega_3+i0^+} + \int d\omega_1 d\omega_2 d\omega_3 A(\omega_1)A(\omega_2)A(\omega_3)\frac{(1-n_F(\omega_2))n_F(\omega_1)n_F(\omega_3)}{\omega+\omega_2-\omega_1-\omega_3+i0^+}\,. \tag{9.53}$$

by numerical integration. As shown in Fig. 9.6(a), $\Sigma_2^{\mathrm{Ret}}(\omega)$ converges rapidly as the number of poles M increases, and 6 poles are already sufficient to produce a result that is visually indistinguishable from the exact result. As shown in Fig. 9.6(b), the error of the simulated result converges exponentially as M increases.

As a final example, we perform self-consistent DMFT calculations on the Bethe lattice. The initial spectral function is chosen to be the semicircular density of states in Eq. (9.50). The simulations are performed at $U = 2$, $t = 1$ and $\beta = 10$. As shown in Fig. 9.7(a), the simulated spectral function $A(\omega)$ converges rapidly as the number of iterations increases, and 5 iterations are already sufficient to produce a converged result. Fig. 9.7(b) shows the converged results as a function of M. We find that 4 poles are already sufficient to obtain a reliable result.

The results shown in this section demonstrate that this complex-pole-based Feynman diagram evaluation technique has significant advantages over imaginary-axis simulations, which not only require many more poles but also suffer from the ill-conditioned analytic continuation process. Although a reliable spectral function can still be re-

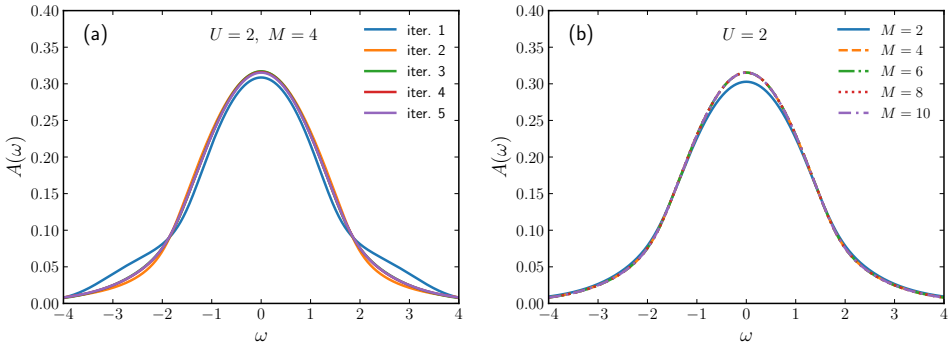

Figure 9.7: Simulated spectral function $A(\omega)$ obtained from the DMFT self-consistency loop. (a) Evolution with the number of iterations for $M = 4$. (b) Converged results for different numbers of poles $M \in \{2, 4, 6, 8, 10\}$.

covered by performing MPM [38, 39], the required precision of the Matsubara data is usually high, and the convergence becomes increasingly slow as β decreases. In contrast, the method presented here does not suffer from this issue.

Finally, we point out that the technique presented here is not restricted to GF2 or to the single-orbital case. For multi-orbital systems, the computational benefits could be even greater by utilizing the shared complex poles developed in this thesis. Moreover, this technique can be readily generalized to other diagrammatic methods, such as GW. This could make real-material simulations much more efficient and reliable, and is therefore a promising direction to explore.

9.4. Conclusions

In this chapter, we explore the application of the complex pole representation to the development of novel quantum impurity solvers. For imaginary-axis Hamiltonian-based methods, we show that the complex pole representation can be used directly to construct a non-Hermitian Hamiltonian for solving non-interacting problems. We take a spinless single-impurity model as an example and demonstrate its reliability and rapid convergence with respect to the number of complex poles. However, since the complex pole representation is valid only in one half of the complex plane, the introduction of interaction terms invalidates this formalism. One therefore has to resort to the conventional

real-pole representation and Hermitian Hamiltonian for interacting cases.

We then use second-order perturbation theory for the SIAM as an example and show that a similar issue persists for imaginary-axis diagrammatic methods, where one has to restrict the poles to the real axis in order to perform pole-based calculations using techniques such as the residue theorem.

Despite this limitation in imaginary-axis simulations, we demonstrate significant success of the complex pole representation in designing new real-axis impurity solvers.

We first present a novel Liouvillian-based impurity solver for calculating steady states by combining MPM [40], the quasi-Lindblad theory [108], and the super-fermion formalism [166]. Since each step is fully controlled, the error of the result can be directly estimated from the accuracy of the input approximation. We take the SIAM as an example and show that even three complex poles can be sufficient to achieve convergence. Therefore, it is worthwhile to incorporate other techniques, such as configuration interaction (CI), to treat more complicated systems [121, 167].

We then use a SIAM GF2 example to demonstrate the direct use of the complex pole representation in real-axis diagrammatic calculations. We show how to combine MPM [40] with the residue theorem to perform the calculations directly on the real-frequency axis. The results converge rapidly as the number of complex poles increases. The computational cost is much lower than that of the corresponding imaginary-axis simulations, and no numerical analytic continuation (NAC) is needed. Therefore, this method offers a significant advantage over imaginary-axis diagrammatic calculations. Its extension to multi-orbital cases and more sophisticated diagrams provides a promising direction for future diagrammatic calculations.

Chapter 10

Mini-Pole Tutorial

This chapter provides a detailed tutorial for the Python package Mini-Pole, which implements the Minimal Pole Method (MPM) introduced in this dissertation. The package is designed to perform numerical analytic continuation (NAC) from Matsubara-axis data, hybridization fitting, and the extraction of compact pole representations from both Matsubara-axis and real-frequency-axis data. It supports both scalar-valued and matrix-valued correlation functions and provides several variants of MPM corresponding to the methods discussed in the previous chapters.

10.1. Installation and Setup

10.1.1. Dependencies

Mini-Pole requires Python >= 3.8, together with the following dependencies:

- NumPy >= 1.21.0. NumPy is a fundamental Python library for scientific computing. It provides efficient multidimensional arrays, linear algebra routines, numerical solvers, and related numerical algorithms.
- SciPy >= 1.7.0. SciPy is a Python library for scientific computing that builds on NumPy and provides additional functionality for linear algebra, optimization, interpolation, integration, and sparse matrix computations.
- Kneed. Kneed is a Python package for detecting knee and elbow points in curves. It is useful for automatically identifying characteristic changes in numerical data,

such as singular-value distributions.

- Matplotlib. Matplotlib is a Python library for creating static, animated, and interactive visualizations, and is widely used for plotting scientific data.

These packages are external dependencies of Mini-Pole and can typically be installed using the `pip install` command.

10.1.2. Installation Commands

The following commands download, build, and install Mini-Pole.

To install the latest unreleased version from source, use

```
$ git clone git@github.com:Green-Phys/MiniPole.git
$ cd MiniPole
$ python3 setup.py install
```

To install the latest released version via pip, use

```
$ pip install mini-pole
```

In general, the unreleased version may contain more functionality than the released version, but it may also be less stable. Users should therefore use it with caution.

After installation, the main classes can be imported as

```
from mini_pole import MiniPole, MiniPoleDLR, MiniPoleRf, MiniPoleRfDPR
```

The appropriate class depends on the task, as described in the following sections.

10.2. Usage

10.2.1. `MiniPole`

The standard MPM algorithm for Matsubara data is implemented in the class `MiniPole`. Note the distinction between the package name, `mini-pole`, and the class name, `MiniPole`. The class is named `MiniPole`, rather than `MiniPoleIf`, where "If" would stand for "imaginary frequency," for historical reasons: the Minimal Pole Method was originally developed for Matsubara data [38, 39].

10.2.1.1. Data preparation

The input data are values of a Green's function on a non-negative uniform Matsubara grid,

$$\{\boldsymbol{G}(i\omega_0), \boldsymbol{G}(i\omega_1), \cdots, \boldsymbol{G}(i\omega_{n_\omega-1})\}, \tag{10.1}$$

where $\omega_n = (2n+1)\pi/\beta$ for fermions and $\omega_n = 2n\pi/\beta$ for bosons. The input array can be either scalar-valued, with shape `(n_w,)`, or matrix-valued, with shape `(n_w, n_orb, n_orb)`, where `n_w` and `n_orb` denote the number of Matsubara points and orbitals, respectively.

If the original data are given on a non-uniform grid, for example from the intermediate representation (IR) [79] or the discrete Lehmann representation (DLR) [82], they should first be transformed to a uniform Matsubara grid [80, 109] before being used as input to `MiniPole`. Alternatively, one can use `MiniPoleDLR`, which is introduced in the next section.

When `symmetry=False`, the algorithm extracts poles in the lower-half complex plane, as explained below. In this case, the optimal value of n_ω is the one that makes the transformed poles $\tilde{\xi}_l$ as well separated as possible. Fortunately, as shown in Sec. 6.3.3, MPM is highly robust with respect to the number of available data points. In practice, it is typical to choose n_ω between β and 10β. When `symmetry=True`, the algorithm extracts poles on the real axis. In this case, one can choose the largest n_ω that is computationally affordable, since the running time increases with n_ω. In practice, choosing $n_\omega = 10\beta$ is usually sufficient.

The code uses the up-down symmetry (3.34) to reconstruct $\boldsymbol{G}(i\omega_n)$ at negative Matsubara frequencies. If the Matsubara data slightly violate this symmetry for numerical reasons, one can pre-symmetrize the data by applying

$$\frac{\boldsymbol{G}(i\omega_n) + [\boldsymbol{G}(-i\omega_n)]^\dagger}{2} \to \boldsymbol{G}(i\omega_n) \tag{10.2}$$

for $n \geq 0$ before passing the data to `MiniPole`.

10.2.1.2. Input parameters

The `MiniPole` class is used as follows:

```
p = MiniPole(G_w, w, n0="auto", err=None, err_type="abs",
             M=None, symmetry=False, compute_const=False)
```

The parameters have the following meanings:

- `G_w` – an `(n_w, n_orb, n_orb)` or `(n_w,)` array containing the Matsubara data.
- `w` – an `(n_w,)` array containing the corresponding real-valued Matsubara grid.
- `n0` – an integer or string corresponding to n_0 in the main text, with default value `"auto"`. If set to `"auto"`, `n0` is selected automatically. If a non-negative integer is provided, `n0` is fixed to that value.
- `err` – a floating-point number specifying the error tolerance for the calculation.
- `err_type` – a string with default value `"abs"`. It specifies the type of error tolerance: `"abs"` for absolute error and `"rel"` for relative error.
- `M` – an optional integer specifying the number of poles in the final result. If it is not specified, the precision from the first ESPRIT step is used to extract poles in the second ESPRIT step.
- `symmetry` – a Boolean variable with default value `False`. It determines whether to preserve the up-down symmetry (3.34) of the Green's function.
- `compute_const` – a Boolean variable with default value `False`. It determines whether to compute the constant term in $\boldsymbol{G}(z) = \sum_l \boldsymbol{A}_l/(z-\xi_l) + \text{const}$. If `False`, the constant term is fixed to zero.

Other parameters, such as `n0_shift`, `G_symmetric`, `plane`, `include_n0`, `k_max`, and `ratio_max`, are advanced options used to control the details of automatic interval selection, symmetrization, pole-weight calculation, and contour integration. They can usually be left at their default values.

If `const` is nonzero and its value is known, it is advisable to subtract it from the input data beforehand and set `compute_const=False`. Cases requiring `compute_const=True` are rare. When this option is used, the computed constant is stored in `p.const`. For simplicity, we use the default value throughout this tutorial.

After the calculation, the recovered complex pole representation can be accessed through `p.pole_weight` and `p.pole_location`. The former stores the pole weights $\boldsymbol{A}_l$ with shape `(M, n_orb, n_orb)`, while the latter stores the shared pole locations ξ_l with shape `(M,)`.

The resulting Green's function is represented as

$$\boldsymbol{G}(z) \approx \sum_{l=1}^{M} \frac{\boldsymbol{A}_l}{z - \xi_l}, \tag{10.3}$$

where M is the number of recovered poles, either specified manually or determined automatically. The real-frequency spectral function can then be evaluated from the boundary value at $z = \omega + i0^+$.

10.2.1.3. Numerical analytic continuation

For numerical analytic continuation (NAC), one usually wants to determine `n0` automatically. This means that the first ESPRIT step is used to approximate the continuous interval

$$[\boldsymbol{G}(i\omega_{n_0}), \boldsymbol{G}(i\omega_{n_\omega - 1})], \tag{10.4}$$

rather than the full set of discrete points

$$\{\boldsymbol{G}(i\omega_0), \boldsymbol{G}(i\omega_1), \cdots, \boldsymbol{G}(i\omega_{n_\omega - 1})\}. \tag{10.5}$$

The first n_0 points are excluded from the first ESPRIT step because including them can lead to an inaccurate approximation on the continuous interval $[\boldsymbol{G}(i\omega_0), \boldsymbol{G}(i\omega_{n_0})]$. Such an error would reduce the reliability of the contour-integral calculation and introduce uncontrolled errors into the final result. Although including these low-frequency points can be useful for hybridization fitting or compact representation, it is usually not beneficial for NAC, where the primary goal is to recover the spectral function reliably. If desired, one may set `include_n0=True` to include the first n_0 Matsubara points when calculating the pole weights. In practice, however, we find that this option provides little improvement.

Unlike in hybridization fitting, the number of resulting poles is usually not fixed in advance for NAC. Therefore, `n0` and `M` can almost always be left at their default values.

For finite-size systems with only delta peaks, one may set `symmetry=True` to restrict the poles to the real axis. In all other cases, one can keep the default value `symmetry=False`, which allows poles in the lower-half complex plane, including the real axis.

Therefore, the use of `MiniPole` for NAC usually reduces to

```
p = MiniPole(G_w, w, err=err, err_type=err_type)
```

so that only the error tolerance `err` and its type `err_type` need to be specified. A useful way to choose these parameters is to inspect the singular-value distribution from the first ESPRIT step, which can be plotted by

```
p.check_svd()
```

Fig. 10.1 shows two typical singular-value distributions. The dashed line indicates the singular value currently used as the cutoff. In practice, the cutoff singular value should be chosen as the last singular value that remains in the first clean exponentially decaying regime of this semilog plot. In Fig. 10.1(b), one can choose a cutoff below the dashed line and obtain a more accurate approximation to the input data. However, doing so overfits the data and produces spurious features in the reconstructed spectral function.

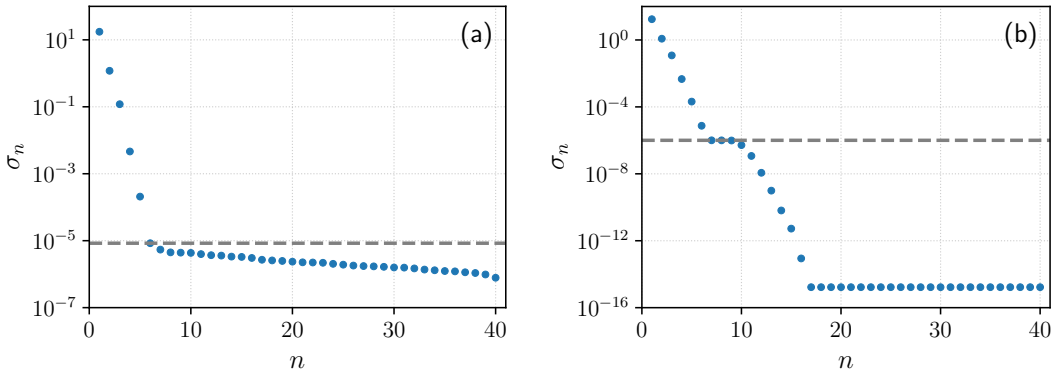

Figure 10.1: Two typical patterns of singular-value distributions in the first ESPRIT step for (a) data with random noise at level 10^{-5} and (b) data without random noise but with systematic bias at level 10^{-6}, which may arise from loss of precision during iterative simulations. The gray lines indicate the recommended cutoffs.

In the first ESPRIT step, we use a minimal number of exponentials to approximate $\boldsymbol{G}(z)$, even though $\boldsymbol{G}(z)$ is expected to have a pole representation rather than a finite exponential representation. Therefore, the singular-value distributions generally

do not exhibit the pattern shown in Fig. 4.1(a). Based on this observation, one may use algorithms such as the Kneedle algorithm [168, 169] to estimate the error tolerance automatically. This is achieved by leaving `err` unspecified, for example by using `p = MiniPole(G_w, w)`. As a result, all three commands below produce the same results shown in Fig. 10.2.

```
p = MiniPole(G_w, w)
p = MiniPole(G_w, w, err=1e-4)
p = MiniPole(G_w, w, err=1e-5, err_type="rel")
```

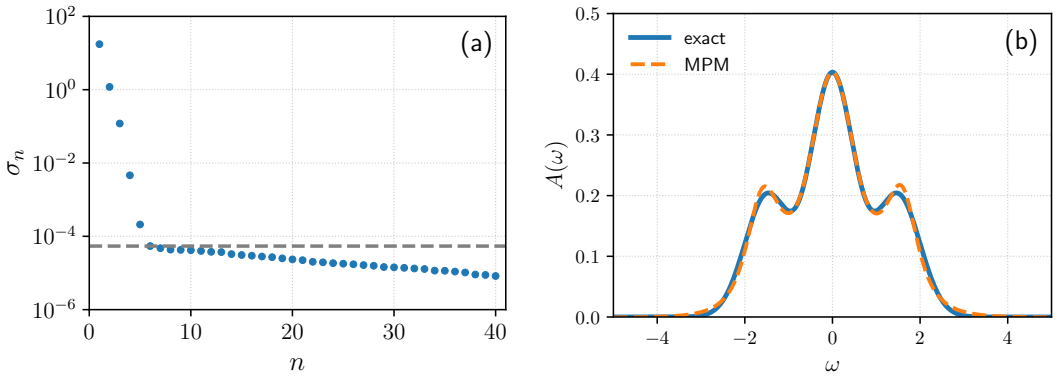

Figure 10.2: (a) Singular-value distribution of the first ESPRIT step for an example with absolute noise level 10^{-4}, corresponding to relative noise level 10^{-5}. The dashed line shows the chosen cutoff, which can be obtained by any of the three commands introduced in the main text. (b) The recovered spectral function, which becomes closer to the exact result as the noise level decreases.

