

國立臺灣大學理學院物理學研究所



博士論文

Department of Physics

College of Science

National Taiwan University

Doctoral Dissertation

多體動力學在張量網路與量子電路中的計算

Simulation of Many-Body Dynamics with Tensor
Networks and Quantum Circuits

洪浩迪

Hao-Ti Hung

指導教授：高英哲 博士

Advisor: Ying-Jer Kao, Ph.D.

中華民國 115 年 2 月

February 2026

國立臺灣大學博士學位論文

口試委員會審定書

PHD DISSERTATION ACCEPTANCE CERTIFICATE

NATIONAL TAIWAN UNIVERSITY

多體動力學在張量網路與量子電路中的計算

Simulation of Many-Body Dynamics with Tensor Networks and Quantum Circuits

本論文係洪浩迪（D11222003）在國立臺灣大學物理學研究所完成之博士學位論文，於民國 115 年 1 月 6 日承下列考試委員審查通過及口試及格，特此證明。

The undersigned, appointed by the Department of Physics, National Taiwan University on 06 (date) 01 (month) 2026 (year) have examined a PhD dissertation entitled above presented by Hao-Ti Hung (D11222003) candidate and hereby certify that it is worthy of acceptance.

口試委員 Oral examination committee :

高英軒

(指導教授 Advisor)

鍾佳民

黃一平

林諭諤

丁子柏





誌謝

本博士論文之完成，承蒙許多教授的指導與協助，以及許多夥伴的幫忙，方得以完成本篇著作，在此由衷感謝。

首先，我感謝我的指導教授高英哲，教授帶領我了解物理領域的學術道路，讓我認識一項項研究如何由發想逐步成形，一篇篇期刊論文是如何經歷同行審查、被接受或被拒稿，最終得以發表。研究題目有的源自既有問題，有的來自一次次學術研討會中的討論與交流，無論是國內或國際的研討會。也謝謝高老師帶領我參與多場國內與國際學術研討會，從中發想研究題目、討論技術難點、宣傳自己的研究，並結識同領域的研究者。我感謝高老師親身帶我走過一個個題目的研究歷程，正如他時常與學生分享的經驗，而我們得以親身參與其中，令我獲益良多，感謝他的付出與提點。也感謝教授提供給我充足的研究經費與研討會補助，讓我在從事研究的同時，無後顧之憂。

除了指導教授，有許許多多的教授、合作者、同學也在研究過程中從旁協助。我感謝 QVM 團隊的教授與夥伴在我 QITE 相關研究上的協助：陳柏中教授在 QVM 團隊的會議安排與主持中，於一次次會議討論中提出許多重要的見解與研究方向，使我在研究上受益良多；感謝曹統在本案中的協助，也感謝張淮竣在本案中提出的建議，以及在赴美國發表研究成果時彼此的照料。

在 Thirring 的案子上，我特別感謝林及仁教授，至今仍十分懷念該研究題目在一次次咖啡廳討論中逐步推進的過程，並感謝老師的熱

情款待；也感謝 Mari Carmen Bañuls 在這個案子上點子的發想、合作、協助與在技術上的指導。並感謝 Krzysztof Cichy 與 Amit Singh 在本案上的合作。在 E_8 symmetry 的案子上，感謝 Johannes Knolle 對於研究方向的發想，也感謝 Isabel Nha Minh Le 提供的技術並協助本案的完成，並感謝 Rongyang Sun 與謝魏子傑教授在本案中的討論與交流。在 QTT 的研究中，感謝鍾佳民教授與陳乾燦的討論，並給予許多寶貴的意見，亦感謝梁晉嘉的協助。

此外，我要特別感謝 Gia-Wei Chern 教授，讓我得以前往美國維吉尼亞大學物理學系進行移地研究，並於期間發想使用 Tensor Network 研究 XXZ Coarsening 的相關問題；亦特別感謝其學生 Ho Jang 在日常物理討論與其他事務上的協助。我也感謝 UVA' s International Center 的室友，在日常生活中的照料與互動。

特別感謝吳愷訢，為 Cytnx 專案的主要作者與貢獻者，我很高興能成為該專案的協作者；也謝謝林昶勝、許可、陳孜穎、Manuel Schneider 在案子上大力的協助。也特別感謝鍾佳民教授、陳柏中教授在這個案子上的討論與付出，以及所有協作的教授與人員。

我感謝林瑜琤教授、許琇娟教授與黃一平教授，在多個場合中討論、發想並給予寶貴的建議，也感謝王喬萱教授在不同方向的發想與關心，以及 Hidehiro Saito、蘇耀霆的交流。感謝譚道璘在我攻讀博士班期間給予的動力與交流，也感謝高文瀚與林育平在美國參加會議時的照顧。感謝孫士傑教授介紹與推薦。

我要特別感謝上述未能一一列出的研究室同學，在博士期間給予我重要的日常協助，包含陳允中、邱翔彬、范宇軒、陳詠鑫、劉晁泓、蘇偉誠、何采昀、廖聲融日常的物理討論，曾晟捷、程大榕日常的幫忙與閒聊。

感謝系上的助理劉家綺，在各項研究會議與行政事務上的協助，

也感謝系上與系辦所有工作人員在我就學期間提供的幫助。感謝教育部提供的獎學金，也感謝臺灣大學 - IBM 量子電腦中心、國網中心、NVIDIA 提供計算資源。



感謝一路以來指導與關心我的師長。最後我要感謝父母、弟妹、祖母、家人、親戚，給我在博士班期間的支持與關心，讓我順利且無後顧之憂地完成我的博士學位。

本博士論文的完成，承蒙許多人的協助，未能一一列舉，不及備載，謝謝你們。





摘要

本論文探討了利用張量網路與量子電路來模擬量子多體系統以及求解偏微分方程 (PDE) 的方法。第一部分著重於在無限與有限長度格點的張量網路架構中使用時間依賴變分原理 (Time-Dependent Variational Principle, TDVP) 演算法，以研究自旋模型的動力學行為。我們透過將 Thirring 模型轉換為 spin-lattice Hamiltonian，並進行複時間演化，探討其動態量子相變 (Dynamical Quantum Phase Transitions, DQPTs)。本研究建立了 DQPT 與 Fisher 零點 (Fisher zeros) 之間的關聯。此外，我們利用有限系統的 TDVP 方法，在 XXZ 模型中進行 quantum quench 後的模擬，觀察到隨著實時間演化而出現的 coarsening 現象。

接著，本論文探討以極座標下的 Quantic Tensor Train (QTT) 表示法求解二維偏微分方程。為處理 Laplacian 算符在原點處的 singularity，我們提出一種方法，使得所有相關運算皆可轉換為矩陣乘積算符 (Matrix Product Operators, MPOs) 的形式。此架構使得我們能夠利用 DMRG 及 TDVP 等演算法，模擬二維簡諧振子與 Gross-Pitaevskii (GP) 方程的基態與動態行為。

最後一部分則研究量子電路方法在多體量子動力學中的應用，特別是 Ising 模型。我們利用受 TDVP 啟發的參數化量子電路進行無限長度 Ising 模型的虛時間演化，將矩陣乘積態 (Matrix Product States, MPS) 轉換為量子電路的表示，並於實際量子硬體上執行模擬。同

時，我們 cost function 與期望值的分佈進行統計分析，並結合 quantum Lanczos 演算法以提升變分最佳化的效果。此外，我們模擬具縱向與橫向磁場的 Ising 模型之實時間演化，以探討其所呈現的 E_8 對稱性。為提升量子電路實作效率，我們亦應用基於張量網路的壓縮技術，優化電路深度，並於真實量子裝置上進行驗證。

綜合而言，本論文展示了張量網路與量子電路方法在模擬複雜量子系統與求解偏微分方程中的高效性與靈活性，為量子模擬的理論研究與實際應用開啟了新的方向。

關鍵字：矩陣乘積態（MPS）、DQPT、Thirring 模型、Coarsening、QTT、量子運算、 E_8 對稱



Abstract

This dissertation explores methods for simulating quantum many-body systems and solving partial differential equations (PDEs) using tensor networks and quantum circuits. The first part focuses on the use of the Time-Dependent Variational Principle (TDVP) algorithm in both infinite- and finite-size lattice systems in tensor networks to study dynamical behavior in spin models. We investigate dynamical quantum phase transitions (DQPTs) in the Thirring and XXZ models by mapping the Thirring model to a spin-lattice Hamiltonian and performing complex-time evolution. A connection between Fisher zeros and DQPTs is established. In addition, we study the coarsening phenomenon in the XXZ model by applying finite-size TDVP following a quantum quench, revealing the emergence of coarsening behavior through real-time evolution.

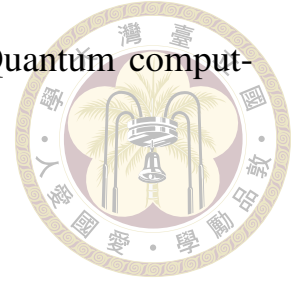
Then, addresses the solution of two-dimensional partial differential equations using the Quantic Tensor Train (QTT) representation under polar coordinates. To manage the singularity at the origin of the Laplacian operator,

we propose a regularization method that enables all relevant operations to be expressed as Matrix Product Operators (MPOs). This framework allows for ground state and dynamical simulations of the two-dimensional simple harmonic oscillator and the Gross–Pitaevskii (GP) equation using DMRG and TDVP algorithms.

The final part investigates quantum circuit approaches to many-body dynamics, with a particular focus on the Ising model. We perform imaginary-time evolution of the infinite-size Ising model using parameterized quantum circuits inspired by TDVP, converting Matrix Product States (MPS) into circuit representations and executing simulations on real quantum hardware. Statistical analysis of cost function distributions and expectation values is conducted, and a quantum Lanczos algorithm is used to improve variational optimization. Additionally, we simulate the real-time evolution of the transverse-field Ising model with a longitudinal field to explore signatures of emergent E_8 symmetry. Tensor network-based compression techniques are applied to optimize the depth of the quantum circuit, further enhancing the efficiency of the implementation on quantum devices.

Together, these studies demonstrate the effectiveness of tensor network and quantum circuit frameworks in simulating complex quantum systems and solving PDEs, paving the way for both theoretical advances and practical applications in quantum simulation.

Keywords: MPS, DQPT, Thirring, Coarsening, QTT, Quantum computing, E_8 Symmetry







List of Publications

Publications

1. Mari Carmen Bañuls, Krzysztof Cichy, **Hao-Ti Hung**, Ying-Jer Kao, C.-J. David Lin and Amit Singh. *Dynamical quantum phase transition and thermal equilibrium in the lattice Thirring model*. Phys. Rev. Research 7, 023194, (2025)
2. Kai-Hsin Wu, Chang-Teng Lin, Ke Hsu, **Hao-Ti Hung**, Manuel Schneider, Chia-Min Chung, Ying-Jer Kao and Pochung Chen. *The Cytex library for tensor networks*. SciPost Phys. Codebases 53 (2025)

Preprints

1. **Hao-Ti Hung**, Isabel Nha Minh Le, Johannes Knolle and Ying-Jer Kao. *Improved Ising Meson Spectroscopy Simulation on a Noisy Digital Quantum Device*. arXiv:2512.02516 (2025).

Manuscripts in Preparation

1. **Hao-Ti Hung**, Tung Tsao, and Ying-Jer Kao. *Quantum Imaginary Time Evolution on an Infinite 1D Lattice*. In preparation.
2. **Hao-Ti Hung** and Ying-Jer Kao. *TDVP in QTT format for PDEs in polar coordinates*. In preparation.



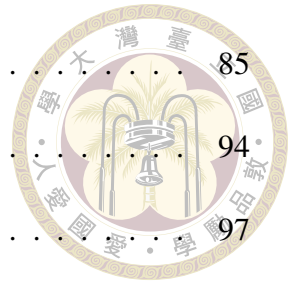


Table of Contents

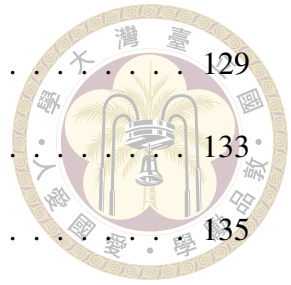
	Page
口試委員會審定書	i
誌謝	iii
摘要	vii
Abstract	ix
List of Publications	xiii
Table of Contents	xv
List of Figures	xix
List of Tables	xxv
Chapter 1 Introduction	1
1.1 Motivation and Role of TDVP	2
1.2 Overview of Dissertation Topics	4
Chapter 2 Dynamical Quantum Phase Transition of the Thirring and XXZ	
Models	7
2.1 Formalism	11
2.1.1 The Thirring Model as a Spin Model	11
2.1.2 Matrix Product State	13

2.1.3	Penalty Term and Matrix Product Operator	14
2.1.4	Brief Introduction of VUMPS and TDVP	15
2.2	Equilibrium Phase Structure of the 1+1-dimensional Thirring Model	17
2.3	The Loschmidt Echo and the Mixed Transfer Matrix	19
2.3.1	Quantum Quench and DQPT	19
2.3.2	Investigation of DQPT in the Lattice Version of the Thirring Model	23
2.3.3	DQPT and Thermalization	25
2.4	DQPTs and Finite-temperature Phase Diagram	26
2.4.1	Energy Threshold and the Inverse Temperature	26
2.5	DQPTs, Fisher Zeros, and the Effective-temperature–Time Plane	35
2.5.1	Benchmark: Fisher Zeros and DQPTs In the XXZ Chain	35
2.5.2	Complex-time Evolution in the Thirring Model	36
2.5.3	Analytic Fisher Zeros in the Transverse-field Ising Chain	39
2.6	Conclusion	41
Chapter 3	Coarsening of the XXZ Model	45
3.1	Quench and Coarsening	45
3.2	Coarsening of the XXZ Model in a Quantum Quench	48
3.3	Statistical Averaging over Random Longitudinal Fields	58
Chapter 4	Solving PED Using Quantics Tensor Train and TDVP	63
4.1	QTT Representation in Cartesian Coordinates	64
4.2	Simulation Results in the Cartesian Coordinate	74
4.3	QTT Representation in Polar Coordinates	80

4.4	Singularity at the Origin	85
4.5	MPO Representation in Polar Coordinates	94
4.6	Simulation Results in Polar Coordinates	97
4.6.1	Heat Equation in Polar Coordinates	98
4.6.2	Schrödinger Equation in the Two-Dimensional Harmonic Oscillator	100
4.7	Rotating Gross–Pitaevskii Equation	105
4.7.1	Rotating Gross–Pitaevskii Equation in Polar Coordinates	105
4.7.2	Increasing the QTT Resolution by Adding Tensor Sites	107
4.7.2.1	Nearest-Neighbor Interpolation	108
4.7.2.2	Linear Interpolation	109
4.7.2.3	Extension to Polar Coordinates	109
4.7.2.4	Application: Refinement in the Rotating GP Simulation	110
4.7.3	Real-Time Dynamics After a Quantum Quench	111
Chapter 5	Quantum Imaginary Time Evolution	115
5.1	The Concept of the TDVP Based on Quantum Circuits	118
5.1.1	The TDVP based on uMPS	118
5.1.2	Mapping between uMPS and Quantum Circuits	123
5.1.3	Real-Time Evolution on 1D Infinite-size	124
5.2	Expand Non-unitary to Unitary Gate	126
5.2.1	Expand Non-unitary to Unitary Gate Using SVD	126
5.2.2	The Construction of the Environment Gates	127
5.2.3	Expansion of the Imaginary-time Evolution Operator	127
5.3	Quantum Imaginary-time Evolution for Infinite Systems	128



5.3.1	Two Approaches to Deal with Non-unitary Operator	129
5.3.2	Quantum Lanczos Algorithm	133
5.4	Quantum Imaginary-time Evolution for Finite System	135
5.5	Simulation Results	138
5.5.1	Results for 1D Infinite-size	138
5.5.2	Results for 1D Finite-size	142
5.6	Statistical Effects of Quantum Measurements	145
5.7	Conclusion	152
Chapter 6	E_8 Symmetry of the Ising Model:Using Quantum Circuits Simulation	155
6.1	E_8 Symmetry in Quantum Ising Model	157
6.2	Using IBMQ Quantum Real Device	163
6.2.1	Evolution Gate	163
6.2.2	Error Mitigation	164
6.3	Riemannian Quantum Circuit Optimization	166
6.4	Masses in Frequency Domain	171
6.5	Detail of Setting on Real Device	174
6.6	Conclusion	176
Chapter 7	Summary and Outlook	179
References		185





List of Figures

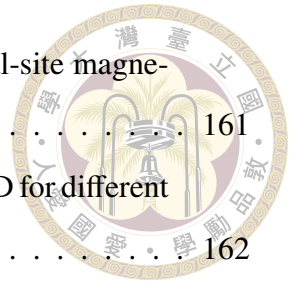
1.1	Schematic overview of the structure of this dissertation.	3
2.1	Zero-temperature phase diagram of the Thirring model in the coupling-mass plane.	10
2.2	The fixed-point equations of the boundary tensors L and R	15
2.3	An illustration of the TDVP.	16
2.4	The return rate function and the full spectrum of the mixed transfer matrix $T(t)$ for $\Delta_2 = -0.9$	21
2.5	The return rate function and the full spectrum of the mixed transfer matrix $T(t)$ for $\Delta_2 = 0.9$	22
2.6	The return rate functions (red) and the chiral condensate (blue) for the cases (a) without and (b) with DQPTs.	24
2.7	Relation $E(\beta)$ between the mean energy and the inverse temperature in the Gibbs state restricted to the zero magnetization sector.	27
2.8	Energy densities corresponding to ground states of varying $m_1 a$ and Δ_1	30
2.9	Threshold energy for DQPTs in the XXZ model.	31
2.10	The DQPTs on the inverse temperature–time plane.	34
2.11	DQPT points and Fisher zeros of the XXZ chain.	37
2.12	DQPT structure in the complex-time plane $z = t - i\tau$ for the Thirring model.	38
2.13	Analytic Fisher zeros of the TFIM Loschmidt amplitude in the complex-time plane $z = t - i\tau$ for two different quenches.	40
3.1	Magnetization of the two-dimensional Ising model in zero field.	46

3.2	Monte Carlo simulation of domain growth in the two-dimensional Ising model at $T = 0$	47
3.3	Phase diagram of the XXZ model.	49
3.4	Real-time evolution following a quench.	50
3.5	Néel-order amplitude of XXZ ground states for various N and J_z (DMRG).	51
3.6	Quantum quench from $J_z = 0.8$ to $J_z = 3$ (DMRG + TDVP, $D = 128$).	52
3.7	Neel order for quantum quench from.	53
3.8	Domain-wall dynamics following a quench from $J_z = 0.8$ to $J_z = 3$	55
3.9	Edge-detection analysis of domain boundaries.	57
3.10	Extracted domain-wall number as a function of time.	58
3.11	Two TDVP simulations with random longitudinal fields.	59
3.12	Statistical averaging of domain-wall number over fifty TDVP runs.	60
3.13	Improved domain-wall identification method.	61
4.1	A one-dimensional function in the QTT representation.	65
4.2	Improving the accuracy of the QTT representation by increasing the number of sampling points.	65
4.3	Heat equation simulations using the QTT representation	75
4.4	Simulation results for the two-dimensional Gross–Pitaevskii equation using the TDVP algorithm.	77
4.5	Tensor-network structure of the TDVP algorithm applied to the 1D Maxwell equations.	78
4.6	Simulation results for the 1D Maxwell equations obtained using the QTT–TDVP method.	79
4.7	QTT representation of a two-dimensional function $f(x, y)$ or $f(r, \theta)$	80
4.8	Test functions used to evaluate the QTT representation in Cartesian and polar coordinates.	81
4.9	Relative QTT approximation errors in Cartesian and polar coordinates.	83
4.10	Bond entanglement entropy of QTT representations in Cartesian and polar coordinates.	84

4.11	Illustration of Cartesian and polar discretizations for evaluating the Laplacian of a two-dimensional function.	86
4.12	Multiple polar grid points (r_0, θ_j) mapping to a single physical origin.	88
4.13	Angular averaging for the polar Laplacian at the origin	89
4.14	TDVP simulation of the heat equation in polar coordinates.	98
4.15	(a) Heat source distribution $S(r, \theta)$ used in the inhomogeneous heat equation (4.68). (b) TDVP simulation of the resulting heat flow at time $t = 0.25$	99
4.16	Comparison of numerical and analytical probability densities for selected eigenstates of the two-dimensional Schrödinger equation in polar coordinates.	101
4.17	Tensor-network structure of the normalization operation.	104
4.18	DMRG simulation of the rotating GP equation with strong nonlinear interactions.	106
4.19	Ground-state density distribution $ \psi(r, \theta) ^2$ of the rotating GP equation.	107
4.20	Nearest-neighbor interpolation in the QTT representation.	108
4.21	Linear interpolation in the QTT representation.	110
4.22	Nearest-neighbor interpolation in polar coordinates.	111
4.23	Adaptive grid refinement for the rotating GP equation using QTT interpolation.	112
4.24	Real-time dynamics of the rotating GP equation after a quantum quench, simulated using QTT–TDVP in polar coordinates.	114
5.1	The tensor diagram of the uMPS.	119
5.2	The concept of the TDVP. See Eq.(5.2)	119
5.3	The transformation of the cost function (Eq. (5.2)) from a TN diagram to a quantum circuit representation.	121
5.4	(a) The fixed-point equations based on the uMPS with evolution operator. (b) and (c) are the fixed-point equations based on quantum circuits.	122
5.5	The mapping between a one-site unit cell of the uMPS and a unitary gate in the quantum circuit.	123

5.6	(a) The left environment of the uMPS. (b) The left environment of the quantum circuit.	124
5.7	The cost function Eq (5.2) represented in a quantum circuit, with respect to a bond dimension of $D = 2$	125
5.8	An example of a PQC with virtual bond dimension $D = 2$	125
5.9	A nonunitary operator can be extended to a unitary gate by introducing an additional qubit, commonly referred to as the ‘ancilla qubit.’	127
5.10	(a) The quantum circuit for the cost function, with the ITE operator requiring replacement by $e^{-iAd\tau}$. (b)The quantum circuit corresponding to Sec.5.2	128
5.11	The cost function represented in TN diagram used to optimize $A(\tau)$	130
5.12	The comparison with two different method introduced.	132
5.13	The ancilla qubits can be reused so they will not increase linearly with time evolution.	137
5.14	(a) The results of ITE on both a simulator and IBMQ real devices. (b) The results of the QLanczos algorithm.	140
5.15	(a) The classical TDVP TN algorithm with bond dimension $D = 2$ and results from the ITE method. (b) The results from the real quantum device ‘ibmq_auckland’, optimized using the QLanczos algorithm.	141
5.16	The solid lines show the classical TN simulation results.	143
5.17	Results from the quantum circuit simulator using reused ancilla qubits. . .	144
5.18	Cost evolution and statistical comparison for the first QITE time step. . .	146
5.19	Cost evolution under measurement-based parameter updates compared with statevector predictions.	148
5.20	Energy evolution and energy distribution over the final 10 time steps from repeated noiseless QITE simulations using measurement-based updates. .	150
5.21	Energy-density distribution and fitted probability laws.	151
6.1	Simulation results obtained via exact diagonalization on a classical computer.	159

6.2	Classical simulation of the real-time evolution of the central-site magnetization $\langle \sigma_{\text{cen}}^z(t) \rangle$	161
6.3	Frequency-domain spectrum $\langle \sigma_{\text{cen}}^z(\omega) \rangle$ obtained via exact ED for different system sizes.	162
6.4	Quantum circuit construction of the real-time evolution operator for the TFIM using first-order Trotter decomposition.	163
6.5	Real-time evolution results on the IBM Quantum device <code>ibm_torino</code> for an 8-site transverse-field Ising model.	165
6.6	TN representation of the overlap between the target unitary operator and the quantum gates.	167
6.7	Simulation results using Riemannian optimization to construct time-evolution circuits with different numbers of layers, executed on a classical simulator.	168
6.8	Real-device simulation results using the IBM Quantum device <code>ibm_torino</code> , where the time-evolution gates are prepared via Riemannian optimization.	169
6.9	Exact real-time evolution starting from different initial states for the transverse-field Ising model with parameters $J = h_x = 1, h_z = 3$	172
6.10	Frequency-domain simulation results on the IBM Quantum device.	173
6.11	Qubit layout of the IBM Quantum device <code>ibm_torino</code>	175
7.1	Overview of completed work and potential future directions discussed in this dissertation.	182







List of Tables

4.1	Error analysis for the two-dimensional heat-equation simulation using different bond dimensions D	75
6.1	E_8 mass spectrum: theory, ED, and IBM Quantum results.	174





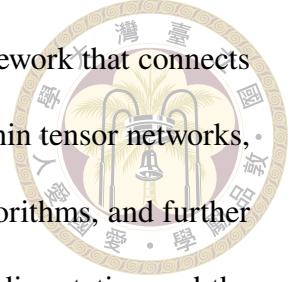
Chapter 1 Introduction

This dissertation explores methods for simulating quantum many-body systems and solving partial differential equations (PDEs) using tensor networks and quantum circuits. It is divided into five main topics, each of which focuses on a different method or application. Although these topics differ in content, they are all closely related through their treatment of time-dependent problems, including both imaginary-time and real-time evolution.

In recent years, quantum computing has rapidly advanced, leading to the development of noisy intermediate-scale quantum (NISQ) devices capable of simulating small-scale quantum systems. Tensor network (TN) techniques, especially matrix product states (MPS), are closely connected to quantum circuits and can often be translated between the two formalisms. Established algorithms such as the time-dependent variational principle (TDVP) and density matrix renormalization group (DMRG) provide robust tools for studying quantum dynamics and ground states. Exploring the connection between TN methods and quantum circuits, particularly how TN algorithms can be mapped onto or hybridized with variational quantum algorithms, represents a new research direction.

A central concept throughout this dissertation is the TDVP algorithm, which is intro-

duced in Section 1.1. Conceptually, TDVP serves as a unifying framework that connects all research topics in this thesis. Starting from TDVP formulated within tensor networks, one can naturally distinguish between infinite-size and finite-size algorithms, and further extend these ideas to quantum circuits. The overall structure of the dissertation and the role of TDVP as a unifying principle are summarized schematically in Fig. 1.1. In Section 1.2, we outline the five research topics discussed in Chapters 2–6 and describe how they are related and structured.



1.1 Motivation and Role of TDVP

The time-dependent variational principle (TDVP) provides a unifying framework for all research topics discussed in this dissertation. Rather than serving only as a numerical algorithm, TDVP plays a central conceptual role by organizing different physical problems and simulation platforms within a common variational perspective.

TDVP approximates the time evolution of a quantum system by projecting the Schrödinger equation onto the tangent space of a variational manifold, typically the matrix product state (MPS) manifold [1]. This projection ensures that the quantum state remains within the chosen variational class while optimally approximating its exact time evolution. TDVP can be formulated for both infinite-size and finite-size systems, leading to distinct but closely related applications.

In this dissertation, infinite-size TDVP is employed to study nonequilibrium phenomena that are most naturally defined in the thermodynamic limit, such as dynamical quantum phase transitions (DQPTs) following a quantum quench. Finite-size TDVP, on the other

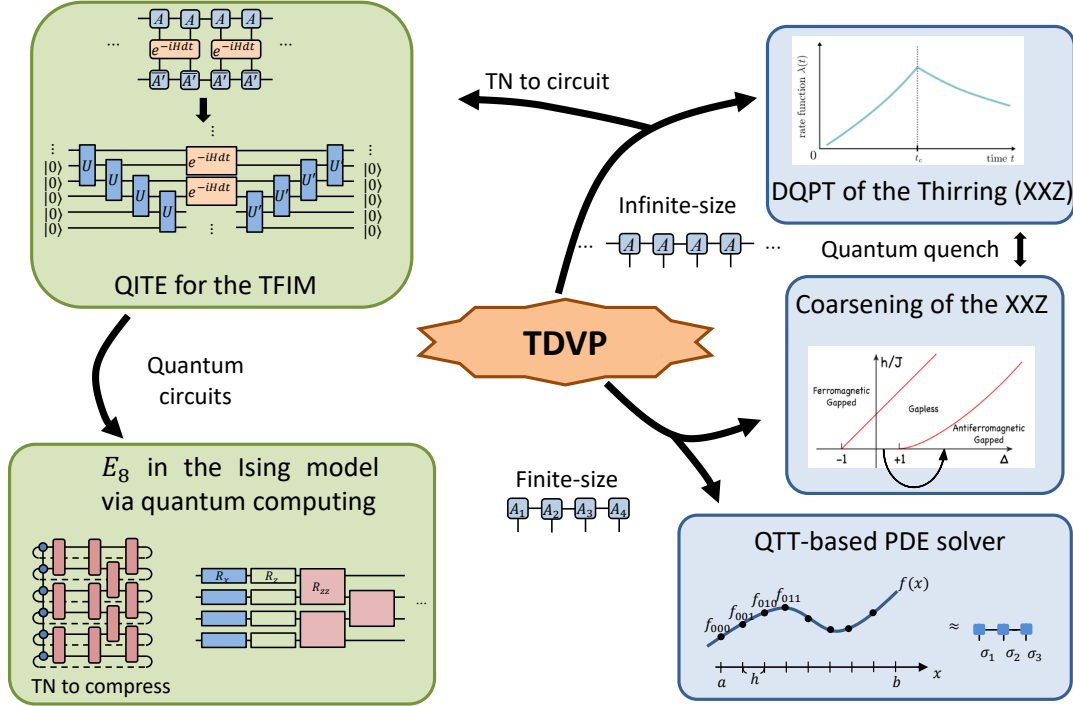
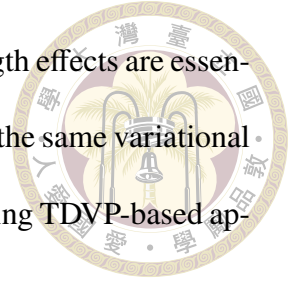


Figure 1.1: Schematic overview of the structure of this dissertation and the role of the TDVP as a unifying framework. Starting from TDVP formulated within tensor networks, both infinite-size (uniform MPS) and finite-size MPS algorithms are developed. Infinite-size TDVP is applied to study DQPTs in the Thirring and XXZ models following quantum quenches, where thermodynamic-limit descriptions are essential. Finite-size TDVP and related MPS techniques are used to investigate coarsening dynamics in the XXZ model and to solve partial differential equations by representing discretized functions as matrix product states, leading to QTT-based PDE solvers. Building on the correspondence between tensor networks and quantum circuits, TDVP-inspired algorithms are translated into the circuit language, enabling quantum imaginary-time evolution for the transverse-field Ising model. We perform real-time quantum simulations of the Ising model with transverse and longitudinal fields to investigate the E_8 structure. Tensor-network techniques are also employed to compress time-evolution operators and optimize quantum circuits.

hand, is used to investigate time-dependent processes where finite-length effects are essential, including coarsening dynamics. Beyond quantum lattice models, the same variational structure allows discretized functions to be represented as MPS, enabling TDVP-based approaches to numerical solutions of partial differential equations.



Motivated by the close correspondence between tensor networks and quantum circuits, TDVP also serves as a conceptual bridge to quantum algorithms. TDVP-inspired ideas are translated into the circuit language to develop variational schemes for imaginary-time evolution and to perform quantum simulations on noisy intermediate-scale quantum devices. Tensor-network techniques also provide tools for circuit optimization and operator compression, which are essential for limited circuit depth.

The relationships between these different formulations and applications, as well as their role within the overall structure of the dissertation, are summarized schematically in Fig. 1.1.

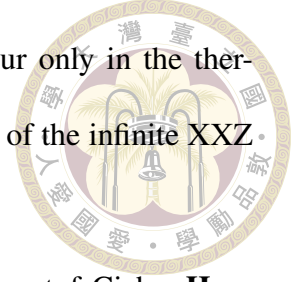
1.2 Overview of Dissertation Topics

The five main research topics of this dissertation are presented in Chapters 2–6 and summarized as follows:

- **Chapter 2: Dynamical Quantum Phase Transitions (DQPT) in the Thirring and XXZ models.**

We study quantum quenches and DQPTs using uMPS and the infinite-size TDVP algorithm. By mapping the Thirring model to a spin-lattice Hamiltonian with a

staggered term, we demonstrate that DQPTs, believed to occur only in the thermodynamic limit, can be observed through real-time evolution of the infinite XXZ model.



This work has been published in: Mari Carmen Bañuls, Krzysztof Cichy, **Hao-Ti Hung**, Ying-Jer Kao, C.-J. David Lin, and Amit Singh, *Dynamical quantum phase transition and thermal equilibrium in the lattice Thirring model*, Phys. Rev. Research **7**, 023194 (2025).

- **Chapter 3: Coarsening dynamics in the XXZ model.**

Using finite-size MPS, DMRG, and TDVP, we investigate whether coarsening-like domain growth can emerge following a quantum quench in the XXZ model.

- **Chapter 4: Solving PDEs using the Quantized Tensor Train (QTT).**

We apply MPS-based algorithms to two-dimensional PDEs, including the harmonic oscillator and the Gross–Pitaevskii equation, using a QTT representation. A particular focus is placed on polar coordinates, where we propose a regularization method to handle the singularity of the Laplacian at the origin.

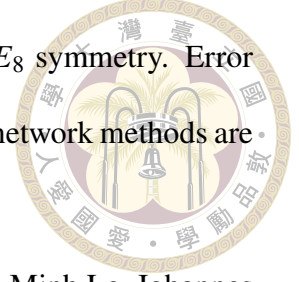
- **Chapter 5: Quantum circuits for imaginary-time evolution.**

We develop parameterized quantum circuits inspired by MPS and apply them to imaginary-time evolution of the Ising model. Using circuit-based TDVP and quantum Lanczos optimization, we approximate ground states and analyze the statistical distribution of energy values obtained from quantum circuit simulations.

- **Chapter 6: Real-time quantum simulation of emergent E_8 symmetry.**

We perform real-time simulations of the transverse-field Ising model with a lon-

itudinal field on IBM quantum devices to explore emergent E_8 symmetry. Error mitigation and circuit compression techniques based on tensor-network methods are applied to observe the characteristic E_8 spectral peaks.



*This work is available as a preprint: **Hao-Ti Hung**, Isabel Nha Minh Le, Johannes Knolle, and Ying-Jer Kao, *Improved Ising Meson Spectroscopy Simulation on a Noisy Digital Quantum Device*, arXiv:2512.02516.*

Each of these topics explores a distinct application, yet they are all unified through the use of time-dependent simulation techniques, TDVP, and the interplay between tensor networks and quantum circuits. This dissertation highlights both the power of these methods for addressing fundamental problems in quantum simulation and numerical computation.



Chapter 2 Dynamical Quantum Phase Transition of the Thirring and XXZ Models

Out-of-equilibrium dynamics is one of the most challenging problems in the study of quantum many-body systems. This funnels the interest towards finding universal behaviors that allow a more comprehensive understanding of the dynamics. A paradigmatic scenario to investigate these questions is that of a quantum quench [2], in which a system is evolved with a Hamiltonian of interest, after having been initialized in a certain (often pure) state. Since the mid-2010s, an active direction of research in this context has been the investigation of dynamical quantum phase transitions (DQPTs) [3,4], which appear as zeros in the Loschmidt echo or return probability, i.e., the probability that the system is found in the initial state during the evolution.

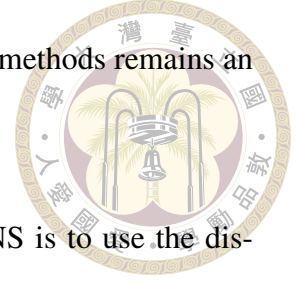
One of the problems for studying out-of-equilibrium dynamics of such complex systems is the scarcity of appropriate tools. Only in certain cases is an analytical solution possible that allows a complete description of the time evolution after a quench [3, 5, 6]. For the most general cases, however, the only possibility is to obtain an approximation

to the dynamics by means of numerical simulations, within the regimes where these are feasible. A relevant role among numerical methods is played by tensor network state (TNS) techniques [7–12]. In particular, in one-dimensional systems, matrix product state (MPS) [13–17] algorithms provide an effective way to explore the time-dependent properties of the Loschmidt echo for systems in the thermodynamic limit [4, 18]. Although the potentially fast entanglement growth [19–21] limits the time that can be reliably simulated, TNS dynamical methods [22–24] enable an accurate picture of the quantity of interest for a moderate time after the quench, which has been exploited to obtain important results regarding DQPTs [18, 25–31].

Signatures of DQPT have been observed experimentally in trapped ion quantum simulators [32] and superconducting qubits [33]. While initial studies suggested that DQPTs occur when the parameter of the Hamiltonian is quenched across an equilibrium phase boundary [3, 4, 25], it was later realized that this is not the general case [18, 34–37], and the nature of DQPTs is more complex and far from being completely understood [30, 38].

The study of out-of-equilibrium physics for quantum field theories (QFTs), defined in the continuum, is highly nontrivial. Equilibrium properties of such models are usually addressed by the path integral formalism and Monte Carlo sampling, which, however, encounters a sign problem in certain scenarios. This limitation has motivated an investigation of applicability of TNS methods to the realm of QFT. In particular, gauge theories in 1+1 dimensions have been systematically addressed using TNS methods (see, e.g., Refs. [39–41] and references therein). Other QFTs and conformal field theories have also been studied in the same manner [42–45]. These investigations have focused on the equilibrium properties of the field theories. Real-time simulations are more scarce [46–50],

and studying the real-time dynamics of a continuum QFT with these methods remains an open problem.



The most straightforward way to study QFT problems with TNS is to use the discretized version of the model Hamiltonian and apply standard TNS methods to find their equilibrium or dynamical properties. Such results depend on different lattice discretization schemes, but by repeating the procedure at different lattice spacings and performing the proper renormalization, it is possible to recover the continuum limit (see, e.g., Refs. [46,51]). In this work, we focus on one such spin chain, corresponding to the discretization of the 1+1- dimensional Thirring model. The resulting spin model is a specialized XXZ Hamiltonian coupled to both staggered and homogeneous magnetic fields. The ground state of the model, as determined in previous work [42] using MPS, exhibits a critical and a gapped phase, separated by a Berezinskii-Kosterlitz-Thoules (BKT) phase transition (see Fig. 2.1). For vanishing bare fermion mass, the model is critical for all couplings.

In this work, we use uniform MPS to study the phenomenology of DQPTs in the spin model for the discretized Thirring model and also XXZ model. We simulate quenches in both phases from initial states with very different properties. We show that a necessary condition for a DQPT to occur within the finite-time window that our numerical method can access is that the energy density of the initial state is above a certain threshold, independent of the phases of the initial and quenched Hamiltonian. However, in the case of the quench to the gapped phase we also identify a connection of DQPTs to the equilibrium phase diagram but at finite temperature. Namely, at the temperature corresponding to the identified energy density threshold, the properties of the thermal equilibrium state undergo a substantial change, with the vanishing of a string correlator that is nonzero in

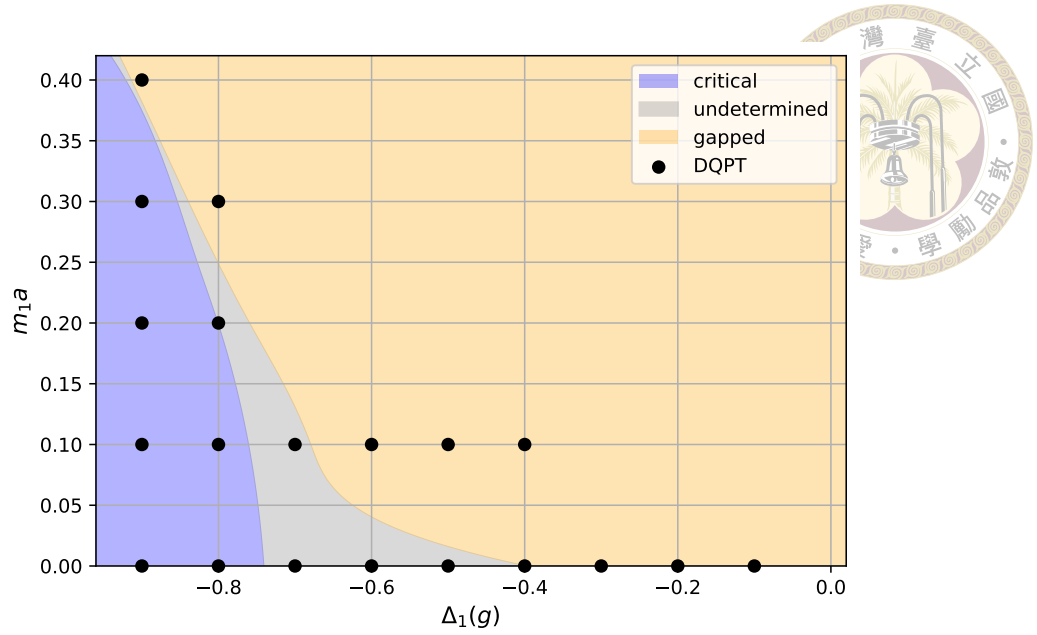
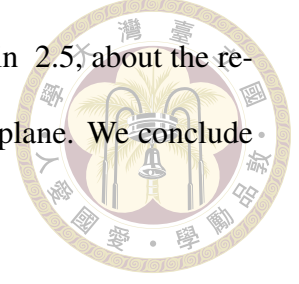


Figure 2.1: Zero-temperature phase diagram of the Thirring model in the coupling-mass plane. The color-coded areas on the plot depict the equilibrium phase through correlator analyses [42]. The blue region is the critical region, where the constant C [defined in Eq. (2.10)] is less than 0.001; the orange region is the gapped phase region, where $C > 0.01$; the gray region is undetermined, where the corresponding $C \in [0.001, 0.01]$. The black point marks the location where we observe the DQPT, starting from the ground state of the Hamiltonian $H_1(m_1a, \Delta_1)$ and quenching to $H_2(m_2a = 0.1, \Delta_2 = 0.9)$.

the ground state. We further show that the times at which DQPTs occur as a function of the energy density of the initial state show a structure reminiscent of the complex zeros of the Loschmidt amplitude [52, 53] and several branches of DQPTs that can be explored from different initial states.

The rest of this chapter is structured as follows. In Sec. 2.1 we introduce the spin model that corresponds to the discretized Thirring Hamiltonian and present the MPS formulation, MPO with penalty term, and the tools and algorithms including VUMPS and TDVP, we have used. Second, we review the equilibrium phase structure of the Thirring model from the previous research in Sec. 2.2. In Sec. 2.3, we discuss the DQPT concept and the use of MPS methods to study them in the model of interest. Our results for DQPTs in different quenches, and the relation of their appearance with the finite temperature phase

diagram are discussed in Sec. 2.4. Finally, we have more discussion in 2.5; about the relation between DQPTs, Fisher zeros and effective-temperature-time plane. We conclude our findings and discuss potential further investigations in Sec. 2.6.