Although the Kneedle algorithm often provides a good estimate of the cutoff singular value, it is not guaranteed to work in all cases. A practical strategy is therefore to decrease `err` gradually until the analytic properties of the recovered spectral function are visibly violated. For the diagonal components of fermionic Green's functions, this typically means that the spectral function ceases to be visually positive. At that point, the input data have been overfitted, and the previous cutoff value usually provides a near-optimal choice.

In principle, one could introduce an additional option to strictly constrain the spectral shape using Eq. (7.17) or Eq. (7.18), even when the input data are inaccurate. In

practice, however, we find that if the unrestricted calculation does not produce a solution that is visually positive, imposing positivity manually does not make the result reliable, even though it may make the spectrum positive. For this reason, this option is not currently included in the code. Instead, the degree to which positivity is violated is used as an indicator of the reliability of the result.

Before concluding this section, we mention that the intermediate steps of `MiniPole` can be checked using

```
p.check_valid()
```

Fig. 10.3 shows a series of plots generated by this command, ordered from left to right and then from top to bottom. To verify that `MiniPole` works as expected, one should check the following points. First, the singular-value cutoff in the first subplot should follow the criterion described above. The dashed line shows the chosen cutoff, while the solid line shows the actual precision of the first ESPRIT approximation; these two lines should be close to each other. Second, the first ESPRIT approximation should be controlled by the solid line. Third, the contour integral in the third subplot should decay continuously without flattening near the final points. Fourth, the singular values in the second ESPRIT step, shown in the fourth subplot, should be truncated using the same principle as in the first subplot. Fifth, the approximation of the contour integral should be well controlled. Sixth, the final approximation in the sixth subplot should also be controlled, possibly with an error about one order of magnitude larger than the target precision because of accumulated errors during the full computation. Seventh, the complex poles in the original plane, shown in the seventh subplot, should be located in the lower-half plane. Finally, the complex poles in the mapped plane, shown in the last subplot, should be visually separated rather than clustered around the same point.

10.2.1.4. Hybridization fitting and compact representation

Both hybridization fitting and compact representation of Matsubara data aim to recover a set of real poles. To achieve this, one can set `symmetry=True` to enforce the up-down symmetry (3.34). Unlike in NAC, the goal here is to optimize the approximation on the discrete points $\{\boldsymbol{G}(i\omega_0), \boldsymbol{G}(i\omega_1), \cdots, \boldsymbol{G}(i\omega_{n_\omega-1})\}$. Therefore, one usually does not want to discard any points in the first ESPRIT step, and one can set `n0=0`.

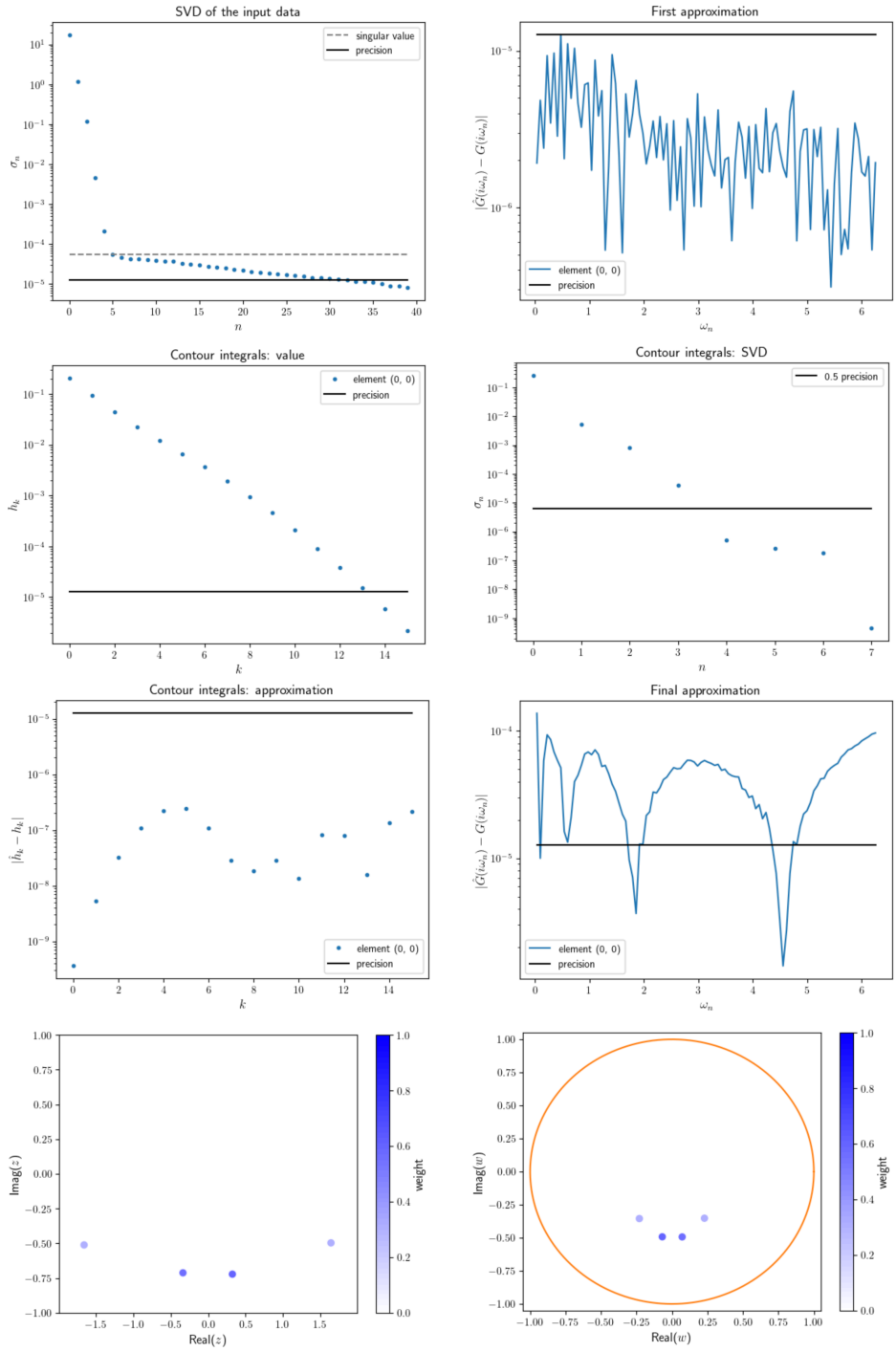

Figure 10.3: A series of plots generated by `p.check_valid()` for checking the validity of the intermediate steps.

For compact representation, one usually does not fix the number of poles. In other words, `M` is kept at its default value, `None`. Therefore, the use of `MiniPole` for compact representation reduces to

```
p = MiniPole(G_w, w, n0=0, err=err, err_type=err_type, symmetry=True)
```

so that one only needs to specify the desired precision level. This is the method used in Ref. [104] to generate the discrete pole representation of MPM.

For hybridization fitting, the number of poles `M` is usually essential. The resulting pole representation is often used to construct an effective model, which is then solved using diagonalization-based techniques. The computational cost of such solvers is limited by the size of the Hilbert space, which is determined by `M`. As a result, the use of `MiniPole` for hybridization fitting reduces to

```
p = MiniPole(G_w, w, n0=0, err=err, err_type=err_type,
             M=M, symmetry=True)
```

The precision level `err` should be set below the final precision achievable by the prescribed number of poles. As long as it is smaller than this value, its precise value does not affect the final approximation quality.

As in NAC, one can use `p.check_valid()` to inspect the intermediate steps and gain confidence in the final result.

10.2.2. `MiniPoleDLR`

The package also provides a DLR-based variant of `MiniPole`, implemented in the class `MiniPoleDLR`. This version starts from the discrete Lehmann representation coefficients [82, 109] rather than from raw Matsubara data.

As mentioned in Sec. 7.4.7, `MiniPoleDLR` can be viewed as a faster version of `MiniPole` with a larger effective `n0`. It usually runs faster, but it may produce less accurate results than `MiniPole` when the input data are not sufficiently accurate. However, when the input precision is more than sufficient, `MiniPoleDLR` is preferable because it can produce spectral functions of comparable quality to those obtained from the original `MiniPole`, while requiring substantially less computational time.

The `MiniPoleDLR` class is used as follows:

```
p = MiniPoleDLR(Al_dlr, xl_dlr, beta, n0, nmax=None, err=None,
                err_type="abs", M=None, symmetry=False)
```

Here, instead of using `G_w` and `w`, one uses `Al_dlr`, which contains the DLR coefficients, `xl_dlr`, which contains the DLR real-frequency grid, and `beta`, which is the inverse temperature. The additional parameter `nmax` represents the upper limit of the Matsubara frequency when `symmetry=False`. Its default value is equal to `beta` and can usually be kept unchanged. Other parameters have the same usage as in `MiniPole`; therefore, `MiniPoleDLR` can be readily used for NAC, hybridization fitting, or compact representation.

10.2.3. `MiniPoleRf`

The standard MPM algorithm for real-frequency fitting [40] is implemented in the class `MiniPoleRf`, where "Rf" stands for "real frequency."

10.2.3.1. Data preparation

The primary object to approximate is the spectral function $A(\omega)$ defined in Eq. (8.1). Although the method can be used to approximate a generic matrix-valued real-frequency Green's function, we focus on $A(\omega)$ for simplicity. Applications to more general functions can be found in the examples folder of the repository.

Currently, the input must be an analytic expression for $A(\omega)$. If the input is sampled on a finite real-frequency grid, one can first construct an analytic approximation, for example using spline interpolation. The quality of this approximation improves as the sampling grid becomes denser. Alternatively, one can use an algorithm such as the AAA algorithm [152, 153] to extract a discrete pole representation and then pass it to `MiniPoleRfDPR`, which is introduced in the next section.

10.2.3.2. Input parameters

The `MiniPoleRf` class is used as follows:

```
p = MiniPoleRf(G_rf, func_type="real", interval_type="infinite",
               w_min=-10, w_max=10, wp_max=1, sing_vals=None,
               err=None, M=None)
```

The parameters have the following meanings:

- `G_rf` – a list of length `n_orb**2` containing analytic expressions for the real-frequency functions.
- `func_type` – a string specifying the type of functions in `G_rf`. It can be either `"real"` for real-valued functions or `"complex"` for complex-valued functions.
- `interval_type` – a string specifying the type of real-frequency interval. It can be either `"infinite"` for the entire real-frequency axis or `"finite"` for a finite interval.
- `w_min` – a floating-point number with default value `-10`. It specifies the lower bound of the finite real-frequency interval, corresponding to $\omega_{\min}$ in the mapping (8.4).
- `w_max` – a floating-point number with default value `10`. It specifies the upper bound of the finite real-frequency interval, corresponding to $\omega_{\max}$ in the mapping (8.4).
- `wp_max` – a floating-point number with default value `1`. It specifies the parameter used in the Möbius transformation for the infinite real-frequency interval, corresponding to ω_p in the mapping (8.3).
- `sing_vals` – an optional list containing the singular points of `G_rf`. If it is not provided, `G_rf` is assumed to have no singular points on the interval of interest.
- `err` – a floating-point number specifying the error tolerance used during the approximation process.
- `M` – an optional integer specifying the number of poles in the final result. If it is not specified, the number of poles is determined automatically from the prescribed error tolerance.

Other parameters, such as `compute_const`, `k_max`, and `Lfactor`, are advanced options used to control the details of constant-term computation, contour integration, and the ESPRIT implementation. They can usually be left at their default values.

Like `MiniPole` and `MiniPoleDLR`, the method returns a minimal pole representation. The pole weights are stored in `p.pole_weight`, a NumPy array of shape `(M, n_orb, n_orb)`,

and the shared pole locations are stored in `p.pole_location`, a NumPy array of shape `(M,)`.

The resulting real-frequency function is represented as

$$A(\omega) \approx \sum_{l=1}^{M} \left(\frac{A_l^{(\mathrm{dn})}}{\omega - \xi_l^{(\mathrm{dn})}} + \frac{A_l^{(\mathrm{up})}}{\omega - \xi_l^{(\mathrm{up})}} \right). \tag{10.6}$$

Here, M denotes the number of recovered poles in the lower-half plane. The precise connection between A_l, ξ_l, M, and the code variables `p.pole_weight`, `p.pole_location`, and `M` depends on whether the mapping (8.3) or the mapping (8.4) is used, as explained below.

10.2.3.3. Conformal mapping (8.3)

When the mapping (8.3) is used, the whole real axis is mapped to the unit circle. Therefore, one should set `interval_type="infinite"`. The parameter `wp_max` is the parameter ω_p in Eq. (8.3) and should be chosen to be comparable to the relevant energy window. Since we focus here on approximating the spectral function $A(\omega)$, which is real-valued, one can keep `func_type` at its default value. It should be changed to `"complex"` whenever the input function is complex-valued. For this mapping, only $\{A_l^{(\mathrm{dn})}, \xi_l^{(\mathrm{dn})}\}$ is mapped inside the unit circle. Therefore, if one wants to specify the number of output poles, one should set `M` equal to the desired number of lower-half-plane poles. A typical usage is therefore

```
p = MiniPoleRf(G_rf, interval_type="infinite", wp_max=wp_max,
               sing_vals=sing_vals, err=err, M=M)
```

As an example, consider a semicircular spectral function centered at the origin with radius 2, `A_w = lambda w: np.sqrt(np.maximum(4 - w**2, 0.0))`. Its energy window is 2, and it has singular points at $\omega = \pm 2$. If one aims to obtain $M = 10$ poles in the lower-half plane, then `MiniPoleRf` can be used as follows:

```
p = MiniPoleRf([A_w], interval_type="infinite", wp_max=2,
               sing_vals=[-2, 2], err=1e-8, M=10)
```

Here, `err` is chosen to be smaller than the final precision produced by the recovered poles. As in Sec. 10.2.1.4, as long as it is smaller than this value, its precise value

does not affect the final approximation quality. The resulting 10 poles $\{A_l^{(\mathrm{dn})}, \xi_l^{(\mathrm{dn})}\}$ are stored in `p.pole_weight` and `p.pole_location`.

10.2.3.4. Conformal mapping (8.4)

When the mapping (8.4) is used, the finite real-frequency interval $[\omega_{\min}, \omega_{\max}]$ is mapped to the unit circle. Therefore, one should set `interval_type="finite"`. The parameters `w_min` and `w_max` correspond to $\omega_{\min}$ and $\omega_{\max}$ in Eq. (8.4). They should be chosen to cover a sufficiently large energy window so that the long-frequency tail of the spectral function is captured. For this mapping, both $\{A_l^{(\mathrm{dn})}, \xi_l^{(\mathrm{dn})}\}$ and $\{A_l^{(\mathrm{up})}, \xi_l^{(\mathrm{up})}\}$ are mapped inside the unit circle. Therefore, if one wants to obtain M lower-half-plane poles, one should set `M` equal to $2M$ in the code. A typical usage is therefore

```
p = MiniPoleRf(G_rf, interval_type="finite", w_min=w_min,
               w_max=w_max, sing_vals=sing_vals, err=err, M=M)
```

As a demonstration, consider again the semicircular spectral function introduced above. If one aims to obtain $M = 10$ poles in the lower-half plane, then `MiniPoleRf` can be used as follows:

```
p = MiniPoleRf([A_w], interval_type="finite", w_min=-6, w_max=6,
               sing_vals=[-2, 2], err=1e-8, M=20)
```

In this case, `p.pole_weight` and `p.pole_location` contain both $\{A_l^{(\mathrm{dn})}, \xi_l^{(\mathrm{dn})}\}$ and $\{A_l^{(\mathrm{up})}, \xi_l^{(\mathrm{up})}\}$, giving 20 poles in total.

10.2.4. `MiniPoleRfDPR`

`MiniPoleRfDPR` is a variant of `MiniPoleRf` for real-frequency fitting [40]. It takes a discrete pole representation as input rather than an analytic expression for the real-frequency function. For example, when the input data are sampled only on a finite grid, one can first apply the AAA algorithm with a small tolerance ε to prioritize approximation accuracy over the number of poles. This usually produces more poles than can be handled efficiently by the target numerical solver. One can then use these poles as the input to `MiniPoleRfDPR` and specify `M` to reduce the number of poles.