2.1 Formalism

2.1.1 The Thirring Model as a Spin Model

In this work we focus on the spin chain formulation corresponding to the lattice discretization of the massive Thirring model in 1+1 dimensions. The action of the original field theory is described by

$$S_{\text{Th}} [\psi, \bar{\psi}] = \int d^2x \left[\bar{\psi} i \gamma^\mu \partial_\mu \psi - m \bar{\psi} \psi - \frac{g}{2} (\bar{\psi} \gamma_\mu \psi) (\bar{\psi} \gamma^\mu \psi) \right],$$

where m is the fermion mass and g is the four-fermion coupling. To facilitate the use of MPS, quantization of the classical theory described by Eq. (2.1) is performed through the canonical method in the Hamiltonian formalism. One subtlety in this formalism is the inclusion of effects of the anomaly in the four-fermion operator. These effects can be easily investigated through the examination of the path-integral measure in the Lagrangian formulation [54]. In the canonical procedure, they need to be accounted for via a nonlocal definition of the currents in the four-fermion operator [55, 56].

The model can be discretized on a one-dimensional spatial lattice using the staggered regularization. For convenience of carrying out numerical computations in this work, the fermionic degrees of freedom in this Hamiltonian are mapped onto spin op-

erators through the Jordan-Wigner transformation. Details of the above procedure can be found in Ref. [42]. Implementation of this strategy turns the Hamiltonian operator of the continuum Thirring model, H_{Th} , into that of the XXZ spin chain coupled to both uniform and staggered magnetic fields, up to a scaling factor,

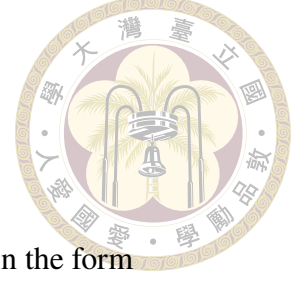
$$H(\tilde{m}_0 a, \Delta) = -\frac{1}{2} \sum_n^{N-2} (S_n^+ S_{n+1}^- + S_{n+1}^+ S_n^-) + a\tilde{m}_0 \sum_n^{N-1} (-1)^n \left(S_n^z + \frac{1}{2} \right) + \Delta(g) \sum_n^{N-1} \left(S_n^z + \frac{1}{2} \right) \left(S_{n+1}^z + \frac{1}{2} \right), \quad (2.1)$$

where a is the lattice spacing, N is the total number of lattice sites, $S_n^\pm = S_n^x \pm iS_n^y$ and S_n^z are the spin matrices ($S_n^i = \sigma^i/2$ with σ^i being the Pauli matrices) at the n th site, and $[S_n^i, S_m^j]_{n \neq m} = 0$. Here $\tilde{m}_0 = m_0/\nu(g)$ with m_0 being the bare counterpart of the mass parameter, m , in Eq. (2.1). The functions $\nu(g)$ and $\Delta(g)$ are the lattice version of wave-function renormalization and the four-fermion coupling [57],

$$\nu(g) = \left(\frac{\pi - g}{\pi} \right) / \sin \left(\frac{\pi - g}{2} \right), \quad \Delta(g) = \cos \left(\frac{\pi - g}{2} \right).$$

Notice that the physical Hamiltonian is obtained by rescaling the dimensionless one in Eq. (2.1) by a factor $\nu(g)/a$. Throughout the rest of the paper, we will work with the lattice Hamiltonian (2.1) and correspondingly define dimensionless time and inverse-temperature parameters, which involve rescaling the physical (dimensionful) magnitudes by $\nu(g)/a$. Additionally, we restrict our consideration to the sector with total spin (magnetization) zero [42] and find the ground state by adding the penalty term (see 2.1.3).

2.1.2 Matrix Product State



A generic quantum state for a system of N spins can be written in the form

$$|\Psi_N\rangle = \sum_{\sigma_1, \dots, \sigma_N} c_{\sigma_1 \dots \sigma_N} |\sigma_1 \dots \sigma_N\rangle, \quad (2.2)$$

which can then be approximated as a MPS with bond dimension D ,

$$|\Psi_N\rangle = \sum_{\sigma_1, \dots, \sigma_N} \text{tr} [M_1^{\sigma_1} M_2^{\sigma_2} \dots M_N^{\sigma_N}] |\sigma_1 \dots \sigma_N\rangle, \quad (2.3)$$

where $M_i^{\sigma_i}$ is a $D \times D$ matrix. For the open boundary condition, $M_1^{\sigma_1}$ and $M_N^{\sigma_N}$ become D -dimensional vectors. In this case, the trace in Eq. (2.3) becomes a simple product of matrices.

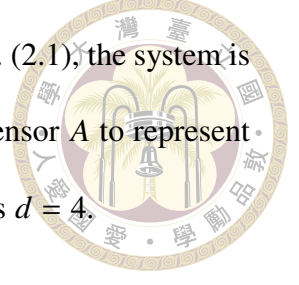
We search for DQPTs in the real-time evolution of Eq. (2.1). For this purpose, it is desirable to perform calculations directly in the thermodynamic limit. This can be accomplished by employing the uniform MPS (uMPS) technique, which allows for the description of the state of an infinite spin chain with translation invariance using one bulk tensor and two appropriate boundary tensors [58, 59]. The uniform MPS can be represented as

$$|\Psi(A)\rangle = \sum_{\sigma} (\dots A^{\sigma_{n-1}} A^{\sigma_n} A^{\sigma_{n+1}} A^{\sigma_{n+2}} \dots) |\sigma\rangle, \quad (2.4)$$

where $A^{\sigma_n} \in \mathbb{C}^{D \times D}$ for $\sigma = 1, \dots, d$. It can be represented diagrammatically as

$$\dots \text{---} \boxed{A}^{\sigma_{n-1}} \text{---} \boxed{A}^{\sigma_n} \text{---} \boxed{A}^{\sigma_{n+1}} \text{---} \boxed{A}^{\sigma_{n+2}} \text{---} \dots, \quad (2.5)$$

where A is the unit cell of the uMPS. Due to the staggered term in Eq. (2.1), the system is translationally invariant with a period of two. Thus, we choose the tensor A to represent two contiguous sites of the chain, and its physical dimension becomes $d = 4$.



2.1.3 Penalty Term and Matrix Product Operator

Since the z component of the total spin corresponds to the total fermion number in the Thirring model, and we are only interested in the zero-charge sector. To enforce this, we introduce a penalty term in the Hamiltonian,

$$\tilde{H}^{\text{penalty}} = H + \lambda \left(\sum_{n=0}^{N-1} S_n^z \right)^2. \quad (2.6)$$

Here λ should be sufficiently large (100 in this work) to ensure that the ground state obtained *via* a variational search is in the sector of vanishing total S^z [60], that is, $\langle S_{tot}^z \rangle = \langle \sum_n S_n^z \rangle = 0$. This enables us to interpret our results in terms of the dual SG theory and the XY model. However, the penalty term will not be turned on during the real-time evolution simulations since $\langle S_{tot}^z \rangle$ is a conserved quantity. In our simulations, this quantity is well preserved, i.e., $\langle S_{tot}^z(t) \rangle = 0$.

In TN algorithms, it is useful to express the Hamiltonian as the MPO. The Hamilto-

nian of Thirring model in Eq. (2.6) can be expressed as an MPO:

$$W^{[n]} = \begin{pmatrix} \mathbb{I} & -\frac{1}{2}S^+ & -\frac{1}{2}S^- & 2\lambda S^z & \Delta S^z & \beta_n S^z + \alpha \mathbb{I} \\ 0 & 0 & 0 & 0 & 0 & S^- \\ 0 & 0 & 0 & 0 & 0 & S^+ \\ 0 & 0 & 0 & \mathbb{I} & 0 & S^z \\ 0 & 0 & 0 & 0 & 0 & S^z \\ 0 & 0 & 0 & 0 & 0 & \mathbb{I} \end{pmatrix}. \quad (2.7)$$



where $\beta_n = \Delta + (-1)^n \tilde{m}_0 a$, $\alpha = \frac{1}{4}(\lambda + \Delta)$. The β_n contains the staggered term. To simplify the simulation, we also combine two-site MPOs into one site, as we did with the uMPS, represented as $W_1 W_2 \rightarrow W$.

2.1.4 Brief Introduction of VUMPS and TDVP

To prepare the ground state in uMPS form, we use the VUMPS algorithm [59]. In VUMPS, we need to construct the effective Hamiltonian through the MPO and the boundary tensors L, R , which satisfy the fixed-point equations (see Fig. 2.2). The tensor diagram

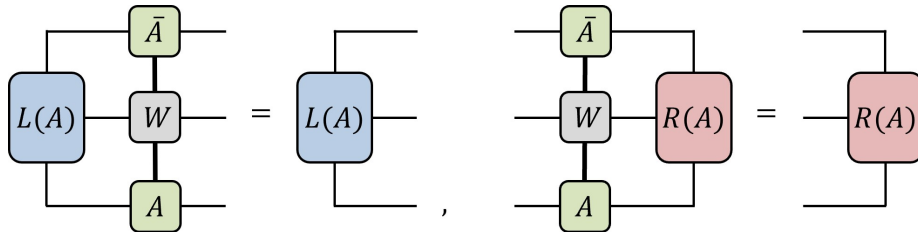
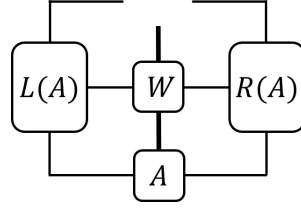


Figure 2.2: The fixed-point equations of the boundary tensors L and R .

of the effective Hamiltonian operating on the unit cell can be expressed as:



(2.8)



The boundary tensors L and R are functions of A . We could use the Lanczos algorithm to obtain the optimal A and then update L and R until it reach the ground state. In our simulation, we turned on the penalty term λ in Eq. (2.7) to guarantee that the ground state we obtained is in the sector of vanishing total $\langle S_z \rangle$.

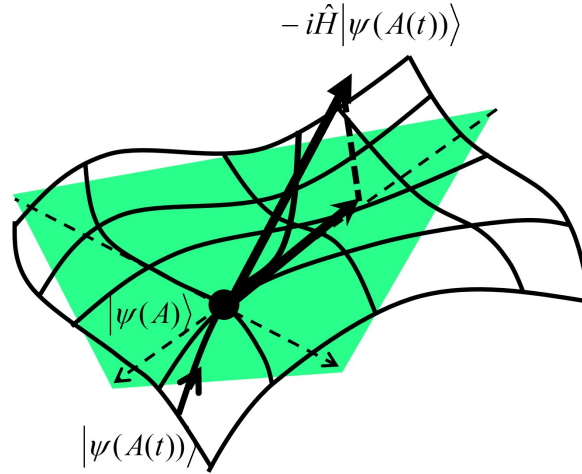


Figure 2.3: An illustration of the TDVP. The curve is the manifold of the uMPS and the colored plane is the tangent space of the current state.

For real-time evolution, we choose the TDVP algorithm, which preserves conserved quantities well [61]. TDVP corresponds to a projection of the evolution state of the Schrödinger equation onto the tangent space of the uMPS manifold $\mathcal{M}$ at the current state (see Fig. 2.3).

We can solve the time derivative of the uMPS $|\Psi(A)\rangle$ through the TDVP equation:

$$\frac{\partial}{\partial t} |\Psi(A)\rangle = -i\hat{P}_{T_{|\Psi(A)\rangle}\mathcal{M}} \hat{H} |\Psi(A)\rangle. \quad (2.9)$$

The left-hand side of Eq. (2.9) gives the time derivative of the tensor A , while the right-hand side gives the projection result: $-iB$, where B is also a tensor that has the same shape as tensor A . Thus, we will get $\dot{A} = -iB$, which can be solved using the Runge-Kutta (RK4) method. Note that for real-time evolution, we turn off the penalty term λ in Eq. (2.7) (setting $\lambda = 0$).

2.2 Equilibrium Phase Structure of the 1+1-dimensional Thirring Model

In this section, we review our study of the zero-temperature phase structure of the Thirring model using MPS [42]. An analysis of renormalization group (RG) equations of the dual sine-Gordon theory [62–64] points to the existence of two phases in the Thirring model. The massless theory is a conformal field theory at all couplings $\Delta(g)$. RG flows in the mass-coupling plane reveal that the line $m = 0$ is a fixed line under RG transformations. However, it is stable/unstable for couplings $\Delta(g)$ below/above certain $\Delta(g_\star)_{m=0} = -\pi/2$. At $m > 0$, the nonzero fermion mass is an irrelevant operator for $\Delta(g) < \Delta(g_\star)_m$ but becomes relevant when $\Delta(g) > \Delta(g_\star)_m$, with the transition coupling $\Delta(g_\star)_m$ being mass dependent. This gives rise to two equilibrium phases:

- (i) $\Delta(g) < \Delta(g_\star)_m$: gapless (critical) phase
- (ii) $\Delta(g) > \Delta(g_\star)_m$: gapped (massive) phase,

separated by a BKT transition. The BKT transition was originally discussed in the context of the XY model [65, 66], which is actually dual to the sine-Gordon and Thirring theories, with the XY model temperature dual to the couplings of these theories. For a more exten-

sive discussion of these aspects, including von Neumann entanglement entropy and chiral condensate, we refer to Ref. [42].



The quantitatively most robust analysis of the phase structure and the determination of $\Delta(g_\star)_m$ is about the string correlator,

$$C_{\text{string}}(x) = \frac{1}{N_x} \sum_n \langle S^z(n) S^z(n+1) \cdots S^z(n+x-1) S^z(n+x) \rangle,$$

which is a string of S^z operators. We note that Ref. [42] erroneously states this string correlator as the fermion-antifermion one, containing $\langle S^+(n) S^z(n+1) \cdots S^z(n+x-1) S^-(n+x) \rangle$ terms. At large-enough distances, the correlator $C_{\text{string}}(x)$ exhibits a power-law decay in the gapless phase, given by $C_{\text{string}}(x) \propto \beta x^{-\alpha}$, and a power-exponential decay in the gapped phase, $C_{\text{string}}(x) \propto B x^{-\eta} A^x$. The crucial distinction that makes the string correlator particularly useful for determining $\Delta(g_\star)_m$ is that it decays to a constant, C , in the gapped phase. In practice, one can analyze both phases based on the power-exponential fitting ansatz:

$$C_{\text{string}}(x) = B x^{-\eta} A^x + C, \quad (2.10)$$

with $A = 1$ found in fits in the gapless phase. Thus, the phase transition when going from the gapless into the gapped phase is manifested by A dropping below 1 and, in the case of the string correlator, by $C > 0$. The onset of the nonzero constant is numerically more robust than the smooth drop of A below 1, accounting for the practical usefulness of the string correlator. As we will see below, this is true also at nonzero temperatures and it will allow us to identify relations between thermal phases of the Thirring model and DQPTs.

As a summary of our findings for the zero-temperature phase structure, we show

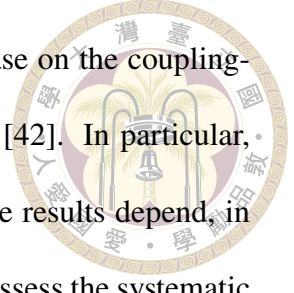


Fig. 2.1. The color-coded regions correspond to the equilibrium phase on the coupling-mass plane through the correlator analyses from our previous work [42]. In particular, we fitted Eq. (2.10) and extracted values of the constant C . Since the results depend, in general, on the fitting interval, we adopted a systematic procedure to assess the systematic uncertainty of C (dominating with respect to other uncertainties, coming from the finite bond dimension and the finite volume). For details of this procedure, we refer again to Ref. [42]. Depending on the value of C , we identified three situations. With $C < 0.001$ or $C > 0.01$, one unambiguously concludes either the critical phase (purple region) or the gapped phase (orange region), respectively. If $C \in [0.001, 0.01]$ (gray region), then the systematic uncertainty does not allow us to draw conclusions about a (non)vanishing value of C . This region of ambiguous values of C is indicated by gray shading and the BKT transition must occur there. We note the onset a nonzero value of C is very smooth at small fermion masses, while it becomes rather sharp with an increasing mass. At very small masses, the implied transition point is consistent with the RG flow analysis [$\Delta(g) \approx -\pi/2$] and at large masses, it moves towards more negative values.

2.3 The Loschmidt Echo and the Mixed Transfer Matrix

2.3.1 Quantum Quench and DQPT

In this work, we investigate DQPT in a quantum quench. A quantum quench is the paradigmatic scenario where to study nonequilibrium dynamics of a quantum system. The system is assumed to be prepared in a certain initial state $|\Psi_1\rangle$, typically the ground state of a Hamiltonian H_1 . From time $t = 0$ on, it is evolved with a different Hamiltonian H_2 .

The evolved state can be written as $|\Psi(t)\rangle = e^{-itH_2} |\Psi_1\rangle$. The key ingredient in the study of DQPT is the Loschmidt amplitude,

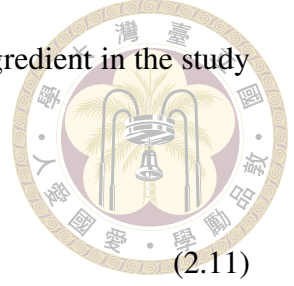
$$G(t) = \langle \Psi_1 | \Psi(t) \rangle = \langle \Psi_1 | e^{-iH_2 t} | \Psi_1 \rangle, \quad (2.11)$$

which is analogous to the boundary partition function [67] $Z_B(z) = \langle \Psi_1 | e^{-zH_2} | \Psi_1 \rangle$ in the complex plane $z \in \mathbb{C}$. The nonanalytical points of the free energy density $f(z) = -\lim_{N \rightarrow \infty} \frac{1}{N} \log Z_B(z)$ are believed to indicate phase transitions [68]. The squared modulus of Loschmidt amplitude in Eq. (2.11), the Loschmidt echo, quantifies the return probability, i.e., the probability of the system being found in the initial state after a certain time t . In the thermodynamic limit ($N \rightarrow \infty$), the Loschmidt echo will approach $e^{-Nr(t)}$, with the rate function

$$r(t) = -\lim_{N \rightarrow \infty} \frac{1}{N} \log |G(t)|^2. \quad (2.12)$$

This function can be interpreted as a dynamical free energy, and can exhibit nonanalyticities. DQPTs are defined as the times when $r(t)$ is nonanalytical [3]. In some cases, DQPTs are associated with quenches across an equilibrium phase transition [3, 69], but such connection to equilibrium is not always present [34], which motivates the identification of DQPTs as a genuinely nonequilibrium effect. The potential observable signature of DQPTs in local and string quantities has also been a subject of intensive studies. [70, 71] The complex phenomenology of DQPTs has motivated studies in multiple models, but the phenomenon is yet far from a full theoretical understanding [4, 18, 28, 72, 73].

The transfer matrix formalism allows for a systematic exploration of the character of DQPTs [18, 26–28, 30]. We focus here on the case where the initial state is a MPS and



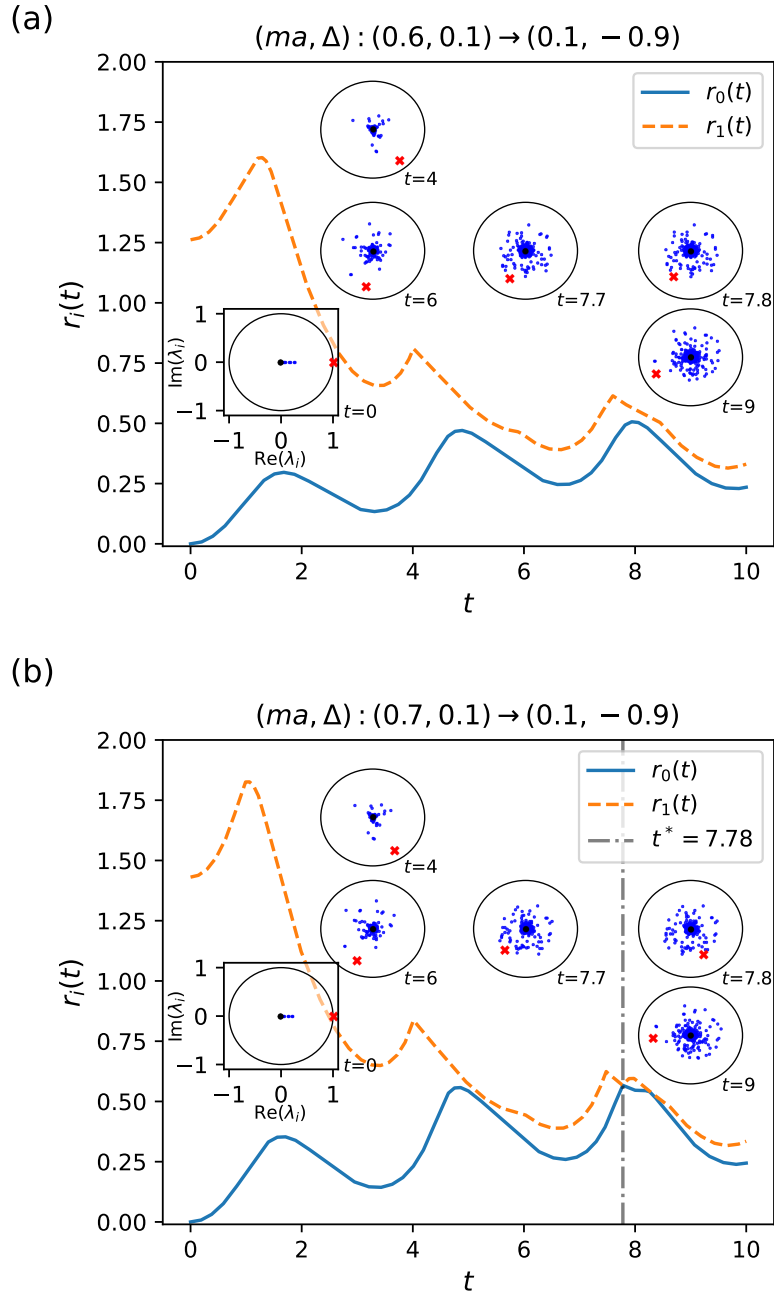
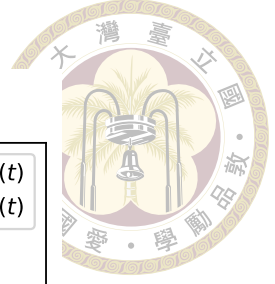


Figure 2.4: The return rate function and the full spectrum of the mixed transfer matrix $T(t)$. The quantity r_i is defined in Eq. (2.16), while r_0 corresponds to the return rate function defined in Eq. (2.12). The insets display the full spectrum of the mixed transfer matrix $T(t)$ defined in Eq. (2.15), with the red cross representing the corresponding dominant eigenvalue. The time t^* (indicated by the gray dash-dotted line) denotes when the first DQPT occurs. This plot shows how the nonanalyticities in the return rate $r_0(t)$ correspond to discontinuous jumps in the phase of the dominant eigenvalue of $T(t)$ (red point). We can notice that in (a), with $(m_1a, \Delta_1) = (0.6, 0.1)$ and $(m_2a, \Delta_2) = (0.1, -0.9)$, there are no DQPTs. However, in (b), a small increase in m_1a to 0.7, with the same quench Hamiltonian, induces a DQPT at $t^* = 7.78$.

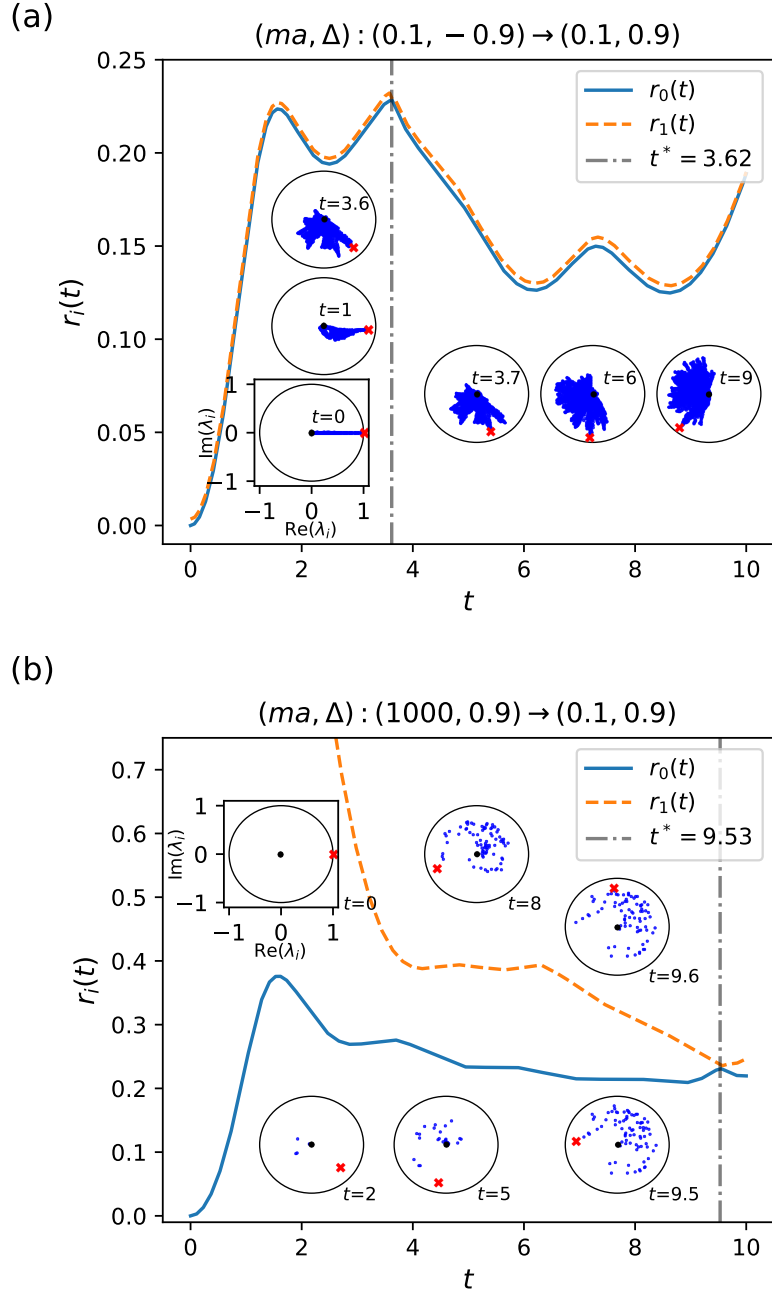


Figure 2.5: Same as Fig. 2.5, but for quenches with $(m_2a, \Delta_2) = (0.1, 0.9)$. Panel (a), with $(m_1a, \Delta_1) = (0.1, -0.9)$, exhibits a DQPT at $t^* = 3.62$, while panel (b), corresponding to $(m_1a, \Delta_1) = (1000, 0.9)$, shows a later DQPT at $t^* = 9.53$.

consider directly the translationally invariant case, i.e., the initial state is parametrized by a single rank-3 tensor A . The Loschmidt amplitude, for a translationally invariant system of N sites, can then be expressed as

$$G(t) = \langle l | T^N(t) | r \rangle = \lambda_0^N, \quad (2.13)$$

where $T(t)$ is the mixed transfer matrix defined by

$$T(t) := \sum_i \bar{A}(0)^\sigma \otimes A(t)^\sigma, \quad (2.14)$$

and $A(t)$ represents the tensor describing the evolved state. The vectors $|l\rangle$ and $|r\rangle$ are the left and right eigenvectors with the leading eigenvalue λ_0 , satisfying $\langle l | r \rangle = 1$. Graphically,

$$T(t) = \begin{array}{c} \text{---} \boxed{\bar{A}(0)} \text{---} \\ | \\ \text{---} \boxed{A(t)} \text{---} \end{array}. \quad (2.15)$$

The Loschmidt rate is thus determined by the spectrum of the (in general non-Hermitian) matrix $T(t)$ and, in the thermodynamic limit, by its dominant eigenvalue $\lambda_0(t)$. DQPTs appear when there is an eigenvalue crossing and the eigenvector associated to the dominant eigenvalue changes [18, 74].

2.3.2 Investigation of DQPT in the Lattice Version of the Thirring Model

Here we focus on the Hamiltonian Eq. (2.1) and consider quenches from the ground state $|\Psi_1\rangle$ of the model with a given set of parameters, $H_1 = H(m_1 a, \Delta_1)$. The quench

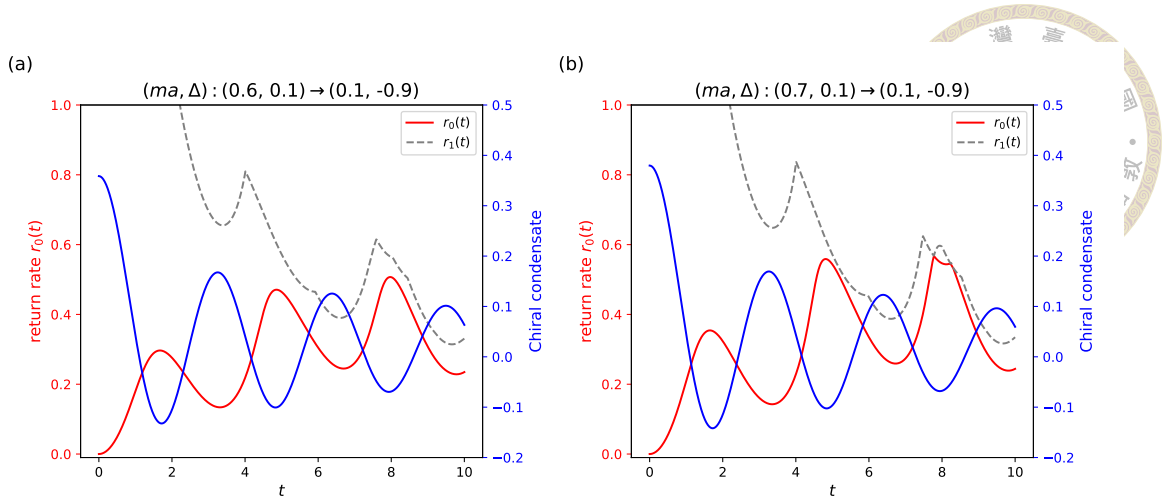


Figure 2.6: The return rate functions (red) and the chiral condensate (blue) for the cases (a) without and (b) with DQPTs. The parameters we choose here are the same as those in Figs. 2.5(a) and 2.5(b).

Hamiltonian, with which the state is evolved, will be denoted as $H_2 = H(m_2a, \Delta_2)$. For a fixed H_2 , we study the real-time evolution of different initial states, $|\Psi_1\rangle$. The latter are represented by their MPS approximation, obtained using the variational uMPS (VUMPS) algorithm [59], and the real-time evolution according to H_2 can be numerically simulated using a standard tensor network algorithm [8, 12, 75]. In particular, we use time-dependent variational principle (TDVP) algorithm [22, 61] to obtain the time-dependent tensors of the MPS representation, $A(t)$. We obtain the spectrum of the mixed transfer matrix, $T(t)$, using Eq. (2.14) and explore the appearance of DQPTs for various quench parameters. To investigate how DQPTs occur, we introduce the following quantity:

$$r_i(t) = -\log |\lambda_i(t)|^2, \quad (2.16)$$

where $\{\lambda_i\}$ is the spectrum of the mixed transfer matrix $T(t)$, satisfying $|\lambda_0| \geq |\lambda_1| \geq \dots$. Notice that r_0 is exactly return rate function defined in Eq. (2.12). Figure 2.5 and 2.4 shows the results for a selection of simulation parameters. The first DQPT occurs when $t = t^*$. The insets show snapshots of the spectrum of $T(t)$, which demonstrate that the nonana-

lytical points in the return rate function in Eq. (2.12) are associated with a discontinuous jump in the phase of the dominant eigenvalue of this mixed transfer matrix [74].

By choosing parameters, $(m_1 a, \Delta_1)$ and $(m_2 a, \Delta_2)$, to be in a different or in the same equilibrium phase, we can simulate quenches across the phase diagram, which we determined in our previous study [42]. We find that, as observed in other models [18, 34], the equilibrium phases of H_1 and H_2 does not determine the presence of DQPTs, which can appear in both types of scenario. In Fig. 2.1, we illustrate that the existence of DQPTs is not equivalent to the equilibrium phase. DQPTs have been demonstrated to occur in quenches that happen entirely within either of the equilibrium phases.

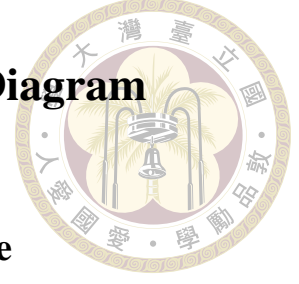
2.3.3 DQPT and Thermalization

Notice that DQPTs and (pre-)thermalization happen at different timescales. In our setup, DQPTs can occur at relatively short times. Hence, we do not expect that local observables have equilibrated yet. Indeed, we observe that local observables are still varying strongly at the time slices when the DQPTs are observed. To illustrate this, we plot in Fig. 2.6 the time evolution of the chiral condensate (or the staggered magnetization in spin model language),

$$\chi \sim \frac{1}{N} \left| \sum_n (-1)^n \left(S_n^z + \frac{1}{2} \right) \right|,$$

in both the cases with or without DQPT. The chiral condensate shows strong time variation at the longest time allowed for our simulation, $t = 10$, in both cases, indicating DQPT and prethermalization or thermalization phenomena occur at different timescales.

2.4 DQPTs and Finite-temperature Phase Diagram



2.4.1 Energy Threshold and the Inverse Temperature

As discussed in the previous section, the appearance of a DQPT after a certain quench is not determined by the fact that the quench crossed an equilibrium phase boundary. Hence the ground state of H_2 is not the most adequate equilibrium reference for all initial states. Since the mean energy of the initial state $E_1 := \langle \Psi_1 | H_2 | \Psi_1 \rangle$ is typically much higher than the ground-state energy of the quench Hamiltonian H_2 , one does not necessarily expect the time-evolved state to be close to the ground state of H_2 , even at the level of local observables. Instead, it is more meaningful to compare the local properties to those of the thermal equilibrium state at the same energy density. If the system equilibrates (in the sense of locally resembling an equilibrium ensemble), then the thermal state should correspond to the maximal entropy state compatible with the conserved quantities¹. Our particular model conserves energy and total magnetization, and we are considering initial states in which the latter vanishes. Thus, the relevant equilibrium state is the Gibbs state with the same energy E_1 , restricted to the subspace of vanishing total magnetization; namely (and up to normalization) $\rho(\beta) \propto P_{s_z=0} e^{-\beta H_2} P_{s_z=0}$, where $P_s = \sum_{n, S_z | n \rangle = s | n \rangle} | n \rangle \langle n |$ is the projector onto the sector of total magnetization s . The inverse temperature $\beta_1 := \beta(E_1)$ corresponding to the state is fixed by the condition

$$E_1 = E(\beta_1) := \frac{\text{tr}(\rho_{\beta_1} H_2)}{\text{tr}(\rho_{\beta_1})}. \quad (2.17)$$

¹A detailed investigation of the equilibration process is beyond the scope of this work.

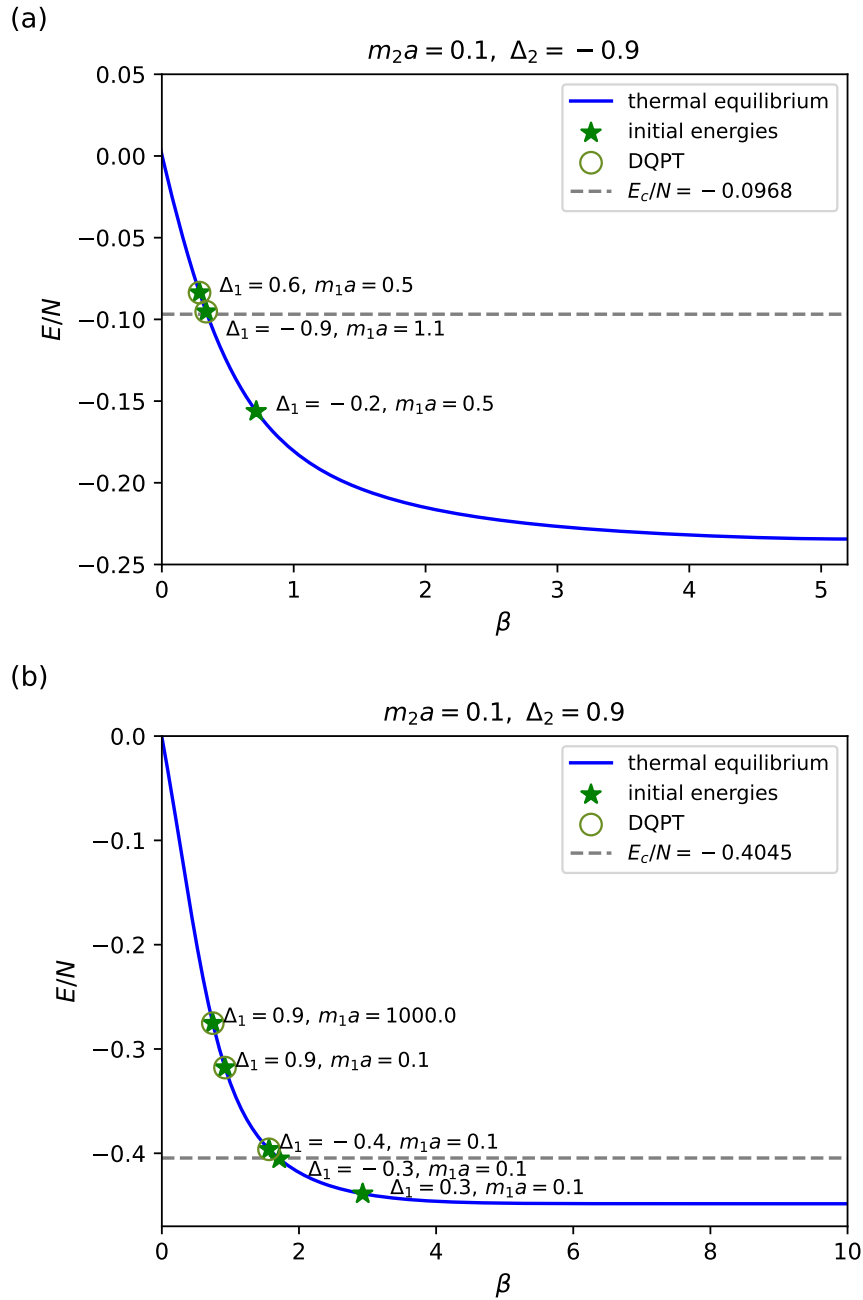
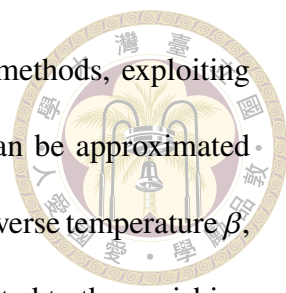


Figure 2.7: Relation $E(\beta)$ between the mean energy and the inverse temperature in the Gibbs state restricted to the zero magnetization sector for $(m_2a, \Delta_2) = (0.1, -0.9)$ (top:critical phase) and $(0.1, 0.9)$ (bottom:gapped phase). We display the energy densities of several explored quenches using star symbols, and circles indicate the initial states where our simulations reveal the occurrence of at least one DQPT. Additionally, the dashed line denotes the lowest value of energy density (E_c/N) at which a DQPT was observed (see also Fig. 2.8). For $(m_2a, \Delta_2) = (0.1, -0.9)$ (top), $E_c/N = -0.0968$ and the corresponding inverse temperature $\beta_c \approx 0.34$; for $(m_2a, \Delta_2) = (0.1, 0.9)$ (bottom), $E_c/N = -0.4045$ and $\beta_c \approx 1.70$.



The determination of β_1 can be achieved using standard TNS methods, exploiting the fact that the thermal equilibrium state of a local Hamiltonian can be approximated efficiently by an MPO [76,77]. More concretely, given a value of the inverse temperature β , the canonical purification of the thermal equilibrium ensemble restricted to the vanishing magnetization sector can be written as $|\Psi(\beta)\rangle := e^{-\beta H_2/2} \sum_{n, S_z|n\rangle=0} |n\rangle_S |n\rangle_A$, where the sum runs over a basis restricted to the sector $S_z = 0$, and the subscripts S and A refer respectively to the system and ancillary degrees of freedom. This state, also known as thermofield double state, can be approximated by a MPS using standard tensor network algorithms [7,8]. This is achieved by preparing a maximally-mixed state of the system and the ancilla, expressed as a simple MPS, where each system site is entangled only with one, next neighbor, ancillary site, and evolving it in imaginary time with an approximation of the operator $e^{-\beta H_2/2}$. In order to restrict the ensemble to a sector of fixed magnetization, we write the projector onto the sector $S_z = 0$ as a MPO and use as initial mixed state, instead of the identity. Additionally, this MPO can also be applied to the intermediate state during the evolution to ensure that truncation errors due to the approximations do not induce undesired magnetization components. For a finite system of size N , the projector onto vanishing magnetization has an exact MPO representation, with bond dimension $N/2$. Applying the aforementioned TNS evolution algorithm in imaginary time to the vectorized projector provides a MPS approximation to $|\Psi(\beta)\rangle$ for varying β , from which the thermal expectation value of a physical observable O can be computed directly as

$$\langle O \rangle_\beta := \frac{\text{tr}(O \rho_\beta)}{\text{tr} \rho_\beta} = \frac{\langle \Psi(\beta) | O \otimes I | \Psi(\beta) \rangle}{\langle \Psi(\beta) | \Psi(\beta) \rangle}, \quad (2.18)$$

where the second factor in the tensor product is the identity acting on the ancillary sys-

tem. In particular, computing the energy along the imaginary time evolution provides an estimate of the function $E(\beta)$, from which we can extract the value β_1 corresponding to a given initial state by inverting it numerically.



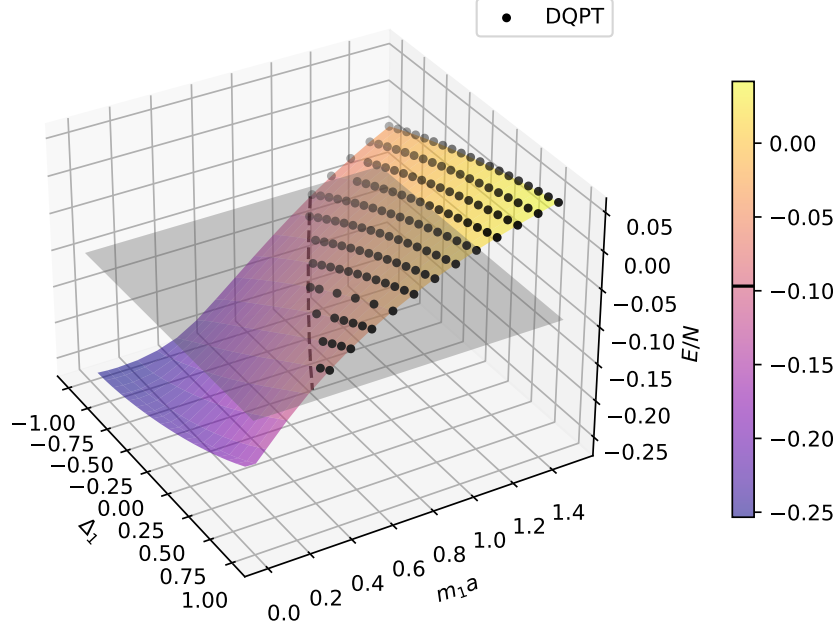
For concreteness, in the rest of the section we focus on two quenches, corresponding to Hamiltonians in each of the phases (see Fig. 2.1). Here we choose $(m_2a, \Delta_2) = (0.1, -0.9)$ in the gapless phase and $(m_2a, \Delta_2) = (0.1, 0.9)$ in the gapped one. For each of them we approximate the thermal equilibrium state in a system of $N = 300$ sites, sufficiently large to neglect finite-size effects in this analysis. The corresponding $E(\beta)$ curves are shown in Fig. 2.7.

For each of the simulated initial states, we can now extract the effective temperature from these curves. The ground states of the different initial Hamiltonians H_1 correspond to initial states with different effective temperatures. Since the latter only depends on the energy E_1 , states in very distant regions of the equilibrium phase diagram may have the same temperature. As an illustration, we show the energy densities of some of the explored quenches with star symbols in Fig. 2.7. We furthermore indicate with circles the initial states for which our simulations show the appearance of (at least one) DQPT. From these results we observe that for DQPTs to be present, the effective temperature of the initial state needs to surpass a certain threshold. This seems to suggest that a decisive factor for the occurrence or not of a DQPT is the amount of excitation energy in the initial state.

To verify this observation, we estimated the value of E_1 for a grid of (m_1a, Δ_1) values, $0 < m_1a \leq 1.5$ and $-1 \leq \Delta_1 \leq 1$, and identified the regions of parameters for which the threshold (denoted as E_c) is surpassed (see Fig. 2.8). We conducted this study for

(a)

$$m_2 a = 0.1, \Delta_2 = -0.9$$



(b)

$$m_2 a = 0.1, \Delta_2 = 0.9$$

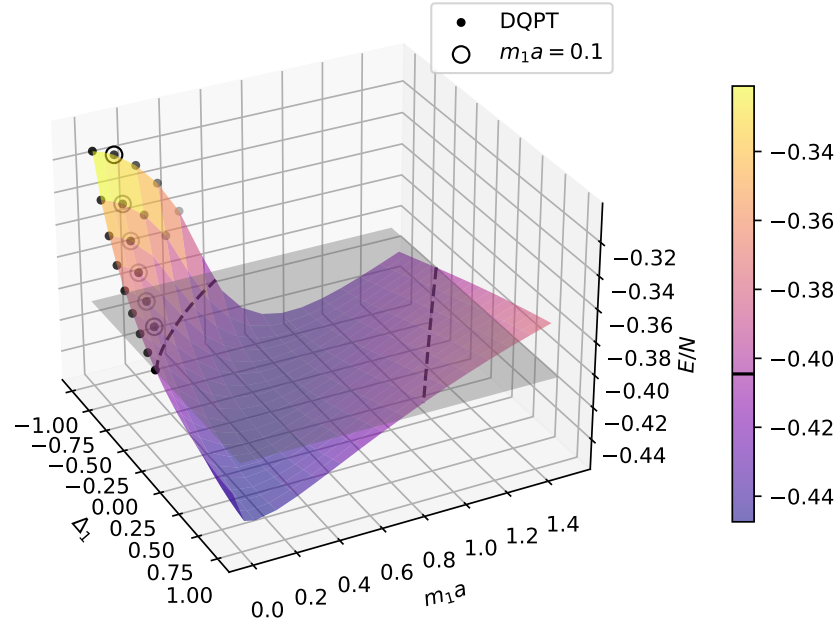


Figure 2.8: Energy densities corresponding to ground states of varying $m_1 a$ and Δ_1 , as a function of these parameters, for the quench to the critical phase ($m_2 a = 0.1$, $\Delta_2 = -0.9$) (top) and the gapped phase ($m_2 a = 0.1$, $\Delta_2 = 0.9$) (bottom). In each case, the dots indicate where DQPT has been observed. The horizontal plane shows the energy threshold, E_c/N , which is also marked as a black line on the color bars. On the right panel, the circles indicate the family of quenches from constant $m_1 a = 0.1$.

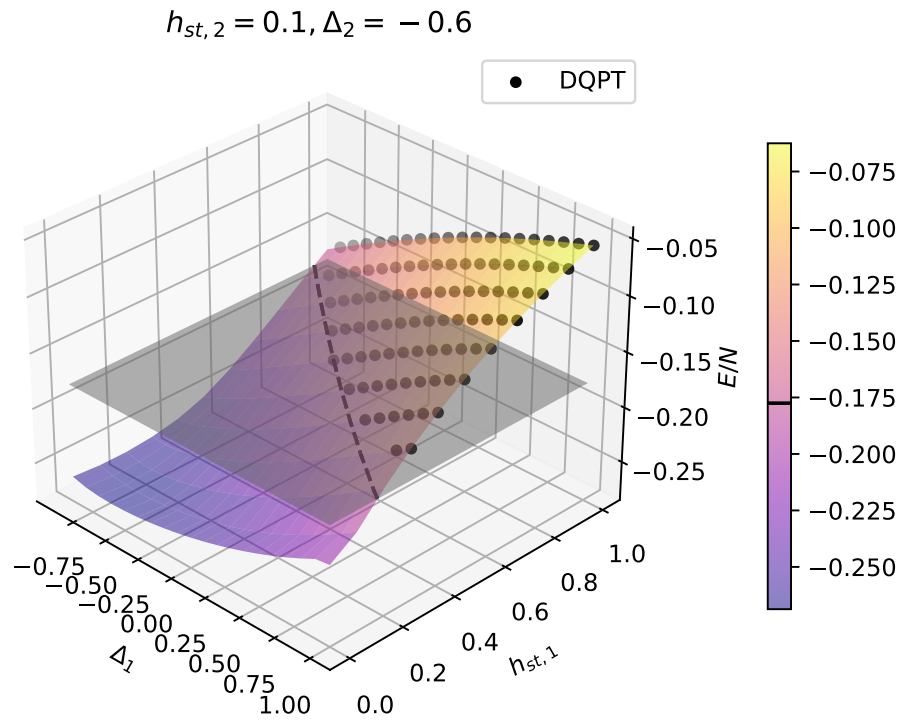
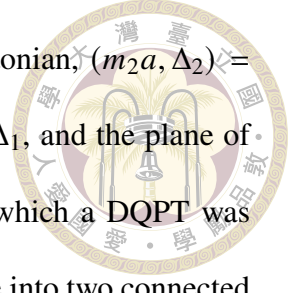


Figure 2.9: Energy landscape of the initial state as a function of the quench parameters $(h_{st,1}, \Delta_1)$ for the XXZ spin chain defined in Eq. (2.19), with the post-quench parameters fixed to $h_{st,2} = 0.1$ and $\Delta_2 = -0.6$. The plane indicates the threshold energy density, defined as the lowest value for which a dynamical quantum phase transition is observed.



both quench Hamiltonians mentioned above. For the gapless Hamiltonian, $(m_2a, \Delta_2) = (0.1, -0.9)$, the value of E_1 increases smoothly with both m_1a and Δ_1 , and the plane of constant $E_c/N \approx -0.0968$, corresponding to the lowest value for which a DQPT was observed. As indicated in Fig. 2.8(a), this divides the parameter space into two connected regions. Consistent with this, we find DQPTs for all initial states sampled from the region with higher energy. A similar observation is obtained for the XXZ spin chain defined in Eq. (2.19), where an analogous threshold energy separating initial states with and without DQPTs can be identified. The corresponding results are shown in Fig. 2.9.

In contrast, for the quench into the gapped phase, $(m_2a, \Delta_2) = (0.1, 0.9)$, the dependence of E_1 on (m_1a, Δ_1) is not monotonic. We find two disjoint regions of parameters with energies above the identified threshold $E_c/N \approx -0.4045$, lying on opposite corners of the parameter space as can be seen in Fig. 2.8(b). One region with negative Δ_1 and small m_1a (overlapping but not coinciding with the critical phase, as shown in Fig. 2.1), and the other region with positive large Δ_1 and m_1a , deep within the gapped regime. We simulate quenches from both regions, as well as intermediate parameters with energy E_1 below the threshold. For the first high-energy region, i.e., negative Δ_1 and small m_1a , we consistently observe DQPTs for energies above the threshold. The second region, with large positive Δ_1 and m_1a , however, seems to exhibit DQPTs only at much later times than the first. An extreme case is shown in Fig. 2.4(b), for the quench from $m_1a = 1000$, $\Delta_1 = 0.9$, for which we observe a DQPT only close to $t \approx 10$. We observe that as m_1a values decrease, the time at which the first DQPT is observed increases even more, so that we need to run long-time simulations to analyze this region. Due to limitations imposed by truncation errors, we can reliably explore times up to only $t \lesssim 14$. Within this window, we

only observe DQPTs for points in the parameter regime with masses outside our original grid. We presume that longer times would identify DQPTs also in the rest of the region above the threshold.



This exploration of the parameter space indicates that for states with similar effective temperature, but corresponding to very different values of $(m_1 a, \Delta_1)$, the DQPTs may occur in very different ways. In order to obtain a more complete picture, we plot in Fig. 2.10, the times at which we observe DQPTs as a function of the effective temperature of the initial state for all the simulated cases. For each of the initial states, we determine β from the energy expectation value $\langle \Psi_1 | H_2 | \Psi_1 \rangle$ through Eq. (2.17). The figure shows not only the first observed nonanalyticity (blue dots), but the times of all subsequently observed DQPTs (with colors indicating the number of the DQPT). These figures are reminiscent of the nonanalyticities shown by the complex Loschmidt rate, when defined on the complex t plane for a fixed initial state [4]. In such case, the nonanalyticities of the rate in the complex-time plane are analogous to Fisher zeros of a boundary partition function and determine the presence and time of occurrence of DQPTs [52, 53]. In our case, the initial state varies and we do not evolve it in imaginary time, but the results suggest that our effective temperature does play an equivalent role, allowing us to explore such complex plane from purely real-time data. However, we also observe that some initial states are not sensitive to some of the DQPT branches. For instance, if we select $(m_1 a, \Delta_1) = (3, 1.3)$ and quenches to $(m_2 a, \Delta_2) = (0.1, 0.9)$ [indicated by the black dashed line and red diamond marks in Fig. 2.10(b)], then the DQPT does not occur at the lowest branch near $t \approx 4$. The generality of this connection for other models thus needs further investigation.

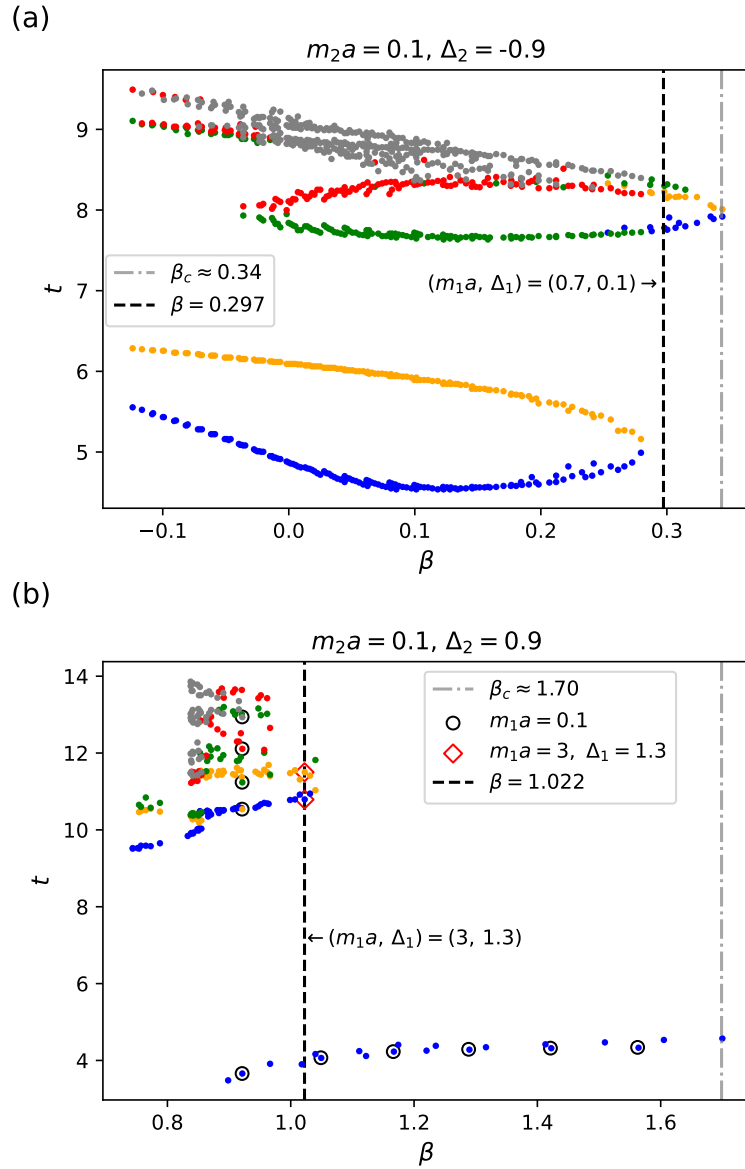
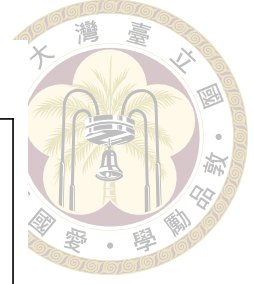
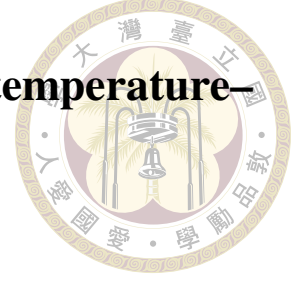


Figure 2.10: The DQPTs on the inverse temperature–time plane. The first, second, third, and fourth occurrences of DQPTs are colored blue, orange, green, and red, respectively. Gray indicates occurrences beyond the fourth. The gray dash-dotted line denotes the critical inverse temperature β_c . (a) We simulate the evolution parameters $m_2a = 0.1$ and $\Delta_2 = -0.9$ (critical phase) and perform a scan for the initial parameters $(m_1a = 0.1, 0.2, \dots, 1.5; \Delta_1 = -0.9, -0.8, \dots, 0.9)$. For example, the dashed line is the case for $m_1a = 0.7$ and $\Delta_1 = 0.1$, which is also specifically shown in Fig. 2.5(b). (b) The simulation results for the evolution parameters $m_2a = 0.1$ and $\Delta_2 = 0.9$ (gapped phase), scanning for the initial parameters as follows: $(m_1a = 0; \Delta_1 = -0.9, -0.8, \dots, -0.1, 0.9)$, $(m_1a = 0.1, 0.2, \dots, 1.5; \Delta_1 = -0.9, -0.8, \dots, 0.9)$, $(m_1a = 1, 2, \dots, 9; \Delta_1 = 1, 1.1, 1.3, 1.5, \dots, 1.9)$, and $(m_1a = 10, 20, 30, 40, 60, 80, 100, 1000; \Delta_1 = 0.9)$. These large masses are chosen to explore the large Δ_1 , large m_1a corner of Fig. 2.8(b), while ensuring that DQPT occurs within our simulation window ($t < 14$). The black dashed line and the red diamond symbols represent the case for $m_1a = 1.3$ and $\Delta_1 = 3$, where DQPT does not occur at the lowest branch near $t \approx 4$.

2.5 DQPTs, Fisher Zeros, and the Effective-temperature–Time Plane



The structure of the DQPT branches in the effective-temperature–time plane, Fig. 2.10, is strongly reminiscent of the Fisher zeros of a boundary partition function in the complex-time plane. In this section, we investigate this connection more concretely by comparing (i) real-time DQPTs, (ii) Fisher zeros obtained from independent methods, and (iii) the pattern of DQPTs in the (β, t) plane. We first benchmark the relation between DQPTs and Fisher zeros in the XXZ chain, where independent results are available, then perform a complex-time evolution study in the Thirring model, and finally analyze the analytically tractable transverse-field Ising model (TFIM) to clarify the role of the initial state and effective temperature.

2.5.1 Benchmark: Fisher Zeros and DQPTs In the XXZ Chain

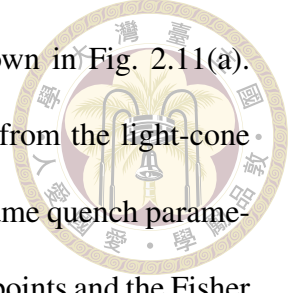
We begin with the spin- $\frac{1}{2}$ XXZ chain with a staggered longitudinal field,

$$H(\Delta, h_{\text{st}}) = J \sum_i \left[\frac{1}{2} (S_i^+ S_{i+1}^- + S_i^- S_{i+1}^+) + \Delta S_i^z S_{i+1}^z \right] + \sum_i (-1)^i h_{\text{st}} S_i^z. \quad (2.19)$$

We consider a quantum quench of the form

$$(\Delta_1, h_{\text{st},1}) = (6, 0.1) \longrightarrow (\Delta_2, h_{\text{st},2}) = (-0.6, 0.1), \quad (2.20)$$

and study the Loschmidt amplitude $G(z) = \langle \Psi_1 | e^{-izH_2} | \Psi_1 \rangle$ for complex time $z = t - i\tau$, where $|\Psi_1\rangle$ is the ground state of H_1 and, H_2 is the quench Hamiltonian.



The resulting DQPT points in the complex-time plane are shown in Fig. 2.11(a). For comparison, in Fig. 2.11(b) we show the Fisher zeros obtained from the light-cone renormalization-group transfer-matrix approach of Ref. [18] for the same quench parameters. We find excellent agreement between the locations of our DQPT points and the Fisher zeros obtained in [18].

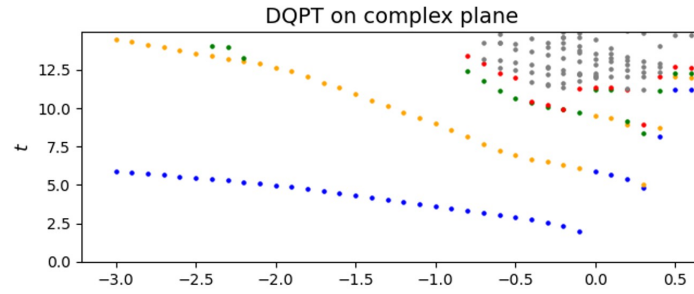
This confirms the Lee–Yang/Fisher picture in this interacting lattice model: the real-time DQPTs occur precisely when a Fisher zero of the analytically continued boundary partition function crosses the real-time axis. In other words, the DQPT points we detect via real-time simulations are consistent with being intersections of Fisher-zero lines with the real axis.

2.5.2 Complex-time Evolution in the Thirring Model

Motivated by the above XXZ benchmark, we next investigate the Thirring model. We consider a quench of the lattice Thirring Hamiltonian (see Fig. 2.12) from parameters $(\Delta_1, m_1) = (0.3, 1.2)$ (which correspond to an effective inverse temperature $\beta \approx 0$) and $(\Delta_1, m_1) = (-0.9, 1.3)$ in the sense of Sec. 2.4, to $(\Delta_2, m_2) = (0.1, -0.9)$, as in our earlier DQPT analysis. We compute the Loschmidt amplitude $G(z)$ for complex times $z = t - i\tau$ using tensor-network techniques adapted to complex-time evolution, and locate the nonanalytic points of the complex rate function. The resulting pattern of zeros and branch-crossings in the complex plane is shown in Fig. 2.12. These points are the natural candidates for Fisher zeros of the Thirring boundary partition function. When we compare Fig. 2.12 to the effective temperature–time diagram of Fig. 2.10(a), we find that the structures are similar but not same.



(a)



(b)

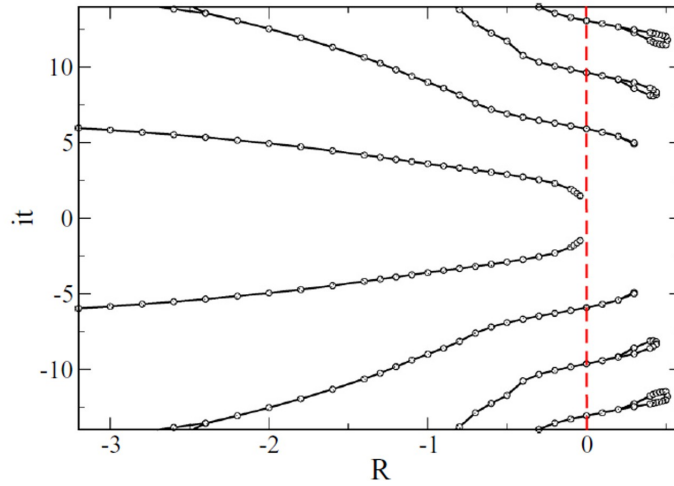
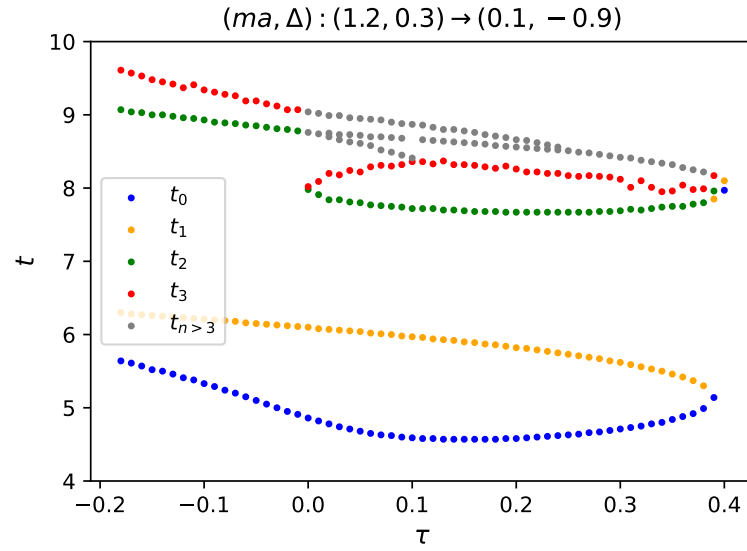


Figure 2.11: (a) DQPT points of the XXZ chain with staggered field (2.19) in the complex-time plane $z = t - i\tau$ for the quench $(\Delta_1, h_{\text{st},1}) = (6, 0.1)$ to $(\Delta_2, h_{\text{st},2}) = (-0.6, 0.1)$ in Eq. (2.20). Symbols indicate the DQPT in the complex plane. (b) Fisher zeros of the same quench, obtained from the light-cone renormalization-group transfer-matrix approach of Ref. [18]. The close agreement between panels (a) and (b).



(a)



(b)

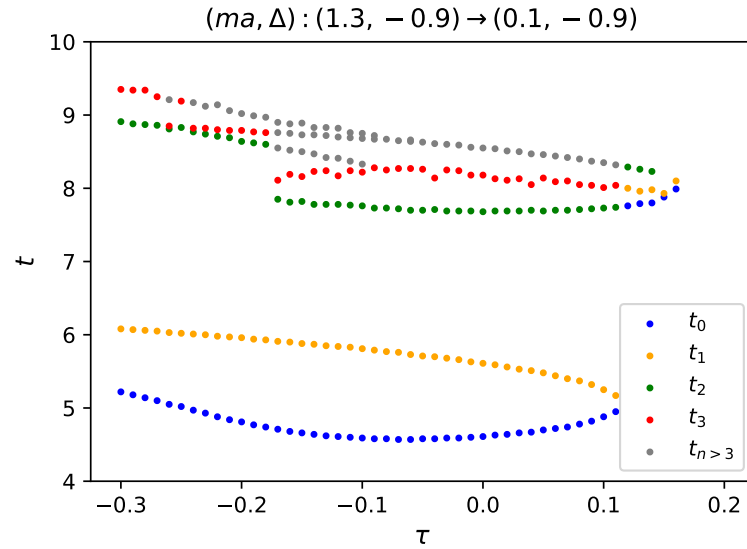
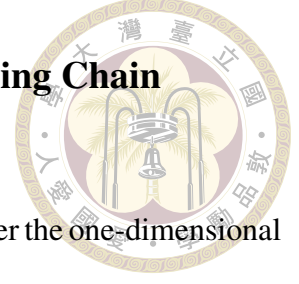


Figure 2.12: DQPT structure in the complex-time plane $z = t - i\tau$ for the Thirring model following a quantum quench. Panel (a) shows the nonanalytic structure starting from the ground state at $(\Delta_1, m_1) = (0.3, 1.2)$ and quenching to $(\Delta_2, m_2) = (0.1, -0.9)$. Panel (b) corresponds to an initial ground state at $(\Delta_1, m_1) = (-0.9, 1.3)$ with the same post-quench Hamiltonian. Symbols indicate the locations of nonanalyticities of the return rate function.

2.5.3 Analytic Fisher Zeros in the Transverse-field Ising Chain



To clarify the origin of these differences, it is instructive to consider the one-dimensional transverse-field Ising model (TFIM),

$$H_{\text{TFIM}} = -J \left(\sum_{\langle i,j \rangle} Z_i Z_j + g \sum_j X_j \right), \quad (2.21)$$

for which the Fisher zeros of the Loschmidt amplitude can be obtained analytically [78].

For a quench from an initial transverse field g_1 to a final one g_2 , the zeros in the complex-time plane $z = t - i\tau$ are given by

$$z_n(k) = \frac{1}{2\epsilon_k(g_1)} \left[\ln \tan^2 \varphi_k + i\pi(2n+1) \right], \quad (2.22)$$

where

$$\varphi_k = \theta_k(g_1) - \theta_k(g_2), \quad \tan[2\theta_k(g)] = \frac{\sin k}{g - \cos k}, \quad (2.23)$$

and

$$\epsilon_k(g) = \sqrt{(g - \cos k)^2 + \sin^2 k} \quad (2.24)$$

is the single-particle dispersion. The Fisher zeros thus depend explicitly on both g_1 and g_2 , i.e. on both the initial state and the post-quench Hamiltonian.

In Fig. 2.13 we show two representative examples of the Fisher zero distribution, for (a) $g_1 = 0.2$, $g_2 = 1.3$ and (b) $g_1 = 0.2$, $g_2 = 1.3$. Even for fixed g_2 , changing the initial condition g_1 modifies the position of the Fisher zeros in the complex plane. This illustrates a key point: different initial states, even at similar effective temperatures, generically generate different sets of Fisher zeros.

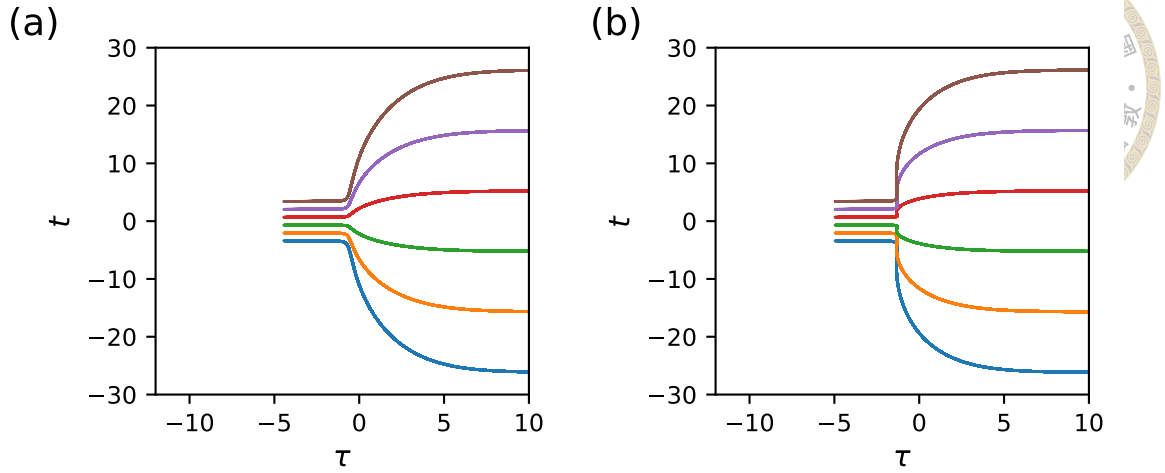


Figure 2.13: Analytic Fisher zeros of the TFIM Loschmidt amplitude in the complex-time plane $z = t - i\tau$ for two different quenches: (a) $g_1 = 0.2$ to $g_2 = 1.3$ and (b) $g_1 = 0.8$ to $g_2 = 1.3$. The Fisher zeros are computed from Eqs. (2.22)–(2.24), following Ref. [78]. The patterns change with the choice of initial field g_1 , illustrating that different initial states lead to different Fisher-zero structures even for the same post-quench Hamiltonian.

The TFIM example makes it clear that the distribution of Fisher zeros in the complex-time plane is not determined solely by the post-quench Hamiltonian H_2 , but also by the initial state, and hence by its effective temperature. In our Thirring model study, varying $(m_1 a, \Delta_1)$ changes both the energy density and the effective inverse temperature β , so that each point in Fig. 2.10 corresponds to a different initial state and, consequently, to a different Fisher-zero structure in the complex plane. From this perspective, the effective-temperature–time diagram in Fig. 2.10 should not be interpreted as the Fisher zeros in a complex plane. Instead, Fig. 2.10 is better viewed as a projection of many different Fisher-zero configurations (one for each initial state) onto the real-time axis, organized by the effective temperature of the corresponding state as shown in Figs. 2.10(a) and 2.12. A more precise characterization of the relation between the (β, t) branch structure and the full Fisher-zero manifolds in interacting field theories such as the Thirring model remains an open problem. We leave a more systematic investigation of this question for future

work.

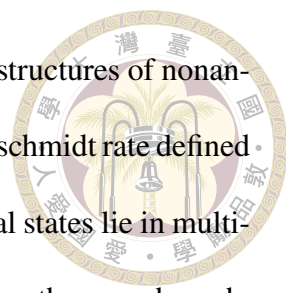


2.6 Conclusion

Using standard TNS methods, we have investigated the occurrence of DQPTs in the lattice discretization of the Thirring model, which corresponds to a spin-1/2 XXZ chain coupled to a uniform and a staggered magnetic field, with two free dimensionless parameters, respectively related to the coupling and the mass. Using the uniform MPS ansatz to work directly in the thermodynamic limit, we have simulated quenches where the evolution is given by a constant Hamiltonian and the initial state is chosen as the ground state for different parameters, in the same or a different phase. We then studied the time evolution of the spectrum of the mixed transfer operator, whose dominant eigenvector directly determines the Loschmidt rate. DQPTs are found when another eigenvalue crosses with the largest one.

We have identified a threshold in the energy density of the initial state below which DQPTs do not occur, at least during the finite times reachable by our simulations. The threshold is observed both for quenches in the gapless and the gapped phases. The energy density of each initial state corresponds to an effective temperature: that of the thermal equilibrium state with the same energy per spin, restricted to the sector of zero magnetization. This thermal state is the one expected to describe the equilibrium values of local observables after the quench in the long-time limit, since energy and magnetization are the only local conserved quantities in the problem.

We have analyzed the position of each observed DQPT (or nonanalyticity of the



Loschmidt rate) in the time-effective-temperature plane. This reveals structures of nonanalyticities reminiscent of the zeros in the complex-time plane of the Loschmidt rate defined for a fixed initial state. Our results show that DQPTs for different initial states lie in multiple well-defined branches. Initial states in different phases can appear on the same branch, which thus is not related to the zero-temperature phase diagram. All DQPT branches start at some minimal effective temperature. In most cases, the lowest temperature edge of a branch corresponds to a smooth merge with another one. However, in the case of the quench with the gapped Hamiltonian, we identify a particular branch, with DQPTs happening at the shortest times, with a sharp end at $\beta_c \sim 1.7$. We have found that this energy threshold corresponds very accurately with a substantial change in the properties of the thermal equilibrium states at the corresponding effective temperature. More concretely, below this temperature, a certain string correlator attains a nonzero value at long distances, a property that also characterizes the ground state in the gapped phase. For this branch, we have thus established a relation between the presence of DQPTs and the equilibrium phase diagram at finite T . We have not found a similar correspondence for the smooth thresholds of branches that merge into each other.

Our study thus suggests that observing the dynamics and analyzing the DQPTs from different initial states can probe properties of the finite-temperature phase diagram, in particular when a sharp threshold is found for the occurrence of these nonanalyticities. Since our simulations are limited to finite-time windows, due to the accumulation of truncation error, and we have not mapped out the complete thermal equilibrium properties at all temperatures, further investigation is needed to determine whether the other observed branches correspond also to some quantifiable equilibrium property. It is also interesting

to investigate whether similar effects appear in other spin models, and the precise correspondence between Fisher zeros and the DQPT lines in our time-effective temperature representation.







Chapter 3 Coarsening of the XXZ Model

3.1 Quench and Coarsening

The phenomenon of domain coarsening (also known as phase-ordering kinetics) arises when a system is rapidly quenched from a homogeneous, high-temperature phase into a region of the phase diagram where multiple symmetry-broken states. A comprehensive review of this subject is given by Bray [79]. An example is provided by the classical 2D Ising model, whose Hamiltonian is

$$H(\sigma) = -J \sum_{\langle i,j \rangle} \sigma_i \sigma_j, \quad \sigma_k \in \{-1, +1\}, \quad (3.1)$$

where the sum runs over nearest-neighbour pairs on a two-dimensional square lattice. At temperatures $T > T_c$, is in homogeneous phase, while for $T < T_c$ the system develops two-phases region. This behavior is illustrated in Fig. 3.1, adapted from Bray [79]. At time $t = 0$, the system is quenched from an initial temperature $T_I > T_c$ to a final temperature $T_F < T_c$, entering the two-phase region where competing domains of opposite magnetization begin to form.

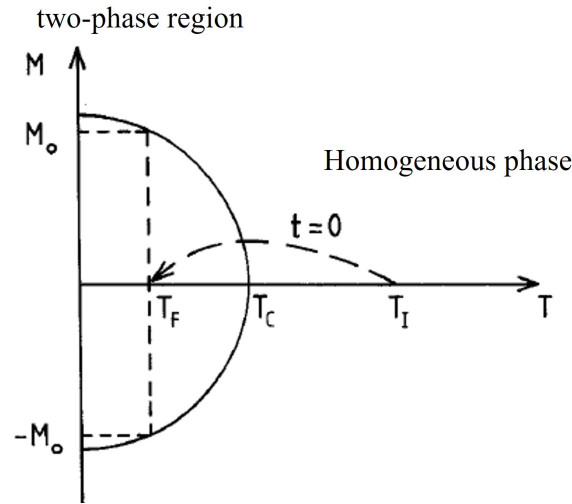


Figure 3.1: Magnetization of the two-dimensional Ising model in zero field. At $T > T_c$ the system is in homogeneous phase, while below T_c is in two-phases region. The arrow indicates a sudden temperature quench at time $t = 0$ from an initial temperature T_I to a final temperature $T_F < T_c$. Adapted from Bray [79].

The microscopic manifestation of coarsening is shown in Fig. 3.2, which displays snapshots of a 256×256 Ising lattice undergoing zero-temperature Monte Carlo dynamics after a quench from $T = \infty$. These images originate from the work of Kissner and are reproduced in Bray's review [79]. At early times, the system consists of many small, irregular domains. As time evolves, domains grow while the total length of domain walls decreases. This behavior is characterized by a single dynamical length scale $L(t)$ that increases algebraically in time,

$$L(t) \sim t^{1/2}. \quad (3.2)$$

The results summarize the phenomenology of classical coarsening: immediately after a quench, the system forms small domains of the new equilibrium phases, and their characteristic length scale grows in time through domain-wall motion.

A natural question is whether analogous behavior occurs in quantum many-body systems out of equilibrium. In particular, a quantum quench, a sudden change of a parameter

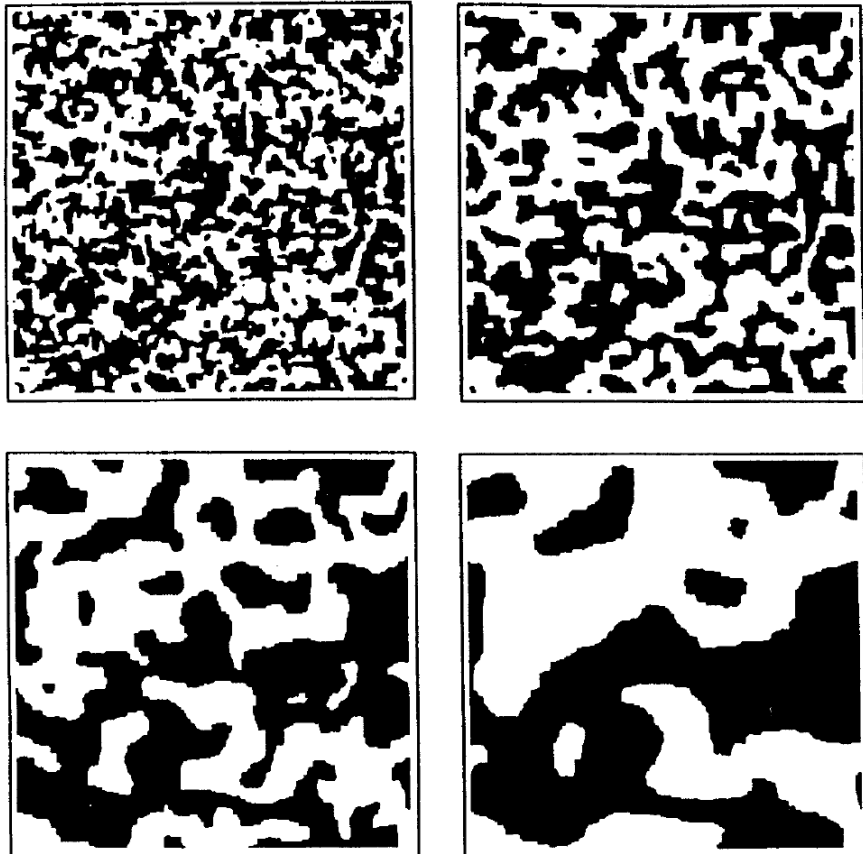


Figure 3.2: Monte Carlo simulation of domain growth in the two-dimensional Ising model at $T = 0$ (from Kissner, reproduced in Bray [79]). The system size is 256×256 . The snapshots correspond to 5, 15, 60, and 200 Monte Carlo steps per spin after a quench from $T = \infty$. Domains grow in size while domain walls shrink, illustrating universal coarsening dynamics.

in the Hamiltonian, can generate spatially inhomogeneous structures. In this chapter, we investigate whether coarsening-like dynamics emerge in the one-dimensional XXZ spin chain. Following the tensor-network framework used in earlier chapters, we prepare initial states using DMRG and study their real-time evolution using the TDVP algorithm. These tools enable us to resolve domain-wall structures and track their evolution, allowing us to examine whether a growing length scale $L(t)$ appears in the quantum dynamics and how it compares to the classical picture.

3.2 Coarsening of the XXZ Model in a Quantum Quench

We now turn to the one-dimensional XXZ spin- $\frac{1}{2}$ chain, whose Hamiltonian is

$$H = J \sum_n \left[\frac{1}{2} (S_n^+ S_{n+1}^- + S_n^- S_{n+1}^+) + \Delta S_n^z S_{n+1}^z \right] + h_z \sum_n S_n^z, \quad (3.3)$$

where J is the exchange coupling, Δ is the anisotropy, and h_z is a longitudinal field. In this chapter we focus on the field-free case $h_z = 0$. To motivate a comparison with the classical Ising quench introduced earlier, we consider quenches that cross the XXZ phase boundary. The schematic phase diagram is shown in Fig. 3.3. For $-1 < \Delta < 1$, the XXZ chain is in a gapless phase, whereas $\Delta > 1$ corresponds to a gapped antiferromagnetic phase with two degenerate Néel-ordered ground states.

A quantum quench is implemented by preparing the ground state of an initial Hamiltonian H_1 and then evolving under a different Hamiltonian H_2 (see also Sec. 2.3):

$$|\psi(t)\rangle = e^{-iH_2 t} |\psi_1\rangle, \quad (3.4)$$

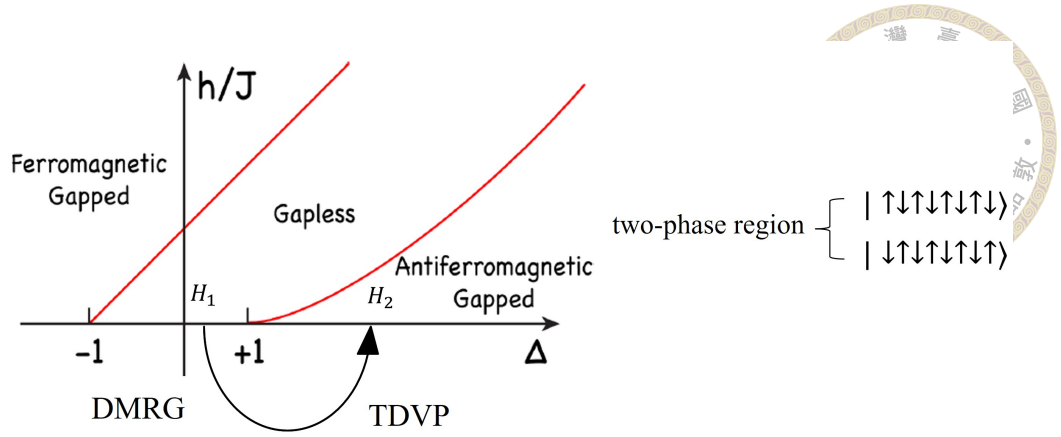


Figure 3.3: Phase diagram of the XXZ model. Adapted from Ref. [80]. The system is gapless for $-1 < \Delta < 1$ and enters a gapped antiferromagnetic phase for $\Delta > 1$. In this chapter we consider quenches from the gapless regime into the gapped regime (black arrow), analogous to the classical temperature quench from disordered to ordered phases.

where $|\psi_1\rangle$ is obtained using DMRG, and the real-time evolution is simulated using the TDVP algorithm. Unlike the infinite-size simulations of the previous chapter 2, our focus here is on finite chains, allowing us to access explicit spatial patterns in the evolving state.