In principle, one could specify the number of poles M directly in the AAA algorithm rather than specifying the tolerance ε. However, as shown in Ref. [40], convergence with respect to the number of poles is usually slower for AAA than for MPM. In practice, `MiniPoleRfDPR` usually produces an approximation comparable to that obtained from `MiniPoleRf`. The choice between the two depends on the form of the available input data and should be made on a case-by-case basis.

The `MiniPoleRfDPR` class is used as follows:

```
p = MiniPoleRfDPR(Al_dpr, xl_dpr, interval_type="infinite",
                  w_min=-10, w_max=10, wp_max=1, err=None,
                  err_type="abs", M=None)
```

Here, the analytic expression `G_rf` is replaced by the discrete pole representation `Al_dpr` and `xl_dpr`, which are related to the spectral function through Eq. (8.1). This workflow is useful when another rational approximation method first produces a large pole representation, and MPM is then used as a compression step to obtain a more compact representation.

10.3. Summary

`Mini-Pole` provides a practical implementation of the complex pole representation and the Minimal Pole Method (MPM) developed in this dissertation. The class `MiniPole` implements the standard procedure for uniformly spaced Matsubara data, including numerical analytic continuation (NAC), imaginary-axis hybridization fitting, and the extraction of compact representations for imaginary-axis Green's functions. `MiniPoleDLR` provides a DLR-based variant of `MiniPole` that is faster and uses a larger effective n_0 for NAC. `MiniPoleRf` applies MPM to real-frequency fitting when the input is an analytic expression for the real-axis function. Finally, `MiniPoleRfDPR` is a variant of `MiniPoleRf` that works directly with discrete pole representations. Together, these tools make it possible to apply the methods developed in this thesis to NAC on the imaginary axis, as well as to hybridization fitting and compact representation on both real- and imaginary-frequency axes, for both scalar-valued and matrix-valued correlation functions.

Chapter 11

Conclusions and Outlook

This dissertation developed the complex pole representation as a unified framework for representing Green's functions and correlation functions in quantum many-body simulations. The central idea is that continuous spectral and correlation functions, which in conventional descriptions are associated with branch cuts or dense sets of real poles, can often be approximated accurately by a small number of complex poles with complex-valued weights. This viewpoint provides both a compact numerical representation and a physically motivated way to organize real-frequency information.

The first part of this dissertation introduced the theoretical and numerical foundations needed for this construction. We reviewed the second-quantization formalism and the single-particle Green's functions used throughout the thesis. The Lehmann representation shows that finite-size Green's functions can be expressed as sums over real poles, while extended systems lead naturally to continuous spectra and broadened spectral features. This observation motivated the complex pole representation developed in Ch. 5. There, we analyzed how complex-valued poles and weights can approximate broadened peaks, how the pole locations are constrained by causality and the Kramers–Kronig relations, and why strict positivity becomes delicate in complex pole representations. We also discussed the matrix-valued extension and the representation of other real-axis Green's functions.

The numerical foundation of the method was provided by Prony-like algorithms, especially ESPRIT and its matrix-valued extension. Combined with holomorphic mappings, these techniques allow the extraction of minimal pole information from sampled data. This combination led to the Minimal Pole Method (MPM), which was the main com-

putational framework developed in this dissertation.

The first main application of MPM was numerical analytic continuation. In Ch. 6, we showed that Matsubara response functions can be approximated by a minimal complex pole representation to within a prescribed precision. The resulting real-frequency spectral functions converge systematically to the exact result as the precision of the Matsubara input data is improved. This behavior is in contrast to the traditional view of analytic continuation as a purely ill-conditioned post-processing problem. The method was shown to work for discrete and continuous spectra, for diagonal and off-diagonal response functions, and for both fermionic and bosonic cases. In the presence of noise, fitting to the known precision of the input data yields physically meaningful spectra that improve as the uncertainty on the Matsubara axis is reduced.

The second main development was the matrix-valued extension presented in Ch. 7. In realistic quantum many-body calculations, Green's functions, self-energies, and susceptibilities are often matrix-valued objects. Treating each element independently may ignore shared analytic structure and lead to inconsistent continuations. The matrix-valued MPM resolves this issue by allowing different matrix elements to share the same pole locations while carrying distinct matrix-valued weights. We further introduced several systematic improvements, including restrictions to real poles when appropriate, constraints on spectral shapes, a combination with the discrete Lehmann representation, the incorporation of prior knowledge, and an automatic procedure for selecting the initial Matsubara point. These developments broaden the applicability of the method while preserving systematic improvability.

The third main application was real-frequency bath fitting and compact representation, discussed in Ch. 8. There, the complex pole perspective was transferred from the Matsubara axis to the real-frequency axis. Spectral densities, real-frequency Green's functions, and bath correlation functions were represented by compact sets of complex poles. This provides an alternative to nonlinear fitting procedures commonly used in quantum impurity models, open quantum systems, and real-time simulations. The resulting representation achieves accuracy comparable to state-of-the-art methods on short-time intervals, while providing a more controlled description of long-time tail behavior.

The fourth main application was the construction of quantum impurity solvers, discussed in Ch. 9. We explored both imaginary-axis and real-axis formulations using the

single-impurity Anderson model as an example. On the imaginary axis, complex poles can be used to construct non-Hermitian Hamiltonians for non-interacting problems. However, once interactions are introduced, the fact that the complex pole representation is valid only in one half of the complex plane becomes a fundamental limitation. Both Hamiltonian-based and diagrammatic imaginary-axis calculations then require additional restrictions that effectively lead back to real-pole-based formulations. On the real axis, by contrast, the complex pole representation is naturally compatible with the analytic structure of retarded, advanced, greater, lesser, time-ordered, anti-time-ordered, and Keldysh Green's functions. This allowed us to construct compact real-axis impurity solvers, including a Liouvillian-based solver for nonequilibrium steady states and a complex-pole-based diagrammatic solver for real-axis perturbation theory. These results show that complex poles can serve not only as tools for pre- and post-processing, but also as active computational degrees of freedom in quantum impurity simulations.

Finally, Ch. 10 presented Mini-Pole, an open-source Python implementation of the methods developed in this dissertation. The package provides tools for analytic continuation, compact representation, hybridization fitting, and pole compression. This software component makes the complex pole framework accessible for practical use and provides a reproducible implementation of the algorithms introduced throughout the dissertation.

Taken together, these results show that numerical analytic continuation, hybridization fitting, compact representation, and quantum impurity solvers are closely related problems. Each seeks to replace a complicated continuum object by a finite set of parameters while preserving the information needed for physical calculations. The complex pole representation provides a common language for these tasks. It also reveals analytic information, such as pole locations and weights, that can be useful not only for numerical approximation but also for physical interpretation and direct many-body simulation.

Several directions remain open for future work. First, although MPM provides a systematically improvable route to the analytic continuation of Matsubara data and can, in principle, recover the spectral function to a prescribed precision when sufficiently accurate input data are available, practical simulations often provide data with only limited precision. This is especially relevant for quantum Monte Carlo simulations, where statistical noise limits the amount of recoverable information. In this situation,

extracting the minimal number of poles may not always yield the most accurate spectral function. In the extreme limit of very noisy data, the method may recover only a single pole, thereby losing finer spectral features. It therefore remains an open question whether there exist alternative representations or regularization strategies that retain systematic improvability while converging more rapidly for certain physically relevant classes of systems.

Second, the real-axis impurity solvers developed in this dissertation are still at an early stage. The Liouvillian-based solver and the complex-pole-based diagrammatic solver demonstrate the feasibility of using complex poles as active computational degrees of freedom, but many extensions remain to be explored. For the Liouvillian-based approach, it will be important to improve the construction of effective open quantum systems, reduce the number of auxiliary degrees of freedom, and develop more efficient non-Hermitian many-body solvers. For the diagrammatic approach, future work may extend the current second-order perturbative calculations to higher-order diagrams, self-consistent schemes, and more realistic multi-orbital impurity models. These developments could provide compact, direct real-axis alternatives to conventional impurity solvers.

Third, the Mini-Pole package provides a practical implementation of the algorithms developed in this dissertation, but many software improvements remain possible. Future versions could include more automated parameter selection, better diagnostics for noisy input data, and tighter integration with existing many-body simulation packages. Such developments would make the complex pole framework easier to apply to realistic problems and would help connect the methods developed here to broader computational workflows.

Finally, the complex pole representation developed in this dissertation may also be useful for two-particle correlation functions, vertex functions, and other quantities that appear in many-body theory. These objects are often high-dimensional and difficult to store, manipulate, or continue to the real axis. Generalizing the complex pole representation to such quantities could possibly provide new tools for studying response functions, collective modes, and effective interactions.

Appendix A

Implementation Details of the Scalar-Valued Minimal Pole Method

The input of our simulations is an odd number of Matsubara points $G(i\omega_n)$ sampled on a uniform grid

$$\{i\omega_{n_0}, i\omega_{n_0+\Delta n}, \cdots, i\omega_{n_0+(N_\omega-1)\Delta n}\}, \tag{A.1}$$

where $\omega_n = \frac{(2n+1)\pi}{\beta}$ for fermions and $\frac{2n\pi}{\beta}$ for bosons, $n_0 \geq 0$ is an integer controlling the number of the first few points we decide to discard (if any), $\Delta n \geq 1$ is an integer controlling the distance of successive sampling points, N_ω is the total number of sampling points and should be an odd number. We find that it is sometimes advantageous to choose n_0 different from 0 (for fermions) or 1 (for bosons). In this case, the final interpolant has to be validated at the discarded points to ensure that they are consistent with the interpolant. For concreteness, in our simulations we choose $\beta = 200$, $N_\omega = 2001$ and $\Delta n = 1$, unless otherwise specified.

For this work, unless specified, the Matsubara data is always obtained from a known spectral function $A(\omega)$ via

$$G(i\omega_n) = \int_{-\infty}^{+\infty} d\omega \frac{A(\omega)}{i\omega_n - \omega} \, . \tag{A.2}$$

After obtaining pole information by our method, the recovered spectral function is

obtained from

$$A_{\text{cont}}(\omega) = -\frac{1}{\pi} \operatorname{Im} \lim_{\eta \to 0^+} \sum_{l=1}^{M} \frac{A_l}{\omega + i\eta - \xi_l} \ . \tag{A.3}$$

And the quality of the analytic continuation is characterized by the ℓ^1 norm of the discrepancy:

$$err(A) = \int_{-\infty}^{+\infty} d\omega |A(\omega) - A_{\text{cont}}(\omega)| \ . \tag{A.4}$$

For broadened peaks, since poles are away from the real axis, we take $\eta \equiv 0$. For delta peaks, η is always chosen to be 0.01 for both visualization and evaluation of $err(A)$, unless otherwise specified.

To facilitate later discussions, two functions, the Gaussian function and the Lorentzian function, are defined here:

$$g(\omega, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{ -\frac{(x-\mu)^2}{2\sigma^2} \right\} \ , \tag{A.5}$$

$$l(\omega, \mu, \gamma) = \frac{1}{\pi} \frac{\gamma}{(\omega - \mu)^2 + \gamma^2} \ . \tag{A.6}$$

A.1. Error Control

In this section, we simulate two models, one for the discrete case with both centered and off-centered delta peaks, the other for the continuous case with multiple-featured broadened peaks. For the former, the spectral function takes the form:

$$A(\omega) = 0.52\delta(\omega) + 0.48\delta(\omega - 1) \ , \tag{A.7}$$

where $\delta(w)$ is the Dirac delta function and the parameter n_0 is chosen to be 30 because of the singularity on the origin. For the later, we choose

$$A(\omega) = 0.2g(\omega, -2, 0.5) + 0.3g(\omega, 0, 0.2) + 0.5g(\omega, 1, 0.5) \tag{A.8}$$

with parameter $n_0 = 0$.

Recovered results are listed in Table A.1 and A.2, where the negligible imaginary part for the discrete case has been discarded for readability. Distinguishing delta peaks and

ε	M	A_l	ξ_l
2.55×10^{-4}	2	0.511333354898343	−0.005233280979342
		0.489020327225432	0.995141438211015
7.98×10^{-6}	2	0.519686962815029	−0.000294257935939
		0.480312064764053	1.000123355664247
1.83×10^{-7}	2	0.519996253000921	−0.000005670340152
		0.480003446182263	1.000008012212157
5.75×10^{-9}	2	0.520000036501130	−0.000000005157088
		0.479999958687507	1.000000183105766
1.60×10^{-10}	2	0.520000015775204	0.000000013352668
		0.479999984193619	0.999999995909646
8.26×10^{-13}	2	0.520000000194287	0.000000000223985
		0.479999999801583	0.999999999848198

Table A.1: Recovered $\{A_l, \xi_l\}$ for Fig. 6.2(a).

broadened peaks can be easily achieved by examining the imaginary part of ξ_l. Poles with negligible weights have also been discarded. Besides, there is subtlety associated with the predetermined error tolerance ε. When ε is given, the program looks for the first singular value from SVD which satisfies $\sigma \leq \varepsilon$. Because we do not distinguish ε and σ in the content of our work and σ is discrete, ε is also discrete in this sense. This is the reason why ε has several digits.

A.2. Dependence on Temperature

In this section, we test the dependence of performance on system temperature. The spectral function being tested is

$$A(\omega) = 0.3l(\omega, -2.5, 0.8) + 0.5l(\omega, 0, 0.5) + 0.2g(\omega, 2.5, 0.8) \ . \tag{A.9}$$

Here, N_ω is fixed at 201, and the precision levels are selected to be $\{10^{-3}, 10^{-6}, 10^{-9}, 10^{-12}\}$. The parameter n_0 for $\beta = 1, 2, 5, 10, 20, 50$ is chosen to be 10, 12, 12, 6, 4, and 2, respectively, regardless of the precision level.

ε	M	A_l	ξ_l
2.36×10^{-3}	3	$0.118058903808435 + 0.182403162681001i$	$-1.204450173782112 - 0.156711479499872i$
		$0.623126378429802 - 0.167121980005347i$	$-0.030090705376894 - 0.293343960793297i$
		$0.315285665426528 - 0.002154502881458i$	$1.048877531307533 - 0.111010313119802i$
1.89×10^{-5}	4	$0.294789142930615 - 0.030683407576160i$	$-1.946519611154891 - 0.483902193898281i$
		$0.195404650709736 - 0.620495075034599i$	$-0.146585427014933 - 0.335878950110814i$
		$-0.029769767413130 + 0.374041799008952i$	$0.177981280484458 - 0.295491343144327i$
		$0.534054140868870 + 0.274005525859195i$	$1.206305940667271 - 0.475472509411325i$
7.70×10^{-7}	5	$0.235930048552452 - 0.078646824324015i$	$-2.136509145947329 - 0.415697321174864i$
		$-0.246758054089975 - 0.960556415150371i$	$-0.174493698730584 - 0.399719372355751i$
		$0.137210004779730 + 1.154077808958321i$	$0.138510559424034 - 0.403564765336938i$
		$0.958566688287725 - 0.395613126648794i$	$0.786369292495720 - 0.692579926695643i$
		$-0.085315624707392 + 0.281372452486608i$	$1.734960150099575 - 0.630973836857521i$
1.04×10^{-9}	8	$-0.000869254038466 - 0.066370182984203i$	$-2.784179904883692 - 0.471055373757732i$
		$0.251842355381606 - 0.000121368742237i$	$-1.943630489721618 - 0.571923148208484i$
		$-0.173694882612084 - 0.087756520613306i$	$-0.806323900759767 - 0.637092057687596i$
		$-1.852955505830634 + 0.578497739787043i$	$-0.262523270780438 - 0.514116078156992i$
		$4.632053629971353 - 0.456112255262277i$	$-0.006147361455948 - 0.525774338640829i$
		$-2.337119090834560 - 0.482155607092420i$	$0.253151690147532 - 0.523458691783798i$
		$0.601579319546361 + 0.449531659601334i$	$1.204948811727744 - 0.733105192605620i$
		$-0.120836019433872 + 0.064484991851313i$	$1.946971017902517 - 0.674348076546615i$
3.54×10^{-11}	9	$-0.035616516363429 - 0.085123955396923i$	$-2.771552714415409 - 0.603953536741066i$
		$0.347803557080295 + 0.027535918871576i$	$-1.928181641845855 - 0.700395552874667i$
		$-0.799529779639230 + 0.045638491582258i$	$-0.640111461899102 - 0.956402202375529i$
		$-1.544146981258659 + 1.760650223719916i$	$-0.292521122921537 - 0.540703723620048i$
		$5.654018960196657 - 2.147975691168267i$	$-0.027662643155237 - 0.544480942308326i$
		$-3.050137945302188 - 0.175769757941538i$	$0.234638870973690 - 0.533138493156489i$
		$-0.000037772920485 + 0.000069877726134i$	$0.548923094314165 - 0.031530914939886i$
		$0.539369205783499 + 0.546534384372247i$	$1.254717651072944 - 0.746514675519833i$
		$-0.111722512793357 + 0.028440919741972i$	$2.007471710919555 - 0.696705820338941i$
2.14×10^{-13}	11	$-0.024837926566027 - 0.004366347509901i$	$-3.118069119689864 - 0.643167785434883i$
		$0.098395494318060 - 0.219620264536520i$	$-2.365185683165514 - 0.716876112030249i$
		$0.185967844930931 + 0.252384802247341i$	$-1.674049451686480 - 0.785786582368154i$
		$0.924104691602651 + 0.664403394148004i$	$-0.397811568145370 - 0.560220818702668i$
		$-4.265344958424443 - 4.603656276549160i$	$-0.157425734776960 - 0.577214844399315i$
		$5.440160882854892 + 6.435022023450554i$	$0.059136250873121 - 0.581043116187629i$
		$-1.677223757359942 - 2.886437519921194i$	$0.288553682833324 - 0.574535166568495i$
		$0.034681571989394 - 0.069689910633521i$	$0.814058514940610 - 0.481768187664549i$
		$0.346643348885711 + 0.325386507633010i$	$1.261498342766544 - 0.664043673218963i$
		$-0.050313356223844 + 0.105058861667653i$	$1.793090553825032 - 0.586679051877780i$
		$-0.012233839679708 + 0.001514730029528i$	$2.364554990881565 - 0.552750542706448i$

Table A.2: Recovered $\{A_l, \xi_l\}$ for Fig. 6.2(b).