To parallel the classical coarsening scenario, we consider quenches from the gapless phase ($\Delta < 1$) into the gapped antiferromagnetic phase ($\Delta > 1$). For even chain lengths, the gapped phase hosts two Néel-ordered configurations,

$$|\uparrow\downarrow\uparrow\downarrow\cdots\rangle, \quad |\downarrow\uparrow\downarrow\uparrow\cdots\rangle.$$

This raises the natural question: can a quantum quench into the antiferromagnetic regime generate coarsening-like domain growth?

To explore this, we performed a first set of TDVP simulations beginning with the ground state at anisotropy $J_z = J\Delta = 0.8$ and quenching to $J_z = 1.2$ (with $J = 1$). The resulting time evolution of the local magnetization is shown in Fig. 3.4. Although Néel-like oscillations appear (highlighted by the white rectangle), their magnitude is extremely

small, suggesting that the quench does not produce strong ordering for these parameters.

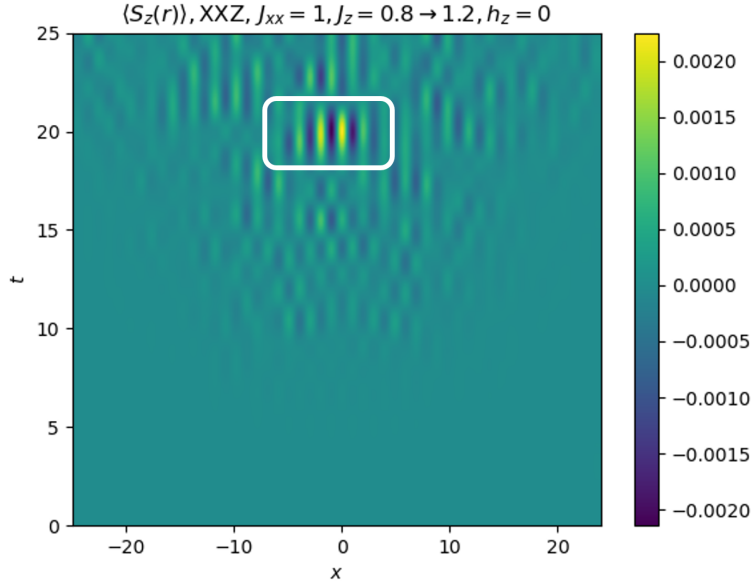


Figure 3.4: Real-time evolution following a quench from $J_z = 0.8$ to $J_z = 1.2$ with bond dimension $D = 64$. Weak Néel-like oscillations (white rectangle) appear after the quench, but the ordering is extremely small. We attribute this to finite-size effects and the relatively weak antiferromagnetic order in the ground state at $J_z = 1.2$.

To clarify the origin of this weak signal, we carried out DMRG calculations for various system sizes N and anisotropy values J_z . For each ground state, we computed the average magnitude of the local magnetization,

$$|S^z|_{\text{avg}} = \frac{1}{N} \sum_{i=1}^N |\langle S_i^z \rangle|. \quad (3.5)$$

The results in Fig. 3.5 demonstrate that the Néel order increases with both J_z and N , as expected when approaching the thermodynamic limit of the gapped phase.

These results reveal that the weak ordering seen in Fig. 3.4 is a consequence of finite-size suppression of Néel order near the phase boundary. To enhance the possibility of observing quantum coarsening dynamics, it is therefore necessary to quench deeper into the antiferromagnetic phase, where the ground-state order is more robust.

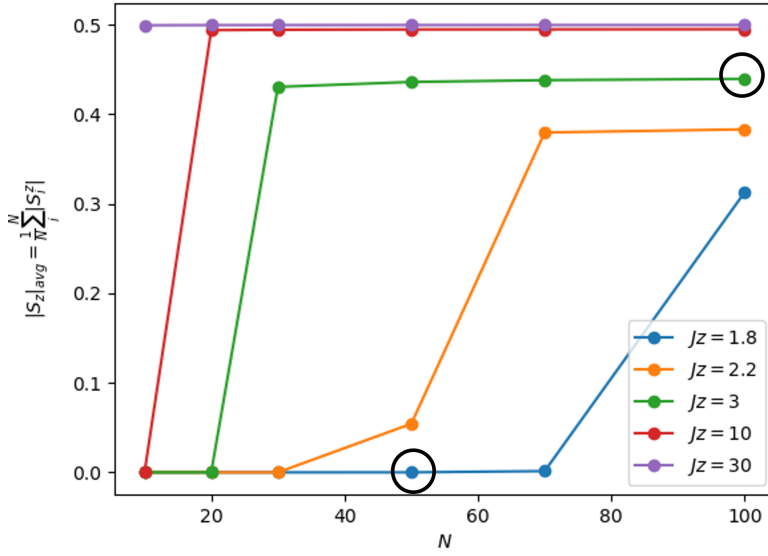


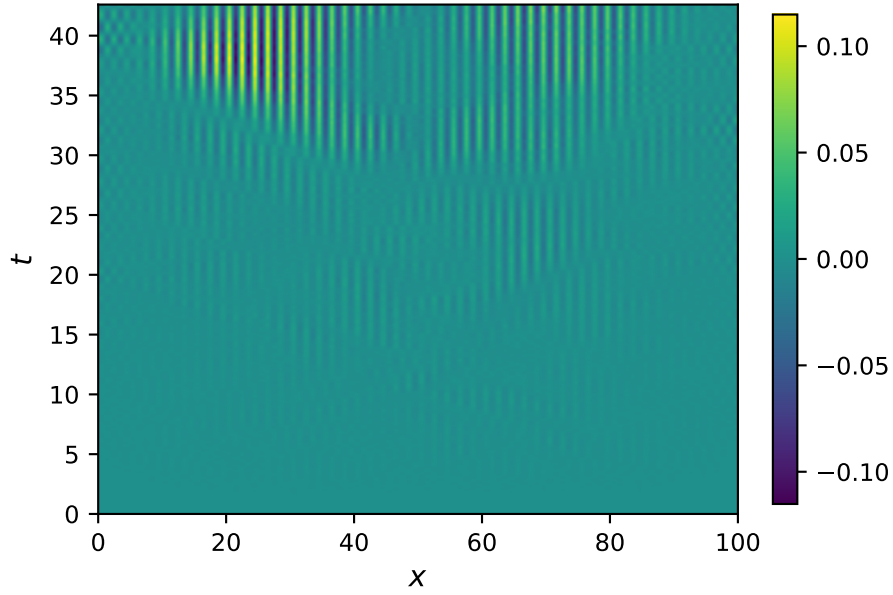
Figure 3.5: Néel-order amplitude of XXZ ground states for various N and J_z (DMRG). The average magnitude $|S^z|_{\text{avg}} = \frac{1}{N} \sum_i |S_i^z|$ increases systematically with both anisotropy J_z and system size N . Left black circle: the quench value $J_z = 1.2$ used in Fig. 3.4 has extremely weak Néel order, explaining the small oscillation amplitude. Right black circle: a larger anisotropy such as $J_z = 3$ for $N = 100$ yields strong Néel order.

We then simulate a deeper quench into the antiferromagnetic phase, starting from the ground state at $J_z = 0.8$ (gapless regime) and evolving under a quench Hamiltonian with $J_z = 3$ and system size $N = 100$ and $dt = 0.02$. The initial state is obtained using DMRG, and the real-time evolution is computed using the TDVP algorithm with bond dimension $D = 128$. The magnetization pattern $\langle S_i^z(t) \rangle$ is displayed in Fig. 3.6(a).

In contrast to the quench into the weakly gapped regime ($J_z = 1.2$), where post-quench Néel order was only of order 10^{-3} , the quench to $J_z = 3$ produces a larger magnetizations reaching values around 0.1. This is consistent with the ground-state analysis in Fig. 3.5, which showed that the staggered order grows rapidly with increasing J_z and system size N . To highlight the structure of the evolving domains, Fig. 3.6(b) shows the same data with a restricted color scale of $[-0.01, 0.01]$.



(a)



(b)

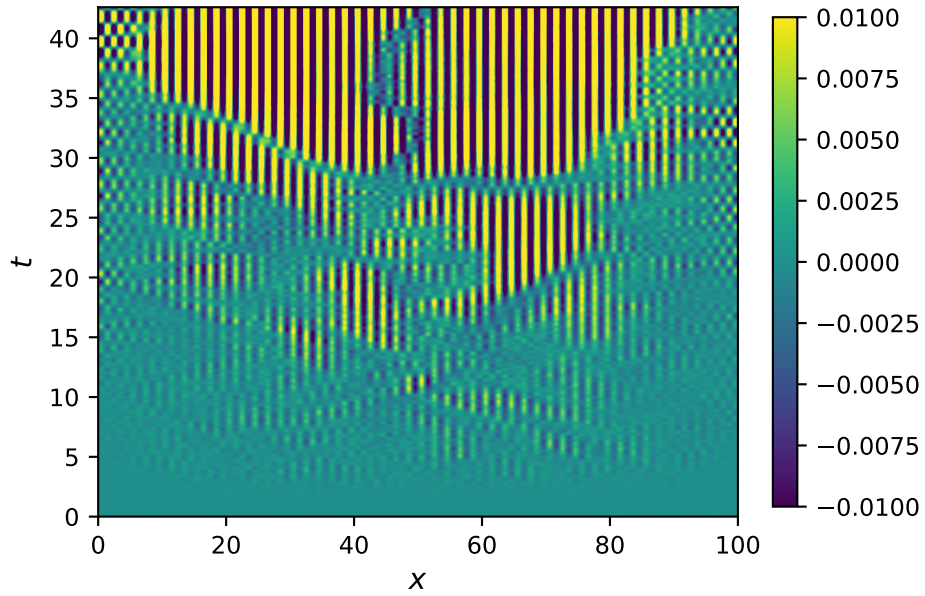


Figure 3.6: Quantum quench from $J_z = 0.8$ to $J_z = 3$ (DMRG + TDVP, $D = 128$). (a) Evolution of $\langle S_i^z \rangle$. (b) Same data with restricted color scale $[-0.01, 0.01]$ to emphasize domain-wall motion and the coarsening of local antiferromagnetic regions.

To better visualize the two Néel sectors, we evaluate the staggered magnetization $(-1)^i \langle S_i^z(t) \rangle$, shown in Fig. 3.7. Blue regions correspond to the $|\uparrow\downarrow\uparrow\downarrow \cdots\rangle$ configuration, red regions to $|\downarrow\uparrow\downarrow\uparrow \cdots\rangle$, and the domain walls are marked in black, identified by the condition $\langle S_i^z \rangle \langle S_{i+1}^z \rangle > 0$. An additional feature visible in Fig. 3.7 is the oscillatory behavior near the boundaries of the chain. These might arise from finite-size effects. To avoid these boundary artifacts, in the subsequent quantitative analysis we restrict attention to the interior region of the chain, specifically sites $i \in [11, 89]$.

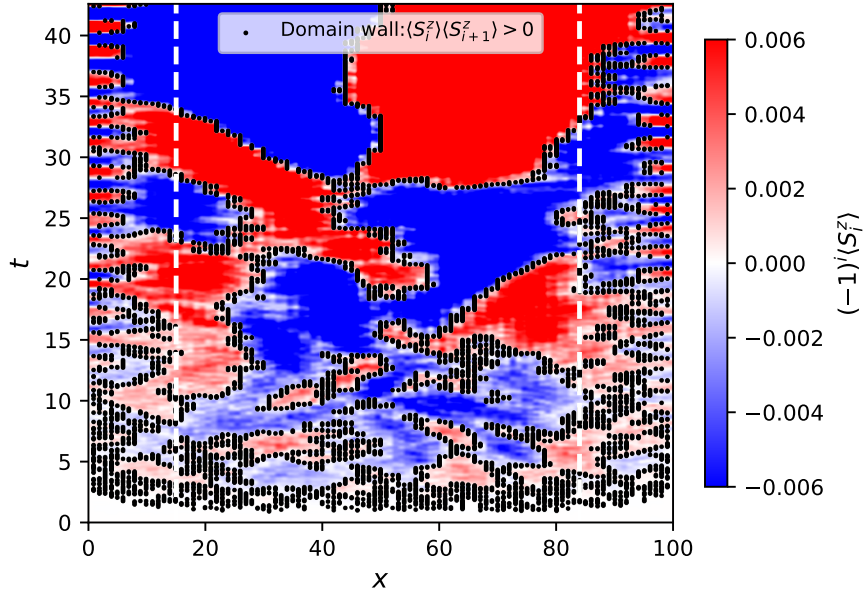
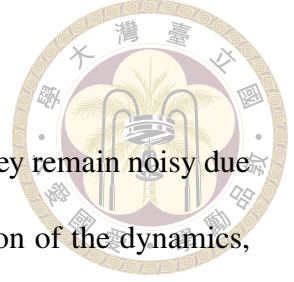


Figure 3.7: Neel order for quantum quench from $J_z = 0.8$ to $J_z = 3$ (DMRG + TDVP, $D = 128$). Staggered magnetization $(-1)^i \langle S_i^z \rangle$, distinguishing the two Néel domains: blue corresponds to $|\uparrow\downarrow\uparrow\downarrow \cdots\rangle$, red to $|\downarrow\uparrow\downarrow\uparrow \cdots\rangle$. Domain walls are marked in black, defined by $\langle S_i^z \rangle \langle S_{i+1}^z \rangle > 0$.

We now analyze the time dependence of the number of domain walls after the quench. As discussed above, boundary effects introduce strong oscillations near the chain edges. To suppress these artifacts, we restrict our analysis to the interior region of the chain, specifically sites $i \in [11, 89]$. Within this window, we count the number of domain walls at each time t using the criterion $\langle S_i^z \rangle \langle S_{i+1}^z \rangle > 0$. The resulting domain-wall number



$N_{\text{DW}}(t)$ is shown in Fig. 3.8(a).

Although the raw data exhibit a clear overall decreasing trend, they remain noisy due to rapid microscopic oscillations. To obtain a smoother representation of the dynamics, we apply a moving average along the time axis. Two representative smoothing windows, mask sizes of 50 and 501, are shown respectively in Fig. 3.8(b) and Fig. 3.8(c). The larger window suppresses short-time fluctuations more effectively and reveals an approximate power-law decay of the domain-wall number.

To quantify this behavior, Fig. 3.8(d) displays the log–log plot of the smoothed data from panel (c). Two distinct regimes are visible. For early to intermediate times ($t < 34$), the data are well described by a power law of the form

$$N_{\text{DW}}(t) \approx 45 t^{-0.7}, \quad t < 34,$$

indicated by the orange fit line in Fig. 3.8(d). Beyond $t \approx 34$, the domain walls within the clipped region have merged into a single surviving domain, and the domain-wall number is

$$N_{\text{DW}}(t) = 1, \quad t \geq 34,$$

The combined fit,

$$N_{\text{DW}}(t) = \begin{cases} 45 t^{-0.7}, & t < 34, \\ 1, & t \geq 34, \end{cases}$$

provides a good description of the observed dynamics, as demonstrated by the agreement with the smoothed data in Fig. 3.8(d).

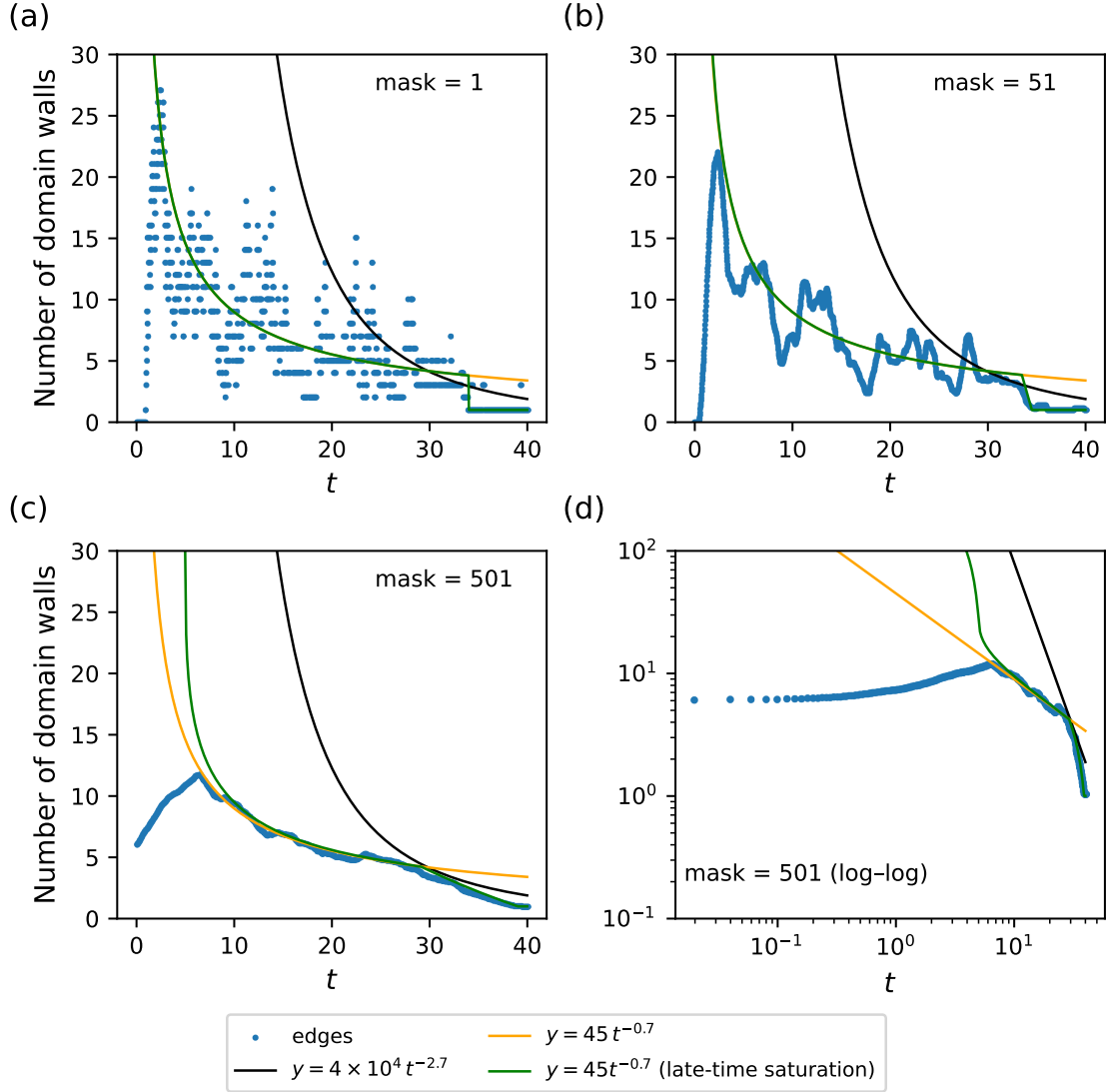


Figure 3.8: Domain-wall dynamics following a quench from $J_z = 0.8$ to $J_z = 3$. Panel (a) shows the raw number of domain walls $N_{DW}(t)$ counted in the clipped region $i \in [11, 89]$. Panels (b) and (c) display the same time series after applying moving averages with mask sizes 50 and 501, respectively, with the larger mask revealing a clearer decay trend. Panel (d) presents a log–log plot of the smoothed data in (c). The orange line indicates a pure power-law decay $N_{DW}(t) \propto t^{-0.7}$ without saturation, while the green line shows the same power-law behavior followed by saturation to a single remaining domain, $N_{DW} = 1$, at late times ($t \geq 34$).

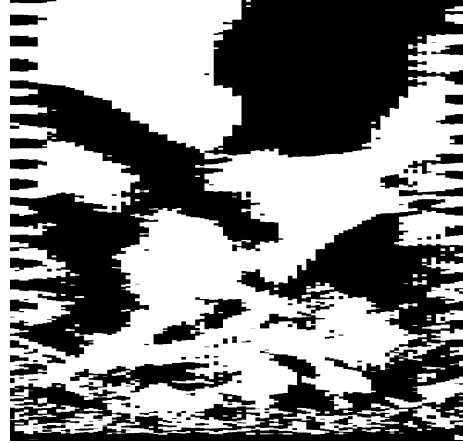
Although the smoothed domain-wall data already exhibit a clear power-law trend, the original (unsmoothed) time series remains significantly fluctuating, as visible in Fig. 3.8(a). This raises the question of whether the fluctuations originate from the procedure used to identify domain walls—namely, the simple criterion $\langle S_i^z \rangle \langle S_{i+1}^z \rangle > 0$ —or whether they reflect intrinsic features of the finite-size quantum dynamics.

To investigate this, we adopt a complementary, more image-based perspective. Instead of determining domain walls directly from the discrete product $\langle S_i^z \rangle \langle S_{i+1}^z \rangle$, we first convert the spatiotemporal magnetization data $\langle S_i^z(t) \rangle$ into a grayscale image, where spatial and temporal variations are naturally represented as intensity gradients. We then apply standard edge-detection algorithms to this image: the Canny edge detector, which responds to edges of all orientations, and the Sobel x operator, which detects horizontal edges corresponding to spatial boundaries. The results are summarized in Fig. 3.9. Panel (a) shows the converted image, (b) the edges detected by the Canny algorithm, and (c) those detected by the Sobel x filter. Fig. 3.10 displays the extracted edge counts as a function of time, converted from pixel units to the physical domain-wall number, together with the domain-wall count obtained previously from the condition $\langle S_i^z \rangle \langle S_{i+1}^z \rangle > 0$.

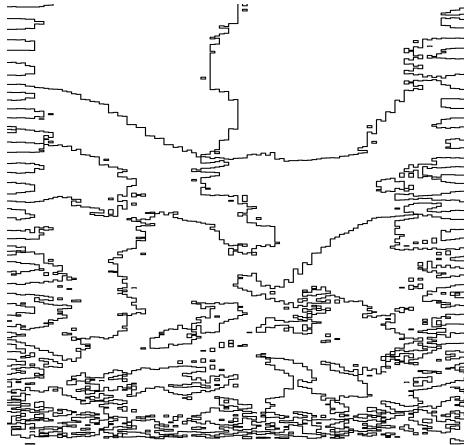
Several observations can be made from Fig. 3.10. First, the Canny detector consistently produces a larger edge count than the Sobel x filter, as expected: Canny identifies edges in all directions (x and y), whereas Sobel x responds only to spatial boundaries relevant for domain walls. Second, both edge-detection methods yield strongly fluctuating results, mirroring the behavior observed in the direct domain-wall count. Thus, applying image-based edge detection does not resolve the fluctuation problem; instead, it suggests that the noisy structure is inherent to the quantum dynamics of this finite-size system rather



(a)



(b)



(c)

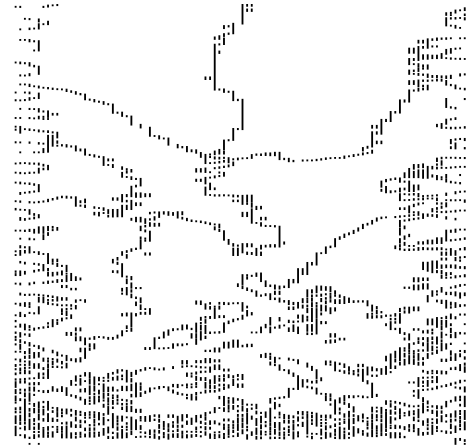


Figure 3.9: Edge-detection analysis of domain boundaries. (a) Image representation of the magnetization data $\langle S_i^z(t) \rangle$ for sites $i \in [11, 89]$, where positive and negative Néel-ordered regions are mapped to black and white pixels, respectively, with corresponding gray values 0 and 255 (see Fig. 3.7). (b) Edges detected from panel (a) using the Canny algorithm with lower and upper thresholds set to 5 and 10, respectively; this method identifies intensity gradients in all directions. (c) Edges detected from panel (a) using the Sobel- x operator, which computes horizontal intensity gradients and is therefore sensitive only to spatial boundaries associated with domain walls.

than an artifact of the boundary-removal or detection method. In the following section, we try to using statistical method to decrease this fluctuated behavior.

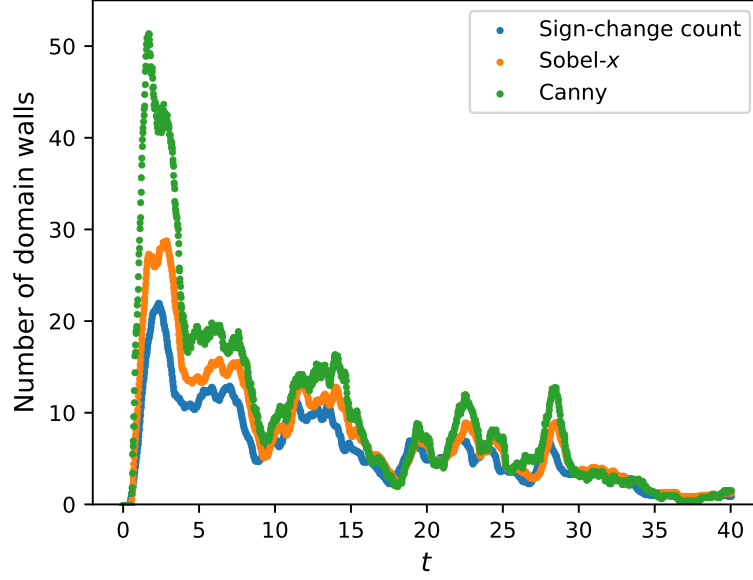


Figure 3.10: Extracted edge counts as a function of time, converted from pixel counts to the physical number of domain walls. All data shown are obtained after applying a moving average with mask size 51 to the time series of $\langle S_i^z(t) \rangle$ within the clipped bulk region $i \in [11, 89]$, including the direct sign-change counting method as well as the Canny and Sobel- x edge-detection approaches. For comparison, the domain-wall numbers extracted using the Canny and Sobel- x methods are based on the edge-detection results shown in Fig. 3.9(b) and (c).

3.3 Statistical Averaging over Random Longitudinal Fields

Since both the direct domain-wall count and the image-based edge-detection methods exhibit strong fluctuations, we perform an additional analysis based on statistical averaging. The idea is to suppress microscopic fluctuations by repeating the TDVP simulation many times, each with a slightly different realization of a weak longitudinal field. Concretely, we keep the initial state fixed (the DMRG ground state at $J_z = 0.8$), but in the quench Hamiltonian we assign to each site a random longitudinal field $h_z[i]$ drawn uniformly from the interval $(-0.01, 0.01)$. We perform fifty independent TDVP simulations

and compute the domain-wall number for each realization.

Two representative evolutions from different random seeds are shown in Fig. 3.11. While the exact positions and trajectories of domain walls differ between realizations, both runs display the same global trend: domain walls annihilate over time and larger ordered regions form.

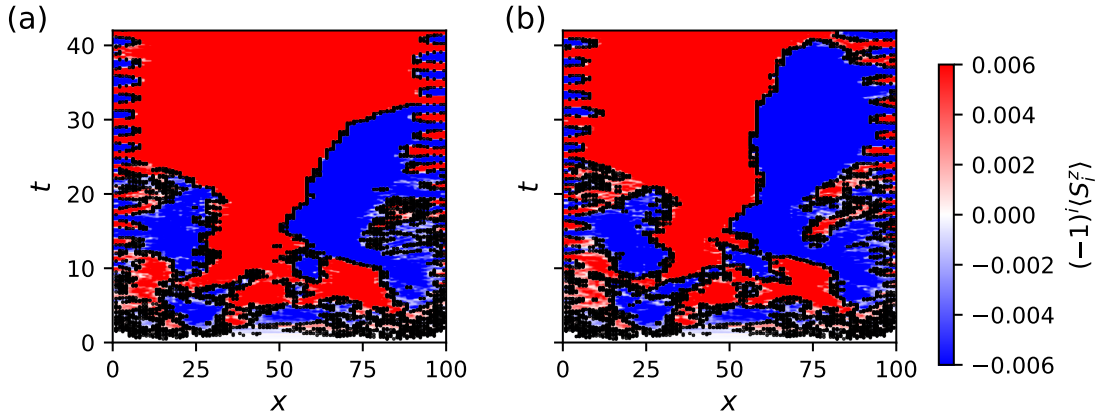


Figure 3.11: Two TDVP simulations with random longitudinal fields. Each simulation starts from the same initial ground state at $J_z = 0.8$, while the quench Hamiltonian is set to $J_z = 3$ and includes independently drawn random on-site fields $h_z[i] \in (-0.01, 0.01)$. Panels (a) and (b) correspond to two different random realizations of the longitudinal field. Although the detailed domain patterns differ, both simulations exhibit clear coarsening behavior.

To reduce fluctuations, we compute the averaged domain-wall number over all fifty realizations. The results are shown in Fig. 3.12. Panel (a) displays the raw average, which is still noisy. To obtain a cleaner trend, we apply moving averages with mask sizes 51 and 501, shown in panels (b) and (c), respectively. With $dt = 0.02$, these mask widths correspond to smoothing windows of 1.0 and 10.0 in physical time units.

In the subsequent analysis, we compare two different domain-wall detection methods. The first (blue curves) uses the direct condition $\langle S_i^z \rangle \langle S_{i+1}^z \rangle > 0$. The second (orange curves) is an improved method designed to suppress artificial splitting of a single domain

wall into many short-lived fluctuating ones. This improved procedure is illustrated in Fig. 3.13, where yellow points correspond to the raw detections while black points show the merged (only counting one time).

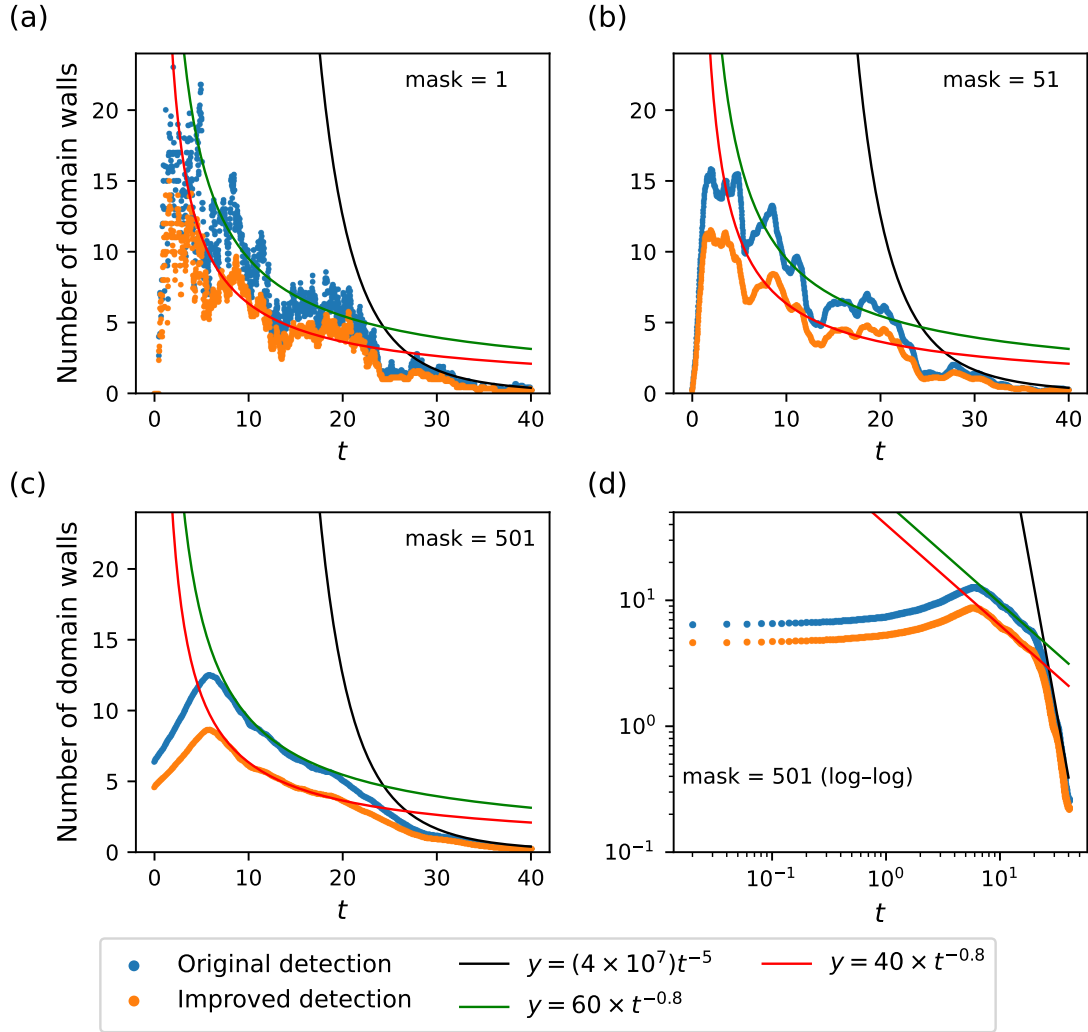
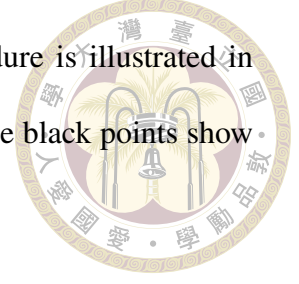


Figure 3.12: Statistical averaging of domain-wall number over fifty TDVP runs. (a) Raw average domain-wall number. (b–c) Moving averages using mask sizes 51 and 501, corresponding to smoothing windows of 1.0 and 10.0 in physical time. Blue lines use the direct condition $\langle S_i^z \rangle \langle S_{i+1}^z \rangle > 0$; orange lines use the improved method illustrated in Fig. 3.13. (d) Log–log plot of panel (c), revealing an exponential decay $N_{\text{DW}}(t) \sim e^{-0.8t}$ for both detection methods.

Finally, the smoothed, statistically averaged data in Fig. 3.12(d) reveal a clean expo-

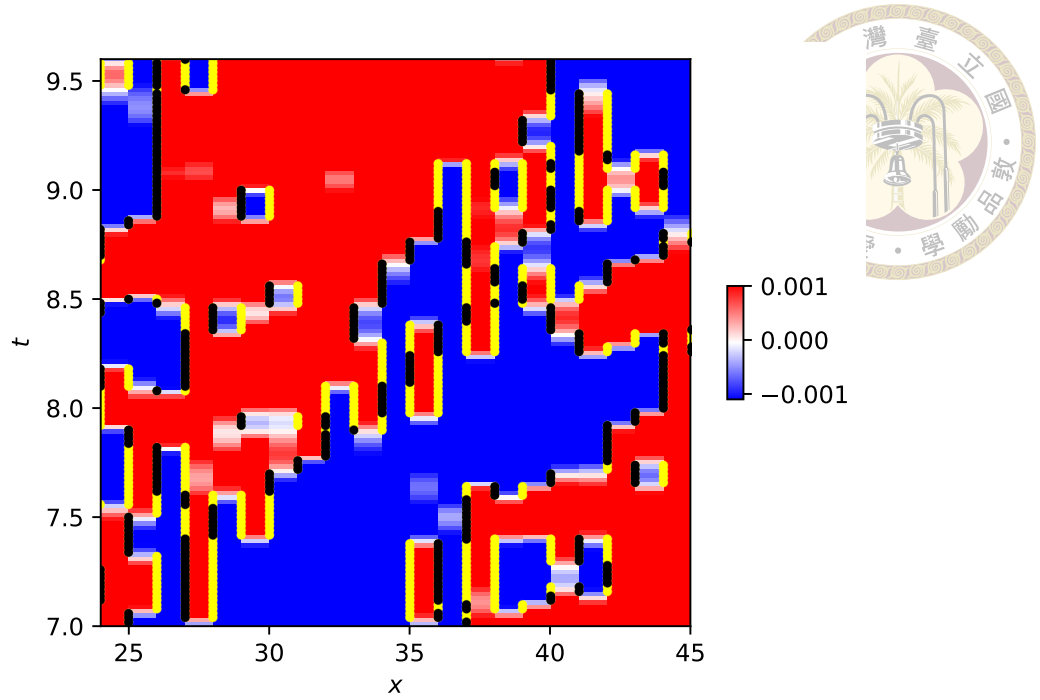


Figure 3.13: Improved domain-wall identification method. Yellow points: raw detections from $\langle S_i^z \rangle \langle S_{i+1}^z \rangle > 0$, which fluctuate rapidly. Black points: improved detections in which contiguous fluctuation segments are merged and counted as a single domain wall. This reduces artificial overcounting, but the statistical averaged decay rate remains the same, as shown in Fig. 3.12(d).

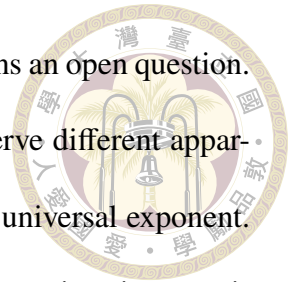
nential decay of the domain-wall number,

$$N_{\text{DW}}(t) \sim e^{-0.8t},$$

independent of whether the direct or improved detection method is used. This exponent differs from the effective power-law $N_{\text{DW}}(t) \sim t^{-0.7}$ observed earlier without statistical averaging.

In conclusion, our simulations demonstrate that coarsening dynamics, well known from classical phase-ordering kinetics, also arise following a quantum quench in the one-dimensional XXZ model. Starting from a gapless initial state and quenching deep into the antiferromagnetic phase, we observe the formation, motion, and annihilation of domain walls, leading to the gradual growth of Néel-ordered regions in qualitative analogy with

classical coarsening. However, the precise scaling of this decay remains an open question. Depending on the analysis method and degree of averaging, we observe different apparent power-law-like trends, and the data do not yet establish a unique universal exponent. Thus, while the XXZ chain exhibits clear signatures of quantum coarsening, its quantitative scaling properties remain unresolved. Determining the true asymptotic behavior of the coarsening length, its might dependence on system size and anisotropy, and its relation to classical coarsening theory are important open problems for future investigation.





Chapter 4 Solving PED Using Quantics Tensor Train and TDVP

The quantics tensor train (QTT) format provides a compact, low-rank tensor representation of high-dimensional data. In this framework, a discretized function is reshaped into a high-order tensor and then approximated as a product of small core tensors with bounded bond dimensions. This structure is mathematically equivalent to a matrix product state (MPS) widely used in quantum many-body physics, and it allows one to represent functions with an effective number of parameters that scales only polynomially with the system size, rather than exponentially. In recent years, QTT has been successfully applied to the numerical solution of partial differential equations (PDEs), where both the solution and the discretized operators can be stored and manipulated efficiently in a low-rank tensor format.

Because of this close connection to MPS, a natural strategy is to transfer mature algorithms from the tensor-network and quantum many-body community to the QTT setting. In particular, methods such as the density-matrix renormalization group (DMRG) and the

time-dependent variational principle (TDVP) have been extensively developed and tested for MPS, providing robust tools for ground-state optimization and real- or imaginary-time evolution on low-rank manifolds. In this work, we follow this philosophy and use these well-established MPS algorithms in the context of QTT representations of PDEs, combining the compression power of QTT with the algorithmic sophistication of tensor-network techniques.

Recent work has shown that the quantum tensor train (QTT) format offers an efficient and systematically compressible representation for solving partial differential equations (PDEs), particularly in high-dimensional settings [81–83].

4.1 QTT Representation in Cartesian Coordinates

To illustrate the basic idea of the quantum tensor train (QTT) representation, consider a one-dimensional function $f(x)$ defined on an interval $[a, b]$. Suppose we discretize the domain using $N = 8$ uniformly spaced sample points. Since $8 = 2^3$, each grid point can be indexed by a three-bit binary label $\sigma_1\sigma_2\sigma_3$, with $\sigma_i \in \{0, 1\}$. In this encoding, the discretized function is stored as a tensor

$$f(x) \longrightarrow f_{\sigma_1\sigma_2\sigma_3} = \{f_{000}, f_{001}, f_{010}, f_{011}, f_{100}, f_{101}, f_{110}, f_{111}\}, \quad (4.1)$$

as illustrated in Fig. 4.1(a). It is immediately clear that $f_{\sigma_1\sigma_2\sigma_3}$ is nothing more than a third-order tensor whose modes each have dimension two.

A central advantage of the QTT format is that this tensor can be decomposed into an MPS, which provides a low-rank factorization of the data. Such a decomposition may be

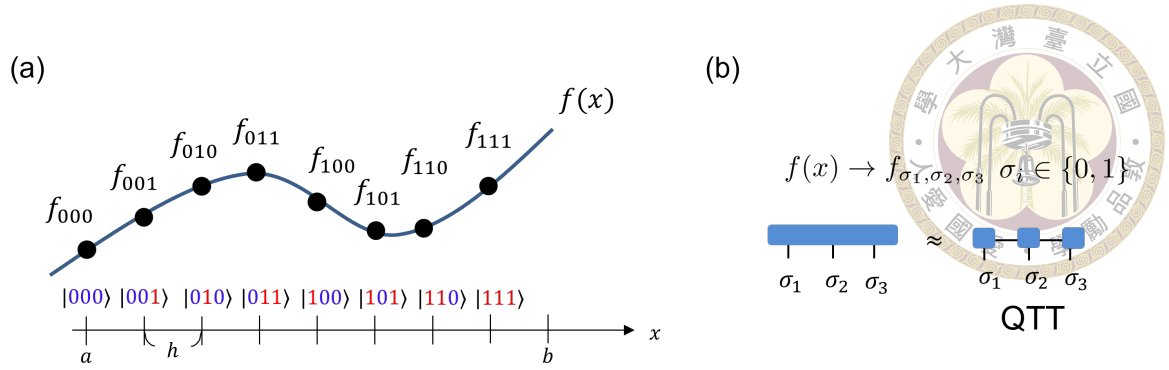


Figure 4.1: (a) Discretization of a one-dimensional function $f(x)$ over the interval $[a, b]$ using $N = 8$ uniformly spaced sample points. Because $8 = 2^3$, each grid point is naturally labeled by a three-bit binary index $\sigma_1\sigma_2\sigma_3 \in \{0, 1\}^3$, yielding the tensor entries $f_{\sigma_1\sigma_2\sigma_3}$. (b) The resulting data array is a third-order tensor of dimension $2 \times 2 \times 2$, which can be compressed into a quantics tensor train (QTT), i.e., a matrix product state with three sites corresponding to the binary indices $\sigma_1, \sigma_2, \sigma_3$.

obtained using successive singular-value decompositions or by more advanced tensor cross interpolation (TCI) methods [84]. The resulting MPS captures the structure of the function with fewer parameters whenever the data is compressible in the QTT basis. Moreover, the accuracy of the representation improves as more sampling points are used. For instance, increasing to $N = 16$ or $N = 32$ corresponds to four or five binary indices, forming a more refined discretization (see Fig. 4.2).