A.3. Dependence on Data Points

We examine the dependence on the number of available data points in this section. The model used is given by Eq. A.9. We evaluate the performance at different precision levels $\{10^{-3}, 10^{-6}, 10^{-9}, 10^{-12}\}$ and various numbers of available points $N_\omega \in \{7, 9, 11, 35, 151, 1001\}$. The parameter n_0 is fixed at 10, and β is fixed at 30.

A.4. Challenging Examples

In this section, we examine density of states in $2d$ tight-binding models, as well as a 'Kondo'-like spectral function with both smooth and sharp features.

In (a), we study the model on the square lattice with nearest-neighbor interaction $t = 1$ and next-nearest-neighbor interaction $t' = 0.2$. Following the convention from Ref. [86], the spectral function can be expressed as

$$A(\omega) = \frac{1}{\pi^2 |t'| \sqrt{z_0(u^2-4)}} K\left(\frac{z_1}{z_0}\right) , \tag{A.10}$$

where $K(m) = \int_0^{\frac{\pi}{2}} d\phi [1 - m \sin^2 \phi]^{-1/2}$ is the complete elliptic integral of the first kind, and

$$z_0 = \begin{cases} q, & 0 < p \leq q \\ p, & 0 < q < p \\ p - q, & q < 0 \end{cases} , \tag{A.11}$$

$$z_1 = \begin{cases} q - p, & 0 < p \leq q \\ p - q, & 0 < q < p \\ p, & q < 0 \end{cases} . \tag{A.12}$$

Here, p and q are defined by two dimensionless parameters $u = t/t'$ and $E = \omega/t$:

$$p = \frac{4u(u-E)}{u^2-4} , \tag{A.13}$$

$$q = \frac{(Eu-4)^2}{4(u^2-4)} . \tag{A.14}$$

And finally, the non-zero range is determined by

$$-4 - \frac{4}{u} \leq E \leq 4 - \frac{4}{u}, \qquad |u| \geq 2. \tag{A.15}$$

Because of the sharp feature in the spectral function, we find that calculating Matsubara data from Eq. A.2 loses lots of precision. So instead, we obtain the input data from

$$G(i\omega_n) = \frac{1}{(2\pi)^2} \int_{-\pi}^{\pi} dk_1 \int_{-\pi}^{\pi} dk_2 \frac{1}{i\omega_n - \epsilon_k} \ , \tag{A.16}$$

where the tight-binding Hamiltonian has the expression

$$\epsilon_k = -2t \cos k_1 - 2t \cos k_2 - 4t' \cos k_1 \cos k_2 \ . \tag{A.17}$$

The simulation is performed at $n_0 = 0$.

In (b), we study the model on the Bethe lattice with interaction $t = 1$. The spectral function in this case is a semicircle:

$$A(\omega) = \frac{1}{2\pi t^2} \sqrt{4t^2 - \omega^2} \ . \tag{A.18}$$

And the Matsubara Green's function has the analytic expression

$$G(i\omega_n) = \frac{i}{2t^2} (\omega_n - \sqrt{\omega_n^2 + 4t^2}) \text{ for } \omega_n > 0 \ . \tag{A.19}$$

This is also simulated at $n_0 = 0$.

In (c), we study the model on an anisotropic triangular lattice with interaction $t = 1$ for two of the three directions and $t' = 0.75$ for the third direction. As shown in Ref. [86], the spectral function has the analytic form

$$A(\omega) = \frac{1}{\pi^2 t' \sqrt{z_0}} K\left(\frac{z_1}{z_0}\right) \ , \tag{A.20}$$

where

$$z_0 = \begin{cases} q, & 0 < \ p \leq q \\ p, & 0 < \ q < p \\ p - q, & q < 0 \end{cases} \ , \tag{A.21}$$

.

$$z_1 = \begin{cases} q-p, & 0 < \quad p \le q \\ p-q, & 0 < \quad q < p \\ p, & \quad q < 0 \end{cases} . \tag{A.22}$$

After the definition of two dimensionless parameters $u = t/t'$ and $E = \omega/t$, p and q can be expressed as

$$r = u\sqrt{u^2 - Eu + 2} \, , \tag{A.23}$$

$$p = 4r \, , \tag{A.24}$$

$$q = \frac{(r-u^2)^2(r^2 - 4u^2 + 2ru^2 + u^4)}{4u^4} \, . \tag{A.25}$$

The non-zero range is

$$-4 - \frac{2}{u} \le E \le u + \frac{2}{u}, \qquad 0 < u \le 2 \, , \tag{A.26}$$

$$-4 - \frac{2}{u} \le E \le 4 - \frac{2}{u}, \qquad u > 2 \, . \tag{A.27}$$

Similarly, we find the Matsubara data calculated from Eq. A.2 is also inaccurate. So instead, we obtain the input data from Eq. A.16 with the Hamiltonian

$$\epsilon_k = -2t\cos k_1 - 2t'\cos k_2 - 2t\cos(k_1 - k_2) \, . \tag{A.28}$$

Simulation is performed at $n_0 = 0$.

Finally, the spectral function in (d) has the form

$$A(\omega) = 0.2g(\omega, -2, 0.5) + 0.6\delta(\omega) + 0.2g(\omega, 2, 0.5) \, , \tag{A.29}$$

with $n_0 = 10$ is chosen for the simulation. For the inset, η is chosen to be a different value 0.001 to give a better visualization of the comparison.

Simulations for Maximum Entropy (MaxEnt) method and stochastic optimization method (SOM) are performed using the programs in Refs. [22] and [31], respectively. The parameter σ_n for MaxEnt is fine-tuned to yield the best possible spectrum, with values set at 10^{-5}, 10^{-4}, 10^{-5} and 10^{-3}, respectively. The weight factor $S(n)$ is set to be 1 for SOM.

A.5. Versatility

In (a) and (b), we perform simulations on fermionic grids and choose $n_0 = 0$ for both cases. For the former, the spectral function reads

$$A(\omega) = -0.5g(\omega, -3, 0.5) + 0.1g(\omega, -1, 1) - 0.1g(\omega, 1, 1) + 0.5g(\omega, 3, 0.5) \ . \tag{A.30}$$

And for the later, the spectral function is

$$A(\omega) = \sum_l A_l \delta(\omega - \xi_l) \tag{A.31}$$

with

$$A_l = \{-0.06842167, 0.30803739, 0.09215373, -0.28216683, -0.09213708, 0.04253446\}, \tag{A.32}$$

$$\xi_l = \{-4.63614978, -2.60745562, 0.86189661, 1.61810008, 3.33810339, 4.02550532\}, \tag{A.33}$$

which comes from the Exact Diagonalization result of a Hubbard dimer system.

For (c) and (d), simulations are performed on bosonic grids with $n_0 = 0$ and 10, respectively. The spectral function of (c) is

$$A(\omega) = -0.5g(\omega, -1, 1) + 0.5g(\omega, 1, 1) \tag{A.34}$$

and of (d) is

$$A(\omega) = \sum_l A_l \delta(\omega - \xi_l) \tag{A.35}$$

with

$$A_l = \{-0.1, -0.3, 0.2, 0.4, -0.4, -0.2, 0.3, 0.1\} \ , \tag{A.36}$$

$$\xi_l = \{-3, -2, -1, -0.02, 0.02, 1, 2, 3\} \ . \tag{A.37}$$

A.6. Noisy Data

We test the noise resistance of our method for both discrete and continuous cases. The discrete case is simulated at $n_0 = 0$ for

$$A(\omega) = 0.6\delta(\omega + 1) + 0.4\delta(\omega - 1) \ , \tag{A.38}$$

and the continuous case is simulated at $n_0 = 10$ for Eq. A.9. Noise is added to the clean data by

$$G_{\text{noisy}}(i\omega_n) = G_{\text{exact}}(i\omega_n)(1 + \delta \times N_{\mathbb{C}}(0, 1)) \ , \tag{A.39}$$

where $\delta \in \{10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}\}$ and $N_{\mathbb{C}}(0, 1)$ is the complex-valued normal Gaussian distribution. The input data we used is also included in the folder.

The parameter σ_n for performing MaxEnt is set to be $0.01 \times |G_{\text{noisy}}(i\omega_n)|$. We do not use $\sigma_n = \delta \times |G_{\text{noisy}}(i\omega_n)|$ for MaxEnt because it will cause the sawtooth noise instability. The weight factor $S(n)$ for SOM is also set to be proportional to $|G_{\text{noisy}}(i\omega_n)|$.

Appendix B

Implementation Details of the Liouvillian-Based Impurity Solver

B.1. Fermionic Operators in Tensor Product Form

In this section, we give a brief introduction to fermionic operators expressed in tensor product format, which is essential for understanding the super-fermion formalism introduced in the next section. Suppose we have two Hilbert spaces $\mathcal{H}_1$ and $\mathcal{H}_2$, which contain fermionic operators $\hat{d}_i^{(\dagger)}$ $(1 \leq i \leq N_1)$ and $\hat{c}_j^{(\dagger)}$ $(1 \leq j \leq N_2)$, respectively. The $\hat{d}^{(\dagger)}$'s and $\hat{c}^{(\dagger)}$'s satisfy their respective anticommutation relations

$$[\hat{d}_{i_1}, \hat{d}_{i_2}^\dagger]_+ = \delta_{i_1 i_2} \ , \quad [\hat{d}_{i_1}, \hat{d}_{i_2}]_+ = [\hat{d}_{i_1}^\dagger, \hat{d}_{i_2}^\dagger]_+ = 0 \ , \quad 1 \leq i_1, i_2 \leq N_1 \ , \tag{B.1}$$

$$[\hat{c}_{j_1}, \hat{c}_{j_2}^\dagger]_+ = \delta_{j_1 j_2} \ , \quad [\hat{c}_{j_1}, \hat{c}_{j_2}]_+ = [\hat{c}_{j_1}^\dagger, \hat{c}_{j_2}^\dagger]_+ = 0 \ , \quad 1 \leq j_1, j_2 \leq N_2 \ , \tag{B.2}$$

where $[\hat{A}, \hat{B}]_+ = \hat{A}\hat{B} + \hat{B}\hat{A}$ is the anticommutator of operators $\hat{A}$ and $\hat{B}$, as described in the main text. These operators can be used to generate states in their respective spaces: $|i\rangle = \hat{d}_{i_1}^\dagger \hat{d}_{i_2}^\dagger \cdots |0\rangle \in \mathcal{H}_1$ and $|j\rangle = \hat{c}_{j_1}^\dagger \hat{c}_{j_2}^\dagger \cdots |0\rangle \in \mathcal{H}_2$, where $i_1, i_2, \cdots$ and $j_1, j_2, \cdots$ are the occupied indices. However, no such relations exist between each $\hat{d}^{(\dagger)}$ and $\hat{c}^{(\dagger)}$, since they act on different spaces.

In practice, $\mathcal{H}_1$ could be the Hilbert space of the impurity system, while $\mathcal{H}_2$ could be the Hilbert space of the bath system. One usually wants to study the total system $\mathcal{H} = \mathcal{H}_1 \otimes \mathcal{H}_2$. If we denote a state in $\mathcal{H}$ by $|i, j\rangle \in \mathcal{H}$, then one needs to define operators $\hat{D}_i^{(\dagger)}$ $(1 \leq i \leq N_1)$ and $\hat{C}_j^{(\dagger)}$ $(1 \leq j \leq N_2)$, both of which act directly on $\mathcal{H}$. Here, we use

capital letters to denote operators acting on the full Hilbert space.

If one defines the normal order as

$$|i,j\rangle = \hat{D}^\dagger_{i_1}\hat{D}^\dagger_{i_2}\cdots\hat{C}^\dagger_{j_1}\hat{C}^\dagger_{j_2}\cdots|0\rangle \in \mathcal{H}\ , \tag{B.3}$$

then the $\hat{D}^{(\dagger)}$'s and $\hat{C}^{(\dagger)}$'s are connected to the $\hat{d}^{(\dagger)}$'s and $\hat{c}^{(\dagger)}$'s via

$$\hat{D}^\dagger_i = \hat{d}^\dagger_i \otimes \hat{\mathbb{1}} \quad , \quad \hat{D}_i = \hat{d}_i \otimes \hat{\mathbb{1}} \quad , \tag{B.4}$$

$$\hat{C}^\dagger_i = (-1)^{\sum_i \hat{d}^\dagger_i \hat{d}_i} \otimes \hat{c}^\dagger_i\ , \quad \hat{C}_i = (-1)^{\sum_i \hat{d}^\dagger_i \hat{d}_i} \otimes \hat{c}_i\ , \tag{B.5}$$

where $\hat{\mathbb{1}}$ is the identity operator acting on the second subspace. One can verify that the $\hat{D}^{(\dagger)}$'s and $\hat{C}^{(\dagger)}$'s satisfy the anticommutation relations

$$[\hat{D}_{i_1}, \hat{D}^\dagger_{i_2}]_+ = \delta_{i_1 i_2}\ , \quad [\hat{D}_{i_1}, \hat{D}_{i_2}]_+ = [\hat{D}^\dagger_{i_1}, \hat{D}^\dagger_{i_2}]_+ = 0\ , \tag{B.6}$$

$$[\hat{C}_{j_1}, \hat{C}^\dagger_{j_2}]_+ = \delta_{j_1 j_2}\ , \quad [\hat{C}_{j_1}, \hat{C}_{j_2}]_+ = [\hat{C}^\dagger_{j_1}, \hat{C}^\dagger_{j_2}]_+ = 0\ , \tag{B.7}$$

$$[\hat{D}_i, \hat{C}^\dagger_j]_+ = [\hat{C}_j, \hat{D}^\dagger_i]_+ = [\hat{D}_i, \hat{C}_j]_+ = [\hat{D}^\dagger_i, \hat{C}^\dagger_j]_+ = 0\ . \tag{B.8}$$

The state can therefore be written as

$$|i,j\rangle = |i\rangle \otimes |j\rangle = \hat{d}^\dagger_{i_1}\hat{d}^\dagger_{i_2}\cdots\otimes\hat{c}^\dagger_{j_1}\hat{c}^\dagger_{j_2}\cdots|0\rangle \in \mathcal{H}\ . \tag{B.9}$$

Thus, both the operators and the states in the total Hilbert space can be written either in terms of the $\hat{D}^{(\dagger)}$'s and $\hat{C}^{(\dagger)}$'s or in terms of the $\hat{d}^{(\dagger)}$'s and $\hat{c}^{(\dagger)}$'s. These two descriptions are equivalent. The benefits of using these two formalisms will become apparent in the next section.

B.2. Super-Fermion Representation

B.2.1. Standard Formalism

Let $\mathcal{H}$ be the Hilbert space of the physical system under study, endowed with fermionic operators a_i and $a^\dagger_i$ for $1 \leq i \leq N$, where the $\hat{}$ signs for operators are omitted in this section for later convenience. Suppose its complete orthonormal basis is $|n\rangle = |n_1 n_2 ... n_N\rangle$, where n is a composite index and its components satisfy $n_i \in \{0,1\}$. Then

the identity operator can be written as $\mathbb{1} = \sum_n |n\rangle\langle n|$ with $\langle m|n\rangle = \delta_{mn}$. According to Refs. [166, 170], one can introduce an identical copy of $\mathcal{H}$ and distinguish its states and operators with $\tilde{}$, i.e., $|\tilde{n}\rangle$ and $\tilde{a}_i^{(\dagger)}$. Then the full space $\mathcal{H}_{\rm tot} = \mathcal{H} \otimes \mathcal{H}$ is called the super-fermion space, with basis $|m, n\rangle = |m\rangle \otimes |\tilde{n}\rangle$.