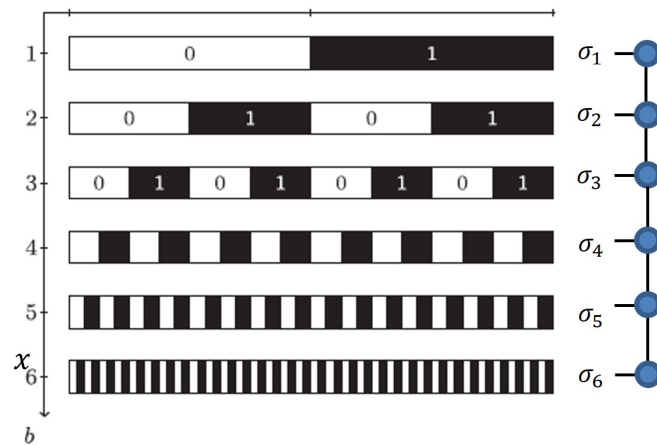


Figure 4.2: Improving the accuracy of the QTT representation by increasing the number of sampling points. When the discretization size increases from $N = 8$ to $N = 16$ or $N = 32$, the number of binary indices grows (from three to four or five), producing a higher-order QTT tensor. This finer discretization allows the tensor-train structure to capture the underlying function with higher fidelity. Adapted from Ref. [85].

A simplest example is that the exponential function $f(x) = e^{ax}$ on the interval $0 \leq x < 1$. Any x in this interval admits a binary fraction expansion

$$x = 0.\sigma_1\sigma_2\cdots\sigma_n = \frac{\sigma_1}{2} + \frac{\sigma_2}{2^2} + \cdots + \frac{\sigma_n}{2^n}, \quad \sigma_i \in \{0, 1\}, \quad (4.2)$$

so that the discretized function values may be written as

$$\begin{aligned} f(x) &= e^{a(0.\sigma_1\sigma_2\cdots\sigma_n)} = e^{a\left(\frac{\sigma_1}{2} + \frac{\sigma_2}{2^2} + \cdots + \frac{\sigma_n}{2^n}\right)} \\ &= \left(e^{a/2}\right)^{\sigma_1} \left(e^{a/2^2}\right)^{\sigma_2} \cdots \left(e^{a/2^n}\right)^{\sigma_n}. \end{aligned} \quad (4.3)$$

This factorization already reveals that the QTT cores of $f(x)$ can be written in fully separated form. Indeed, each bit σ_k contributes an independent multiplicative factor,

$$A^{\sigma_k=0} = [1], \quad A^{\sigma_k=1} = \left[e^{a/2^k}\right], \quad k = 1, \dots, n,$$

so that the full QTT representation of $f(x)$ becomes a rank-one MPS,

$$f_{\sigma_1\sigma_2\cdots\sigma_n} = A^{\sigma_1}A^{\sigma_2}\cdots A^{\sigma_n}. \quad (4.4)$$

This example demonstrates how certain smooth or separable functions naturally admit extremely compact QTT representations. Here, the exponential function requires only n scalar-valued cores, corresponding to a tensor train of bond dimension one ($D = 1$), and more complicated functions may require larger bond dimensions.

In order to solve partial differential equations within the QTT framework, we require several basic operations on functions represented in the tensor-train format. Because a QTT representation is mathematically equivalent to an MPS, all operations are naturally

expressed in terms of contractions and manipulations of local tensor cores. Below we summarize the fundamental algebraic and differential operations used throughout this work.



1. Addition and scalar multiplication

Given two functions f and g represented in QTT form with cores $\{A_i^{\sigma_i}\}$ and $\{B_i^{\sigma_i}\}$, respectively, we may construct their sum $h(x) = f(x) + g(x)$ directly in the tensor-train format. The corresponding tensor-network diagram for the addition operation can be expressed as

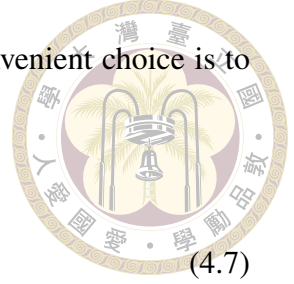
$$\begin{array}{c} C^{\sigma_1} \quad C^{\sigma_2} \quad C^{\sigma_n} \\ \bullet \quad \bullet \quad \bullet \quad \bullet \\ | \quad | \quad | \quad | \\ h(x) \end{array} = \begin{array}{c} A^{\sigma_1} \quad A^{\sigma_2} \quad A^{\sigma_n} \\ \bullet \quad \bullet \quad \bullet \quad \bullet \\ | \quad | \quad | \quad | \\ f(x) \end{array} + \begin{array}{c} B^{\sigma_1} \quad B^{\sigma_2} \quad B^{\sigma_n} \\ \bullet \quad \bullet \quad \bullet \quad \bullet \\ | \quad | \quad | \quad | \\ g(x) \end{array} . \quad (4.5)$$

the tensors $\{C_i^{\sigma_i}\}$ representing h are obtained by taking the direct sum of the tensors of f and g :

$$C^{\sigma_1} = \begin{bmatrix} A^{\sigma_1} & B^{\sigma_1} \end{bmatrix}, \quad C^{\sigma_i} = \begin{bmatrix} A^{\sigma_i} & 0 \\ 0 & B^{\sigma_i} \end{bmatrix}, \quad i = 2, \dots, n-1, \quad C^{\sigma_n} = \begin{bmatrix} A^{\sigma_n} \\ B^{\sigma_n} \end{bmatrix}. \quad (4.6)$$

This block structure ensures that contracting the resulting tensor train reproduces the function $h(x) = f(x) + g(x)$ exactly. Since the bond dimension doubles under this construction, an SVD-based compression step is typically applied afterward to reduce the tensor-train rank.

Scalar multiplication in the QTT format is straightforward. Given a constant a and a QTT representation of a function $f(x)$, which QTT is $\{A_i^{\sigma_i}\}$, the product $af(x)$ may be



obtained simply by multiplying one of the local tensors by a . A convenient choice is to scale the first tensor,

$$C^{\sigma_1} = a A^{\sigma_1}, \quad C^{\sigma_i} = A^{\sigma_i}, \quad i = 2, \dots, n, \quad (4.7)$$

and the resulting collection $\{C_i^{\sigma_i}\}$ is then the QTT representation of the scaled function $af(x)$. The TN diagram can be represented as

$$\begin{array}{c} aA^{\sigma_1} \quad A^{\sigma_2} \quad A^{\sigma_n} \\ \bullet \quad \bullet \quad \bullet \quad \bullet \\ | \quad | \quad | \quad | \end{array} . \quad (4.8)$$

The same technique applies to MPOs. For MPO addition, the local tensors are combined by direct sums in the same manner as the QTT case, and for scalar multiplication one simply multiplies any single local tensor of the MPO by the constant. The corresponding TN diagram can be represented as

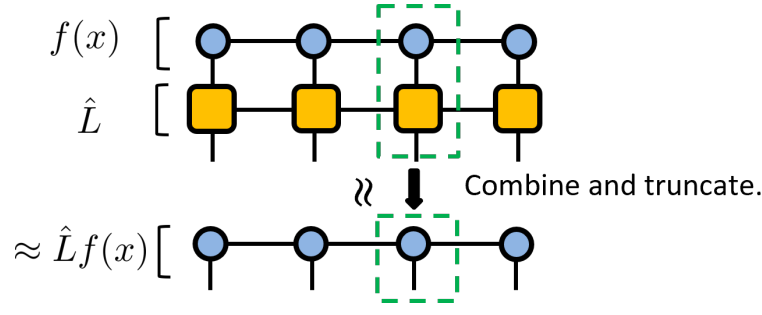
$$\begin{array}{c} \begin{array}{c} \text{Green squares} \\ \text{Red squares} \end{array} + \begin{array}{c} \text{Red squares} \\ \text{Green squares} \end{array} = \begin{array}{c} \text{Blue squares} \\ \text{Block matrix} \end{array} \\ \text{[Green block]} \dots \text{[Green block]} + \text{[Red block]} \dots \text{[Red block]} \rightarrow \text{[Green block]} \dots \text{[Red block]} \rightarrow \text{[Block matrix]} \end{array} . \quad (4.9)$$

2. MPO–MPS operation

A linear operator $\hat{L}$ acting on a function $f(x)$ produces a new function

$$\hat{L}f(x), \quad (4.10)$$

in the usual sense. In the tensor-network setting, the operator $\hat{L}$ is represented as a MPO, and the function f is represented as an MPS. Applying $\hat{L}$ to f is carried out by the MPO–MPS operation: This produces a new sequence of tensors that together represent the function $\hat{L}f(x)$. The entire procedure is illustrated in the tensor-network diagram is



$$\approx \hat{L}f(x) \quad (4.11)$$

During this contraction, bond dimensions generally increase, so an SVD-based truncation is applied afterward to maintain an efficient representation.

3. Pointwise multiplication

Given two functions $f(x)$ and $g(x)$ represented in QTT form with local tensors $\{A_i^{\sigma_i}\}$ and $\{B_i^{\sigma_i}\}$, respectively, we want to construct the QTT representation of their pointwise product [85]

$$h(x) = f(x) g(x) \approx (A^{\sigma_1} A^{\sigma_2} \dots A^{\sigma_n}) (B^{\sigma_1} B^{\sigma_2} \dots B^{\sigma_n}). \quad (4.12)$$

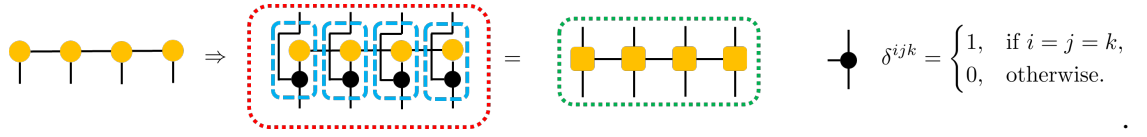
We introduce the super Kronecker delta

$$\delta^{ijk} = \begin{cases} 1, & \text{if } i = j = k, \\ 0, & \text{otherwise,} \end{cases} \quad (4.13)$$

which enforces equality of three binary indices. We then transfer the QTT representation of $g(x)$ into a matrix-product operator (MPO) by attaching the super Kronecker delta to each local tensor. For each site i , we define the MPO tensor

$$W_i^{\sigma_i \sigma'_i} = \sum_{\sigma''_i} \delta^{\sigma_i \sigma'_i \sigma''_i} B_i^{\sigma''_i}, \quad (4.14)$$

or



$$\delta^{ijk} = \begin{cases} 1, & \text{if } i = j = k, \\ 0, & \text{otherwise.} \end{cases} \quad (4.15)$$

The pointwise multiplication of two QTT-represented functions can be expressed directly as

$$\begin{aligned} f(x)g(x) &= A^{\sigma_1} A^{\sigma_2} \dots A^{\sigma_n} B^{\sigma_1} B^{\sigma_2} \dots B^{\sigma_n} \\ &= \sum_{\sigma'} \sum_{\sigma''} \left(\delta^{\sigma_1 \sigma'_1 \sigma''_1} B^{\sigma''_1} \right) \left(\delta^{\sigma_2 \sigma'_2 \sigma''_2} B^{\sigma''_2} \right) \dots \left(\delta^{\sigma_n \sigma'_n \sigma''_n} B^{\sigma''_n} \right) A^{\sigma'_1} A^{\sigma'_2} \dots A^{\sigma'_n} \\ &= \sum_{\sigma'} \left(\delta^{\sigma_1 \sigma'_1} B^{\sigma'_1} \right) \left(\delta^{\sigma_2 \sigma'_2} B^{\sigma'_2} \right) \dots \left(\delta^{\sigma_n \sigma'_n} B^{\sigma'_n} \right) A^{\sigma'_1} A^{\sigma'_2} \dots A^{\sigma'_n}. \end{aligned} \quad (4.16)$$

This contraction is precisely the MPO–MPS operation shown in Eq. 4.11.

4. Inner product

For functions sampled on a uniform grid $\{x_i\}$ with grid spacing h , the inner product is approximated as

$$\int f(x) g(x) dx \approx h \sum_i f(x_i) g(x_i). \quad (4.17)$$

Let the QTT representations of f and g be given by local tensors $\{A_i^{\sigma_i}\}$ and $\{B_i^{\sigma_i}\}$, respectively. Using the binary encoding of the index $i \leftrightarrow (\sigma_1, \dots, \sigma_n)$, the discrete sum appearing in Eq. (4.17) becomes

$$\begin{aligned} \sum_i f(x_i) g(x_i) &\approx \sum_{\sigma_1, \dots, \sigma_n} A^{\sigma_1} A^{\sigma_2} \dots A^{\sigma_n} B^{\sigma_1} B^{\sigma_2} \dots B^{\sigma_n} \\ &= \left(\sum_{\sigma_1} A^{\sigma_1} B^{\sigma_1} \right) \left(\sum_{\sigma_2} A^{\sigma_2} B^{\sigma_2} \right) \dots \left(\sum_{\sigma_n} A^{\sigma_n} B^{\sigma_n} \right). \end{aligned}$$

The tensor-network diagram is

$$\sum_i f(x_i) g(x_i) \approx \begin{array}{ccccccc} & A^{\sigma_1} & A^{\sigma_2} & \dots & A^{\sigma_n} & & \\ & \bullet & \bullet & \dots & \bullet & & \\ & | & | & \dots & | & & \\ & \bullet & \bullet & \dots & \bullet & & \\ & B^{\sigma_1} & B^{\sigma_2} & \dots & B^{\sigma_n} & & \end{array} \quad (4.18)$$

Multiplying by the grid spacing h yields the final approximation of the continuous inner product.

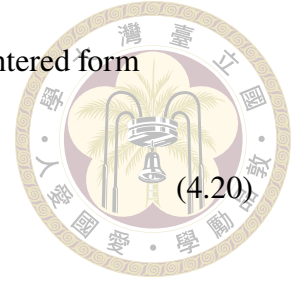
6. Derivatives and Laplacian

A key ingredient in solving partial differential equations is the ability to apply differential operators efficiently. For a one-dimensional function discretized on a uniform grid $\{x_i\}$ with spacing h , the second derivative is approximated by the central finite-difference formula

$$\frac{\partial^2}{\partial x^2} f(x_i) = \frac{1}{h^2} [f(x_{i+1}) - 2f(x_i) + f(x_{i-1}))] + O(h^2). \quad (4.19)$$

Similarly, the first derivative can be approximated by the standard centered form

$$\frac{\partial}{\partial x} f(x_i) = \frac{1}{2h} [f(x_{i+1}) - f(x_{i-1})] + O(h^2). \quad (4.20)$$



The discretized second derivative operator may be written in matrix form as

$$\frac{\partial^2}{\partial x^2} \approx \frac{1}{h^2} \begin{pmatrix} -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & 1 & \ddots & \ddots & \\ & & \ddots & \ddots & 1 \\ & & & 1 & -2 \end{pmatrix}, \quad (4.21)$$

and the first derivative has a similar form with $(1, -1)$ off-diagonal entries. Both operators are special cases of the general tri-diagonal operator

$$T = \begin{pmatrix} \alpha & \beta & & & \\ \gamma & \alpha & \beta & & \\ & \gamma & \ddots & \ddots & \\ & & \ddots & \ddots & \beta \\ & & & \gamma & \alpha \end{pmatrix}, \quad (4.22)$$

where (α, β, γ) take specific values for each differential operator.

Any tri-diagonal operator of the form (4.22) can be written compactly as a matrix-

product operator (MPO) [86]. The MPO tensors for sites $i = 2, \dots, n - 1$ are given by

$$O_i = \begin{pmatrix} I & \sigma^- & \sigma^+ \\ 0 & \sigma^+ & 0 \\ 0 & 0 & \sigma^- \end{pmatrix}, \quad (4.23)$$

where σ^+ and σ^- act on the local binary index and shift it upward or downward by one, selecting contributions from $(i + 1)$ and $(i - 1)$ grid points. The boundary tensors take the form

$$O_1 = \begin{pmatrix} I & \sigma^- & \sigma^+ \end{pmatrix}, \quad O_n = \begin{pmatrix} \alpha I + \beta \sigma^+ + \gamma \sigma^- \\ \gamma \sigma^+ \\ \beta \sigma^- \end{pmatrix}, \quad (4.24)$$

such that contracting $O_1, O_2, \dots, O_n$ reproduces the full tri-diagonal matrix (4.22).

Once the derivative operator is expressed as an MPO, acting on a QTT representation of a function becomes a standard MPO–MPS multiplication, as described in the previous section. If $f(x)$ has QTT tensors $\{A_i^{\sigma_i}\}$, then the derivative $(\partial_x f)(x)$ or $(\partial_x^2 f)(x)$ is obtained by contracting the MPO $\{O_i^{\sigma_i \sigma'_i}\}$ with the MPS $\{A_i^{\sigma'_i}\}$ and (optionally) performing truncation to control the bond dimension. Thus, both first- and second-order derivatives, including the one-dimensional Laplacian, are naturally and efficiently implemented within the QTT/MPS framework. For multidimensional problems, such as the 2D Laplacian, the operator is obtained simply by taking tensor products of the corresponding 1D operators with identity matrices. For example, in Cartesian coordinates,

$$\nabla^2 \approx \left(\frac{\partial^2}{\partial x^2} \right) \otimes I + I \otimes \left(\frac{\partial^2}{\partial y^2} \right),$$

so the 2D Laplacian MPO is constructed directly from the 1D second-derivative MPO and the identity operator.



4.2 Simulation Results in the Cartesian Coordinate

We now present the simulation results in Cartesian coordinates. As a first example, we consider the heat equation

$$u_t(\mathbf{x}, t) = \alpha \nabla^2 u(\mathbf{x}, t), \quad (4.25)$$

with initial condition and boundary condition

$$u(\mathbf{x}, t = 0) = 1, \quad \partial u = 0. \quad (4.26)$$

The corresponding numerical results are shown in Fig. 4.3. For the two-dimensional case, we also perform an error analysis for different bond dimensions. The error is defined as the maximum absolute difference between the TDVP results and the exact diagonalization (ED) solution at the final evolution time. The numerical error values are summarized in Table 4.1. We note that the error remains on the order of 10^{-8} even without any bond-dimension truncation. This residual discrepancy most likely originates from machine-precision rounding in our complex-valued implementation.

We next present simulation results for the two-dimensional Gross–Pitaevskii (GP) equation,

$$\partial_t \psi(\mathbf{x}, t) = -\frac{1}{2} \nabla^2 \psi(\mathbf{x}, t) + V(\mathbf{x}) \psi(\mathbf{x}, t) + \beta |\psi(\mathbf{x}, t)|^2 \psi(\mathbf{x}, t), \quad (4.27)$$

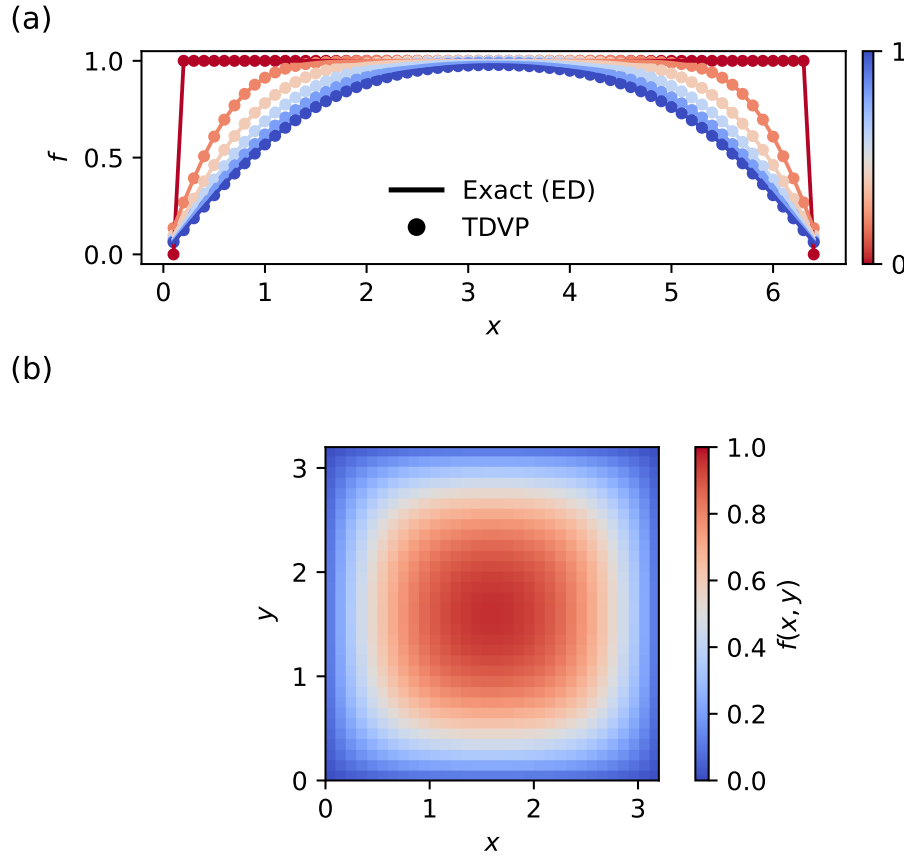


Figure 4.3: Simulation results for the heat equation obtained using the QTT representation without bond-dimension truncation. (a) One-dimensional case with $N = 6$ tensor sites. Solid lines correspond to the exact solution obtained by exact-diagonalization (ES), while markers denote results from the TDVP. (b) Two-dimensional case evolved to $t = 0.25$ using bond dimension $D = 32$, with $N = 5 + 5$ tensor sites for the two spatial directions; only TDVP results are shown. In both simulations, the spatial discretization is $dx = 0.1$ and the initial condition is $u(\mathbf{x}, 0) = 1$. Accordingly, the spatial domain is $x \in [0, 6.4)$ in (a), corresponding to 2^6 sampling points, and $(x, y) \in [0, 3.2) \times [0, 3.2)$ in (b).

Bond dimension D	Error ($\max u_{\text{TDVP}} - u_{\text{ED}} $)
2	4.89×10^{-2}
4	3.96×10^{-6}
8	5.24×10^{-8}
16	2.02×10^{-8}
32	1.87×10^{-8}

Table 4.1: Error analysis for the two-dimensional heat-equation simulation using different bond dimensions D . The error is defined as the maximum absolute difference between TDVP results and the exact diagonalization (ED) benchmark at the final simulation time.

evolved using the TDVP algorithm. The results are summarized in Fig. 4.4. Figure 4.4(a) corresponds to the harmonic trapping potential

$$V(x, y) = -\frac{x^2 + y^2}{2}, \quad (4.28)$$

while Fig. 4.4(b) shows the case of an anisotropic saddle potential,

$$V(x, y) = \frac{x^2 - y^2}{2}. \quad (4.29)$$

In both cases the initial state is chosen to be a Gaussian wave packet,

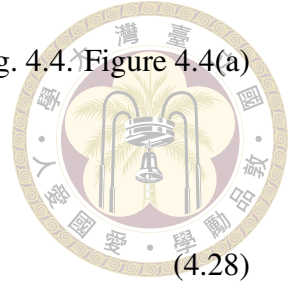
$$\psi(x, y, t = 0) = \frac{1}{\sqrt{\sigma\pi}} \exp\left[-\frac{x^2 + y^2}{2\sigma}\right], \quad (4.30)$$

and the evolution is carried out up to $t = 5.6$. The QTT representation uses $6 + 6$ tensor sites for the two spatial directions. The resulting dynamics obtained via TDVP show agreement with known results in the literature (see Ref. [87]), demonstrating that the QTT–TDVP approach accurately only using bond dimension $D = 6$, captures both the linear and nonlinear contributions of the GP equation.

As another example demonstrating that TDVP can naturally handle coupled partial differential equations, we consider the Maxwell equations in free space. For a charge-free region, the fields satisfy

$$\frac{\partial \mathbf{E}}{\partial t} = \frac{1}{\varepsilon} \nabla \times \mathbf{H}, \quad \frac{\partial \mathbf{H}}{\partial t} = -\frac{1}{\mu} \nabla \times \mathbf{E}. \quad (4.31)$$

Specializing to one spatial dimension and free space with $\varepsilon = \mu = 1$, the system reduces



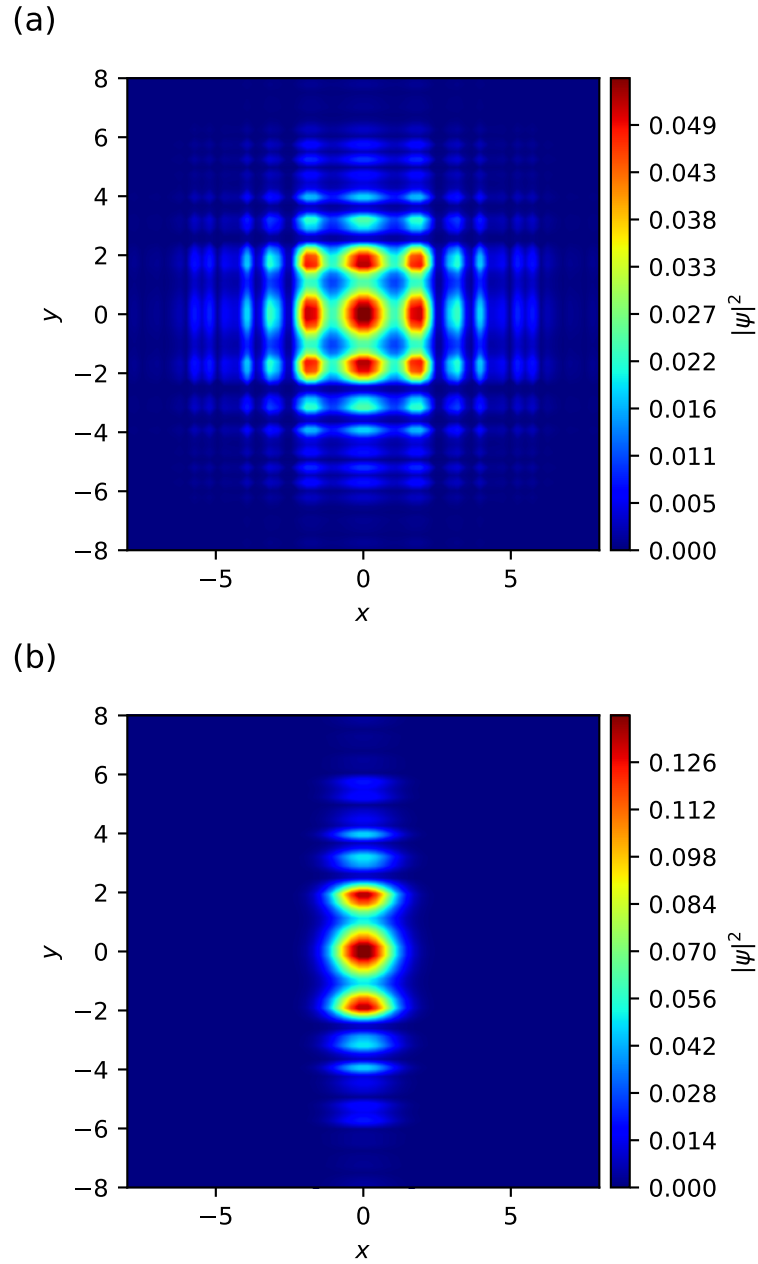


Figure 4.4: Simulation results for the two-dimensional Gross–Pitaevskii equation ($\beta = 1$) using the TDVP algorithm with bond dimension $D = 6$. (a) Evolution under the harmonic potential $V(x, y) = -(x^2 + y^2)/2$. (b) Evolution under the saddle potential $V(x, y) = (x^2 - y^2)/2$. In both cases the initial state is the Gaussian $\psi(x, y, 0) = \frac{1}{\sqrt{\sigma\pi}} \exp[-(x^2 + y^2)/(2\sigma)]$. The QTT representation uses $6 + 6$ tensor sites for the two spatial directions, and the simulation is evolved to time $t = 5.6$.

to

$$\frac{\partial E_x}{\partial t} = -\frac{\partial H_y}{\partial z}, \quad \frac{\partial H_y}{\partial t} = -\frac{\partial E_x}{\partial z}. \quad (4.32)$$



Unlike the previous scalar PDE examples, this system involves two coupled fields, so we represent $E_x(z, t)$ and $H_y(z, t)$ using two separate QTT tensor trains and evolve them simultaneously using two coupled TDVP equations. The computational structure is illustrated in Fig. 4.5.

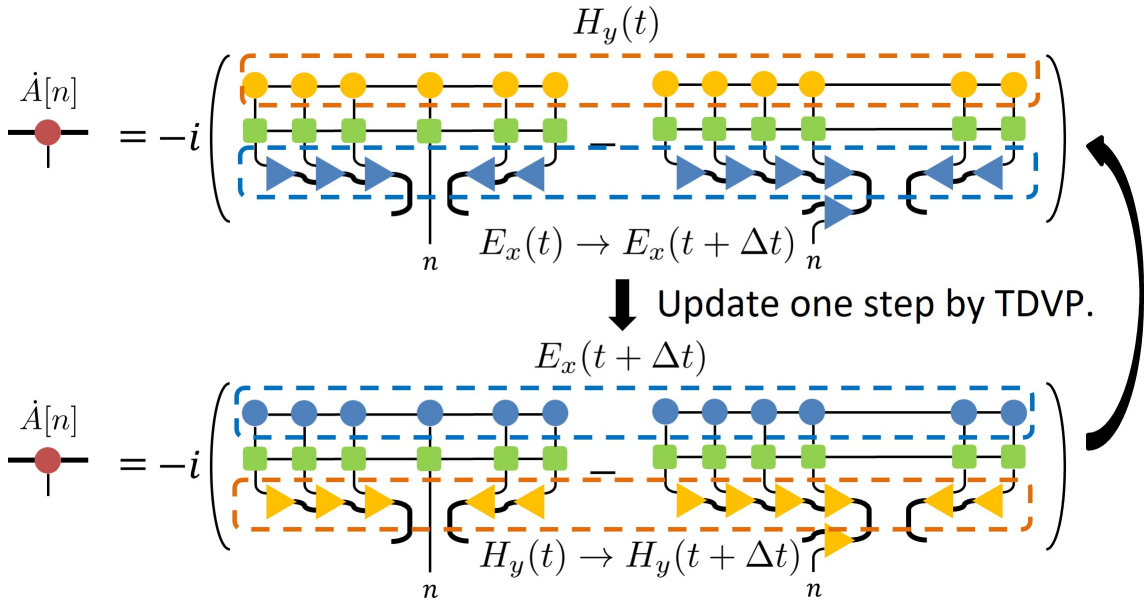
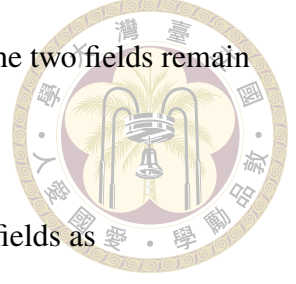


Figure 4.5: Tensor-network structure of the TDVP algorithm applied to the 1D Maxwell equations. Two QTT tensor trains are used to represent the coupled fields: the electric field $E_x(z, t)$ (shown in blue) and the magnetic field $H_y(z, t)$ (shown in yellow). Each TDVP update requires constructing the projection operator for one field using the current state of the other: the TDVP projection for E_x is built from H_y , while the projection for H_y is built from E_x . The two fields are then updated alternately using the Euler method.

To time-evolve the coupled system, we solve the two TDVP equations using a simple Euler method with time step $dt = 0.01$. At each step, we first update the electric field $E_x(z, t)$ using the TDVP projection onto the tangent space generated by $H_y(t)$, obtaining $E_x(t + dt)$. Then, using this updated electric field, we evolve the magnetic field $H_y(z, t)$ via its corresponding TDVP projection. This sequential update procedure reflects the in-

trinsic coupling structure of Maxwell' s equations and ensures that the two fields remain dynamically consistent.



We impose periodic boundary conditions and choose the initial fields as

$$E_x(z, 0) = \exp\left[-\frac{(z - 0.5)^2}{2 \cdot 0.05^2}\right], \quad H_y(z, 0) = 0, \quad (4.33)$$

corresponding to a localized Gaussian electric pulse. The numerical solution, shown in Fig. 4.6, exhibits the expected physical behavior: the electric field splits and propagates freely along $\pm z$ with unit speed, while the magnetic field is generated through the spatial derivative of the electric field. This example demonstrates that the QTT–TDVP framework can efficiently simulate coupled vector PDEs in addition to scalar equations.

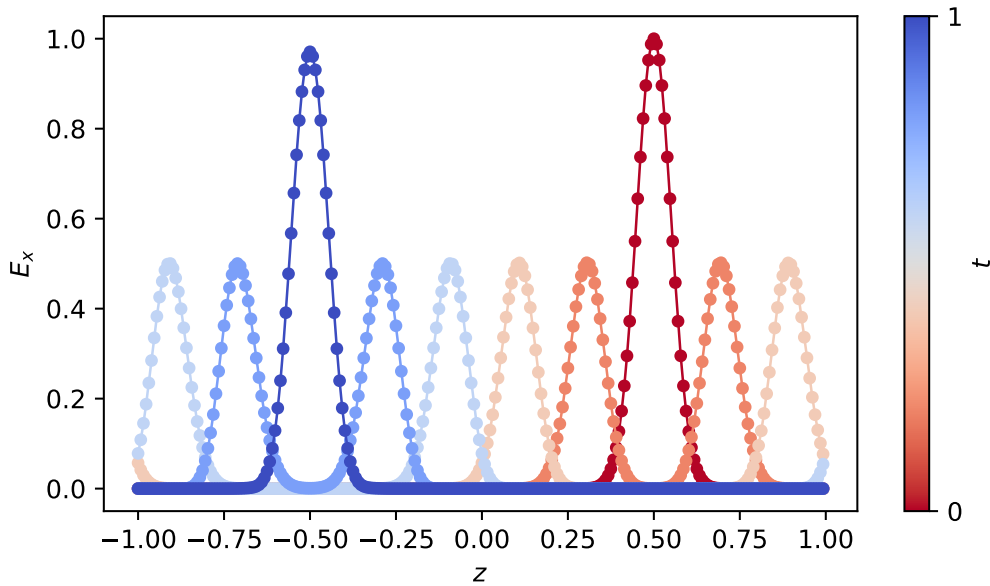


Figure 4.6: Simulation results for the 1D Maxwell equations obtained using the QTT–TDVP method. Periodic boundary conditions are imposed. The initial electric field is $E_x(z, 0) = \exp\left[-(z - 0.5)^2/(2 \cdot 0.05^2)\right]$ and the initial magnetic field is $H_y(z, 0) = 0$. Time evolution is carried out using a forward Euler update with time step $dt = 0.01$ until $t = 1.0$. The results show that the initial Gaussian electric pulse splits into two components propagating freely in the $\pm z$ directions with unit speed.

4.3 QTT Representation in Polar Coordinates



The quantics tensor train (QTT) format provides a compact representation of discretized functions by exploiting low-rank tensor structures. In its simplest setting, a one-dimensional function $f(x)$ sampled on a grid can be efficiently encoded in QTT form, and this construction extends naturally to higher-dimensional functions such as $f(x, y)$ or $f(x, y, z)$. These representation has been widely studied in Cartesian coordinates, where it enables efficient storage and manipulation of high-dimensional data. In contrast, much less attention has been devoted to formulating QTT representations in curvilinear coordinate systems [88], such as polar or spherical coordinates. Developing such representations is essential for applying QTT methods to problems in which symmetry or boundary conditions are more naturally expressed in non-Cartesian coordinates.

A two-dimensional function $f(x, y)$ is represented in the QTT format as illustrated in Fig. 4.7. In this representation, we denote the maximum bond dimensions as D_L , D_M ,

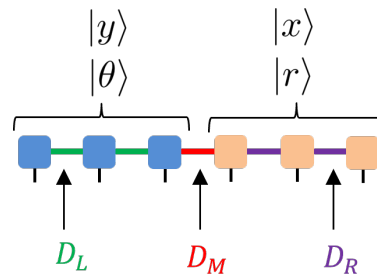


Figure 4.7: QTT representation of a two-dimensional function $f(x, y)$ or $f(r, \theta)$.

and D_R , corresponding respectively to the x (or r) part, the intermediate bond between the $x(r)$ and $y(\theta)$ components, and the $y(\theta)$ part. To assess the performance of the QTT representation in Cartesian and polar coordinates, we consider two test functions with

radial symmetry. Case (a) is a Gaussian function,

$$f(r, \theta) = \frac{1}{\sqrt{\sigma\pi}} e^{-r^2/2\sigma}, \quad (4.34)$$

which exhibits smooth radial decay. Case (b) is

$$f(r, \theta) = \sin\left(\frac{1}{r}\right) + \frac{\sin(r)}{1+r^2}, \quad (4.35)$$

which contains strong oscillations near the origin. For each case, the function is discretized

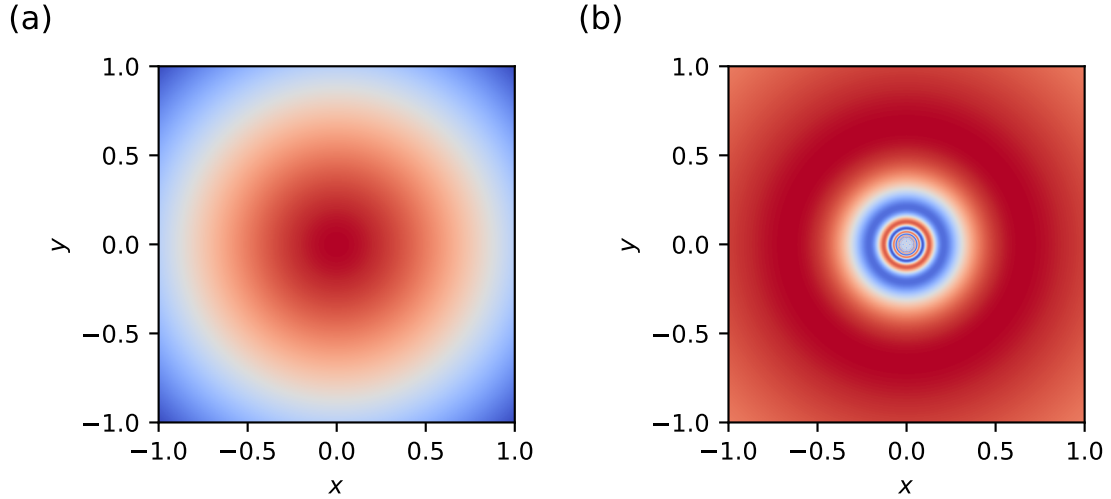


Figure 4.8: Test functions used to evaluate the QTT representation in Cartesian and polar coordinates, corresponding to Eq. (4.34) and Eq. (4.35). (a) Gaussian function, which is smooth and radially symmetric. (b) Oscillatory function, which exhibits strong oscillations near the origin.

and mapped into a full tensor $c_{\sigma_1 \dots \sigma_N}$, which is subsequently converted into QTT form by successive singular value decompositions (SVD). Each dimension is represented by 10 sites, such that the two-dimensional functions are encoded as a 20-site MPS. The bond dimension is truncated to χ to control the accuracy of the representation. To quantify the accuracy of the QTT approximation in Cartesian and polar coordinates, we define the

relative error in the Frobenius norm as

$$\varepsilon = \frac{\|f_{\text{QTT}} - f_{\text{exact}}\|_F}{\|f_{\text{exact}}\|_F}.$$



In addition, we investigate the entanglement properties of the representation by computing the entanglement entropy across each bond. Let $\{s_i\}$ denote the singular values obtained from the SVD after normalization, such that $\sum_i s_i^2 = 1$. The entropy is then given by

$$S = - \sum_i s_i^2 \log s_i^2.$$

For the Cartesian coordinate system, we set the bond dimensions to $D_L = D_M = D_R = \chi$ (see Fig. 4.7). In contrast, for the polar coordinate system we use $D_L = \chi$ while $D_M = D_R = 2$. The simulation results are shown in Fig. 4.9 and Fig. 4.10. As seen in Fig. 4.9, for the Gaussian function the Cartesian and polar representations both achieve small errors without requiring large bond dimensions. However, for the oscillatory function, which depends strongly on the radial coordinate, the Cartesian representation requires $\chi \approx 80$ to reach errors close to machine precision, whereas the polar representation achieves the same accuracy with only $\chi \approx 15$ in the radial sector (with $D_R = 2$ for the angular part). This leads to a substantial memory saving in the polar coordinate system. The entanglement entropy analysis in Fig. 4.10 further confirms this behavior: in the polar representation, significant entanglement is present only in the radial bonds, while the entanglement entropy for the angular bonds are zeros.

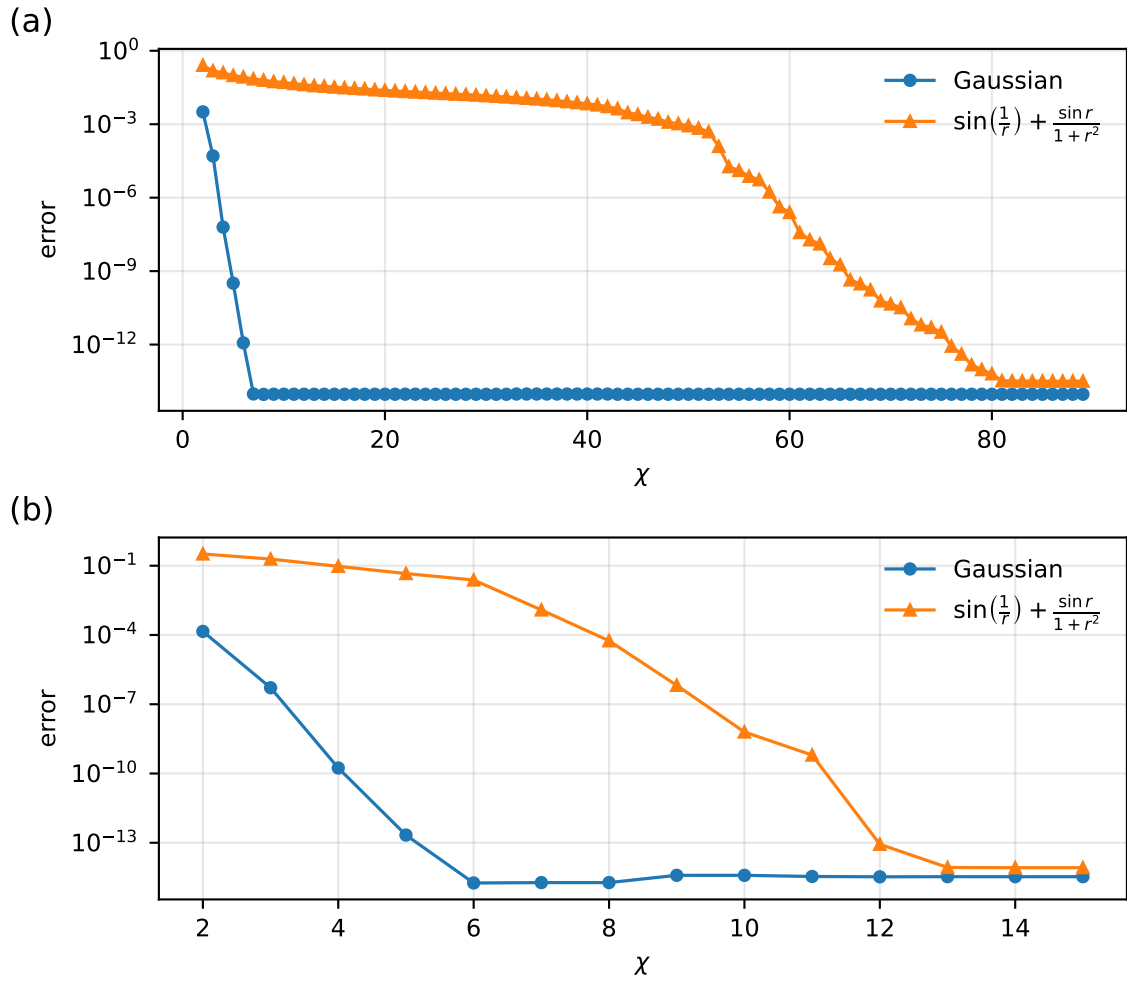
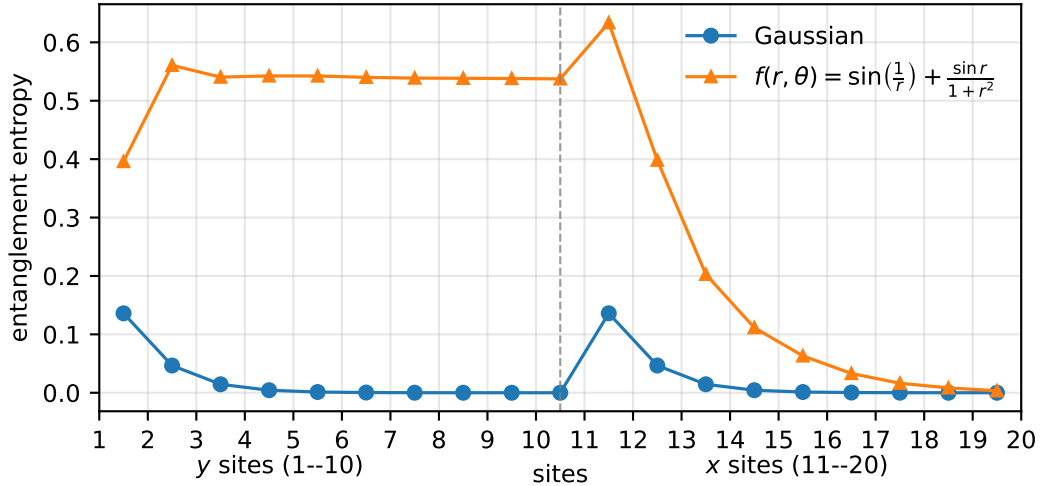


Figure 4.9: Relative approximation error of the QTT representation as a function of bond dimension χ for two-dimensional functions. (a) Cartesian coordinate representation. (b) Polar coordinate representation. Results are shown for a Gaussian test function and for the oscillatory function $f(r, \theta) = \sin\left(\frac{1}{r}\right) + \frac{\sin r}{1+r^2}$.



(a)



(b)

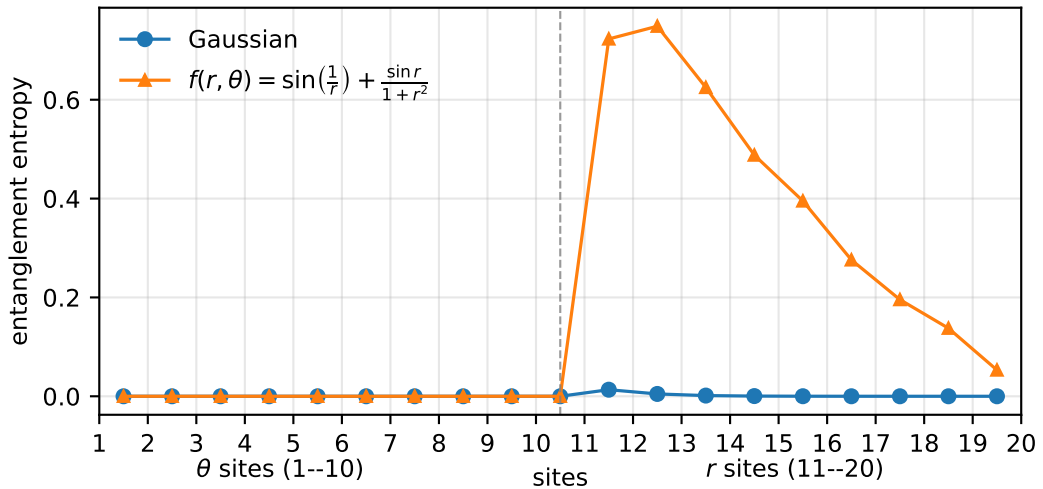


Figure 4.10: Entanglement entropy across the bonds of the QTT representation in Cartesian and polar coordinates. (a) Cartesian representation, where tensor sites 1–10 correspond to the y direction and sites 11–20 to the x direction. (b) Polar representation, where tensor sites 1–10 correspond to the angular coordinate θ and sites 11–20 to the radial coordinate r . In the polar representation, only the radial bonds show significant entanglement, while the angular bonds remain unentangled.

4.4 Singularity at the Origin



Several PDEs we consider, such as the heat equation, the Schrödinger equation, and the Gross–Pitaevskii (GP) equation, involve the Laplacian operator. As shown in the previous section, when the system exhibits radial symmetry, it is advantageous to represent the state in polar coordinates within the QTT framework, since the angular direction requires only a small bond dimension. However, the Laplacian operator in polar coordinates,

$$\nabla^2 = \frac{\partial^2}{\partial r^2} + \frac{1}{r} \frac{\partial}{\partial r} + \frac{1}{r^2} \frac{\partial^2}{\partial \theta^2}, \quad (4.36)$$

is singular at the origin due to the $1/r$ and $1/r^2$ terms. In this section, we describe how to regularize this singular behavior.

We examine the discretization of the Laplacian operator for a two-dimensional function $f : \mathbb{R}^2 \rightarrow \mathbb{R}$. To illustrate the differences between Cartesian and polar discretizations, we consider two sample points as shown in Fig. 4.11(a). When using a Cartesian grid, the Laplacian at a grid point (x_i, y_j) is approximated by combining the values of f at the left–right and up–down neighboring points,

$$\begin{aligned} \nabla^2 f(x_i, y_j) \approx & \frac{1}{\Delta x^2} [f(x_{i-1}, y_j) - 2f(x_i, y_j) + f(x_{i+1}, y_j)] \\ & + \frac{1}{\Delta y^2} [f(x_i, y_{j-1}) - 2f(x_i, y_j) + f(x_i, y_{j+1})]. \end{aligned} \quad (4.37)$$

This expression is well-defined everywhere on the grid, including the origin, because the Cartesian Laplacian contains no coordinate singularities.

In polar coordinates, the Laplacian at a point (r_i, θ_j) with $r_i > 0$ is computed using

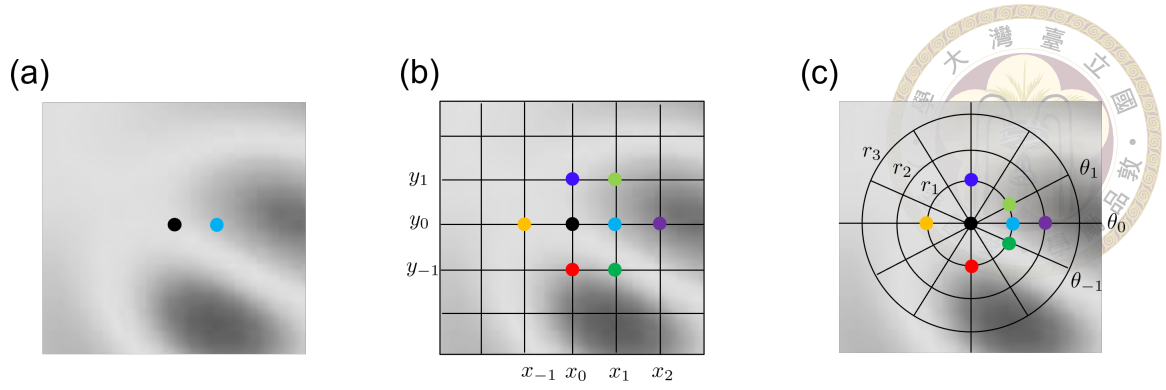


Figure 4.11: Illustration of Cartesian and polar discretizations for evaluating the Laplacian of a two-dimensional function. (a) Two representative sample points of the function $f(\mathbf{x})$: a blue point and a black point. (b) Cartesian discretization of the same region, where both the blue and black points lie on a regular grid and their Laplacians can be estimated directly using the standard finite-difference stencil in Eq. (4.37). (c) Polar discretization of the same region, where grid points are arranged along radial and angular directions. For the blue point at $r_i > 0$, the Laplacian can be evaluated directly using the polar finite-difference formula in Eq. (4.38). However, the black point lies at the origin $r_0 = 0$, where the polar Laplacian becomes singular and cannot be evaluated directly. In this case, the Laplacian at the origin is instead estimated using a Cartesian finite-difference stencil and subsequently re-expressed in (r, θ) form, as described in Eq. 4.44.

the neighboring values in both the radial and angular directions,

$$\begin{aligned} \nabla^2 f(r_i, \theta_j) \approx & \frac{1}{\Delta r^2} [f(r_{i-1}, \theta_j) - 2f(r_i, \theta_j) + f(r_{i+1}, \theta_j)] \\ & + \frac{1}{2r_i \Delta r} [f(r_{i+1}, \theta_j) - f(r_{i-1}, \theta_j)] \\ & + \frac{1}{r_i^2 \Delta \theta^2} [f(r_i, \theta_{j-1}) - 2f(r_i, \theta_j) + f(r_i, \theta_{j+1})]. \end{aligned} \quad (4.38)$$

This formula is valid for all $r_i > 0$, but it becomes singular at $r_0 = 0$ because of the terms $1/r_i$ and $1/r_i^2$. Consequently, the polar form of the Laplacian cannot be evaluated directly at the origin. This geometric difference between Cartesian and polar discretizations is illustrated schematically in Fig. 4.11.

To express the polar discretization in a compact form, we rewrite its components in matrix notation. The radial second derivative, angular second derivative, and radial first

derivative operators are represented as

$$\partial_{rr} \approx \frac{1}{\Delta r^2} \begin{pmatrix} -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & 1 & \ddots & \ddots & \\ & & \ddots & \ddots & 1 \\ & & & 1 & -2 \end{pmatrix} \equiv D_{rr}, \quad (4.39)$$

$$\partial_{\theta\theta} \approx \frac{1}{\Delta\theta^2} \begin{pmatrix} -2 & 1 & & & 1 \\ 1 & -2 & 1 & & \\ & 1 & \ddots & \ddots & \\ & & \ddots & \ddots & 1 \\ 1 & & & 1 & -2 \end{pmatrix} \equiv D_{\theta\theta}, \quad (4.40)$$

$$\partial_r \approx \frac{1}{2\Delta r} \begin{pmatrix} 0 & 1 & & & \\ -1 & 0 & 1 & & \\ & -1 & \ddots & \ddots & \\ & & \ddots & \ddots & 1 \\ & & & -1 & 0 \end{pmatrix} \equiv D_r. \quad (4.41)$$

Defining the diagonal matrix of radii,

$$R = \text{diag}(r_0, r_1, \dots, r_{n-1}), \quad (4.42)$$

the Laplacian on a polar grid can be written in the Kronecker-product form

$$\nabla^2 \approx D_{rr} \otimes I + R^{-1} D_r \otimes I + R^{-2} \otimes D_{\theta\theta}. \quad (4.43)$$



This expression makes the singular behavior explicit: R^{-1} and R^{-2} diverge at $r_0 = 0$.

Although the polar formula cannot be applied at the origin, the Laplacian there can still be determined by recognizing that (r_0, θ_j) for all θ_j corresponds to the same Cartesian point $(x_0, y_0) = (0, 0)$. We therefore evaluate the origin Laplacian using the Cartesian expression, but interpret the required neighboring points in polar coordinates. Each neighbor of the origin lies at a radius r_1 with angular offsets of $\theta_j, \theta_j + \pi/2, \theta_j + \pi$, and $\theta_j + 3\pi/2$. For a general angle θ_j , this leads to

$$\begin{aligned} \nabla^2 f(r_0, \theta_j) \approx & \frac{1}{\Delta r^2} [f(r_1, \theta_j + \pi) - 2f(r_0, \theta_j) + f(r_1, \theta_j)] \\ & + \frac{1}{\Delta r^2} [f(r_1, \theta_j - \frac{\pi}{2}) - 2f(r_0, \theta_j) + f(r_1, \theta_j + \frac{\pi}{2})]. \end{aligned} \quad (4.44)$$

Thus, the Laplacian at the origin requires only four surrounding points on the circle r_1 for each angle θ_j .

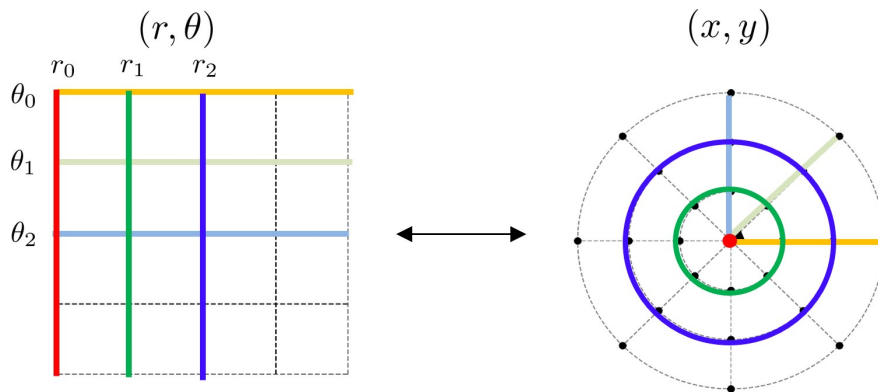


Figure 4.12: Illustration of the non-unique representation of the origin in polar coordinates. Left: Discretized polar grid indexed by (r_i, θ_j) , where all points (r_0, θ_j) (highlighted by the red vertical line) correspond to distinct tensor entries in the polar discretization. Right: Mapping of the same grid to the physical x - y plane. All points (r_0, θ_j) collapse to the same physical location at the origin (marked by the red dot), while points with r_1 and r_2 form concentric rings.

A further complication arises from the fact that the origin is not uniquely represented in polar coordinates. As illustrated in Fig. 4.12, all grid points of the form (r_0, θ_j) cor-

respond to the same physical location (x_0, y_0) in Cartesian coordinates. Although these points are identical geometrically, in a numerical computation, they are treated as distinct entries of the discretized function in polar coordinates. As a consequence, if we evaluate $\nabla^2 f(r_0, \theta_j)$ separately for each angle using the Cartesian-based construction derived above, the updated values $f(r_0, \theta_j)$ may differ across j . This contradicts the physical requirement that the function must be single-valued at the origin and remain rotationally symmetric.

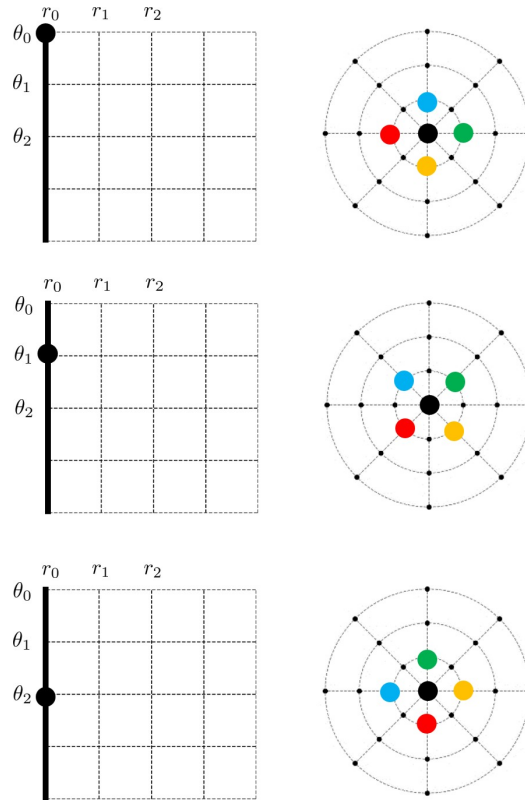


Figure 4.13: Angular dependence of the Laplacian construction at the origin in polar coordinates. Each row corresponds to a different angular index θ_j . Left: polar grid representation showing the origin at (r_0, θ_j) . Right: the corresponding real-space neighborhood used to evaluate directional second derivatives. Because each θ_j defines a different Cartesian neighborhood around the origin, the ideal Laplacian at r_0 must be constructed by averaging over all angular directions, as described in Eq. (4.45).

A more careful inspection reveals that the ideal polar Laplacian at the origin should be constructed by including contributions from all angular directions simultaneously. Be-

cause all (r_0, θ_j) represent the same point, the correct value of the Laplacian must be obtained by averaging the directional second derivatives coming from every θ_j (see Fig. 4.13).

Denoting the number of angular points by N_θ , we obtain

$$\begin{aligned} \nabla^2 f(r_0, \theta) \approx \frac{1}{\Delta r^2} \frac{1}{N_\theta} \sum_{j=0}^{N_\theta-1} & \left[f(r_1, \theta_j + \pi) - 2f(r_0, \theta_j) + f(r_1, \theta_j) \right. \\ & \left. + f(r_1, \theta_j - \frac{\pi}{2}) - 2f(r_0, \theta_j) + f(r_1, \theta_j + \frac{\pi}{2}) \right]. \end{aligned} \quad (4.45)$$

Rearranging the terms separates the contributions at r_1 and at r_0 ,

$$\begin{aligned} \nabla^2 f(r_0, \theta) = & \frac{1}{\Delta r^2} \frac{1}{N_\theta} \left[\sum_i f(r_1, \theta_i + \pi) + \sum_i f(r_1, \theta_i) + \sum_i f(r_1, \theta_i - \frac{\pi}{2}) + \sum_i f(r_1, \theta_i + \frac{\pi}{2}) \right] \\ & - \frac{1}{\Delta r^2} \frac{1}{N_\theta} \left[4 \sum_i f(r_0, \theta_i) \right]. \end{aligned} \quad (4.46)$$

Because a shift of the angular index corresponds to the same set of points,

$$\sum_i f(r_1, \theta_i + \alpha) = \sum_i f(r_1, \theta_i), \quad \text{for any shift } \alpha, \quad (4.47)$$

the expression simplifies to

$$\begin{aligned} \nabla^2 f(r_0, \theta) &= \frac{1}{\Delta r^2} \frac{1}{N_\theta} \left[4 \sum_i f(r_1, \theta_i) - 4 \sum_i f(r_0, \theta_i) \right] \\ &= \frac{4}{\Delta r^2} \left(\frac{1}{N_\theta} \sum_i f(r_1, \theta_i) - \frac{1}{N_\theta} \sum_i f(r_0, \theta_i) \right). \end{aligned} \quad (4.48)$$

Using the angular averaging operator

$$O_{\text{avg}} g(r_0, \theta_i) = \frac{1}{N_\theta} \sum_{j=0}^{N_\theta-1} g(r_0, \theta_j), \quad (4.49)$$

the ideal Laplacian at the origin can be written compactly as

$$\nabla^2 f(r_0, \theta) = O_{\text{avg}} \tilde{\nabla}^2 f(r_0, \theta), \quad (4.50)$$



where $\tilde{\nabla}^2$ is the Laplacian assuming local radial symmetry at r_0 :

$$\tilde{\nabla}^2 f(r_0, \theta_i) = \frac{1}{\Delta r^2} [4f(r_1, \theta_i) - 4f(r_0, \theta_i)]. \quad (4.51)$$

In practical numerical evolution, however, we do not first apply O_{avg} and then update f . Instead, the algorithm proceeds in the following order:

1. we update the function using the locally radial operator $\tilde{\nabla}^2$, and
2. we apply O_{avg} after the update to enforce that all angular components at r_0 remain identical.

This ordering is used because we strictly constrain all values $f(r_0, \theta_i)$ to be equal for every i , and therefore the averaging step must be applied after the update to restore this single-valuedness at the origin.

Let $f(t)$ denote the function at time t satisfied $f(t + dt) = \nabla^2 f(t)$. One time step of size dt is implemented as

$$\begin{aligned} f(t + dt) &= O_{\text{avg}} \tilde{\nabla}^2 f(t) = O_{\text{avg}} \left(f(t) + dt \tilde{\nabla}^2 f(t) \right) + O(dt^2) \\ &= f(t) + dt O_{\text{avg}} \tilde{\nabla}^2 f(t) + O(dt^2). \end{aligned} \quad (4.52)$$

Because we enforce

$$O_{\text{avg}} f(t) = f(t) \quad (\text{all } \theta \text{ components at } r_0 \text{ equal}), \quad (4.53)$$



the evolution becomes

$$f(t + dt) = f(t) + dt \nabla^2 f(t) + O(dt^2), \quad (4.54)$$

since the correct Laplacian satisfies

$$\nabla^2 = O_{\text{avg}} \tilde{\nabla}^2. \quad (4.55)$$

Thus the “update $\rightarrow$ averaging” procedure is accurate to second order in dt and correctly reproduces the Laplacian at the origin.

With this understanding, we may now write the full discretized Laplacian in polar coordinates as

$$\nabla^2 f(r_i, \theta_j) \approx \begin{cases} \frac{1}{\Delta r^2} [4f(r_1, \theta_j) - 4f(r_0, \theta_j)], & i = 0, \\ \frac{1}{\Delta r^2} [f(r_{i-1}, \theta_j) - 2f(r_i, \theta_j) + f(r_{i+1}, \theta_j)] \\ \quad + \frac{1}{2r_i \Delta r} [f(r_{i+1}, \theta_j) - f(r_{i-1}, \theta_j)] \\ \quad + \frac{1}{r_i^2 \Delta \theta^2} [f(r_i, \theta_{j-1}) - 2f(r_i, \theta_j) + f(r_i, \theta_{j+1})], & i > 0. \end{cases} \quad (4.56)$$

The matrix representation of the full discretized operator becomes



$$\begin{aligned}
 \nabla^2 \approx & \frac{1}{\Delta r^2} \begin{pmatrix} -4 & 4 & & & \\ 1 & -2 & 1 & & \\ & 1 & \ddots & \ddots & \\ & & \ddots & \ddots & 1 \\ & & & 1 & -2 \end{pmatrix} \otimes I \\
 & + R^P \frac{1}{2\Delta r} \begin{pmatrix} 0 & 1 & & & \\ -1 & 0 & 1 & & \\ & -1 & \ddots & \ddots & \\ & & \ddots & \ddots & 1 \\ & & & -1 & 0 \end{pmatrix} \otimes I \\
 & + (R^P)^2 \otimes \frac{1}{\Delta \theta^2} \begin{pmatrix} -2 & 1 & & 0 & 1 \\ 1 & -2 & 1 & & 0 \\ & 1 & \ddots & \ddots & \\ 0 & & \ddots & \ddots & 1 \\ 1 & 0 & & 1 & -2 \end{pmatrix},
 \end{aligned} \tag{4.57}$$

where

$$R^P = \text{diag}(0, r_1^{-1}, r_2^{-1}, \dots), \tag{4.58}$$

is a pseudo-inverse of $R = \text{diag}(r_0, r_1, \dots)$ in which the undefined entries at the origin have been replaced by zero to regularize the operator.

This hybrid construction resolves the coordinate singularity at r_0 , enforces the single-

valuedness of the function at the origin, and retains the full accuracy of the polar-coordinate discretization for all other points. In the next section, we incorporate this operator into the MPO representation used for TDVP evolution.



4.5 MPO Representation in Polar Coordinates

In the previous section we derived the discretized Laplacian in polar coordinates, Eq. (4.57), which is suitable for QTT–TDVP evolution. To use this operator within tensor-network simulations, we must express it as a matrix product operator (MPO). Two ingredients are essential: (i) constructing the Laplacian operator itself as an MPO, including the special treatment of the origin, and (ii) constructing the angular-averaging operator O_{avg} as an MPO to enforce single-valuedness at r_0 .

The most delicate part of the MPO construction arises from the first and final term in Eq. (4.57). The first part contains

$$L' = \begin{pmatrix} -4 & 4 & & \\ 1 & -2 & 1 & \\ & 1 & \ddots & 1 \\ & & 1 & -2 \end{pmatrix}, \quad (4.59)$$

whose top row differs from the standard second-derivative stencil. To rewrite L' in a form

that can be factorized into an MPO, we decompose it as

$$L' = \underbrace{\begin{pmatrix} -2 & 1 & & \\ 1 & -2 & 1 & \\ & 1 & \ddots & 1 \\ & & 1 & -2 \end{pmatrix}}_L - 2 \underbrace{\begin{pmatrix} 1 & 0 & & \\ 0 & \ddots & & \\ & & \ddots & \\ & & & 0 \end{pmatrix}}_{a \otimes \cdots \otimes a \otimes a} + 3 \underbrace{\begin{pmatrix} 0 & 1 & & \\ 0 & 0 & & \\ & & \ddots & \\ & & & 0 \end{pmatrix}}_{a \otimes \cdots \otimes a \otimes \sigma^+}. \quad (4.60)$$

Here we use the projectors and ladder operators

$$a = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \quad \sigma^+ = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \quad \sigma^- = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}. \quad (4.61)$$

With this decomposition, the Laplacian block L' can be implemented using a three-channel MPO. Its left, bulk, and right tensors take the form

$$L' = \begin{pmatrix} W_L & a & a \end{pmatrix} \begin{pmatrix} W_M & 0 & 0 \\ 0 & a & 0 \\ 0 & 0 & a \end{pmatrix}^{n-2} \begin{pmatrix} W_R \\ -2a \\ 3\sigma^+ \end{pmatrix}, \quad (4.62)$$

where W_L , W_M and W_R generate the standard second-derivative stencil L , which can be constructed using Eq. (4.23) and Eq. (4.24). The additional two channels propagate the a and σ^+ contributions necessary to reconstruct the modified first row in Eq. (4.59). For the final term in Eq. (4.57), the angular part contains the periodic Laplacian, which in matrix

form reads

$$L' = \begin{pmatrix} -2 & 1 & & 1 \\ 1 & -2 & 1 & \\ & 1 & \ddots & \ddots \\ & & \ddots & -2 & 1 \\ 1 & & & 1 & -2 \end{pmatrix}. \quad (4.63)$$



This can be decomposed into the open-boundary Laplacian plus two corner couplings:

$$L' = \underbrace{\begin{pmatrix} -2 & 1 & & \\ 1 & -2 & 1 & \\ & 1 & \ddots & \ddots \\ & & \ddots & -2 & 1 \\ & & & 1 & -2 \end{pmatrix}}_L + \underbrace{\begin{pmatrix} 0 & 0 & & 1 \\ & 0 & & \\ & & \ddots & \\ & & & 0 \\ & & & & 0 \end{pmatrix}}_{\sigma^+ \otimes \dots \otimes \sigma^+} + \underbrace{\begin{pmatrix} 0 & & & \\ & 0 & & \\ & & \ddots & \\ & & & 0 \\ 1 & & & & 0 \end{pmatrix}}_{\sigma^- \otimes \dots \otimes \sigma^-}. \quad (4.64)$$

Here σ^+ and σ^- are the 2×2 ladder operators

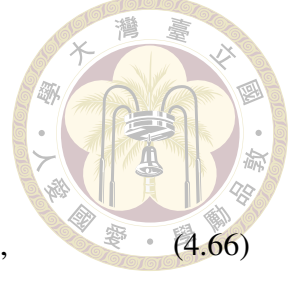
$$\sigma^+ = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \quad \sigma^- = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}, \quad (4.65)$$

whose tensor products generate the upper-right and lower-left corner elements of the periodic Laplacian.

Using a construction analogous to the non-periodic case, L' admits a three-channel

MPO representation of the form

$$L' = \begin{pmatrix} W_L & \sigma^+ & \sigma^- \end{pmatrix} \begin{pmatrix} W_M & 0 & 0 \\ 0 & \sigma^+ & 0 \\ 0 & 0 & \sigma^- \end{pmatrix}^{n-2} \begin{pmatrix} W_M \\ \sigma^+ \\ \sigma^- \end{pmatrix}, \quad (4.66)$$



where W_L and W_M are the MPO tensors corresponding to the open-boundary Laplacian L , and the second and third channels propagate the σ^+ and σ^- contributions needed to generate the periodic corner couplings in Eq. (4.64).

For the pseudo-inverse R^P , we first generate a QTT representation R^P using the tensor cross interpolation (TCI) algorithm, and then convert it to an MPO using the QTT-to-MPO procedure described earlier (see Sec. 4.1). Then we complete the Laplacian represented as MPOs.

4.6 Simulation Results in Polar Coordinates

With the polar-coordinate Laplacian and the corresponding MPO construction developed in the previous sections, we now apply the QTT-TDVP framework to time-evolve several representative partial differential equations. We consider three classes of PDEs commonly used in mathematical physics: the heat equation, the Schrödinger equation for the two-dimensional harmonic oscillator, and the Gross–Pitaevskii (GP) equation.

It is important to note that the Laplacian operator in polar coordinates, as defined in Eq. (4.57), is in general not Hermitian. The same is true for first-order derivative operators appearing in the considered equations. Consequently, the effective local generators

encountered during the TDVP evolution are not guaranteed to be Hermitian. While the standard TDVP algorithm often employs a Lanczos-based approximation to construct the local time-evolution operator under the assumption of a Hermitian operator [89], in the present work we instead adopt a fourth-order Runge–Kutta (RK4) scheme to update the local tensors. This integrator does not rely on Hermiticity and is therefore applicable to the more general, non-Hermitian operators arising in our polar-coordinate formulation.

4.6.1 Heat Equation in Polar Coordinates

We first examine the two-dimensional heat equation in polar coordinates,

$$\partial_t u(r, \theta, t) = \alpha \nabla^2 u(r, \theta, t), \quad (4.67)$$

Figure 4.14 shows the TDVP simulation results for the initial condition $u(r, \theta, 0) = 1$, evolved to time $t = 0.25$. The solution remains radially symmetric throughout the evolution, demonstrating that the tensor representation preserves rotational symmetry and properly handles the coordinate singularity at the origin.

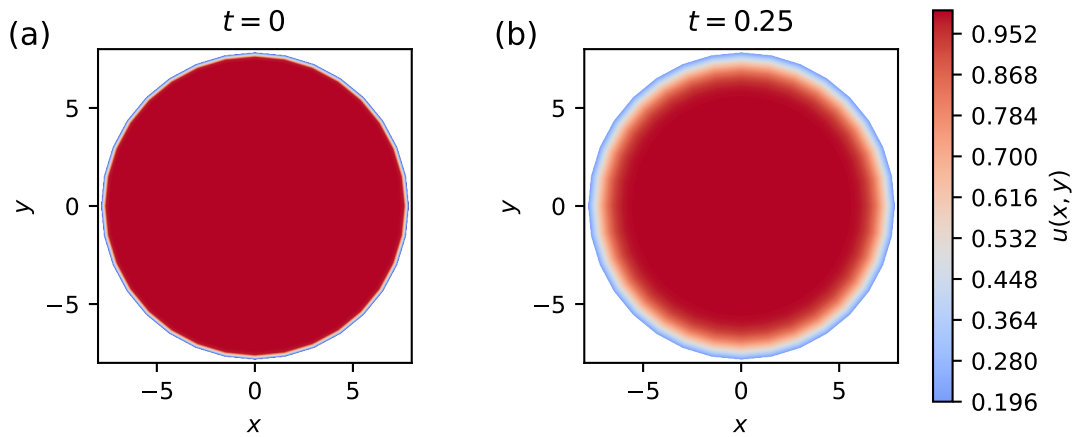


Figure 4.14: TDVP simulation of the heat equation (4.67) in polar coordinates. (a) The initial state is $u(r, \theta, 0) = 1$, and (b) the evolution is shown at $t = 0.25$.

We also consider the heat equation with a source term,

$$\partial_t u(r, \theta, t) = \alpha \nabla^2 u(r, \theta, t) + S(r, \theta), \quad (4.68)$$



Figure 4.15(a) shows the spatial profile of the source term $S(r, \theta)$, and Fig. 4.15(b) shows the corresponding solution evolved to time $t = 0.25$ using TDVP.

In the TDVP simulation, the virtual bond dimension is kept as small as $D = 10$. The TDVP results agree with a reference solution computed using a standard fourth-order Runge–Kutta (RK4) finite-difference solver. The maximum pointwise error is approximately 10^{-8} , which is consistent with the machine-precision limitations associated with complex-valued arithmetic in the QTT implementation.

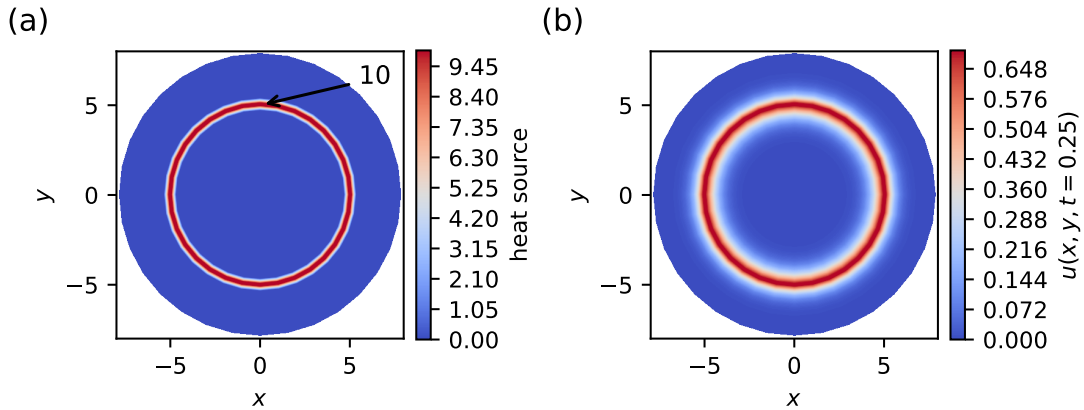


Figure 4.15: (a) Heat source distribution $S(r, \theta)$ used in the inhomogeneous heat equation (4.68). (b) TDVP simulation of the resulting heat flow at time $t = 0.25$. The maximal deviation from a reference RK4 finite-difference solver is $\sim 10^{-8}$, reflecting only machine-precision errors.

4.6.2 Schrödinger Equation in the Two-Dimensional Harmonic Oscillator



We next consider the two-dimensional quantum harmonic oscillator in polar coordinates. The Hamiltonian is

$$H = -\frac{1}{2}\nabla^2 + \frac{1}{2}r^2. \quad (4.69)$$

To benchmark the accuracy of the QTT representation and our MPO construction of the polar Laplacian, we compare the numerical results with the analytical eigenfunctions,

$$\psi_{n_r, m}(r, \phi) = \sqrt{\frac{n_r!}{\pi (n_r + |m|)!}} r^{|m|} e^{-r^2/2} L_{n_r}^{(|m|)}(r^2) e^{im\phi}, \quad (4.70)$$

where $L_{n_r}^{(|m|)}$ are the generalized Laguerre polynomials. The corresponding eigenvalues are

$$E_{n_r, m} = 1 + n_r + |m|. \quad (4.71)$$

Ground state. We first compute the ground state $\psi_{0,0}$ using the DMRG algorithm with the MPO form of the Hamiltonian (4.69).¹ Figure 4.16(a) shows the DMRG result for the probability density $|\psi_{0,0}|^2$, and Fig. 4.16(b) shows the analytical expression from Eq. (4.70).

¹In this DMRG calculation, the effective local operator is non-Hermitian. Consequently, the standard Lanczos diagonalization cannot be used, and we instead adopt the Arnoldi algorithm for performing the local updates.

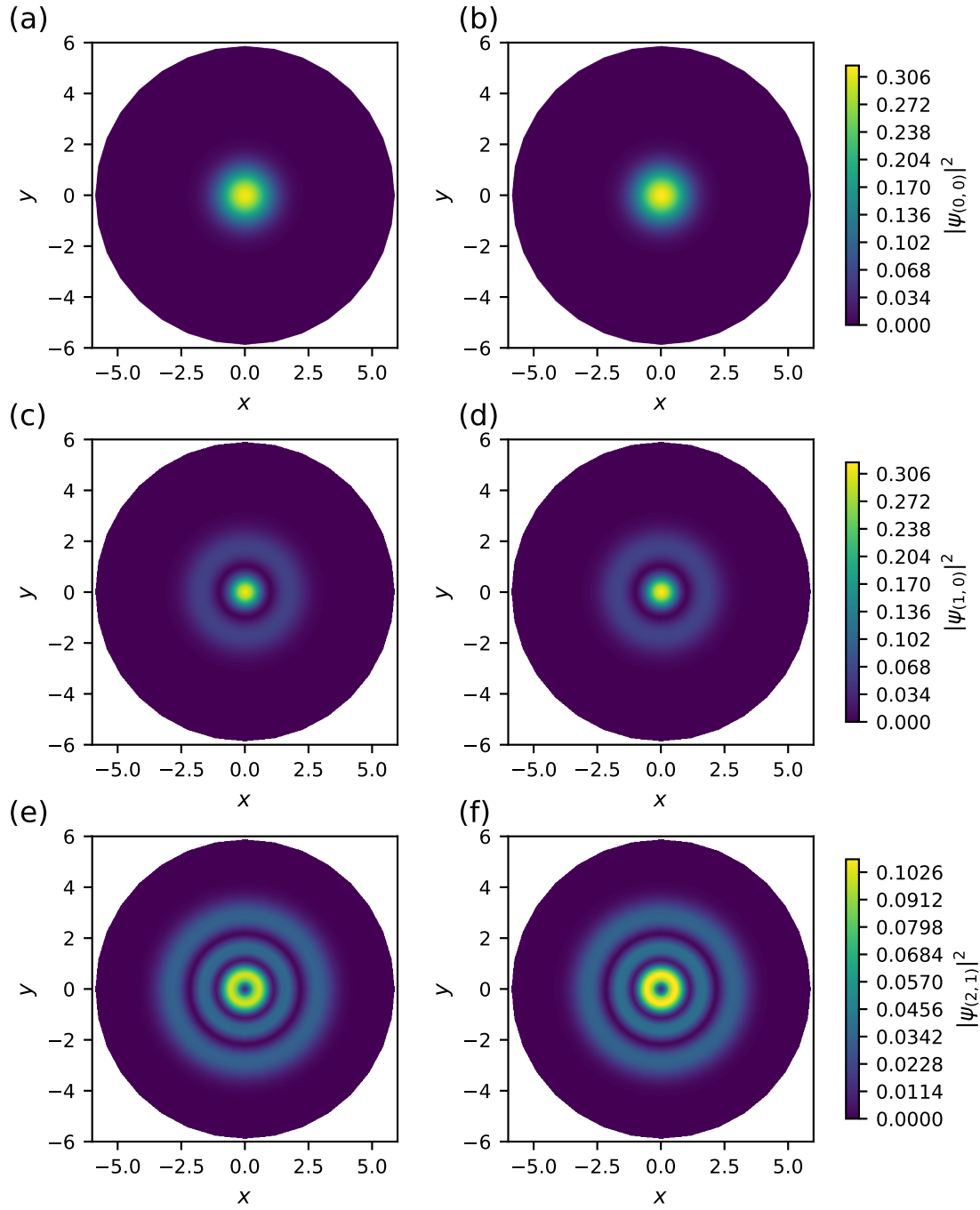
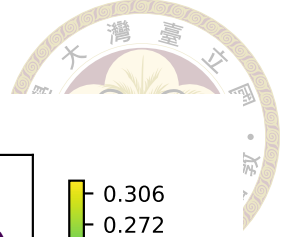


Figure 4.16: Comparison of numerical and analytical probability densities $|\psi_{(n,m)}(x, y)|^2$ for selected eigenstates of the two-dimensional Schrödinger equation in polar coordinates. Panels (a,c,e) show numerical results obtained using the QTT-based methods, while panels (b,d,f) show the corresponding analytical densities from Eq. (4.70). The first row corresponds to the ground state $\psi_{(0,0)}$ obtained via QTT-DMRG, the second row shows the excited state $\psi_{(0,1)}$ constructed using MPO ladder operators, and the third row corresponds to the state $\psi_{(2,1)}$.

Excited states via ladder operators. To obtain excited states, we use ladder operators.

In Cartesian coordinates, the standard harmonic-oscillator ladder operators are

$$a_x = \frac{1}{\sqrt{2}}(x + \partial_x), \quad a_x^\dagger = \frac{1}{\sqrt{2}}(x - \partial_x), \quad (4.72)$$

$$a_y = \frac{1}{\sqrt{2}}(y + \partial_y), \quad a_y^\dagger = \frac{1}{\sqrt{2}}(y - \partial_y). \quad (4.73)$$

Thus,

$$|\phi_{n_x, n_y}\rangle \propto (a_x^\dagger)^{n_x} (a_y^\dagger)^{n_y} |\psi_{0,0}\rangle, \quad E = 1 + n_x + n_y. \quad (4.74)$$

In polar coordinates, the corresponding ladder operators [90] are

$$a_+^\dagger = \frac{1}{2} e^{-i\phi} \left(r - \partial_r + \frac{i}{r} \partial_\phi \right), \quad (4.75)$$

which raises the angular momentum m , and the radial-raising operator

$$K_+ = \frac{1}{4} \left(\nabla^2 + r^2 - 2r\partial_r - 2 \right). \quad (4.76)$$

Both operators are linear combinations of the Laplacian, spatial derivatives, and multiplication by r , and therefore can be represented directly as MPOs using the construction described earlier. Excited states are generated according to

$$|\psi_{n_r, m}\rangle \propto (K_+)^{n_r} (a_+^\dagger)^m |\psi_{0,0}\rangle, \quad m > 0, \quad (4.77)$$

with energies $E_{n_r, m} = 1 + n_r + |m|$. Figure 4.16(c) shows the QTT-computed density $|\psi_{0,1}|^2$, and Fig. 4.16(d) shows the analytical density. Similarly, Fig. 4.16(e,f) shows the numerical and analytical results for $|\psi_{2,1}|^2$. In all cases, the polar QTT simulation reproduces the

correct nodal structure and radial oscillations.