One can define a so-called left-vacuum vector $|I\rangle = \sum_n |n\rangle \otimes |\tilde{n}\rangle$, so that a generic operator $\rho = \sum_{mn} \rho_{mn}|m\rangle\langle n|$ acting on the original space can be transformed one-to-one into a state vector $|\rho\rangle$ in $\mathcal{H}_{\rm tot}$, i.e.,

$$|\rho\rangle = \sum_{mn} \rho_{mn}|m\rangle \otimes |\tilde{n}\rangle = (\rho \otimes \mathbb{1})|I\rangle \in \mathcal{H}_{\rm tot} \ , \tag{B.10}$$

and its inner product with $|I\rangle$ gives the trace of the original operator:

$$\langle I|\rho\rangle = (\sum_n \langle n| \otimes \langle\tilde{n}|)(\rho \otimes \mathbb{1})(\sum_m |m\rangle \otimes |\tilde{m}\rangle) = {\rm Tr}[\rho] \ . \tag{B.11}$$

One can define the counterparts of $a_i^{(\dagger)}$ and $\tilde{a}_i^{(\dagger)}$ that act on the total space. Their actions on the basis states are

$$A_i|m, n\rangle = (a_i \otimes \mathbb{1})|m, n\rangle \ , \tag{B.12}$$
$$A_i^\dagger|m, n\rangle = (a_i^\dagger \otimes \mathbb{1})|m, n\rangle \ , \tag{B.13}$$
$$\tilde{A}_i|m, n\rangle = (-1)^{\sum_i (m_i - n_i)+1}(\mathbb{1} \otimes \tilde{a}_i)|m, n\rangle \ , \tag{B.14}$$
$$\tilde{A}_i^\dagger|m, n\rangle = (-1)^{\sum_i (m_i - n_i)}(\mathbb{1} \otimes \tilde{a}_i^\dagger)|m, n\rangle \ . \tag{B.15}$$

One can verify that this definition guarantees $(A_i^\dagger)^\dagger = A_i$ and $(\tilde{A}_i^\dagger)^\dagger = \tilde{A}_i$, and that these operators satisfy the anticommutation relations

$$[A_i, A_j^\dagger]_+ = \delta_{ij} \ , \quad [A_i, A_j]_+ = [A_i^\dagger, A_j^\dagger]_+ = 0 \ , \tag{B.16}$$
$$[\tilde{A}_i, \tilde{A}_j^\dagger]_+ = \delta_{ij} \ , \quad [\tilde{A}_i, \tilde{A}_j]_+ = [\tilde{A}_i^\dagger, \tilde{A}_j^\dagger]_+ = 0 \ , \tag{B.17}$$
$$[A_i, \tilde{A}_j^\dagger]_+ = [\tilde{A}_j, A_i^\dagger]_+ = [A_i, \tilde{A}_j]_+ = [A_i^\dagger, \tilde{A}_j^\dagger]_+ = 0 \ , \tag{B.18}$$

while their actions on the basis states give the occupation numbers

$$A_i^\dagger A_i|m,n\rangle = m_i|m,n\rangle \ , \tag{B.19}$$

$$\tilde{A}_i^\dagger \tilde{A}_i|m,n\rangle = n_i|m,n\rangle \ . \tag{B.20}$$

The power of this formalism is that their actions on the left-vacuum vector can be interchanged via

$$A_i|I\rangle = \tilde{A}_i^\dagger|I\rangle \ , \quad A_i^\dagger|I\rangle = -\tilde{A}_i|I\rangle \ , \tag{B.21}$$

so that superoperators acting on $\mathcal{H}$ can be transformed into operators acting on $\mathcal{H}_{\rm tot}$ by using the $A^{(\dagger)}$'s and $\tilde{A}^{(\dagger)}$'s. For example, consider a superoperator that acts on a density operator ρ as $a_i\rho a_j^\dagger$. In the full space, it can be written as

$$\begin{aligned}(a_i\rho a_j^\dagger)\otimes\mathbb{1}|I\rangle &= (a_i\otimes\mathbb{1})(\rho\otimes\mathbb{1})(a_j^\dagger\otimes\mathbb{1})|I\rangle \\ &= A_i(\rho\otimes\mathbb{1})A_j^\dagger|I\rangle = -A_i(\rho\otimes\mathbb{1})\tilde{A}_j|I\rangle \\ &= -A_i\tilde{A}_j(\rho\otimes\mathbb{1})|I\rangle = -A_i\tilde{A}_j|\rho\rangle \ ,\end{aligned} \tag{B.22}$$

where we have assumed that the density operator ρ is a particle-number-conserving operator acting on the original space, so that it commutes with $\tilde{A}_j$. As a result, this superoperator can be expressed as the operator $-A_i\tilde{A}_j$ acting on the state $|\rho\rangle \in \mathcal{H}_{\rm tot}$. This technique is therefore essential for rewriting superoperators as operators acting on the enlarged space, whose properties can then be studied using matrix diagonalization techniques.

Despite the convenience introduced so far, note that the defined operators $\tilde{A}^{(\dagger)}$'s are not creation and annihilation operators in the usual sense when acting on $|m,n\rangle$, i.e.,

$$|m,n\rangle \neq (A_1^\dagger)^{m_1}(A_2^\dagger)^{m_2}\cdots(A_N^\dagger)^{m_N}(\tilde{A}_1^\dagger)^{n_1}(\tilde{A}_2^\dagger)^{n_2}\cdots(\tilde{A}_N^\dagger)^{n_N}|0\rangle \ , \tag{B.23}$$

where $m_i, n_i \in \{0,1\}$, as mentioned before. This point can also be verified by comparing the difference between Eqs. (B.14) and (B.15) with Eq. (B.5). In practice, if one has a large system and wants to perform bit-wise operations on states, this formalism can become inconvenient, since the actions of $\tilde{A}_i^{(\dagger)}$ depend on both the occupation numbers before and after this index. In the next section, we introduce an equivalent formalism that can be more convenient.

B.2.2. Alternative Formalism

Suppose O is a generic operator acting on $\mathcal{H}$. One can verify that $O \otimes \mathbb{1}|I\rangle = \mathbb{1} \otimes \tilde{O}^T|I\rangle$, where T denotes the transpose and $\tilde{}$ is used to distinguish the mirrored space from the original space. As clarified in the main text, we always adopt an occupation-number basis without any complex-valued prefactors of norm 1, so that the matrix representations of $a^{(\dagger)}$ are always real-valued, i.e., $a_i^T = a_i^\dagger$. Then, by taking O to be a_i and $a_i^\dagger$, we obtain the conjugation relations

$$a_i \otimes \mathbb{1}|I\rangle = \mathbb{1} \otimes \tilde{a}_i^\dagger|I\rangle \ , \quad a_i^\dagger \otimes \mathbb{1}|I\rangle = \mathbb{1} \otimes \tilde{a}_i|I\rangle \ . \tag{B.24}$$

Now Eq. (B.22) can be rewritten as

$$\begin{aligned}(a_i \rho a_j^\dagger) \otimes \mathbb{1}|I\rangle &= (a_i \otimes \mathbb{1})(\rho \otimes \mathbb{1})(a_j^\dagger \otimes \mathbb{1})|I\rangle \\ &= (a_i \otimes \mathbb{1})(\rho \otimes \mathbb{1})(\mathbb{1} \otimes \tilde{a}_j)|I\rangle \\ &= (a_i \otimes \mathbb{1})(\mathbb{1} \otimes \tilde{a}_j)(\rho \otimes \mathbb{1})|I\rangle \\ &= a_i \otimes \tilde{a}_j|\rho\rangle \ ,\end{aligned} \tag{B.25}$$

so that the same superoperator can be expressed as the operator $a_i \otimes \tilde{a}_j$ acting on the state $|\rho\rangle \in \mathcal{H}_{\text{tot}}$. This representation is more convenient because the prefactor from $\tilde{a}_j$ acting on a state $|m, n\rangle$ depends only on $\sum_{i=1}^{j-1} n_i$, rather than on $\sum_{i=1}^{N}(m_i - n_i)$. In our implementation, we always use this formalism.

B.3. Explicit Form of the Liouvillian

For the single-impurity Anderson model (SIAM) studied in Sec. 9.3.1.2, by applying the conclusions from Sec. B.2.2, the explicit expression for the Liouvillian $\hat{\mathcal{L}}$ is

$$\begin{aligned}
\hat{\mathcal{L}} = & -i(H_{\text{aux}} \otimes \mathbb{1} - \mathbb{1} \otimes \tilde{H}_{\text{aux}}^T) \\
& + \sum_{k_1\sigma} (\boldsymbol{\gamma}_1)_{k_1\sigma} (2c_{k_1\sigma} \otimes \tilde{c}_{k_1\sigma} - c_{k_1\sigma}^\dagger c_{k_1\sigma} \otimes \mathbb{1} - \mathbb{1} \otimes \tilde{c}_{k_1\sigma}^\dagger \tilde{c}_{k_1\sigma}) \\
& + \sum_{k_2\sigma} (\boldsymbol{\gamma}_2)_{k_2\sigma} (2c_{k_2\sigma}^\dagger \otimes \tilde{c}_{k_2\sigma}^\dagger - c_{k_2\sigma} c_{k_2\sigma}^\dagger \otimes \mathbb{1} - \mathbb{1} \otimes \tilde{c}_{k_2\sigma} \tilde{c}_{k_2\sigma}^\dagger) \\
& + \sum_{k_1\sigma} (\boldsymbol{M}_1)_{k_1\sigma} (2d_\sigma \otimes \tilde{c}_{k_1\sigma} + 2c_{k_1\sigma} \otimes \tilde{d}_\sigma - c_{k_1\sigma}^\dagger d_\sigma \otimes \mathbb{1} - d_\sigma^\dagger c_{k_1\sigma} \otimes \mathbb{1} - \mathbb{1} \otimes \tilde{d}_\sigma^\dagger \tilde{c}_{k_1\sigma} - \mathbb{1} \otimes \tilde{c}_{k_1\sigma}^\dagger \tilde{d}_\sigma) \\
& + \sum_{k_2\sigma} (\boldsymbol{M}_2)_{k_2\sigma} (2d_\sigma^\dagger \otimes \tilde{c}_{k_2\sigma}^\dagger + 2c_{k_2\sigma}^\dagger \otimes \tilde{d}_\sigma^\dagger - c_{k_2\sigma} d_\sigma^\dagger \otimes \mathbb{1} - d_\sigma c_{k_2\sigma}^\dagger \otimes \mathbb{1} - \mathbb{1} \otimes \tilde{d}_\sigma \tilde{c}_{k_2\sigma}^\dagger - \mathbb{1} \otimes \tilde{c}_{k_2\sigma} \tilde{d}_\sigma^\dagger)
\end{aligned} , \tag{B.26}$$

where the hats $\hat{}$ on the operators on the right-hand side have been omitted to avoid confusion with the tildes $\tilde{}$.

For large systems, $\hat{\mathcal{L}}$ is usually not constructed explicitly. Instead, it is implemented as a black-box operation that takes a vector $v \in \mathcal{H}_{\text{tot}}$ as input and outputs another vector $\hat{\mathcal{L}}v \in \mathcal{H}_{\text{tot}}$. Since $\hat{\mathcal{L}}$ is a non-Hermitian operator, one also needs a black-box operation for $\hat{\mathcal{L}}^\dagger v \in \mathcal{H}_{\text{tot}}$ to perform algorithms such as the non-Hermitian Lanczos method, which will be introduced in Sec. B.4. For completeness, we therefore provide the explicit expression for $\hat{\mathcal{L}}^\dagger$:

$$\begin{aligned}
\hat{\mathcal{L}}^\dagger = & +i(H_{\text{aux}} \otimes \mathbb{1} - \mathbb{1} \otimes \tilde{H}_{\text{aux}}^T) \\
& + \sum_{k_1\sigma} (\boldsymbol{\gamma}_1)_{k_1\sigma} (2c_{k_1\sigma}^\dagger \otimes \tilde{c}_{k_1\sigma}^\dagger - c_{k_1\sigma}^\dagger c_{k_1\sigma} \otimes \mathbb{1} - \mathbb{1} \otimes \tilde{c}_{k_1\sigma}^\dagger \tilde{c}_{k_1\sigma}) \\
& + \sum_{k_2\sigma} (\boldsymbol{\gamma}_2)_{k_2\sigma} (2c_{k_2\sigma} \otimes \tilde{c}_{k_2\sigma} - c_{k_2\sigma} c_{k_2\sigma}^\dagger \otimes \mathbb{1} - \mathbb{1} \otimes \tilde{c}_{k_2\sigma} \tilde{c}_{k_2\sigma}^\dagger) \\
& + \sum_{k_1\sigma} (\boldsymbol{M}_1)_{k_1\sigma} (2d_\sigma^\dagger \otimes \tilde{c}_{k_1\sigma}^\dagger + 2c_{k_1\sigma}^\dagger \otimes \tilde{d}_\sigma^\dagger - d_\sigma^\dagger c_{k_1\sigma} \otimes \mathbb{1} - c_{k_1\sigma}^\dagger d_\sigma \otimes \mathbb{1} - \mathbb{1} \otimes \tilde{c}_{k_1\sigma}^\dagger \tilde{d}_\sigma - \mathbb{1} \otimes \tilde{d}_\sigma^\dagger \tilde{c}_{k_1\sigma}) \\
& + \sum_{k_2\sigma} (\boldsymbol{M}_2)_{k_2\sigma} (2d_\sigma \otimes \tilde{c}_{k_2\sigma} + 2c_{k_2\sigma} \otimes \tilde{d}_\sigma - d_\sigma c_{k_2\sigma}^\dagger \otimes \mathbb{1} - c_{k_2\sigma} d_\sigma^\dagger \otimes \mathbb{1} - \mathbb{1} \otimes \tilde{c}_{k_2\sigma} \tilde{d}_\sigma^\dagger - \mathbb{1} \otimes \tilde{d}_\sigma \tilde{c}_{k_2\sigma}^\dagger)
\end{aligned} , \tag{B.27}$$

where we have used the fact that $\boldsymbol{\gamma}_1$, $\boldsymbol{\gamma}_2$, $\boldsymbol{M}_1$, and $\boldsymbol{M}_2$ are all real-valued.

These parameters can be extracted from the MPM results $\{A_{k_1\sigma}, \xi_{k_1\sigma}\}$ and $\{A_{k_2\sigma}, \xi_{k_2\sigma}\}$

via

$$
\begin{aligned}
(\boldsymbol{V}_1)_{k_1\sigma} - i(\boldsymbol{M}_1)_{k_1\sigma} &= \sqrt{-2\pi i A^{(\mathrm{dn})}_{k_1\sigma}} \ , \\
(\boldsymbol{V}_2)_{k_2\sigma} + i(\boldsymbol{M}_2)_{k_2\sigma} &= \sqrt{+2\pi i A^{(\mathrm{up})}_{k_2\sigma}} \ , \\
(\boldsymbol{Z}_1)_{k_1\sigma} = (\boldsymbol{\gamma}_1)_{k_1\sigma} + i(\boldsymbol{\varepsilon}_1)_{k_1\sigma} &= +i\xi^{(\mathrm{dn})}_{k_1\sigma} \ , \\
(\boldsymbol{Z}_2)_{k_2\sigma} = (\boldsymbol{\gamma}_2)_{k_2\sigma} - i(\boldsymbol{\varepsilon}_2)_{k_2\sigma} &= -i\xi^{(\mathrm{up})}_{k_2\sigma} \ .
\end{aligned}
\tag{B.28}
$$

B.4. Non-Hermitian Lanczos Algorithm

B.4.1. Theoretical Background

The non-Hermitian Lanczos algorithm is a generalization of the Hermitian Lanczos algorithm to non-Hermitian matrices [161]. It builds a pair of biorthogonal bases for the two Krylov subspaces

$$
\mathcal{K}_m(\hat{\mathcal{L}}, v_1) = \mathrm{span}\{v_1, \hat{\mathcal{L}}v_1, \cdots, \hat{\mathcal{L}}^{m-1}v_1\} \tag{B.29}
$$

and

$$
\mathcal{K}_m(\hat{\mathcal{L}}^\dagger, w_1) = \mathrm{span}\{w_1, \hat{\mathcal{L}}^\dagger w_1, \cdots, (\hat{\mathcal{L}}^\dagger)^{m-1}w_1\} \ , \tag{B.30}
$$

where $\hat{\mathcal{L}}$ is a non-Hermitian square matrix of size $M \times M$, following the notation introduced in the main text; $m \leq M$ is an integer; and v_1 and w_1 are column vectors of size M.

The algorithm is performed as described in Alg. 6. It produces a pair of biorthogonal bases $\{v_i\}$ and $\{w_i\}$ satisfying the orthonormality relation

$$
w_i^\dagger v_j = \delta_{ij} \quad \text{for} \quad 1 \leq i, j \leq m \ , \tag{B.31}
$$

where $\dagger$ denotes the Hermitian conjugate. It also produces a tridiagonal matrix

$$
T_m = \begin{pmatrix}
\alpha_1 & \beta_2 & & & \\
\delta_2 & \alpha_2 & \beta_3 & & \\
 & \delta_3 & \alpha_3 & \ddots & \\
 & & \ddots & \ddots & \beta_m \\
 & & & \delta_m & \alpha_m
\end{pmatrix} \ , \tag{B.32}
$$

Algorithm 6 Non-Hermitian Lanczos Algorithm

Input: $\tilde{v}_1$, $\tilde{w}_1$, $\hat{\mathcal{L}}$, $m_{\max}$, and ε.
Output: $\alpha \in \mathbb{C}^m$, $\beta \in \mathbb{C}^{m-1}$, $\delta \in \mathbb{C}^{m-1}$, and $c \in \mathbb{C}$.
Compute $c = \tilde{w}_1^\dagger \tilde{v}_1$.
Initialize $w_1 = c_w \tilde{w}_1$, $v_1 = c_v \tilde{v}_1$, with $c_w = \frac{\sqrt{\|\tilde{v}_1\|}}{\sqrt{\|\tilde{w}_1\|}} \frac{1}{\sqrt{|c|}}$ and $c_v = \frac{1}{c \times c_w}$.
Set $\beta_1 = 0$, $\delta_1 = 0$, $w_0 = v_0 = 0$.
for $j = 1, 2, \cdots, m_{\max}$ **do**
    $\tilde{v}_{j+1} = \hat{\mathcal{L}} v_j - \beta_j v_{j-1}$.
    $\alpha_j = w_j^\dagger \tilde{v}_{j+1}$.
    $\tilde{v}_{j+1} = \tilde{v}_{j+1} - \alpha_j v_j$.
    $\tilde{w}_{j+1} = \hat{\mathcal{L}}^\dagger w_j - \alpha_j^* w_j - \delta_j w_{j-1}$.
    $\delta_{j+1} = \sqrt{|\tilde{w}_{j+1}^\dagger \tilde{v}_{j+1}|}$.
    if $\delta_{j+1} < \varepsilon$ **then**
        Stop the iteration.
    $\beta_{j+1} = \tilde{w}_{j+1}^\dagger \tilde{v}_{j+1} / \delta_{j+1}$.
    $w_{j+1} = \tilde{w}_{j+1} / \beta_{j+1}^*$.
    $v_{j+1} = \tilde{v}_{j+1} / \delta_{j+1}$.

such that the non-Hermitian matrix $\hat{\mathcal{L}}$ is tridiagonalized as

$$W_m^\dagger \, \hat{\mathcal{L}} \, V_m = T_m \ , \tag{B.33}$$

where $V_m = (v_1 \ v_2 \ \cdots \ v_m)$ and $W_m = (w_1 \ w_2 \ \cdots \ w_m)$ are $M \times m$ matrices. In the algorithm, we distinguish unnormalized vectors from normalized ones by placing a tilde on the unnormalized vectors. The initial vectors $\tilde{v}_1$ and $\tilde{w}_1$ are normalized so that v_1 and w_1 have the same norm. The coefficients δ_j and β_j are chosen so that v_j and w_j for $j > 1$ have the same norm as well.

Suppose T_m has an eigenvalue λ_i with right eigenvector x_i and left eigenvector y_i, i.e.,

$$T_m x_i = \lambda_i x_i \ , \tag{B.34}$$

$$T_m^\dagger y_i = \lambda_i^* y_i \ . \tag{B.35}$$

When T_m has the same size as $\hat{\mathcal{L}}$, i.e., $m = M$, λ_i is also an eigenvalue of $\hat{\mathcal{L}}$, with right eigenvector $V_m x_i$ and left eigenvector $W_m y_i$. When $m \ll M$, λ_i, $V_m x_i$, and $W_m y_i$ can still provide good estimates of the extreme eigenvalues of $\hat{\mathcal{L}}$ and their corresponding

eigenvectors [161]. For these m approximate eigenvectors to be biorthonormal, i.e., $(W_m y_i)^\dagger (V_m x_j) = \delta_{ij}$, one only needs to construct biorthonormal eigenvectors of T_m, i.e., $y_i^\dagger x_j = \delta_{ij}$.

B.4.2. Green's Function Evaluation

Because of the spin symmetry in the SIAM considered in Sec. 9.3.1.3, we only need to focus on $\sigma = \uparrow$. To calculate the first term of Eq. (9.37), we set $\tilde{w}_1 = \hat{d}_\uparrow^\dagger |I_0\rangle$, $\tilde{v}_1 = \hat{d}_\uparrow^\dagger |\rho_{\text{aux}}(\infty)\rangle$ and perform Alg. 6 on the block $(+1,+1)$ to obtain T_m as well as the normalization constant c. Suppose it has eigenvalue λ_i $(1 \le i \le m)$ with right eigenvector x_i and left eigenvector y_i, as described in the previous section. Then R and L can be approximated by $V_m(x_1 \cdots x_m)$ and $W_m(y_1 \cdots y_m)$, respectively. For clarity, we temporarily omit the superscript $(+1,+1)$, and the first term can be written as

$$
\begin{aligned}
& v_{1L}^\dagger \begin{pmatrix} \frac{1}{\omega - i\lambda_1} & & & \\ & \frac{1}{\omega - i\lambda_2} & & \\ & & \ddots & \\ & & & \frac{1}{\omega - i\lambda_m} \end{pmatrix} v_{1R} \\
&= c\, w_1^\dagger V_m \begin{pmatrix} x_1 & x_2 & \cdots & x_m \end{pmatrix} \begin{pmatrix} \frac{1}{\omega - i\lambda_1} & & & \\ & \frac{1}{\omega - i\lambda_2} & & \\ & & \ddots & \\ & & & \frac{1}{\omega - i\lambda_m} \end{pmatrix} \begin{pmatrix} y_1^\dagger \\ y_2^\dagger \\ \vdots \\ y_m^\dagger \end{pmatrix} W_m^\dagger v_1 \\
&= c \begin{pmatrix} 1 & 0 & \cdots & 0 \end{pmatrix} \begin{pmatrix} x_1 & x_2 & \cdots & x_m \end{pmatrix} \begin{pmatrix} \frac{1}{\omega - i\lambda_1} & & & \\ & \frac{1}{\omega - i\lambda_2} & & \\ & & \ddots & \\ & & & \frac{1}{\omega - i\lambda_m} \end{pmatrix} \begin{pmatrix} y_1^\dagger \\ y_2^\dagger \\ \vdots \\ y_m^\dagger \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \\
&= c \begin{pmatrix} (x_1)_1 & (x_2)_1 & \cdots & (x_m)_1 \end{pmatrix} \begin{pmatrix} \frac{1}{\omega - i\lambda_1} & & & \\ & \frac{1}{\omega - i\lambda_2} & & \\ & & \ddots & \\ & & & \frac{1}{\omega - i\lambda_m} \end{pmatrix} \begin{pmatrix} (y_1)_1^* \\ (y_2)_1^* \\ \vdots \\ (y_m)_1^* \end{pmatrix} .
\end{aligned}
\tag{B.36}
$$

Similarly, one can set $\tilde{w}_1 = \hat{d}_\uparrow |I_0\rangle$, $\tilde{v}_1 = \hat{d}_\uparrow |\rho_{\text{aux}}(\infty)\rangle$ and perform Alg. 6 on the block

$(-1,-1)$ to obtain the second term. As a result, Eq. (9.37) can be simplified to

$$G^{\mathrm{Ret}}_{\uparrow\uparrow}(\omega) = c^{(+1,+1)} \sum_{j=1}^{m} \frac{(x_j^{(+1,+1)})_1 (y_j^{(+1,+1)})_1^*}{\omega - i\lambda_j^{(+1,+1)}} + c^{(-1,-1)*} \sum_{j=1}^{m} \frac{(x_j^{(-1,-1)})_1^* (y_j^{(-1,-1)})_1}{\omega - i(\lambda_j^{(-1,-1)})^*} \ , \quad \text{(B.37)}$$

where the superscripts $(+1,+1)$ and $(-1,-1)$ are used to distinguish the two blocks.

In practice, when performing the non-Hermitian Lanczos algorithm, we only input the actions of the operators $\hat{\mathcal{L}}$ and $\hat{\mathcal{L}}^\dagger$, which can be implemented by bit-wise operations, rather than constructing the full matrices explicitly. Moreover, we only need T_m and c as outputs, without storing the large rectangular matrices V_m and W_m. After that, we simply diagonalize the small matrix T_m and substitute the results into Eq. (B.37) to obtain the final Green's function.

Bibliography

[1] J. W. Negele. *Quantum many-particle systems*. CRC Press, 2018.

[2] A. L. Fetter and J. D. Walecka. *Quantum theory of many-particle systems*. Courier Corporation, 2012.

[3] G. Mahan. *Many-Particle Physics*. Physics of Solids and Liquids. Springer US, 2013.

[4] A. Szabo and N. S. Ostlund. *Modern quantum chemistry: introduction to advanced electronic structure theory*. Courier Corporation, 2012.

[5] P. Coleman. *Introduction to Many-Body Physics*. Cambridge University Press, 2015.

[6] G. Stefanucci and R. van Leeuwen. *Nonequilibrium Many-Body Theory of Quantum Systems: A Modern Introduction*. Cambridge University Press, 2013.

[7] L. Hedin. "New Method for Calculating the One-Particle Green's Function with Application to the Electron-Gas Problem". In: *Physical Review* 139 (3A Aug. 1965), A796–A823. DOI: 10.1103/PhysRev.139.A796.

[8] N. E. Dahlen and R. van Leeuwen. "Self-consistent solution of the Dyson equation for atoms and molecules within a conserving approximation". In: *The Journal of Chemical Physics* 122.16 (2005), p. 164102. DOI: 10.1063/1.1884965.

[9] J. J. Phillips and D. Zgid. "Communication: The description of strong correlation within self-consistent Green's function second-order perturbation theory". In: *The Journal of Chemical Physics* 140.24 (2014), p. 241101. DOI: 10.1063/1.4884951.

[10] R. Blankenbecler, D. J. Scalapino, and R. L. Sugar. "Monte Carlo calculations of coupled boson-fermion systems. I". In: *Physical Review D* 24 (8 Oct. 1981), pp. 2278–2286. DOI: 10.1103/PhysRevD.24.2278.

[11] L. Pollet. "Recent developments in quantum Monte Carlo simulations with applications for cold gases". In: *Reports on Progress in Physics* 75.9 (Aug. 2012), p. 094501. DOI: 10.1088/0034-4885/75/9/094501.

[12] E. Gull, A. J. Millis, A. I. Lichtenstein, A. N. Rubtsov, M. Troyer, and P. Werner. "Continuous-time Monte Carlo methods for quantum impurity models". In: *Reviews of Modern Physics* 83 (2 May 2011), pp. 349–404. DOI: 10.1103/RevModPhys.83.349.

[13] M. Jarrell and J. E. Gubernatis. "Bayesian inference and the analytic continuation of imaginary-time quantum Monte Carlo data". In: *Physics Reports* 269.3 (1996), pp. 133–195. DOI: 10.1016/0370-1573(95)00074-7.

[14] G. A. Baker Jr. and P. Graves-Morris. *Padé Approximants*. 2nd. Vol. 59. Encyclopedia of Mathematics and its Applications. Cambridge: Cambridge University Press, 1996, pp. xiv+746.

[15] H. J. Vidberg and J. W. Serene. "Solving the Eliashberg equations by means of N-point Padé approximants". In: *Journal of Low Temperature Physics* 29.3 (1977), pp. 179–192. DOI: 10.1007/BF00655090.

[16] J. Fei, C.-N. Yeh, and E. Gull. "Nevanlinna analytical continuation". In: *Physical Review Letters* 126.5 (2021), p. 056402. DOI: 10.1103/PhysRevLett.126.056402.

[17] J. Fei, C.-N. Yeh, D. Zgid, and E. Gull. "Analytical continuation of matrix-valued functions: Carathéodory formalism". In: *Physical Review B* 104.16 (2021), p. 165111. DOI: 10.1103/PhysRevB.104.165111.

[18] C. E. Creffield, E. G. Klepfish, E. R. Pike, and S. Sarkar. "Spectral Weight Function for the Half-Filled Hubbard Model: A Singular Value Decomposition Approach". In: *Physical Review Letters* 75 (3 July 1995), pp. 517–520. DOI: 10.1103/PhysRevLett.75.517.

[19] K. S. D. Beach, R. J. Gooding, and F. Marsiglio. "Reliable Padé analytical continuation method based on a high-accuracy symbolic computation algorithm". In: *Physical Review B* 61 (8 Feb. 2000), pp. 5147–5157. DOI: 10.1103/PhysRevB.61.5147.

[20] O. Gunnarsson, M. W. Haverkort, and G. Sangiovanni. "Analytical continuation of imaginary axis data for optical conductivity". In: *Physical Review B* 82 (16 Oct. 2010), p. 165125. DOI: 10.1103/PhysRevB.82.165125.

[21] D. Bergeron and A.-M. S. Tremblay. "Algorithms for optimized maximum entropy and diagnostic tools for analytic continuation". In: *Physical Review E* 94 (2 Aug. 2016), p. 023303. DOI: 10.1103/PhysRevE.94.023303.

[22] R. Levy, J. LeBlanc, and E. Gull. "Implementation of the maximum entropy method for analytic continuation". In: *Computer Physics Communications* 215 (2017), pp. 149–155. DOI: 10.1016/j.cpc.2017.01.018.

[23] G. J. Kraberger, R. Triebl, M. Zingl, and M. Aichhorn. "Maximum entropy formalism for the analytic continuation of matrix-valued Green's functions". In: *Physical Review B* 96 (15 Oct. 2017), p. 155128. DOI: 10.1103/PhysRevB.96.155128.

[24] H. Shao and A. W. Sandvik. "Progress on stochastic analytic continuation of quantum Monte Carlo data". In: *Physics Reports* 1003 (2023), pp. 1–88. DOI: 10.1016/j.physrep.2022.11.002.

[25] A. W. Sandvik. "Stochastic method for analytic continuation of quantum Monte Carlo data". In: *Physical Review B* 57 (17 May 1998), pp. 10287–10290. DOI: 10.1103/PhysRevB.57.10287.

[26] A. S. Mishchenko, N. V. Prokof'ev, A. Sakamoto, and B. V. Svistunov. "Diagrammatic quantum Monte Carlo study of the Fröhlich polaron". In: *Physical Review B* 62 (10 Sept. 2000), pp. 6317–6336. DOI: 10.1103/PhysRevB.62.6317.

[27] K. Vafayi and O. Gunnarsson. "Analytical continuation of spectral data from imaginary time axis to real frequency axis using statistical sampling". In: *Physical Review B* 76 (3 July 2007), p. 035115. DOI: 10.1103/PhysRevB.76.035115.

[28] S. Fuchs, M. Jarrell, and T. Pruschke. "Application of Bayesian inference to stochastic analytic continuation". In: *Journal of Physics: Conference Series* 200.1 (Jan. 2010), p. 012041. DOI: 10.1088/1742-6596/200/1/012041.

[29] O. Goulko, A. S. Mishchenko, L. Pollet, N. Prokof'ev, and B. Svistunov. "Numerical analytic continuation: Answers to well-posed questions". In: *Physical Review B* 95 (1 Jan. 2017), p. 014102. DOI: 10.1103/PhysRevB.95.014102.

[30] J. Otsuki, M. Ohzeki, H. Shinaoka, and K. Yoshimi. "Sparse modeling approach to analytical continuation of imaginary-time quantum Monte Carlo data". In: *Physical Review E* 95 (6 June 2017), 061302(R). DOI: 10.1103/PhysRevE.95.061302.

[31] I. Krivenko and M. Harland. "TRIQS/SOM: Implementation of the stochastic optimization method for analytic continuation". In: *Computer Physics Communications* 239 (2019), pp. 166–183. DOI: 10.1016/j.cpc.2019.01.021.

[32] E. Vitali, M. Rossi, L. Reatto, and D. E. Galli. "Ab initio low-energy dynamics of superfluid and solid ^{4}He". In: *Physical Review B* 82 (17 Nov. 2010), p. 174510. DOI: 10.1103/PhysRevB.82.174510.

[33] D. Huang and Y. F. Yang. "Learned optimizers for analytic continuation". In: *Physical Review B* 105 (7 Feb. 2022), p. 075112. DOI: 10.1103/PhysRevB.105.075112.

[34] J. Yao, C. Wang, Z. Yao, and H. Zhai. "Noise enhanced neural networks for analytic continuation". In: *Machine Learning: Science and Technology* 3.2 (May 2022), p. 025010. DOI: 10.1088/2632-2153/ac6f44.

[35] Z. Huang, E. Gull, and L. Lin. "Robust analytic continuation of Green's functions via projection, pole estimation, and semidefinite relaxation". In: *Physical Review B* 107 (7 Feb. 2023), p. 075151. DOI: 10.1103/PhysRevB.107.075151.

[36] L. Ying. "Pole recovery from noisy data on imaginary axis". In: *Journal of Scientific Computing* 92.3 (2022), p. 107. DOI: 10.1007/s10915-022-01963-z.

[37] L. Ying. "Analytic continuation from limited noisy Matsubara data". In: *Journal of Computational Physics* 469 (2022), p. 111549. DOI: 10.1016/j.jcp.2022.111549.

[38] L. Zhang and E. Gull. "Minimal pole representation and controlled analytic continuation of Matsubara response functions". In: *Physical Review B* 110.3 (2024), p. 035154. DOI: 10.1103/PhysRevB.110.035154.

[39] L. Zhang, Y. Yu, and E. Gull. "Minimal pole representation and analytic continuation of matrix-valued correlation functions". In: *Physical Review B* 110 (23 Dec. 2024), p. 235131. DOI: 10.1103/PhysRevB.110.235131.

[40] L. Zhang, A. Erpenbeck, Y. Yu, and E. Gull. "Minimal pole representation for spectral functions". In: *The Journal of Chemical Physics* 162.21 (June 2025), p. 214111. DOI: 10.1063/5.0273763.