All simulations in this section use a tensor network with $5 + 5$ sites in the radial and angular directions and a bond dimension of $D = 32$ without any truncation. A subtle but important issue arises when applying TDVP to different types of partial differential equations: the question of normalization. For equations such as the heat equation and the Maxwell equations discussed in the previous sections, the evolving quantity is not a wavefunction. Consequently, the solution is not defined up to an arbitrary multiplicative constant, and its overall scale has physical meaning. This implies that, during TDVP evolution, we cannot freely renormalize the singular values after each SVD step. In other words, the common MPS operation

$$S \leftarrow \frac{S}{\|S\|}$$

is not allowed, because rescaling the function would incorrectly modify the physical solution of the PDE. Thus, for non-quantum PDEs, the TDVP algorithm must preserve the absolute scale of the tensor train throughout the evolution.

In contrast, for the Schrödinger equation the evolved object ψ is a wavefunction, and may therefore be rescaled at will as long as the correct physical normalization is restored after each update. In Cartesian coordinates, the normalization condition

$$\int |\psi|^2 dx dy = 1 \approx \sum_{x_i, y_i} |\psi(x_i, y_i)|^2 \Delta x \Delta y = 1 \quad (4.78)$$

can be evaluated directly using the QTT inner-product operations discussed previously. In

polar coordinates, however, the normalization integral contains the Jacobian factor r :

$$\int |\psi|^2 r dr d\theta = 1 \approx \sum_{r_i, \theta_i} |\psi(r_i, \theta_i)|^2 r_i \Delta r \Delta \theta = 1, \quad (4.79)$$

which means that an additional MPO representing multiplication by r is required. Figure 4.17 illustrates the corresponding tensor-network structures in both Cartesian and polar coordinates.

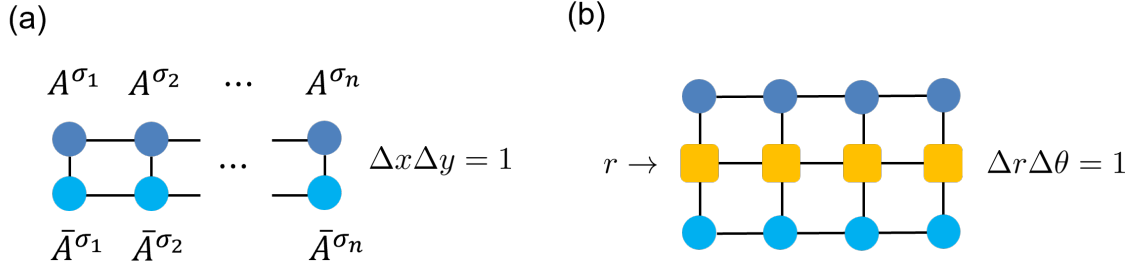


Figure 4.17: Tensor-network structure of the normalization operation. (a) In Cartesian coordinates, normalization is performed using the standard QTT inner-product contraction as in Eq. (4.78). (b) In polar coordinates, the Jacobian factor r must be included, requiring an additional MPO for the pointwise multiplication by r , as in Eq. (4.79).

This distinction becomes especially important in the next section when we consider the Gross–Pitaevskii equation. Because the GP equation contains the nonlinear interaction term $\beta|\psi|^2\psi$, any incorrect global scaling of $\psi(t)$ feeds back into the dynamics and produces an unphysical evolution. Ensuring the correct normalization condition

$$\int |\psi(t)|^2 d\mathbf{x} = 1$$

is therefore essential for obtaining accurate and stable TDVP simulations.

4.7 Rotating Gross–Pitaevskii Equation



4.7.1 Rotating Gross–Pitaevskii Equation in Polar Coordinates

We finally demonstrate that the polar-coordinate QTT–TDVP framework is capable of simulating nonlinear quantum fluids by considering the rotating Gross–Pitaevskii (GP) equation,

$$i\partial_t\psi(r, \theta, t) = \left[-\frac{1}{2}\nabla^2 + \frac{1}{2}r^2 + \beta|\psi|^2 - \Omega L_z\right]\psi, \quad (4.80)$$

where β controls the nonlinear interaction strength and Ω is the rotation frequency. Unlike the linear PDEs discussed previously, this equation is strongly nonlinear. When β becomes large, standard variational ground-state solvers such as DMRG often fail to converge to a unique stationary solution. Indeed, as shown in Fig. 4.18, the DMRG iterations oscillate between two distinct probability-density profiles: odd sweeps converge to one displaced distribution, while even sweeps converge to another located on the opposite side. This alternating pattern indicates that the nonlinear landscape yields multiple competing solutions and that DMRG cannot stabilize a single ground state under strong interaction and rotation.

Because of this convergence problem, we instead obtain the ground state using imaginary-time evolution within TDVP. During imaginary-time propagation, the wavefunction must remain normalized after each update. As discussed in the end of the previous section, normalization in polar coordinates requires the correct measure

$$\int |\psi|^2 r dr d\theta = 1,$$

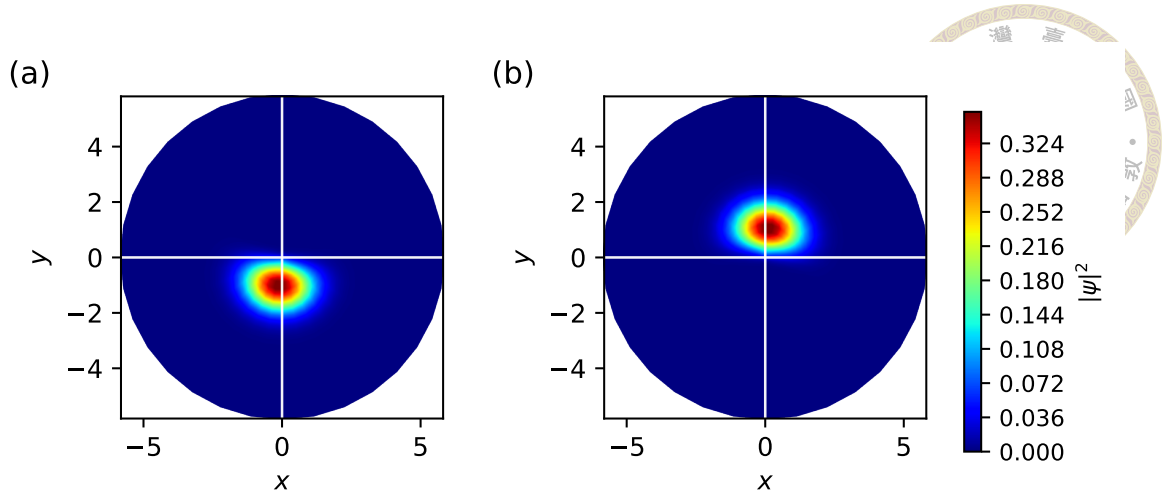


Figure 4.18: DMRG simulation of the rotating GP equation with strong nonlinear interactions ($\beta = 100$, $\Omega = 0.8$). (a) Probability density $|\psi|^2$ at the 499th sweep (representative of all odd sweeps). (b) Probability density at the 500th sweep (representative of all even sweeps). The DMRG solution swings between two displaced profiles rather than converging, showing that variational optimization becomes unstable for strongly nonlinear GP systems.

which we enforce using an MPO-based normalization operator at every TDVP step. This is essential because the nonlinear interaction term $\beta|\psi|^2$ is extremely sensitive to the magnitude of ψ ; any normalization drift produces incorrect interaction energy and prevents convergence.

We solve Eq. (4.80) using imaginary-time TDVP with time step $d\tau = 10^{-3}$, evolving to $\tau = 50$, with interaction strength $\beta = 100$ and rotation frequency $\Omega = 0.8$. The QTT grid consists of $N = 5 + 5$ tensor sites (five radial and five angular), and the bond dimension is fixed to $D = 32$ without truncation. Starting from a random QTT initial state and renormalizing after each step, the imaginary-time evolution converges to a stationary solution with chemical potential $\mu \approx 4.79$. The resulting density profile, shown in Fig. 4.19, exhibits three quantized vortices—a hallmark of rapidly rotating Bose–Einstein condensates.

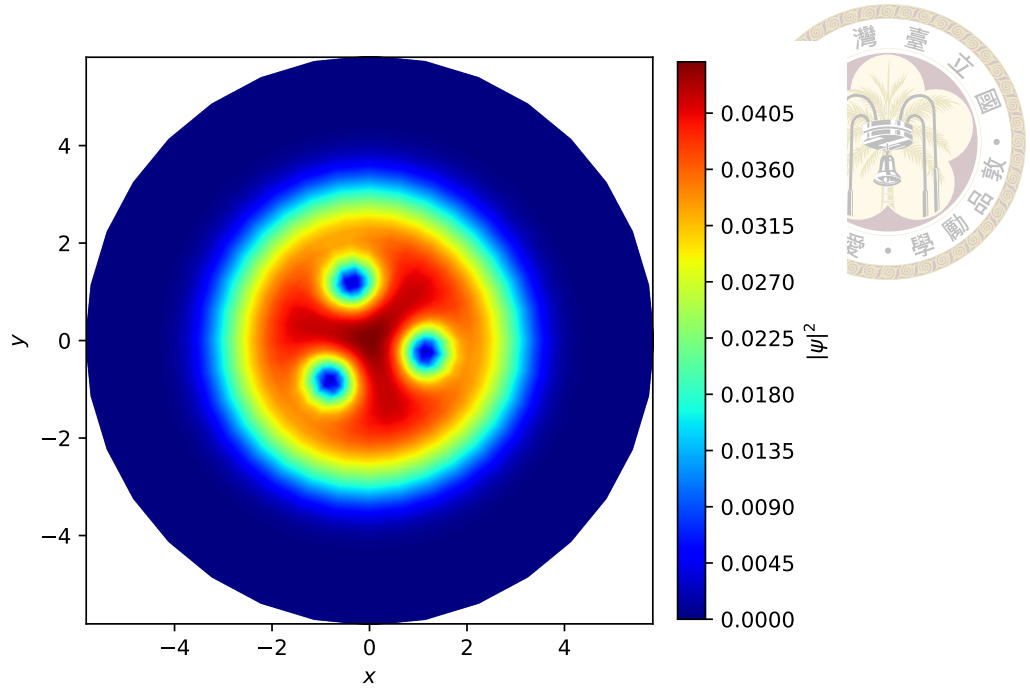


Figure 4.19: Ground-state density distribution $|\psi(r, \theta)|^2$ of the rotating GP equation, obtained using imaginary-time TDVP in polar coordinates. Parameters: $d\tau = 10^{-3}$, $\beta = 100$, $\Omega = 0.8$, QTT lattice $N = 5 + 5$, bond dimension $D = 32$, total imaginary time $\tau = 50$. The renormalization MPO is applied at every update. The converged state exhibits three quantized vortices.

4.7.2 Increasing the QTT Resolution by Adding Tensor Sites

In the QTT representation, increasing the number of tensor sites directly increases the number of sampling points used to discretize the underlying function. Adding a single site doubles the resolution; adding two sites increases the resolution by a factor of four, and so on. This property makes QTT particularly suitable for adaptive-resolution simulations, where one first obtains a coarse but accurate approximation and then refines the discretization.

In this subsection, we describe how to enlarge the QTT grid by adding tensor sites while preserving the functional structure. We consider a one-dimensional function $f(x)$ with QTT tensors $\{A_1^{\sigma_1}, A_2^{\sigma_2}, \dots, A_n^{\sigma_n}\}$, and explain how to construct a new QTT with $n+1$



sites representing an interpolated function. Two interpolation strategies are considered: nearest-neighbor interpolation and linear interpolation.

4.7.2.1 Nearest-Neighbor Interpolation

The simplest way to refine the discretization is to copy the value of the nearest existing sample point. In the QTT representation, this is achieved by appending an identity tensor to the end of the tensor train, as illustrated in Fig. 4.20(a). Because the appended tensor has bond dimension 1, this produces a new QTT state in which each value of $f(x)$ is duplicated at the finer grid.

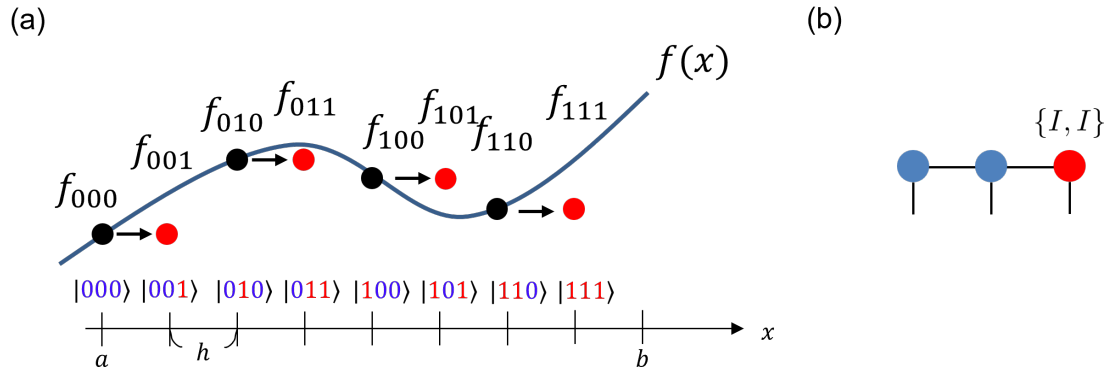


Figure 4.20: (a) QTT representation of a one-dimensional function $f(x)$ on 2^n grid points. (b) After adding one tensor site, the resolution doubles and the new grid values are obtained by nearest-neighbor interpolation. Appending an identity tensor copies each sample value to the refined grid.

To maintain numerical stability for later TDVP evolution, we must also increase the bond dimension of the new QTT. This is performed using an MPO–MPS contraction with an identity MPO,

$$\text{MPO} = \left(I, \begin{pmatrix} I & 0 \\ 0 & I \end{pmatrix}, \dots, \begin{pmatrix} I \\ I \end{pmatrix} \right),$$

followed by an SVD truncation to the desired bond dimension. Because this MPO is the

identity operator, the QTT values remain unchanged.



4.7.2.2 Linear Interpolation

A more accurate refinement method is linear interpolation. The procedure begins in the same manner as nearest-neighbor interpolation: we first append an identity tensor to introduce new sampling points. However, rather than using an identity MPO to expand the bond dimension in the second step, we instead apply an MPO corresponding to a tridiagonal matrix

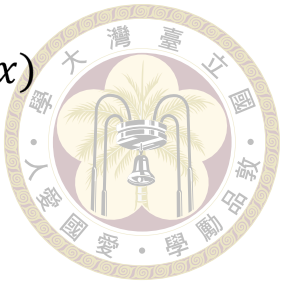
$$T = \begin{pmatrix} 0.5 & 0.5 & 0 & & \\ 0 & 0.5 & 0.5 & 0 & \\ & 0 & 0.5 & 0.5 & \\ & & \ddots & \ddots & \\ & & & 0 & 1 \end{pmatrix}, \quad (4.81)$$

which averages each sample with its nearest neighbor (the last element remains unchanged).

Matrix (4.81) is a special case of the general tridiagonal MPO construction discussed earlier in Eq. (4.23) and (4.24). After contracting this MPO with the expanded QTT and performing SVD truncation, the resulting QTT represents the linearly interpolated function, as shown in Fig. 4.21.

4.7.2.3 Extension to Polar Coordinates

In two-dimensional polar coordinates, the QTT tensor network contains two logical directions: the radial index and the angular index. To refine the grid, we must add tensor sites to both directions. Figure 4.22 illustrates nearest-neighbor interpolation: panel (a)



shows the original sampling points and their QTT structure, while panel (b) shows the enlarged QTT after appending identity tensors to both the radial and angular directions. This duplicates each function value into newly inserted sampling points.

This interpolation procedure allows us to refine the grid during the imaginary-time TDVP simulation of the rotating GP equation. Figure 4.23 summarizes the refinement steps:

2. (b) After adding one site to each direction ($N = 6 + 6$) using nearest-neighbor interpolation.

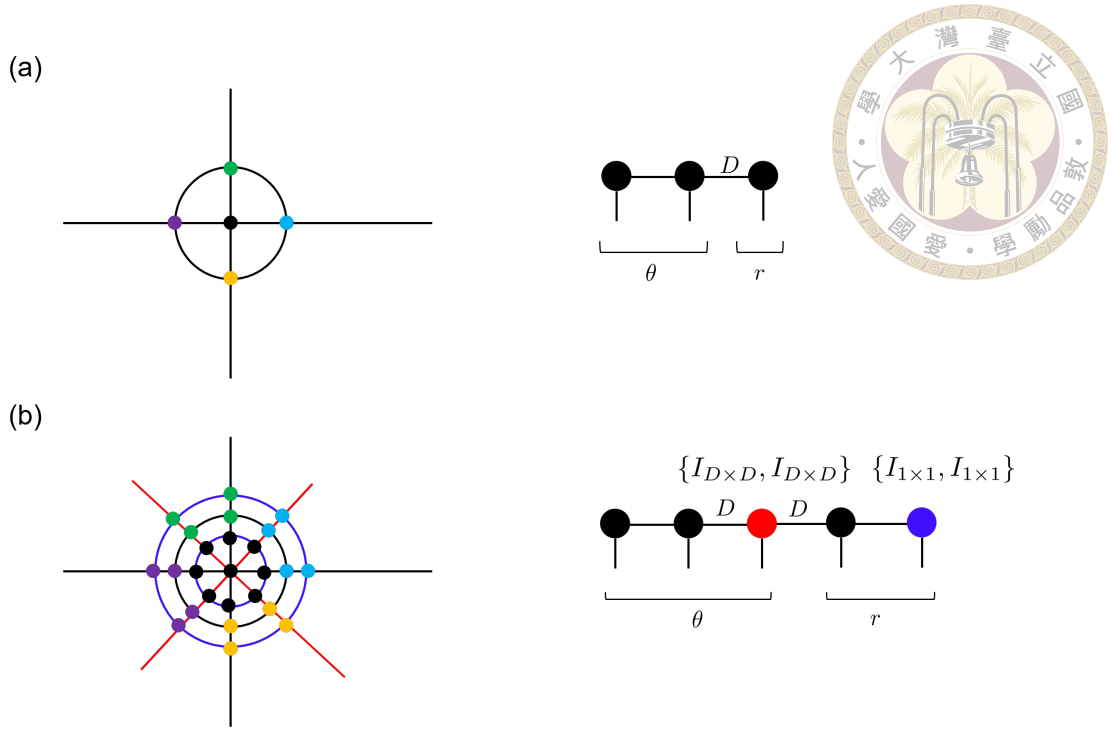


Figure 4.22: Nearest-neighbor interpolation in polar coordinates. (a) Original QTT representation of $\psi(r, \theta)$ with N_r radial sites and N_θ angular sites. (b) After adding one identity tensor to the radial direction and one to the angular direction, the resolution becomes $(2N_r) \times (2N_\theta)$, with new grid values obtained by copying their nearest neighbors.

$\tau = 3$.

4. (d) Adding another site to each direction to obtain $N = 7 + 7$.

5. (e) Final refinement using $d\tau = 10^{-6}$ for $\tau = 0.06$.

6. (f) Visualization of panel (e) in Cartesian (x, y) coordinates.

The time step is reduced at each refinement stage to satisfy the Courant–Friedrichs–Lewy condition $d\tau \sim (dx)^2$, ensuring TDVP stability on finer grids.

4.7.3 Real-Time Dynamics After a Quantum Quench

Having established that the QTT–TDVP framework can accurately obtain ground states of the rotating Gross–Pitaevskii (GP) equation, we now investigate its capability

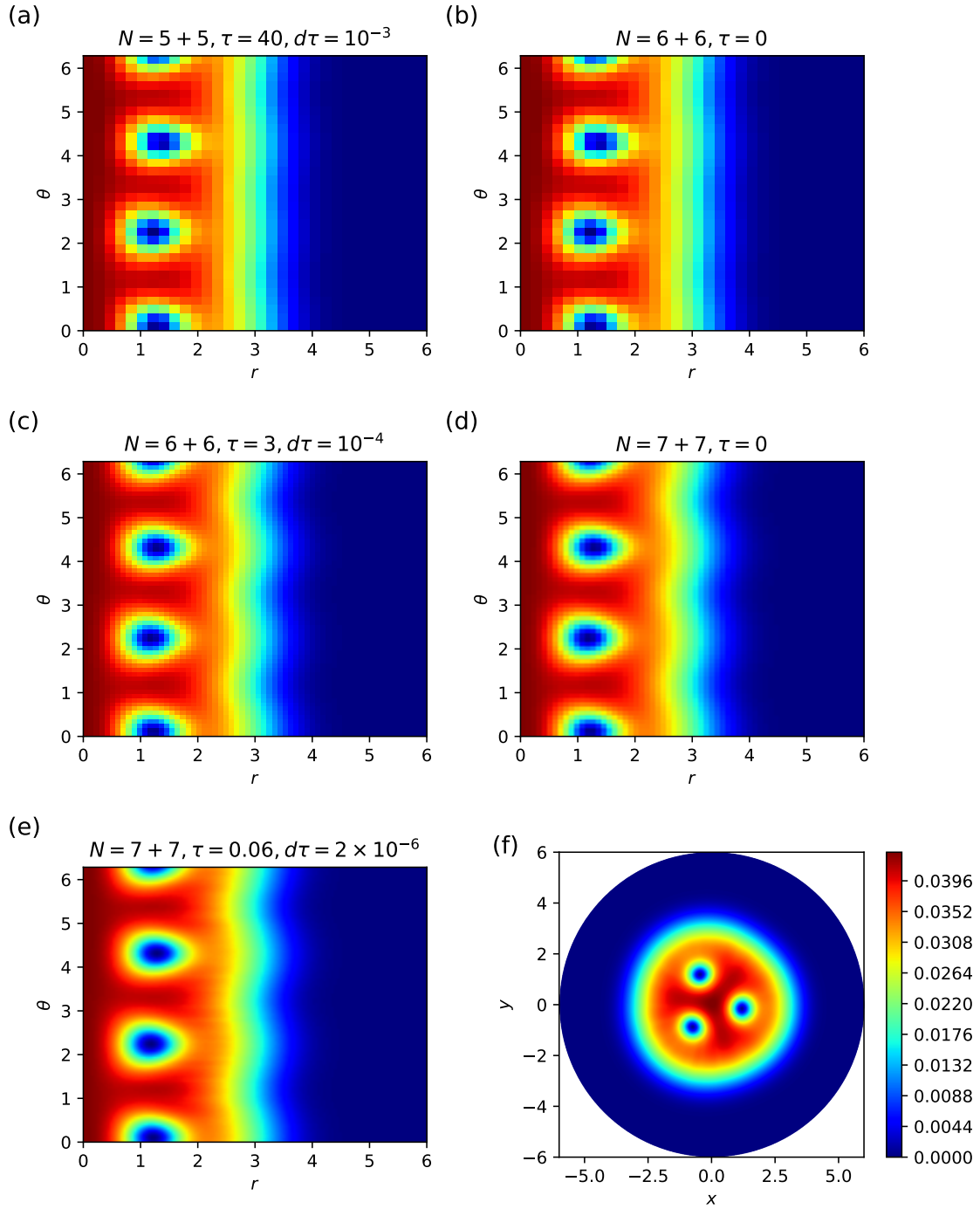


Figure 4.23: Adaptive grid refinement for the rotating GP equation using QTT interpolation. Panels (a)–(e) show the density $|\psi(r, \theta)|^2$ as the number of QTT sites is increased from $N = 5 + 5$ to $N = 7 + 7$ using nearest-neighbor interpolation, followed by short imaginary-time TDVP evolution after each refinement. Panel (f) displays the final state in Cartesian coordinates. This demonstrates that QTT grid refinement enables accurate and efficient continuation of TDVP simulation on progressively finer discretizations.

to simulate nonequilibrium real-time dynamics following a quantum quench, analogous to the studies performed in Cartesian coordinates in the previous chapter. We prepare the initial state as the ground state of the initial Hamiltonian corresponding to a strongly rotating condensate with parameters

$$\beta = 100, \quad \Omega_{\text{initial}} = 0.8,$$

which is obtained via imaginary-time TDVP evolution with step size $d\tau = 10^{-3}$ up to a total imaginary time $\tau = 50$. At real time $t = 0$, we abruptly quench the rotation frequency to

$$\Omega_{\text{final}} = 0.1,$$

while keeping the nonlinear interaction strength β fixed. The subsequent real-time evolution is simulated using the TDVP algorithm with time step $dt = 10^{-3}$.

The simulation results on a grid of $N = 5 + 5$ tensor sites (radial + angular) are shown in Fig. 4.24. Panel (a) displays the ground state of the initial Hamiltonian with $(\beta = 100, \Omega_{\text{initial}} = 0.8)$. Panels (b)–(d) show the real-time evolution of this initial state under the quench Hamiltonian $(\beta = 100, \Omega_{\text{final}} = 0.1)$ at times $t = 2$, $t = 4$, and $t = 6$, respectively.

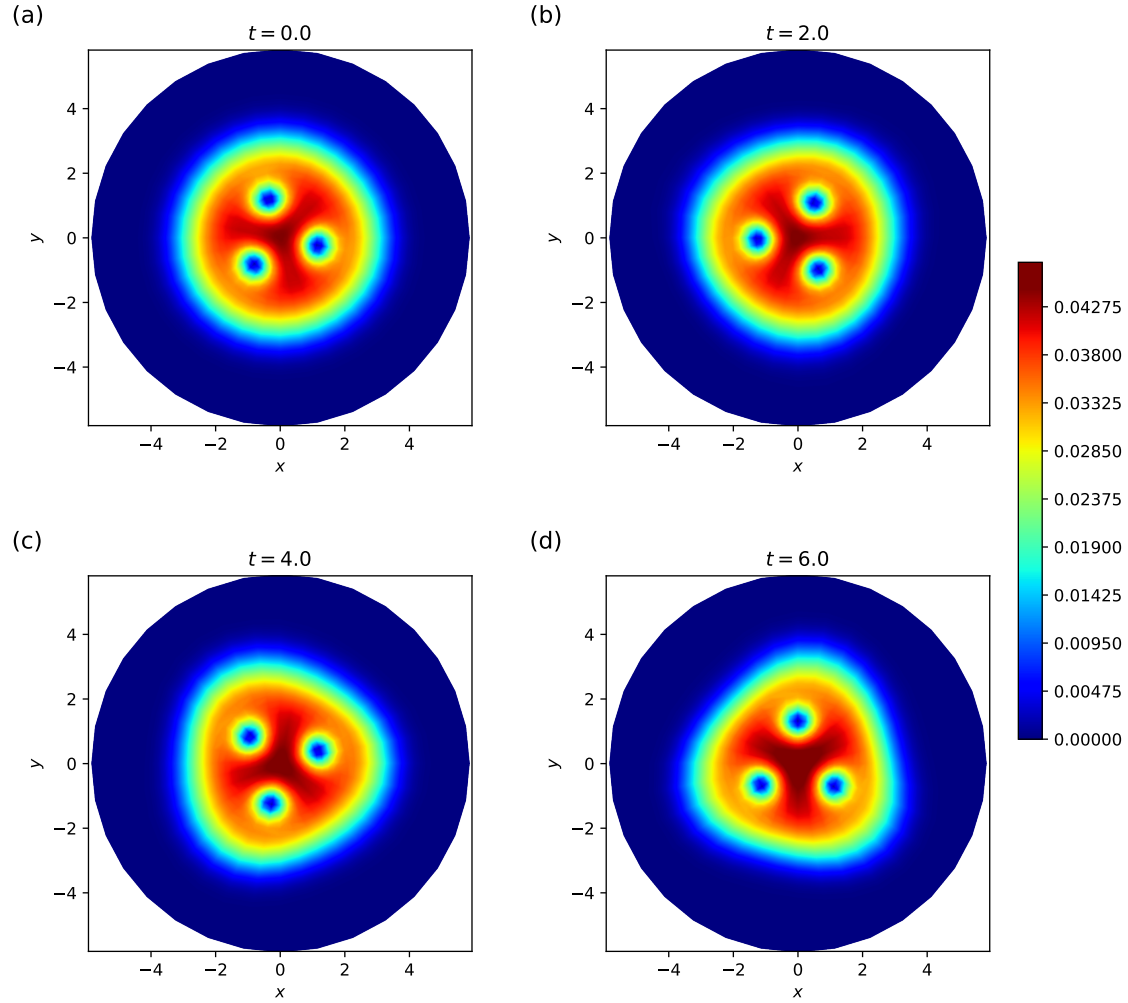


Figure 4.24: Real-time dynamics of the rotating GP equation after a quantum quench, simulated using QTT–TDVP in polar coordinates. (a) Ground state of the initial Hamiltonian with $(\beta = 100, \Omega_{\text{initial}} = 0.8)$. (b)–(d) Real-time evolution under the quench Hamiltonian $(\beta = 100, \Omega_{\text{final}} = 0.1)$ at times $t = 2, t = 4$, and $t = 6$. The vortex configuration rotates with angular velocity approximately $|\Delta\Omega| = 0.7$. Simulation parameters: imaginary-time preparation with $d\tau = 10^{-3}$ up to $\tau = 50$, followed by real-time evolution with $dt = 10^{-3}$.

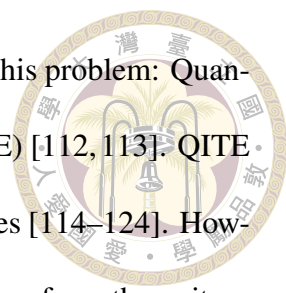


Chapter 5 Quantum Imaginary Time Evolution

The recent advent of noisy intermediate-scale quantum (NISQ) computers [91] introduces new opportunities to study quantum many-body systems. These devices have spurred the development of various quantum algorithms, with the quantum-inspired tensor network algorithms playing a crucial role.

As matrix product states (MPSs) [92, 93], which have proven highly successful in simulation quantum systems in one dimension [94–97], can be directly encoded as quantum circuits, several classical time evolution algorithms of the MPS such as time-evolving block decimation (TEBD) [98–100] and the time-dependent variational principle (TDVP) [1, 101] have been translated into quantum circuits to simulate quantum dynamics on NISQ devices [102–108].

Algorithms for real-time evolution can utilize the unitary nature of quantum gates to simulate the dynamics of quantum systems [104, 109–111]. However, extending these methods to the imaginary-time evolution (ITE), a common approach for finding the ground state of a quantum system, is more challenging. The difficulty lies in the fact that the ITE operator is non-unitary, making it incompatible with the inherently unitary nature of quan-



tum gates. Three primary approaches have been proposed to address this problem: Quantum ITE (QITE), Variational ITE (VITE), and Probabilistic ITE (PITE) [112, 113]. QITE approximates the non-unitary evolution through the use of unitary gates [114–124]. However, the ground-state energy obtained inevitably contains errors arising from the unitary approximation of the non-unitary operator. PITE resolves this issue by expanding the ITE operator into a unitary form, at the expense of introducing an additional ancilla qubit for each unitary gate used to represent the non-unitary operator [110, 112, 113, 125–134], and thereby eliminates the approximation error present in QITE. Although the ancilla qubits can be reused by measuring and resetting them, so that the total number of qubits does not grow linearly with the evolution time but only depends on the system size, the number of qubits that must be measured does increase linearly with the number of time steps. Consequently, the total number of shots required to achieve a given accuracy also grow. VITE, on the other hand, employs a variational approach in which the quantum state is represented by a parameterized ansatz, and the parameters are iteratively optimized to simulate ITE [135–140]. Here, the optimization depends on the definition of the cost function. If the cost value is obtained directly from the statevector in simulation, it can be computed exactly. However, on real quantum devices, the cost must be estimated through repeated measurements, where expectation values are inferred from outcome probabilities. This measurement process introduces statistical fluctuations, which significantly influence the convergence of the variational optimization.

Each of these methods provides a unique solution to the implementation of ITE on quantum circuits. While they offer practical ways to realize imaginary-time evolution, they have so far been primarily restricted to finite-size systems. In this work, we demonstrate

how to perform ITE on quantum circuits for an infinite-size 1D quantum system. Our approach is based on the concept of the TDVP, where the variational parameters of the quantum circuit are updated at each time step. In this way, the number of quantum gates does not grow with the evolution time, in the same spirit as VITE. For implementing the non-unitary ITE operator, we explore two strategies: (i) approximating the operator using the unitary gate, as in QITE, and (ii) introducing an additional ancilla qubit to construct the operator exactly, as in PITE.

In this chapter, the method we develop is also a quantum imaginary-time evolution on quantum circuits (applied to infinite 1D systems), and for convenience we will simply refer to it as QITE, although it should be distinguished from the QITE scheme discussed in the previous paragraph. We present results obtained both from statevector simulations and from measurement-based simulations. In the latter case, since the cost value is estimated via repeated measurements, it represents a statistical quantity that follows a distribution rather than a fixed value. We therefore analyze how the distribution of cost values impacts the distribution of the energy density during the evolution. Finally, we implement our algorithm on IBM Quantum devices and employ the Quantum Lanczos (QLanczos) Algorithm [114, 115, 119, 120, 122] to optimize the ground-state energy, showing that the energy converges toward the ground-state value during the ITE.

The remaining sections are organized as follows: In Sec. 5.1, we review the key concepts from Ref. [104], which introduces the TDVP framework based on quantum circuits and demonstrates real-time evolution in infinite 1D quantum systems. In Sec. 5.2, we discuss how a non-unitary gate can be embedded into a unitary operation using a single ancilla qubit. In Sec. 5.3 and Sec. 5.4, we explore the implementation of ITE on quantum circuits

for infinite- and finite-size systems, respectively, addressing the challenges posed by non-unitary operators by presenting two distinct approaches (similar to QITE and PITE) and refining the simulation results with the QLanczos algorithm. We then present simulation results obtained from both statevector and measurement-based approaches, including implementations on IBM Quantum real devices, in Sec. 5.5. In Sec. 5.6, we provide a detailed discussion of the statistics of the cost values and energy density arising from measurement-based estimations. Finally, we summarize our findings and offer concluding remarks in Sec. 5.7.

5.1 The Concept of the TDVP Based on Quantum Circuits

In this section, we will review the main ideas presented in the Ref. [104], focusing on how to perform the real-time evolution on quantum a circuit based on TDVP. We begin by introducing the concept of the TDVP as applied to uniform Matrix Product States (uMPS) [1, 141]. Following this, the TDVP framework is adapted from its original uMPS-based formulation to one based on quantum circuits, enabling implementation on quantum devices.

5.1.1 The TDVP based on uMPS

The uMPS is a powerful ansatz for representing translationally invariant states in an infinite 1D system. It is based on the idea that the quantum state can be described by repeatedly applying the same tensor across all lattice sites. In general, the uMPS can be

expressed as:

$$\sum_{\{i\},\{s\}} \prod_n \cdots A_{i_{n-1},i_n}^{s_{n-1}} A_{i_n,i_{n+1}}^{s_n} \cdots |\cdots s_{n-1} s_n \cdots\rangle, \quad (5.1)$$

where all rank-3 tensors A are identical. The corresponding tensor diagram of the uMPS is shown in Fig.(5.1). The uMPS span a subspace or manifold within the entire Hilbert

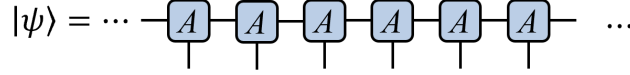


Figure 5.1: The tensor diagram of the uMPS.

space. We denote the uMPS as $|\psi(X(t))\rangle$, where $X(t)$ is a set of parameters that define the specific form of the state at time t . Here, X specifically refers to the parameters that define the state in terms of the uMPS. The state after a small real-time evolution dt with the Hamiltonian H is given by $e^{-iHdt}|\psi(X(t))\rangle$. The goal is to identify a uMPS $|\psi(X(t+dt))\rangle$ that closely approximates the evolved state $e^{-iHdt}|\psi(X(t))\rangle$, which is achieved by solving the equation:

$$X(t+dt) = \arg \max_{X(t+dt)} \langle \psi(X(t+dt)) | e^{-iHdt} | \psi(X(t)) \rangle. \quad (5.2)$$

It can be regarded as a projection onto the manifold of the uMPS (refer to the Fig.(5.2)).

This projection embodies the core idea of the TDVP ¹.

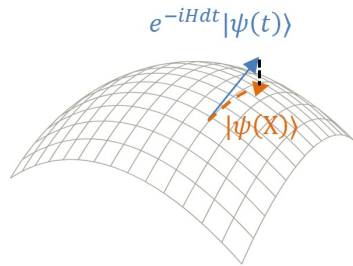


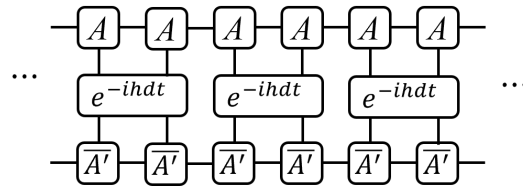
Figure 5.2: The concept of the TDVP. See Eq.(5.2)

¹Unlike the projection onto the tangent space of the uMPS, as described in [1], this approach involves a direct projection onto the manifold.

If the Hamiltonian consists only of nearest-neighbor terms, $H = \sum_n h^{[n,n+1]}$, the time-evolution operator e^{-iHt} can be decomposed through the Suzuki–Trotter decomposition:

$$\left(e^{-i \sum_{n \in \text{odd}} h^{[n,n+1]} \Delta t} \cdot e^{-i \sum_{n \in \text{even}} h^{[n,n+1]} \Delta t} \right)^m, \quad (5.3)$$

where $\Delta t = t/m$. In TN language, it can be viewed as the application of odd and even layer operators. By alternating between these odd and even layers, the decomposition allows for an approximation of the full time evolution. Due to translational invariance, we can limit our consideration to a single layer. Consequently, Eq. (5.2) can be represented as an infinite-chain in the tensor diagram:



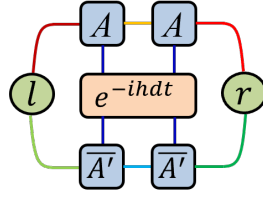
The diagram shows an infinite chain of tensors. The top row consists of a sequence of A tensors. The bottom row consists of a sequence of A' tensors. Between the A and A' tensors, there is a vertical line representing a contraction. In the middle, there is a box labeled $e^{-i h dt}$ that is connected to the vertical line. The entire chain is enclosed in large curly braces on the right side. The top brace is labeled $e^{-i H_{\text{odd}} dt} |\psi(X(t))\rangle$ and the bottom brace is labeled $\langle \psi(X(t + dt/2)|$. Ellipses (...) are placed at the beginning and end of the chain to indicate it is infinite.

$$\dots \left\{ \begin{array}{l} e^{-i H_{\text{odd}} dt} |\psi(X(t))\rangle \\ \langle \psi(X(t + dt/2)| \end{array} \right. \quad (5.4)$$

Please note that a single layer evolves the system by only $dt/2$, as the full time step dt requires applying the same layer twice. The infinite-chain in Eq. (5.4) can be simplified by introducing environment tensors. In Fig. 5.3(a), the tensors l and r represent the environment, which can be determined by solving the fixed-point equations. Once these tensors are obtained, the tensor diagram in Eq. (5.4) can be further simplified, which only contains seven tensors rather than an infinite-chain, as illustrated in Fig. 5.3(a).

To construct the environment gates L and R , as shown in Fig. 5.3(b), we first convert the unitary gate U into a unit cell of the uMPS representation. We then solve the fixed-point equations, as illustrated in Fig. 5.4(a), to obtain the environment tensors l and r . In practice, we solve the fixed-point equation by simulating the leading eigenvector. Due to

(a)



(b)

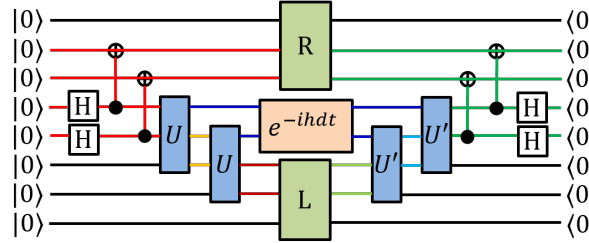


Figure 5.3: The transformation of the cost function (Eq. (5.2)) from a TN diagram to a quantum circuit representation. On a quantum computer, the bra state $\langle 000 \dots |$ cannot be directly used. Instead, we estimate the overlap by doing the measurement of the quantum state.

the finite time step used in the evolution, the leading eigenvalue will be slightly smaller than 1, rather than exactly 1. Once the leading left and right eigenvectors, l and r , are determined, we embed them into the unitary gates L and R , following the process outlined in Sec. 5.2.2. Here L and R are also satisfied the fixed-point equations as shown in Fig. 5.4(b) and (c).