[41] E. Koch. "Exact Diagonalization and Lanczos Method". In: *Many-Body Methods for Real Materials*. Ed. by E. Pavarini, E. Koch, and S. Zhang. Vol. 9. Modeling and Simulation. Jülich: Verlag des Forschungszentrum Jülich, 2019. Chap. 7.

[42] P.-O. Löwdin. "On the non-orthogonality problem connected with the use of atomic wave functions in the theory of molecules and crystals". In: *The Journal of Chemical Physics* 18.3 (1950), pp. 365–375. DOI: 10.1063/1.1747632.

[43] P.-O. Löwdin. "Quantum theory of many-particle systems. I. Physical interpretations by means of density matrices, natural spin-orbitals, and convergence problems in the method of configurational interaction". In: *Physical Review* 97.6 (1955), p. 1474. DOI: 10.1103/PhysRev.97.1474.

[44] R. De L. Kronig. "On the theory of dispersion of x-rays". In: *Journal of the Optical Society of America* 12.6 (1926), pp. 547–557. DOI: 10.1364/JOSA.12.000547.

[45] H. A. Kramers. "La diffusion de la lumiere par les atomes". In: *Atti Cong. Intern. Fisica (Transactions of Volta Centenary Congress) Como*. Vol. 2. 1927, pp. 545–557.

[46] J. S. Toll. "Causality and the dispersion relation: logical foundations". In: *Physical Review* 104.6 (1956), p. 1760. DOI: 10.1103/PhysRev.104.1760.

[47] G. R. de Prony. "Essai experimental et analytique: sur les lois de la dilatabilite des fluides elastique et sur celles de la force expansive de la vapeur de l'eau et de la vapeur de l'alkool, a differentes temperatures". In: *Journal Polytechnique ou Bulletin du Travail fait a l'Ecole Centrale des Travaux Publics* (1795).

[48] A. Moitra. "Super-Resolution, Extremal Functions and the Condition Number of Vandermonde Matrices". In: *Proceedings of the 47th Annual ACM Symposium on Theory of Computing*. STOC '15 (2015), pp. 821–830. DOI: 10.1145/2746539.2746561.

[49] G. Beylkin and L. Monzón. "On approximation of functions by exponential sums". In: *Applied and Computational Harmonic Analysis* 19.1 (2005), pp. 17–48. DOI: 10.1016/j.acha.2005.01.003.

[50] G. Beylkin and L. Monzón. "Approximation by exponential sums revisited". In: *Applied and Computational Harmonic Analysis* 28.2 (2010), pp. 131–149. DOI: 10.1016/j.acha.2009.08.011.

[51] Y. Hua and T. K. Sarkar. "Matrix pencil method for estimating parameters of exponentially damped/undamped sinusoids in noise". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 38.5 (1990), pp. 814–824. DOI: 10.1109/29.56027.

[52] T. K. Sarkar and O. Pereira. "Using the matrix pencil method to estimate the parameters of a sum of complex exponentials". In: *IEEE Antennas and Propagation Magazine* 37.1 (1995), pp. 48–55. DOI: 10.1109/74.370583.

[53] D. Potts and M. Tasche. "Parameter estimation for nonincreasing exponential sums by Prony-like methods". In: *Linear Algebra and its Applications* 439.4 (2013), pp. 1024–1039. DOI: 10.1016/j.laa.2012.10.036.

[54] R. Roy and T. Kailath. "ESPRIT-estimation of signal parameters via rotational invariance techniques". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 37.7 (1989), pp. 984–995. DOI: 10.1109/29.32276.

[55] R. A. Horn and C. R. Johnson. *Matrix analysis*. Cambridge University Press, 2012.

[56] H. Takahashi, S. Rudge, C. Kaspar, M. Thoss, and R. Borrelli. "High accuracy exponential decomposition of bath correlation functions for arbitrary and structured spectral densities: Emerging methodologies and new approaches". In: *The Journal of Chemical Physics* 160.20 (May 2024), p. 204105. DOI: 10.1063/5.0209348.

[57] S. Dyatlov and M. Zworski. *Mathematical theory of scattering resonances*. Vol. 200. American Mathematical Soc., 2019.

[58] M. Asakawa, Y. Nakahara, and T. Hatsuda. "Maximum entropy analysis of the spectral functions in lattice QCD". In: *Progress in Particle and Nuclear Physics* 46.2 (2001), pp. 459–508. DOI: 10.1016/S0146-6410(01)00150-8.

[59] R.-A. Tripolt, P. Gubler, M. Ulybyshev, and L. von Smekal. "Numerical analytic continuation of Euclidean data". In: *Computer Physics Communications* 237 (2019), pp. 129–142. DOI: 10.1016/j.cpc.2018.11.012.

[60] A. Rothkopf. "Bryan's Maximum Entropy Method—Diagnosis of a Flawed Argument and Its Remedy". In: *Data* 5.3 (2020). DOI: 10.3390/data5030085.

[61] A. Filinov. "Correlation effects and collective excitations in bosonic bilayers: Role of quantum statistics, superfluidity, and the dimerization transition". In: *Physical Review A* 94 (1 July 2016), p. 013603. DOI: 10.1103/PhysRevA.94.013603.

[62] M. Boninsegni and D. M. Ceperley. "Density fluctuations in liquid4He. Path integrals and maximum entropy". In: *Journal of Low Temperature Physics* 104.5 (1996), pp. 339–357. DOI: 10.1007/BF00751861.

[63] S. Saccani, S. Moroni, and M. Boninsegni. "Excitation Spectrum of a Supersolid". In: *Physical Review Letters* 108 (17 Apr. 2012), p. 175301. DOI: 10.1103/PhysRevLett.108.175301.

[64] T. Dornheim, S. Groth, J. Vorberger, and M. Bonitz. "Ab initio Path Integral Monte Carlo Results for the Dynamic Structure Factor of Correlated Electrons: From the Electron Liquid to Warm Dense Matter". In: *Physical Review Letters* 121 (25 Dec. 2018), p. 255001. DOI: 10.1103/PhysRevLett.121.255001.

[65] A. Östlin, L. Chioncel, and L. Vitos. "One-particle spectral function and analytic continuation for many-body implementation in the exact muffin-tin orbitals method". In: *Physical Review B* 86 (23 Dec. 2012), p. 235107. DOI: 10.1103/PhysRevB.86.235107.

[66] Z. Ž. Osolin and R. Žitko. "Padé approximant approach for obtaining finite-temperature spectral functions of quantum impurity models using the numerical renormalization group technique". In: *Physical Review B* 87 (24 June 2013), p. 245135. DOI: 10.1103/PhysRevB.87.245135.

[67] J. Schött, I. L. M. Locht, E. Lundin, O. Grånäs, O. Eriksson, and I. Di Marco. "Analytic continuation by averaging Padé approximants". In: *Physical Review B* 93 (7 Feb. 2016), p. 075104. DOI: 10.1103/PhysRevB.93.075104.

[68] X.-J. Han, H.-J. Liao, H.-D. Xie, R.-Z. Huang, Z.-Y. Meng, and T. Xiang. "Analytic Continuation with Padé Decomposition". In: *Chinese Physics Letters* 34.7 (July 2017), p. 077102. DOI: 10.1088/0256-307x/34/7/077102.

[69] R. K. Bryan. "Maximum entropy analysis of oversampled data problems". In: *European Biophysics Journal* 18.3 (1990), pp. 165–174. DOI: 10.1007/BF02427376.

[70] K. S. D. Beach. *Identifying the maximum entropy method as a special limit of stochastic analytic continuation*. 2004.

[71] A. Gaenko, A. Antipov, G. Carcassi, T. Chen, X. Chen, Q. Dong, L. Gamper, J. Gukelberger, R. Igarashi, S. Iskakov, M. Könz, J. LeBlanc, R. Levy, P. Ma, J. Paki, H. Shinaoka, S. Todo, M. Troyer, and E. Gull. "Updated core libraries of the ALPS project". In: *Computer Physics Communications* 213 (2017), pp. 235–251. DOI: 10.1016/j.cpc.2016.12.009.

[72] M. Rumetshofer, D. Bauernfeind, and W. von der Linden. "Bayesian parametric analytic continuation of Green's functions". In: *Physical Review B* 100 (7 Aug. 2019), p. 075137. DOI: 10.1103/PhysRevB.100.075137.

[73] J.-H. Sim and M. J. Han. "Maximum quantum entropy method". In: *Physical Review B* 98 (20 Nov. 2018), p. 205102. DOI: 10.1103/PhysRevB.98.205102.

[74] K. Yoshimi, J. Otsuki, Y. Motoyama, M. Ohzeki, and H. Shinaoka. "SpM: Sparse modeling tool for analytic continuation of imaginary-time Green's function". In: *Computer Physics Communications* 244 (2019), pp. 319–323. DOI: 10.1016/j.cpc.2019.07.001.

[75] N. Joukowsky. "Über die konturen der Tragflächen der Drachenflieger". In: *Zeitschrift für Flugtechnik und Motorluftschiffahrt* 1.22 (1910), pp. 281–285.

[76] L. Boehnke, H. Hafermann, M. Ferrero, F. Lechermann, and O. Parcollet. "Orthogonal polynomial representation of imaginary-time Green's functions". In: *Physical Review B* 84 (7 Aug. 2011), p. 075145. DOI: 10.1103/PhysRevB.84.075145.

[77] A. A. Kananenka, A. R. Welden, T. N. Lan, E. Gull, and D. Zgid. "Efficient Temperature-Dependent Green's Function Methods for Realistic Systems: Using Cubic Spline Interpolation to Approximate Matsubara Green's Functions". In: *Journal of Chemical Theory and Computation* 12.5 (May 2016), pp. 2250–2259. DOI: 10.1021/acs.jctc.6b00178.

[78] E. Gull, S. Iskakov, I. Krivenko, A. A. Rusakov, and D. Zgid. "Chebyshev polynomial representation of imaginary-time response functions". In: *Physical Review B* 98 (7 Aug. 2018), p. 075127. DOI: 10.1103/PhysRevB.98.075127.

[79] H. Shinaoka, J. Otsuki, M. Ohzeki, and K. Yoshimi. "Compressing Green's function using intermediate representation between imaginary-time and real-frequency domains". In: *Physical Review B* 96 (3 July 2017), p. 035147. DOI: 10.1103/PhysRevB.96.035147.

[80] J. Li, M. Wallerberger, N. Chikano, C.-N. Yeh, E. Gull, and H. Shinaoka. "Sparse sampling approach to efficient ab initio calculations at finite temperature". In: *Physical Review B* 101 (3 Jan. 2020), p. 035144. DOI: 10.1103/PhysRevB.101.035144.

[81] H. Shinaoka, N. Chikano, E. Gull, J. Li, T. Nomoto, J. Otsuki, M. Wallerberger, T. Wang, and K. Yoshimi. "Efficient ab initio many-body calculations based on sparse modeling of Matsubara Green's function". In: *SciPost Physics Lecture Notes* (2022), p. 63. DOI: 10.21468/SciPostPhysLectNotes.63.

[82] J. Kaye, K. Chen, and O. Parcollet. "Discrete Lehmann representation of imaginary time Green's functions". In: *Physical Review B* 105 (23 June 2022), p. 235115. DOI: 10.1103/PhysRevB.105.235115.

[83] S. Iskakov, C.-N. Yeh, P. Pokhilko, Y. Yu, L. Zhang, G. Harsha, V. Abraham, M. Wen, M. Wang, J. Adamski, T. Chen, E. Gull, and D. Zgid. "Green/Weak-Coupling: Implementation of fully self-consistent finite-temperature many-body perturbation theory for molecules and solids". In: *Computer Physics Communications* 306 (2025), p. 109380. DOI: 10.1016/j.cpc.2024.109380.

[84] L. Zhang and E. Gull. *Green-Phys/PronyAC: v0.1.* Version 0.1. June 2024. DOI: 10.5281/zenodo.11520534.

[85] L. Zhang and E. Gull. *Green-Phys/PronyAC.* https://github.com/Green-Phys/pronyac. June 2024.

[86] Y. Yu, S. Iskakov, and E. Gull. "Heating and cooling in self-consistent many-body simulations". In: *Physical Review B* 108 (15 Oct. 2023), p. 155116. DOI: 10.1103/PhysRevB.108.155116.

[87] A. F. Kemper, C. Yang, and E. Gull. "Denoising and Extension of Response Functions in the Time Domain". In: *Physical Review Letters* 132 (16 Apr. 2024), p. 160403. DOI: 10.1103/PhysRevLett.132.160403.

[88] E. Gull and A. J. Millis. "Pairing glue in the two-dimensional Hubbard model". In: *Physical Review B* 90 (4 July 2014), 041110(R). DOI: 10.1103/PhysRevB.90.041110.

[89] A. Reymbaut, D. Bergeron, and A.-M. S. Tremblay. "Maximum entropy analytic continuation for spectral functions with nonpositive spectral weight". In: *Physical Review B* 92 (6 Aug. 2015), 060509(R). DOI: 10.1103/PhysRevB.92.060509.

[90] A. Reymbaut, A.-M. Gagnon, D. Bergeron, and A.-M. S. Tremblay. "Maximum entropy analytic continuation for frequency-dependent transport coefficients with nonpositive spectral weight". In: *Physical Review B* 95 (12 Mar. 2017), 121104(R). DOI: 10.1103/PhysRevB.95.121104.

[91] K. Nogaki and H. Shinaoka. "Bosonic Nevanlinna Analytic Continuation". In: *Journal of the Physical Society of Japan* 92.3 (2023), p. 035001. DOI: 10.7566/JPSJ.92.035001.

[92] C. Yue and P. Werner. "Maximum entropy analytic continuation of anomalous self-energies". In: *Physical Review B* 108 (22 Dec. 2023), p. L220503. DOI: 10.1103/PhysRevB.108.L220503.

[93] X. Wang, E. Gull, L. de'Medici, M. Capone, and A. J. Millis. "Antiferromagnetism and the gap of a Mott insulator: Results from analytic continuation of the self-energy". In: *Physical Review B* 80 (4 July 2009), p. 045101. DOI: 10.1103/PhysRevB.80.045101.

[94] N. Blümer. "Mott-Hubbard metal insulator transition and optical conductivity in high dimensions". PhD thesis. Jan. 2003.

[95] C.-N. Yeh, S. Iskakov, D. Zgid, and E. Gull. "Fully self-consistent finite-temperature *GW* in Gaussian Bloch orbitals for solids". In: *Physical Review B* 106 (23 Dec. 2022), p. 235104. DOI: 10.1103/PhysRevB.106.235104.

[96] A. Georges, G. Kotliar, W. Krauth, and M. J. Rozenberg. "Dynamical mean-field theory of strongly correlated fermion systems and the limit of infinite dimensions". In: *Reviews of Modern Physics* 68 (1 Jan. 1996), pp. 13–125. DOI: 10.1103/RevModPhys.68.13.

[97] G. Kotliar, S. Y. Savrasov, K. Haule, V. S. Oudovenko, O. Parcollet, and C. A. Marianetti. "Electronic structure calculations with dynamical mean-field theory". In: *Reviews of Modern Physics* 78 (3 Aug. 2006), pp. 865–951. DOI: 10.1103/RevModPhys.78.865.

[98] A. A. Kananenka, E. Gull, and D. Zgid. "Systematically improvable multiscale solver for correlated electron systems". In: *Physical Review B* 91 (12 Mar. 2015), p. 121111. DOI: 10.1103/PhysRevB.91.121111.

[99] D. Zgid and E. Gull. "Finite temperature quantum embedding theories for correlated systems". In: *New Journal of Physics* 19.2 (Feb. 2017), p. 023047. DOI: 10.1088/1367-2630/aa5d34.

[100] Y. Burnier and A. Rothkopf. "Bayesian Approach to Spectral Function Reconstruction for Euclidean Quantum Field Theories". In: *Physical Review Letters* 111 (18 Oct. 2013), p. 182003. DOI: 10.1103/PhysRevLett.111.182003.

[101] Y. Lu, M. Höppner, O. Gunnarsson, and M. W. Haverkort. "Efficient real-frequency solver for dynamical mean-field theory". In: *Physical Review B* 90 (8 Aug. 2014), p. 085102. DOI: 10.1103/PhysRevB.90.085102.

[102] L. Huang and S. Liang. "Stochastic pole expansion method for analytic continuation of the Green's function". In: *Physical Review B* 108 (23 Dec. 2023), p. 235143. DOI: 10.1103/PhysRevB.108.235143.

[103] L. Huang and S. Liang. "Reconstructing lattice QCD spectral functions with stochastic pole expansion and Nevanlinna analytic continuation". In: *Physical Review D* 109 (5 Mar. 2024), p. 054508. DOI: 10.1103/PhysRevD.109.054508.

[104] D. Gazizova, L. Zhang, E. Gull, and J. P. F. LeBlanc. "Feynman diagrammatics based on discrete pole representations: A path to renormalized perturbation theories". In: *Physical Review B* 110 (7 Aug. 2024), p. 075158. DOI: 10.1103/PhysRevB.110.075158.

[105] T. Barthel, U. Schollwöck, and S. R. White. "Spectral functions in one-dimensional quantum systems at finite temperature using the density matrix renormalization group". In: *Physical Review B* 79 (24 June 2009), p. 245101. DOI: 10.1103/PhysRevB.79.245101.

[106] Y. Tian and S. R. White. "Matrix product state recursion methods for computing spectral functions of strongly correlated quantum systems". In: *Physical Review B* 103 (12 Mar. 2021), p. 125142. DOI: 10.1103/PhysRevB.103.125142.

[107] C. C. Reeves, J. Yin, Y. Zhu, K. Z. Ibrahim, C. Yang, and V. ě. Vl ček. "Dynamic mode decomposition for extrapolating nonequilibrium Green's-function dynamics". In: *Physical Review B* 107 (7 Feb. 2023), p. 075107. DOI: 10.1103/PhysRevB.107.075107.

[108] G. Park, Z. Huang, Y. Zhu, C. Yang, G. K.-L. Chan, and L. Lin. "Quasi-Lindblad pseudomode theory for open quantum systems". In: *Physical Review B* 110 (19 Nov. 2024), p. 195148. DOI: 10.1103/PhysRevB.110.195148.

[109] J. Kaye, K. Chen, and H. U. Strand. "libdlr: Efficient imaginary time calculations using the discrete Lehmann representation". In: *Computer Physics Communications* 280 (2022), p. 108458. DOI: 10.1016/j.cpc.2022.108458.

[110] P. Pokhilko, C.-N. Yeh, and D. Zgid. "Iterative subspace algorithms for finite-temperature solution of Dyson equation". In: *The Journal of Chemical Physics* 156.9 (Mar. 2022), p. 094101. DOI: 10.1063/5.0082586.

[111] D. P. Arovas, E. Berg, S. A. Kivelson, and S. Raghu. "The Hubbard Model". In: *Annual Review of Condensed Matter Physics* Volume 13 (2022), pp. 239–274. DOI: 10.1146/annurev-conmatphys-031620-102024.

[112] M. Qin, T. Schäfer, S. Andergassen, P. Corboz, and E. Gull. "The Hubbard Model: A Computational Perspective". In: *Annual Review of Condensed Matter Physics* Volume 13 (2022), pp. 275–302. DOI: 10.1146/annurev-conmatphys-090921-033948.

[113] P. Werner, A. Comanac, L. de'Medici, M. Troyer, and A. J. Millis. "Continuous-Time Solver for Quantum Impurity Models". In: *Physical Review Letters* 97 (7 Aug. 2006), p. 076405. DOI: 10.1103/PhysRevLett.97.076405.

[114] J. VandeVondele and J. Hutter. "Gaussian basis sets for accurate calculations on molecular systems in gas and condensed phases". In: *The Journal of Chemical Physics* 127.11 (2007), p. 114105. DOI: 10.1063/1.2770708.

[115] S. Goedecker, M. Teter, and J. Hutter. "Separable dual-space Gaussian pseudopotentials". In: *Physical Review B* 54.3 (1996), p. 1703. DOI: 10.1103/PhysRevB.54.1703.

[116] Q. Sun, X. Zhang, S. Banerjee, P. Bao, M. Barbry, N. S. Blunt, N. A. Bogdanov, G. H. Booth, J. Chen, Z.-H. Cui, et al. "Recent developments in the PySCF program package". In: *The Journal of Chemical Physics* 153.2 (2020), p. 024109. DOI: 10.1063/5.0006074.

[117] S. Iskakov, A. Hampel, N. Wentzell, and E. Gull. "TRIQS/Nevanlinna: Implementation of the Nevanlinna Analytic Continuation method for noise-free data". In: *Computer Physics Communications* 304 (2024), p. 109299. DOI: 10.1016/j.cpc.2024.109299.

[118] L. Zhang, Y. Yu, and E. Gull. *Green-Phys/MiniPole: v0.1.1*. Version v0.1.1. Oct. 2024. DOI: 10.5281/zenodo.13936894.

[119] L. Zhang, Y. Yu, and E. Gull. *Green-Phys/MiniPole*. https://github.com/Green-Phys/MiniPole. Oct. 2024.

[120] M. Caffarel and W. Krauth. "Exact diagonalization approach to correlated fermions in infinite dimensions: Mott transition and superconductivity". In: *Physical Review Letters* 72 (10 Mar. 1994), pp. 1545–1548. DOI: 10.1103/PhysRevLett.72.1545.

[121] D. Zgid, E. Gull, and G. K.-L. Chan. "Truncated configuration interaction expansions as solvers for correlated quantum impurity models and dynamical mean-field theory". In: *Physical Review B* 86 (16 Oct. 2012), p. 165128. DOI: 10.1103/PhysRevB.86.165128.

[122] A. Shee and D. Zgid. "Coupled Cluster as an Impurity Solver for Green's Function Embedding Methods". In: *Journal of Chemical Theory and Computation* 15.11 (Nov. 2019), pp. 6010–6024. DOI: 10.1021/acs.jctc.9b00603.

[123] T. Zhu, C. A. Jiménez-Hoyos, J. McClain, T. C. Berkelbach, and G. K.-L. Chan. "Coupled-cluster impurity solvers for dynamical mean-field theory". In: *Physical Review B* 100 (11 Sept. 2019), p. 115154. DOI: 10.1103/PhysRevB.100.115154.

[124] E. Koch, G. Sangiovanni, and O. Gunnarsson. "Sum rules and bath parametrization for quantum cluster theories". In: *Physical Review B* 78 (11 Sept. 2008), p. 115102. DOI: 10.1103/PhysRevB.78.115102.

[125] D. Sénéchal. "Bath optimization in the cellular dynamical mean-field theory". In: *Physical Review B* 81 (23 June 2010), p. 235125. DOI: 10.1103/PhysRevB.81.235125.

[126] A. Liebsch and H. Ishida. "Temperature and bath size in exact diagonalization dynamical mean field theory". In: *Journal of Physics: Condensed Matter* 24.5 (Dec. 2011), p. 053201. DOI: 10.1088/0953-8984/24/5/053201.

[127] C. Mejuto-Zaera, L. Zepeda-Núñez, M. Lindsey, N. Tubman, B. Whaley, and L. Lin. "Efficient hybridization fitting for dynamical mean-field theory via semi-definite relaxation". In: *Physical Review B* 101 (3 Jan. 2020), p. 035143. DOI: 10.1103/PhysRevB.101.035143.

[128] Y. Tanimura and R. Kubo. "Time Evolution of a Quantum System in Contact with a Nearly Gaussian-Markoffian Noise Bath". In: *Journal of the Physical Society of Japan* 58.1 (1989), pp. 101–114. DOI: 10.1143/JPSJ.58.101.

[129] E. Arrigoni, M. Knap, and W. von der Linden. "Nonequilibrium Dynamical Mean-Field Theory: An Auxiliary Quantum Master Equation Approach". In: *Physical Review Letters* 110 (8 Feb. 2013), p. 086403. DOI: 10.1103/PhysRevLett.110.086403.

[130] Y. Tanimura. "Numerically "exact" approach to open quantum dynamics: The hierarchical equations of motion (HEOM)". In: *The Journal of Chemical Physics* 153.2 (July 2020), p. 020901. DOI: 10.1063/5.0011599.

[131] H. Shinaoka, J. Otsuki, K. Haule, M. Wallerberger, E. Gull, K. Yoshimi, and M. Ohzeki. "Overcomplete compact representation of two-particle Green's functions". In: *Physical Review B* 97 (20 May 2018), p. 205111. DOI: 10.1103/PhysRevB.97.205111.

[132] J. Kaye and D. Golež. "Low rank compression in the numerical solution of the nonequilibrium Dyson equation". In: *SciPost Physics* 10 (2021), p. 091. DOI: 10.21468/SciPostPhys.10.4.091.

[133] X. Dong, E. Gull, and H. U. R. Strand. "Excitations and spectra from equilibrium real-time Green's functions". In: *Physical Review B* 106 (12 Sept. 2022), p. 125153. DOI: 10.1103/PhysRevB.106.125153.

[134] J. Kaye, Z. Huang, H. U. R. Strand, and D. Gole ž. "Decomposing Imaginary-Time Feynman Diagrams Using Separable Basis Functions: Anderson Impurity Model Strong-Coupling Expansion". In: *Physical Review X* 14 (3 Aug. 2024), p. 031034. DOI: 10.1103/PhysRevX.14.031034.

[135] R. Smit. *Hyperfunction formulation of many body Green's functions and the Matsubara formalism*. 2022.

[136] S. Goswami, K. Barros, and M. R. Carbone. "Physically interpretable approximations of many-body spectral functions". In: *Physical Review E* 109 (1 Jan. 2024), p. 015302. DOI: 10.1103/PhysRevE.109.015302.

[137] L. Huang and C. Yue. "Barycentric rational function approximation made simple: A fast analytic continuation method for Matsubara Green's functions". In: *Physical Review B* 111 (12 Mar. 2025), p. 125139. DOI: 10.1103/PhysRevB.111.125139.

[138] Y. Tanimura. "Nonperturbative expansion method for a quantum system coupled to a harmonic-oscillator bath". In: *Physical Review A* 41 (12 June 1990), pp. 6676–6687. DOI: 10.1103/PhysRevA.41.6676.

[139] A. Ishizaki and Y. Tanimura. "Quantum Dynamics of System Strongly Coupled to Low-Temperature Colored Noise Bath: Reduced Hierarchy Equations Approach". In: *Journal of the Physical Society of Japan* 74.12 (2005), pp. 3131–3134. DOI: 10.1143/JPSJ.74.3131.

[140] J. Hu, R.-X. Xu, and Y. Yan. "Communication: Padé spectrum decomposition of Fermi function and Bose function". In: *The Journal of Chemical Physics* 133.10 (Sept. 2010), p. 101106. DOI: 10.1063/1.3484491.

[141] J. Hu, M. Luo, F. Jiang, R.-X. Xu, and Y. Yan. "Padé spectrum decompositions of quantum distribution functions and optimal hierarchical equations of motion construction for quantum open systems". In: *The Journal of Chemical Physics* 134.24 (2011), p. 244106. DOI: 10.1063/1.3602466.

[142] C. Duan, Q. Wang, Z. Tang, and J. Wu. "The study of an extended hierarchy equation of motion in the spin-boson model: The cutoff function of the sub-Ohmic spectral density". In: *The Journal of Chemical Physics* 147.16 (2017), p. 164112. DOI: 10.1063/1.4997669.

[143] A. Erpenbeck, C. Hertlein, C. Schinabeck, and M. Thoss. "Extending the hierarchical quantum master equation approach to low temperatures and realistic band structures". In: *The Journal of Chemical Physics* 149.6 (2018), p. 064106. DOI: 10.1063/1.5041716.

[144] L. Cui, H.-D. Zhang, X. Zheng, R.-X. Xu, and Y. Yan. "Highly efficient and accurate sum-over-poles expansion of Fermi and Bose functions at near zero temperatures: Fano spectrum decomposition scheme". In: *The Journal of Chemical Physics* 151.2 (2019), p. 024110. DOI: 10.1063/1.5096945.

[145] M. Xu, Y. Yan, Q. Shi, J. Ankerhold, and J. T. Stockburger. "Taming Quantum Noise for Efficient Low Temperature Simulations of Open Quantum Systems". In: *Physical Review Letters* 129 (23 Nov. 2022), p. 230601. DOI: 10.1103/PhysRevLett.129.230601.

[146] Z.-H. Chen, Y. Wang, X. Zheng, R.-X. Xu, and Y. Yan. "Universal time-domain Prony fitting decomposition for optimized hierarchical quantum master equations". In: *The Journal of Chemical Physics* 156.22 (2022), p. 221102. DOI: 10.1063/5.0095961.

[147] D. Suess, A. Eisfeld, and W. T. Strunz. "Hierarchy of Stochastic Pure States for Open Quantum System Dynamics". In: *Physical Review Letters* 113 (15 Oct. 2014), p. 150403. DOI: 10.1103/PhysRevLett.113.150403.

[148] D. Tamascelli, A. Smirne, S. F. Huelga, and M. B. Plenio. "Nonperturbative Treatment of non-Markovian Dynamics of Open Quantum Systems". In: *Physical Review Letters* 120 (3 Jan. 2018), p. 030402. DOI: 10.1103/PhysRevLett.120.030402.

[149] A. Dorda, M. Nuss, W. von der Linden, and E. Arrigoni. "Auxiliary master equation approach to nonequilibrium correlated impurities". In: *Physical Review B* 89 (16 Apr. 2014), p. 165105. DOI: 10.1103/PhysRevB.89.165105.

[150] D. Werner, J. Lotze, and E. Arrigoni. "Configuration interaction based nonequilibrium steady state impurity solver". In: *Physical Review B* 107 (7 Feb. 2023), p. 075119. DOI: 10.1103/PhysRevB.107.075119.

[151] W. Metzner and D. Vollhardt. "Correlated Lattice Fermions in $d = \infty$ Dimensions". In: *Physical Review Letters* 62 (3 Jan. 1989), pp. 324–327. DOI: 10.1103/PhysRevLett.62.324.

[152] Y. Nakatsukasa, O. Sète, and L. N. Trefethen. "The AAA Algorithm for Rational Approximation". In: *SIAM Journal on Scientific Computing* 40.3 (2018), A1494–A1522. DOI: 10.1137/16M1106122.

[153] Y. Nakatsukasa and L. N. Trefethen. "An Algorithm for Real and Complex Rational Minimax Approximation". In: *SIAM Journal on Scientific Computing* 42.5 (2020), A3157–A3179. DOI: 10.1137/19M1281897.

[154] T. Brandes. "Quantum Dissipation". In: *Lectures on the Background to Quantum Information*. Lecture notes. 2003. Chap. 7.

[155] U. Weiss. *Quantum dissipative systems*. World Scientific, 2012.

[156] P. L. Walters, T. C. Allen, and N. Makri. "Direct determination of discrete harmonic bath parameters from molecular dynamics simulations". In: *Journal of Computational Chemistry* 38.2 (2017), pp. 110–115. DOI: 10.1002/jcc.24527.

[157] D. Jasrasaria and E. Rabani. "Circumventing the phonon bottleneck by multiphonon-mediated hot exciton cooling at the nanoscale". In: *npj Computational Materials* 9.1 (Aug. 2023), p. 145. DOI: 10.1038/s41524-023-01102-8.

[158] D. Jasrasaria and E. Rabani. "Interplay of Surface and Interior Modes in Exciton–Phonon Coupling at the Nanoscale". In: *Nano Letters* 21.20 (Oct. 2021), pp. 8741–8748. DOI: 10.1021/acs.nanolett.1c02953.

[159] I. S. Dunn, R. Tempelaar, and D. R. Reichman. "Removing instabilities in the hierarchical equations of motion: Exact and approximate projection approaches". In: *The Journal of Chemical Physics* 150.18 (May 2019), p. 184109. DOI: 10.1063/1.5092616.

[160] L. Zhang, A. Erpenbeck, Y. Yu, and E. Gull. *Green-Phys/MiniPole: v0.4*. Version v0.4. Apr. 2025. DOI: 10.5281/zenodo.15121302.

[161] Y. Saad. *Numerical Methods for Large Eigenvalue Problems*. Society for Industrial and Applied Mathematics, 2011. DOI: 10.1137/1.9781611970739.

[162] L. N. Trefethen and D. Bau III. *Numerical Linear Algebra*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 1997. DOI: 10.1137/1.9780898719574.

[163] Z. Huang, G. Park, G. K.-L. Chan, and L. Lin. "Coupled Lindblad Pseudomode Theory for Simulating Open Quantum Systems". In: *Physical Review Letters* 136 (9 Mar. 2026), p. 090403. DOI: 10.1103/qfkp-jc7s.

[164] G. Lindblad. "On the generators of quantum dynamical semigroups". In: *Communications in Mathematical Physics* 48.2 (1976), pp. 119–130. DOI: 10.1007/BF01608499.

[165] V. Gorini, A. Kossakowski, and E. C. G. Sudarshan. "Completely positive dynamical semigroups of N-level systems". In: *Journal of Mathematical Physics* 17.5 (May 1976), pp. 821–825. DOI: 10.1063/1.522979.

[166] A. A. Dzhioev and D. S. Kosov. "Super-fermion representation of quantum kinetic equations for the electron transport problem". In: *The Journal of Chemical Physics* 134.4 (Jan. 2011), p. 044121. DOI: 10.1063/1.3548065.

[167] A. Go and A. J. Millis. "Adaptively truncated Hilbert space based impurity solver for dynamical mean-field theory". In: *Physical Review B* 96 (8 Aug. 2017), p. 085139. DOI: 10.1103/PhysRevB.96.085139.

[168] V. Satopaa, J. Albrecht, D. Irwin, and B. Raghavan. "Finding a "Kneedle" in a Haystack: Detecting Knee Points in System Behavior". In: *2011 31st International Conference on Distributed Computing Systems Workshops*. 2011, pp. 166–171. DOI: 10.1109/ICDCSW.2011.20.

[169] K. Arvai. *kneed*. Version 0.8.6. Mar. 2026. DOI: 10.5281/zenodo.6496267.

[170] M. Schmutz. "Real-time Green's functions in many body problems". In: *Zeitschrift für Physik B Condensed Matter* 30.1 (1978), pp. 97–106. DOI: 10.1007/BF01323673.