We have demonstrated how to convert the environment tensors l and r into the environment gates L and R by extending the non-unitary operator to a unitary one with the addition of an ancilla qubit. This method ensures that the non-unitary tensors are embedded into a unitary gate for quantum circuit implementation (Please refer to Sec. 5.2). However, since we only consider a bond dimension of $D = 2$, there is another way to do

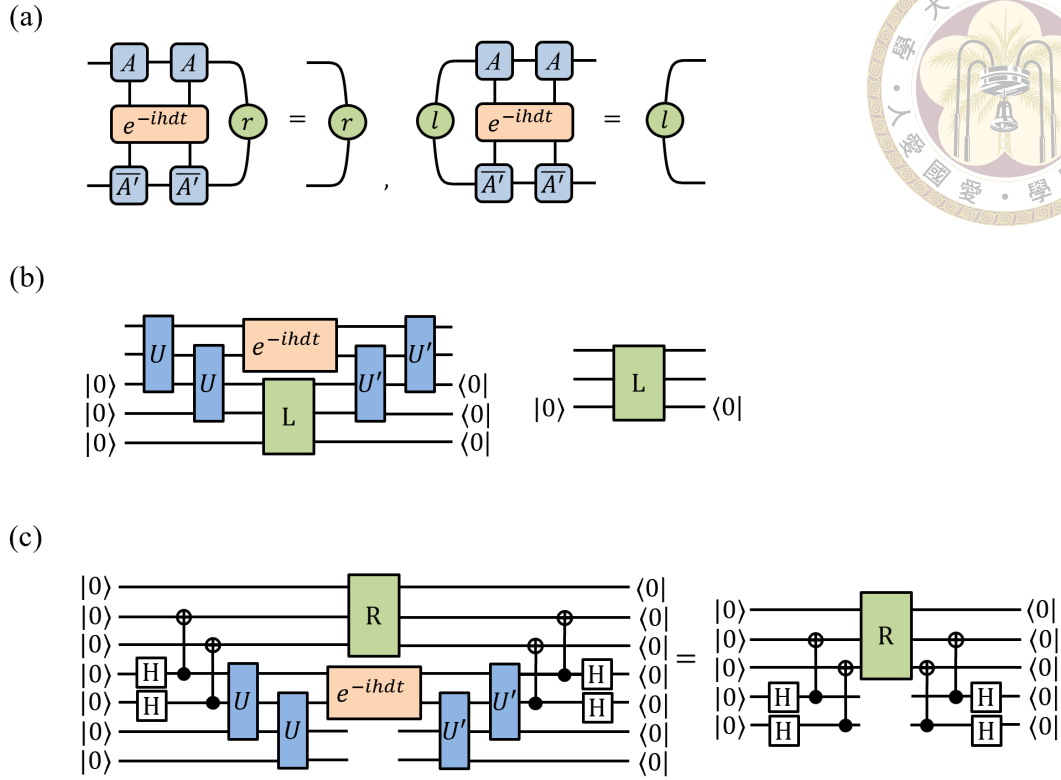


Figure 5.4: (a) The fixed-point equations based on the uMPS with evolution operator. (b) and (c) are the fixed-point equations based on quantum circuits.

this (see the github repository of Ref. [104]). We can choose L so that

$$\text{if } l = \begin{pmatrix} (\beta) & 0 \\ (\alpha) & 1 \end{pmatrix} \begin{pmatrix} a & b \\ c & d \end{pmatrix}, \text{ set } L = \begin{pmatrix} (\delta\beta) & 00 & 01 & 10 & 11 \\ & a & b & d^* & -c^* \\ & c & d & -b^* & a^* \\ & & \mathbf{n}_0^\dagger & & \\ & & \mathbf{n}_1^\dagger & & \end{pmatrix} \quad (5.5)$$

where $\{\mathbf{n}_0, \mathbf{n}_1\}$ is a orthonormal basis of the null space of $\begin{pmatrix} a & b & d^* & -c^* \\ c & d & -b^* & a^* \end{pmatrix}$. Here, l has

been normalize so $|a|^2 + |b|^2 + |c|^2 + |d|^2 = 1$, and it is easy to check that L is satisfied

the condition in Eq.(5.11). We have confirmed that both methods yield the same results

for ITE.

5.1.2 Mapping between uMPS and Quantum Circuits



To perform simulations on quantum devices, the cost function in Eq. (5.2) must be represented in the form of a quantum circuit rather than a tensor diagram (see Fig.5.3(a)).

This requires a mapping from the uMPS tensors to unitary gates in a quantum circuit.

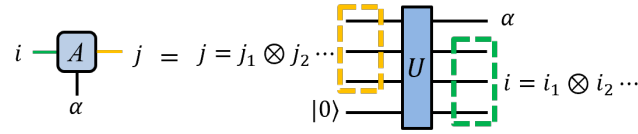


Figure 5.5: The mapping between a one-site unit cell of the uMPS and a unitary gate in the quantum circuit.

For a uMPS with virtual bond dimension $D = 2^N$ and physical bond dimension $d = 2$, as illustrated in Fig. 5.5, a mapping can be established to a unitary gate in a quantum circuit with $N + 1$ qubits. This mapping between uMPS and quantum circuits automatically enforces the canonical form of the tensors [104]. The environment tensors l and r are generally not unitary matrices and therefore cannot be directly represented as quantum gates. However, by expanding the dimension space, we can construct unitary gates that satisfy the necessary properties of the environment. Continuing with the example of the left environment tensor l , let's consider $l_{\alpha,\beta}$ as an $D \times D$ matrix. To convert this non-unitary tensor into a unitary gate, we expand $l_{\alpha,\beta}$ to a new tensor $L_{(\alpha,\gamma),(\beta,\delta)}$ with dimensions $2D \times 2D$, where $\gamma, \delta \in \{0, 1\}$ (see Fig. 5.6). The goal is to construct a unitary gate $L_{(\alpha,\gamma),(\beta,\delta)}$ such that it satisfies two conditions:

$$\begin{cases} L \text{ is unitary, namely } LL^\dagger = I \\ L_{\alpha,\gamma=0,\beta,\delta=0} = l_{\alpha,\beta} \end{cases} \quad (5.6)$$

The first condition ensures that the constructed gate L is unitary, and the second condition makes sure that the expanded tensor represents the original environment tensor l , embedding it in a higher-dimensional space. The right environment R also needs to satisfy this condition when embedding tensor r . The detail technique will be discussed in Sec. 5.2.

After the environment quantum gate has been constructed, the cost function in Eq. (5.2) can be fully represented as a quantum circuit. This circuit, which includes both the unitary gates derived from the uMPS and the environment tensors as quantum gates, is shown in Fig. 5.3(b). This representation will be used in the simulation of ITE on quantum circuits.

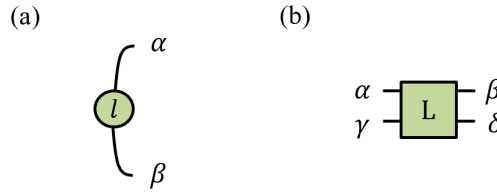


Figure 5.6: (a) The left environment of the uMPS. (b) The left environment of the quantum circuit.

5.1.3 Real-Time Evolution on 1D Infinite-size

The cost function has been successfully constructed using a quantum circuit. In general, we can consider any bond dimension $D = 2^N$. In our case, we consider $D = 2$, which means the total number of qubits required is only 6. The cost function represented in Fig. 5.3 corresponds to the overlap between the quantum circuit's output and the state $\langle 000 \cdots |$. This overlap can be estimated by measuring the probability of obtaining all zeros in the output state, as illustrated in Fig. 5.7.

To perform the real-time evolution, we parametrize the unitary gate U in Fig. 5.3(b)

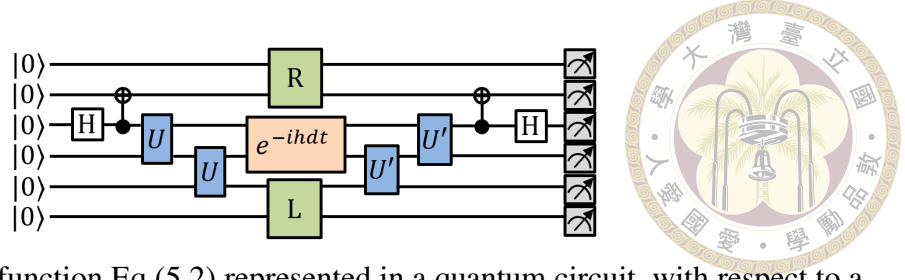


Figure 5.7: The cost function Eq (5.2) represented in a quantum circuit, with respect to a bond dimension of $D = 2$.

as a variational ansatz, shown in Fig. 5.8. It is also known as a parametrized quantum circuit (PQC). The unitary gate U represents the current state $|\psi(X(t))\rangle$, and the goal is to find the optimized unitary U' corresponding to the state $|\psi(X(t + dt/2))\rangle$, such that the overlap represented by the circuit in Fig.5.3(b) is maximized. The cost function is computed on a quantum simulator or quantum real devices, while the optimization process to determine U' is performed on a classical computer. Once the optimal U' is found for the current time step, then replace the current U with U' . Repeated this process for the next time step and iteratively update U to track the quantum dynamics over time. As mentioned in Ref. [104], this approach has been successful in performing real-time evolution on simulator via statevector and accurately simulating the correct dynamical quantum phase transition (DQPT) [78, 142] for the transverse field Ising model.

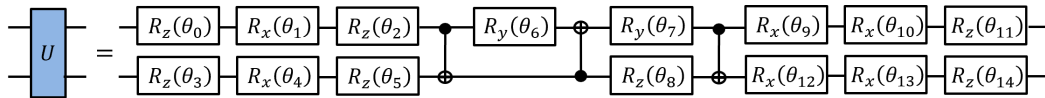


Figure 5.8: An example of a PQC with virtual bond dimension $D = 2$. In this work, we use this ansatz, which is from the GitHub repository associated with the paper in Ref. [104], in our simulation. R_i is the rotation operator along the i -axis.

5.2 Expand Non-unitary to Unitary Gate



5.2.1 Expand Non-unitary to Unitary Gate Using SVD

Any non-unitary $n \times n$ operator M can be converted into a unitary operator by embedding it into a larger $2n \times 2n$ matrix. The resulting unitary matrix U will have the form:

$$U = \begin{pmatrix} M & A \\ B & C \end{pmatrix}. \quad (5.7)$$

Perform the singular value decomposition (SVD) on M :

$$M = WSV^\dagger. \quad (5.8)$$

Subsequently, construct the unitary gate U :

$$U = \begin{pmatrix} M & W\sqrt{I - S^2}V^\dagger \\ W\sqrt{I - S^2}V^\dagger & -M \end{pmatrix}. \quad (5.9)$$

We can directly verify that U is unitary:

$$\begin{aligned} & UU^\dagger \\ &= \begin{pmatrix} WDV^\dagger & W\sqrt{I - D^2}V^\dagger \\ W\sqrt{I - D^2}V^\dagger & -WDV^\dagger \end{pmatrix} \begin{pmatrix} VDW^\dagger & V\sqrt{I - D^2}W^\dagger \\ V\sqrt{I - D^2}W^\dagger & -VDW^\dagger \end{pmatrix} \\ &= \begin{pmatrix} WD^2W^\dagger + W(I - D^2)W^\dagger & 0 \\ 0 & WD^2W^\dagger + W(I - D^2)W^\dagger \end{pmatrix} \\ &= I \end{aligned} \quad (5.10)$$

This implies that for the non-unitary gate M with N qubits (see Fig. 5.9), it can be unitarized as $U(M)$ with $N + 1$ qubits. This property is useful when dealing with a non-unitary gate. We use this property in two respects. The first aspect is that we can use it to construct the environment gates L and R . The other aspect is that we can use it to deal with the imaginary time evolution operator.

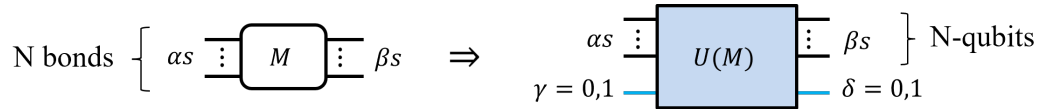


Figure 5.9: A nonunitary operator can be extended to a unitary gate by introducing an additional qubit, commonly referred to as the ‘ancilla qubit.’

5.2.2 The Construction of the Environment Gates

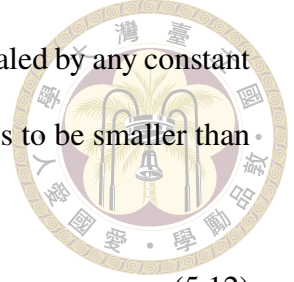
Consider the left environment gate L as an example. If the tensor $l_{\alpha,\beta}$ (See Fig.5.6(a)) has been solved from the fixed-point equation in the Fig. 5.4(a), the environment gate $L_{\alpha,\gamma,\beta,\delta}$ (See Fig. 5.6(b)) needs to satisfied:

$$\begin{cases} L \text{ is unitary, namely } LL^\dagger = I \\ L_{\alpha,\gamma=0,\beta,\delta=0} = l_{\alpha,\beta} \end{cases} \quad (5.11)$$

Applying the approach mentioned in Sec. 5.2, we can obtain the environment unitary gate L . The right environment gate R can be obtained in the similar way.

5.2.3 Expansion of the Imaginary-time Evolution Operator

We can use this way to expand the imaginary-time evolution into a unitary gate as well. However, the largest singular value may not be smaller than 1, which is the only



condition for this approach. Fortunately, the imaginary-time can be scaled by any constant so that the largest singular value of the imaginary-time operator scales to be smaller than 1.

$$e^{-d\tau H} = RDV \rightarrow ae^{-d\tau H} = R(aD)V \quad (5.12)$$

This is possible because the evolution state can be normalized later. After expanding the imaginary-time evolution operator into a unitary gate, the original quantum circuit (see Fig. 5.10(a)) can be transformed into the new one (see Fig. 5.10(b)).

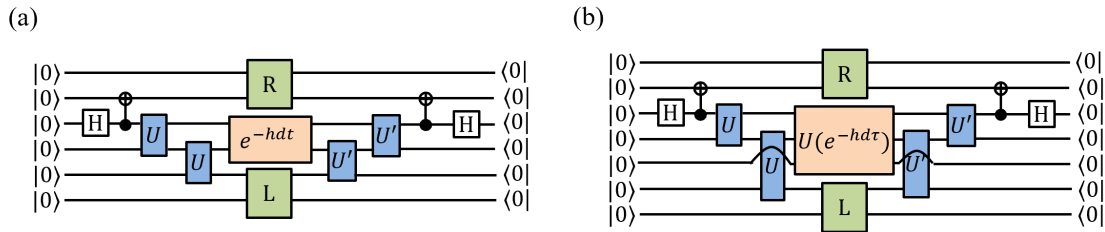


Figure 5.10: (a) The quantum circuit for the cost function, with the ITE operator requiring replacement by $e^{-iAd\tau}$. (b) The quantum circuit corresponding to Sec. 5.2

5.3 Quantum Imaginary-time Evolution for Infinite Systems

Although real-time evolution can be successfully implemented using the statevector representation in quantum circuits, applying ITE presents a more challenging problem since the non-unitary of the ITE operator. Furthermore, applying these methods on quantum real devices is even more difficult. In this section, we will address the challenge of dealing with the nonunitary nature of the ITE operator and show how to optimize the simulation results using the QLanczos algorithm.

5.3.1 Two Approaches to Deal with Non-unitary Operator



To address the challenge of non-unitary operators in ITE, we have developed two primary approaches. The first method involves approximating the non-unitary gate with a unitary one, as in the QITE algorithm [114]. This approach works by identifying a time-dependent Hermitian operator $A(\tau)$, which is then used to construct a unitary operator $e^{-id\tau A(\tau)}$ that closely approximates the ITE operator. Specifically, the unitary operator is designed to satisfy the relation

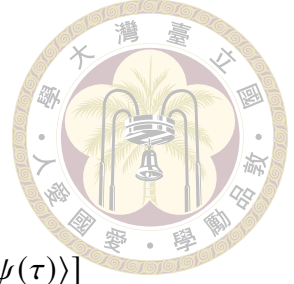
$$e^{-iA(\tau)d\tau}|\psi(\tau)\rangle \approx e^{-Hd\tau}|\psi(\tau)\rangle. \quad (5.13)$$

Since non-unitary operators do not preserve the norm of the state, and the operator $A(\tau)$ is determined by minimizing the difference between the true ITE evolution and the unitary approximation, $A(\tau)$ can be solved by the following equation:

$$A(\tau) = \arg \min_{A(\tau)} \left\| \frac{e^{-Hd\tau}|\psi(\tau)\rangle}{\sqrt{\langle\psi(\tau)|e^{-2Hd\tau}|\psi(\tau)\rangle}} - e^{-iA(\tau)d\tau}|\psi(\tau)\rangle \right\|. \quad (5.14)$$

Once this operator $A(\tau)$ is found, the non-unitary ITE operator $e^{-Hd\tau}$ can be replaced by the unitary operator $e^{-iA(\tau)d\tau}$ in the quantum circuit, ensuring that all gates within the circuit, as shown in Fig. 5.3(b), remain unitary. The steps for implementing this algorithm are outlined in the Algorithm 1. The optimization problem to find $A(\tau)$ is solved using tensor network techniques.

To approximate the non-unitary gate with a unitary in Eq. (5.14), We minimize the following expression:



$$\begin{aligned}
A(\tau) &= \arg \min_{A(\tau)} \left\| \frac{e^{-Hd\tau} |\psi(\tau)\rangle}{\sqrt{\langle \psi(\tau) | e^{-2Hd\tau} | \psi(\tau) \rangle}} - e^{-iA(\tau)d\tau} |\psi(\tau)\rangle \right\| \\
&= \arg \min_{A(\tau)} \left[2 - \frac{1}{N} \langle \psi(\tau) | e^{-Hd\tau} e^{-iA(\tau)d\tau} + e^{iAd\tau} e^{-H(\tau)d\tau} | \psi(\tau) \rangle \right] \\
&= \arg \min_{A(\tau)} \left[\langle \psi(\tau) | e^{-Hd\tau} e^{-iA(\tau)d\tau} + e^{iAd\tau} e^{-H(\tau)d\tau} | \psi(\tau) \rangle \right] \\
&= \arg \min_{A(\tau)} \left[\langle \psi(\tau) | O | \psi(\tau) \rangle \right],
\end{aligned} \tag{5.15}$$

where $N = \langle \psi(\tau) | e^{-2Hd\tau} | \psi(\tau) \rangle$ and $O = e^{-Hd\tau} e^{-iA(\tau)d\tau} + e^{iAd\tau} e^{-H(\tau)d\tau}$. Since the operator $e^{-Hd\tau}$ is non-unitary and does not preserve the norm of the state, which is why normalization is required in the first step of Eq. (5.15). Since O is also non-unitary, we cannot directly use quantum circuits to construct the cost function in Eq. (5.15). Instead, we use TN to construct the cost function, as illustrated in Fig. 5.11.

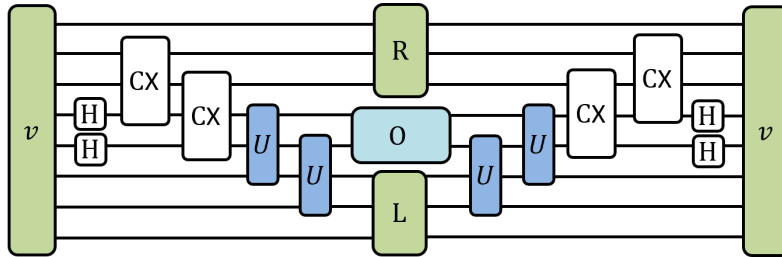


Figure 5.11: The cost function represented in TN diagram used to optimize $A(\tau)$. CX is the CNOT gate, depicted as a tensor. The input and output states v correspond to the all-zero state $|000 \cdots\rangle$, where all elements are zero, except for the element with all-zero indices, which has a value of one.

Here we parametrize the operator A as a $d^2 \times d^2$ Hermitian matrix, where d is the

physical bond dimension matrix. For example, for $d = 2$, the matrix is constructed as:

$$\begin{pmatrix} p_0 & p_1 + ip_2 & p_3 + ip_4 & p_5 + ip_6 \\ p_1 - ip_2 & p_7 & p_8 + ip_9 & p_{10} + ip_{11} \\ p_3 - ip_4 & p_8 - ip_9 & p_{12} & p_{13} + ip_{14} \\ p_5 - ip_6 & p_{10} - ip_{11} & p_{13} - ip_{14} & p_{15} \end{pmatrix}. \quad (5.16)$$



We then use the minimization method from the Python package `scipy.optimize.minimize` to find the parameters $[p_0, \dots, p_{15}]$ that minimize the cost function in Fig. 5.11. After obtaining the optimized A in Eq. (5.15), we can construct the quantum circuit, just like the one in the left of Fig. 5.10, and use it to find the optimized U' for the next time step.

In comparison with the method described in Sec. 5.2.3, we have presented the corresponding statevector simulation results in Fig. 5.12(a). As discussed in Sec. 5.5, the approach we used to approximate the ITE operator to a unitary one resulted in an unexpected increase in energy during the ITE process. We suspect that this issue arises from the errors introduced when approximating the ITE operator to a unitary form. To further investigate, we also simulated the error

$$\Delta(\tau) = \left\| \frac{e^{-Hd\tau} |\psi(\tau)\rangle}{\sqrt{\langle \psi(\tau) | e^{-2Hd\tau} | \psi(\tau) \rangle}} - e^{-iA(\tau)d\tau} |\psi(\tau)\rangle \right\|, \quad (5.17)$$

and displayed the results in Fig. 5.12(b). It can be observed that the error and energy follow similar trends, leading us to conclude that the energy error is likely caused by the approximation error.

The other method involves adding one extra ancilla qubit, similar to the PITE algorithm. This approach allows the expansion of the non-unitary ITE operator into a unitary

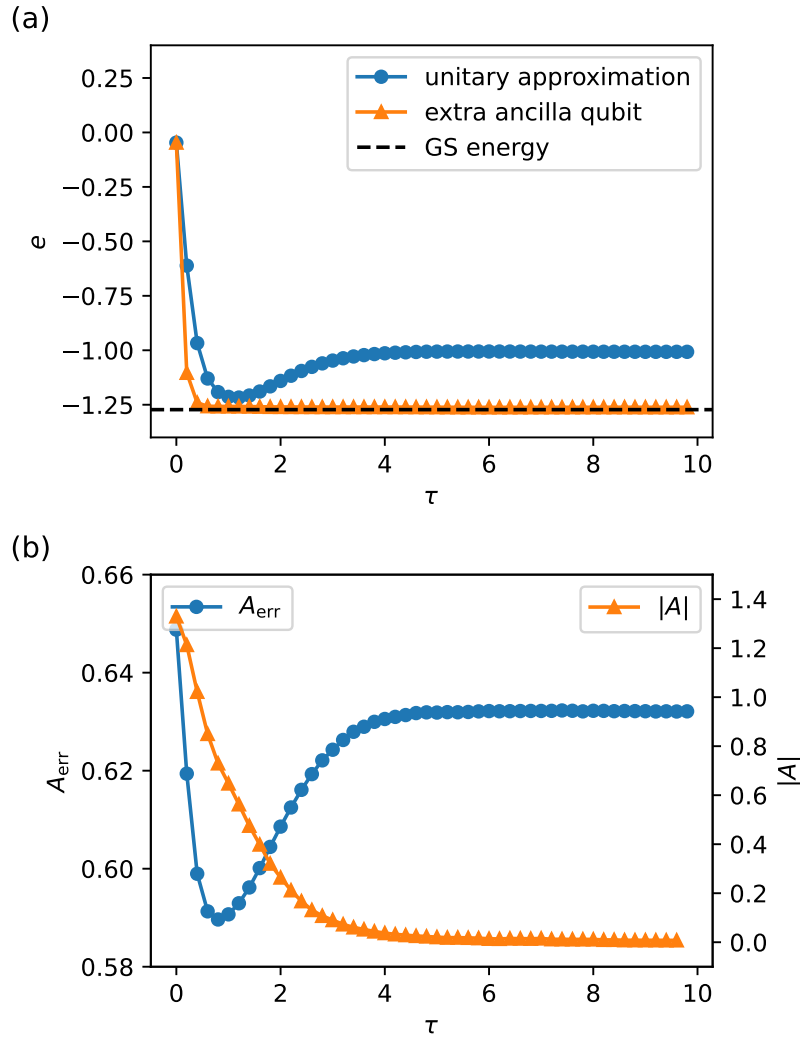


Figure 5.12: The comparison with two different method introduced. (b) The cost function value introduced in the Fig. 5.11 . It represent the error between the ITE operator $e^{-Hd\tau}$ and e^{-iAdt} .

Algorithm 1 QITE algorithm 1

- Step 1: Given the current U , find the environment R and L , and construct the cost function as Eq. 5.14 to find the optimized operator $A(\tau)$.
- Step 2: Construct the cost function, the quantum circuit as the one Fig.5.3(b) but replace the unitary gate $e^{-iHd\tau}$ with $e^{-iA(\tau)d\tau}$. Then find the optimized U' to maximize the cost function.
- Step 3: Update U as U' .
-

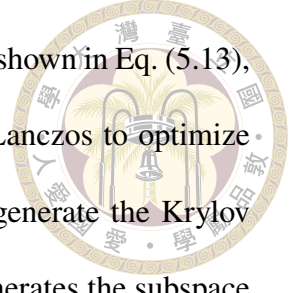
one by extending the quantum system from n qubits to $n + 1$ qubits. In our case, where the bond dimension $D = 2$, this means that we need to use 7 qubits to perform ITE on the quantum circuit, rather than the 6 qubits required for real-time evolution or previous method, as shown in Fig. 5.7. In order to implement a non-unitary operator M on a quantum computer, the operator is embedded within a larger unitary matrix. The unitary operator $U(M)$ is constructed as a block encoding [134, 143], where the non-unitary M is embedded as a submatrix within the unitary matrix, defined as:

$$U(M) = \begin{pmatrix} M & \sqrt{1 - M^2} \\ \sqrt{1 - M^2} & -M \end{pmatrix}. \quad (5.18)$$

Here, $\sqrt{1 - M^2}$ is formally defined in terms of the singular value decomposition (SVD) of M . It can then be straightforwardly verified that $U^\dagger U = I$, as long as the singular values are less than or equal to 1, which may always be achieved by rescaling M to some M/α . If M is ITE operator, this rescaling does not change or affect the optimization of the cost function used to find the new state $|\psi(X(\tau + d\tau/2))\rangle$. Additionally, this method can be utilized to construct environment gates L and R , by setting $M = l$ and r respectively, as described in Sec. 5.1.2. Since the singular values of l and r are always less than or equal to 1, rescaling is not necessary in these cases. For more detailed explanations of these methods, please refer to Sec. 5.2.

5.3.2 Quantum Lanczos Algorithm

The Quantum Lanczos (QLanczos) algorithm, first introduced by Ref. [114], was developed to address the limitations of the QITE method, particularly the errors that arise



when approximating the non-unitary ITE operator by a unitary one as shown in Eq. (5.13), and converges more rapidly than QITE. In our work, we employ QLanczos to optimize our simulation results. Unlike traditional Lanczos algorithms that generate the Krylov subspace from vectors like $|\phi\rangle, |H|\phi\rangle, H^2|\phi\rangle, \dots$, etc., QLanczos generates the subspace directly from the ITE. Specifically, the subspace is constructed from the sequence of vectors $|\phi_l\rangle = e^{-ld\tau H} |\phi_0\rangle$. Here, the regularization parameter s is introduced to ensure that the overlap between all the basis vectors $\{|\phi_l\rangle\}$ stays below a specified threshold s . If this overlap exceeds s , the corresponding state will be excluded from the basis. Once the basis is selected, The Hamiltonian matrix $H_{l,l'} = \langle\phi_l|\hat{H}|\phi_{l'}\rangle$ and the norm matrix $S_{l,l'} = \langle\phi_l|\phi_{l'}\rangle$ are constructed based on the chosen basis. These matrices are then used to solve the generalized eigenvalue problem $\mathbf{H}\mathbf{x} = E\mathbf{S}\mathbf{x}$, where the smallest eigenvalue provides an approximation of the ground state energy. The whole algorithm is shown in List. 2.

Algorithm 2 Quantum Lanczos algorithm

- Step 1: Given the result of QITE algorithm, i.e. value of different time steps, set $|\phi_0\rangle = |\phi_{l=0}\rangle$, it is the first element of the basis sequence.
- Step 2: Find the next $|\phi_{l=l}\rangle$, s.t. $|\langle\phi_{l=l}|\phi_0\rangle| < s$, where s is a regularization parameter $0 < s < 1$, then add $|\phi_{l=l}\rangle$ into the chosen basis sequence $\{|\Phi_l\rangle\}$.
- Step 3: Set $|\phi_0\rangle = |\phi_{l=l}\rangle$ obtained from Step 2 and then repeat Step 2 until the desired number of basis is reached.
- Step 4: Calculate $S_{l,l'} = \langle\Phi_l|\Phi_{l'}\rangle$, $H_{l,l'} = \langle\Phi_l|\hat{H}|\Phi_{l'}\rangle$ and solve the generalized eigenvalue equation $\mathbf{H}\mathbf{x} = E\mathbf{S}\mathbf{x}$, the smallest solution of E becomes an approximation of the ground state energy.
-

In our simulation, we save the unitary gates for each time step and then convert them into the unit cell of uMPS form (see Fig. 5.5). We then simulate the norm matrix elements $S_{l,l'}$ by calculating the leading eigenvalue of the transfer matrix corresponding to each uMPS. For the Hamiltonian matrix elements $H_{l,l'}$, We utilize standard tensor network

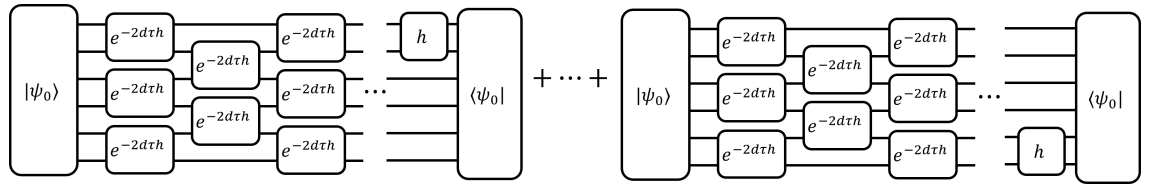
techniques, such as constructing the network as shown in the Fig. 5.3(a), but with the replacement of the real-time evolution operator by the local Hamiltonian h . The QLanczos algorithm is employed to refine and optimize these simulation results, with a detailed discussion provided in Sec. 5.5.

5.4 Quantum Imaginary-time Evolution for Finite System

In this section, we explore the ITE based on quantum circuits to simulate 1D finite-size systems. Unlike our approach for 1D infinite-size systems discussed in Sec. 5.3, this method does not require any variational ansatz. Since the Hamiltonian H commutes with the evolution operator $e^{-\tau H}$, the energy of the quantum system can be expressed as

$$\langle \psi(\tau) | H | \psi(\tau) \rangle = \langle \psi_0 | e^{-\tau H} H e^{-\tau H} | \psi_0 \rangle = \langle \psi_0 | H e^{-2\tau H} | \psi_0 \rangle. \quad (5.19)$$

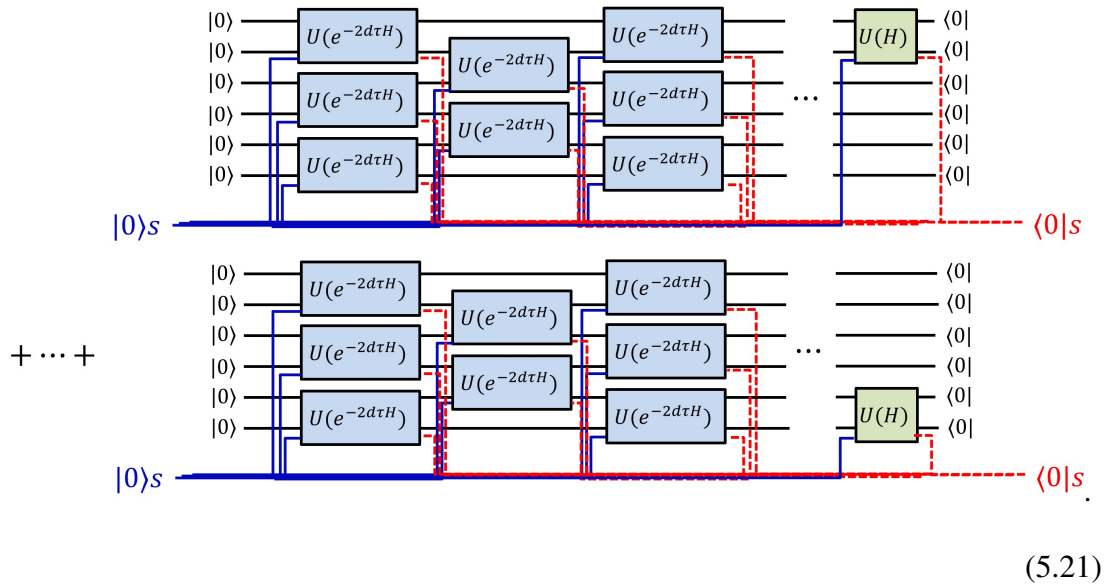
By applying the Suzuki–Trotter decomposition, Eq. (5.19) can be represented as the tensor network diagram:



$$\langle \psi_0 | e^{-\tau H} H e^{-\tau H} | \psi_0 \rangle = \langle \psi_0 | H e^{-2\tau H} | \psi_0 \rangle. \quad (5.20)$$

However, it is important to note that the ITE operator $e^{-2d\tau H}$ and Hamiltonian H are both non-unitary in general. To handle this, we can employ the technique discussed in Sec. 5.3.1, where an ancilla qubit is added to every operator $e^{-2d\tau H}$ and H , expanding

the operator to a unitary form while preserving the correct physical manipulation. If we consider the initial state as $|000\cdots\rangle$, this implies that the elements of the tensor ψ_0 in Eq. (5.20) will be zero except when all indices are 0, in which case it will be 1. Consequently, this tensor network diagram can be explicitly converted into a quantum circuit as follows:



Note that given that each ITE operator requires an additional ancilla qubit, the total number of ancilla qubits needed will increase linearly with the number of time steps. More concretely, the required number of ancilla qubits is $\lceil \frac{n_l(2N-1)}{2} \rceil$, where n_l is the number of evolution layer and N is the system size. However, it's important to note that these ancilla qubits can be reused. After each ITE operator is applied, a measurement is performed, the result is saved, and the qubit is reset to the zero state for reuse in the next ITE operator expansion. This reuse strategy reduces the required number of ancilla qubits to just $N - 1$, independent of the number of time steps, as illustrated in Fig. 5.13.

It is important to delve into the meaning of the measurement process illustrated in 5.13. When we consider the computational basis $\{|\psi_i\rangle\}$ and specify $|\psi_0\rangle = |000\cdots\rangle$ is

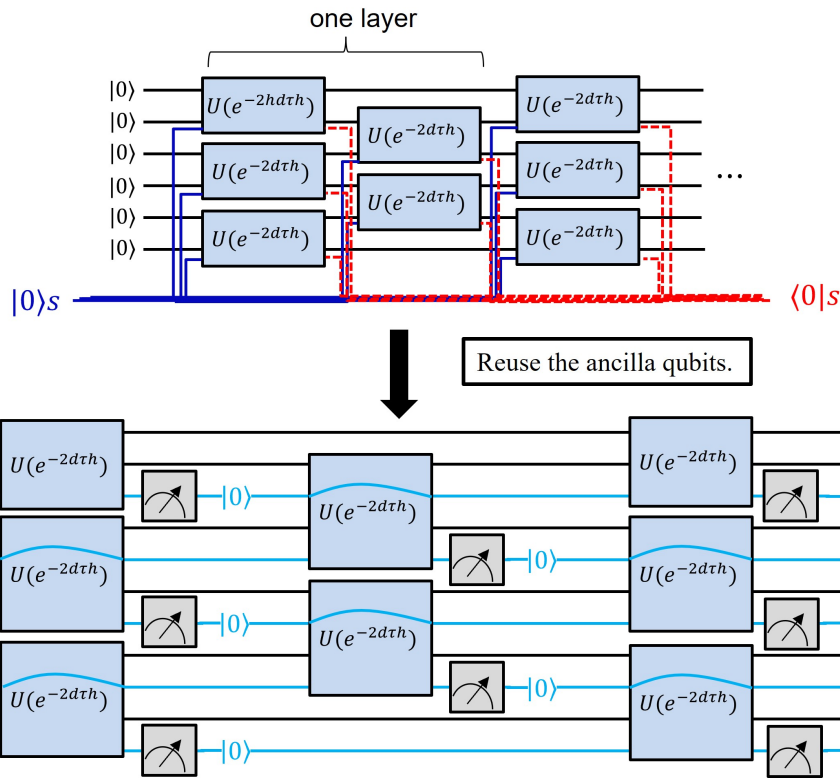


Figure 5.13: The ancilla qubits can be reused so they will not increase linearly with time evolution.

the all-zero state, we assume that $e^{-2\tau H}|\psi_0\rangle = \sum_i c_i |\psi_i\rangle$. The energy of the quantum system, as defined in Eq. (5.19), is then given by:

$$e = \langle \psi_0 | H e^{-2\tau H} | \psi_0 \rangle = \langle \psi_0 | \left(\sum_i c_i |\psi_i\rangle \right) = c_0. \quad (5.22)$$

This implies that the coefficient of the all-zero state, c_0 , represents the system's energy. However, when we perform the measurement, it only provides information about the probability P_0 that the state remains in the all-zero state, where $P_0 = |c_0|^2 = e^2$. As a result, the measurement process allows us to estimate only the absolute value of the energy, $|e|$, rather than its exact value.



5.5 Simulation Results

In this section, we will present all the simulation results we have obtained. The results include simulations for both 1D infinite-size and 1D finite-size systems. Specifically, we consider the Transverse Field Ising Model (TFIM) with the Hamiltonian

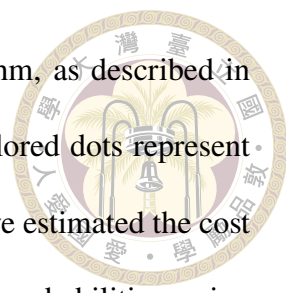
$$H = \sum_i -J\sigma_i^z\sigma_{i+1}^z + h\sigma_i^x, \quad (5.23)$$

where $J = 1$ and $h = 1$.

5.5.1 Results for 1D Infinite-size

In our simulations of the 1D infinite-size system, we applied the algorithms described in the previous section, with a bond dimension $D = 2$, to handle the non-unitary nature of the ITE operator. The first approach approximates the ITE operator to a unitary form, $e^{-id\tau A(\tau)}$, where $A(\tau)$ is a Hermitian operator obtained from the optimization of Eq. (5.14). The results of this approach are presented in Fig.5.14. The solid black line represents the statevector simulation. We observe that the energy unexpectedly increases at a certain time around $\tau \approx 1$, a phenomenon also reported in other studies [114, 117, 121]. This issue is may due to the limited correlation domain size [114] in our infinite-size system simulations, which is restricted to 2. Unlike finite-size systems, where the correlation domain size can be increased to mitigate this effect, our simulations remain constrained by this limitation.

To address the noise and estimation errors that arise from approximating the non-



unitary operator to a unitary one, we applied the QLanczos algorithm, as described in Sec. 5.3.2, to optimize the simulation results. In Fig.5.14(a), the colored dots represent the results from quantum simulators and IBMQ real devices, where we estimated the cost value in Eq. (5.2) (with H replaced by $A(\tau)$) through measurement probabilities, using 8192 samples (2^{13}). This cost function can be represented through a 6-qubit circuit, as shown in Fig. 5.7 (with h replaced by $A(\tau)$). Unlike the statevector simulation, which can directly calculate the exact cost value, the measurement-based approach cannot obtain the exact cost value, leading to noisier results. However, after applying the QLanczos algorithm, as shown in Fig.5.14(b), the optimized results closely match the exact energy obtained from the TN algorithm using the VUMPS [144] method with bond dimension $D = 2$. The black crosses in Fig.5.14(b) indicate the QLanczos bases selected during the optimization. Notably, the final basis chosen occurs at $\tau = 4.2$, beyond which no new basis states are selected, and the energy remains unchanged through subsequent Lanczos iterations.

We also implemented the algorithm to handle the non-unitary ITE operator by adding an extra ancilla qubit. Unlike the 6-qubit circuit previously discussed, this method requires a 7-qubit circuit. The statevector simulation results for this approach are represented by the orange triangles shown in Fig. 5.15(a). Additionally, ITE simulations were performed using the tensor-network algorithm and the TDVP method [1] with bond dimension $D = 2$, both implemented with the Cytex library [145]. The results from this classical TN algorithm are depicted by the blue points in Fig. 5.15(a), and they closely match the statevector simulation results. We also conducted simulations on an IBMQ real device; however, as expected, the results are noisy due to the fact that measurement-based quantum circuits

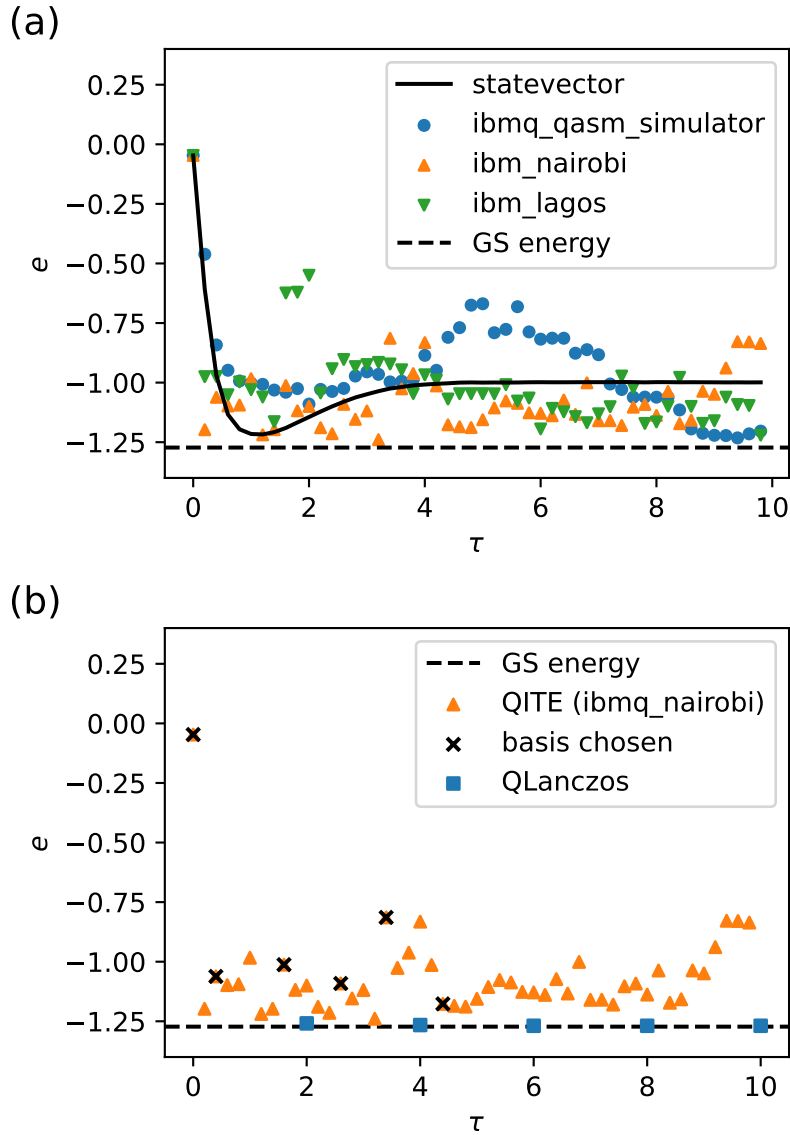
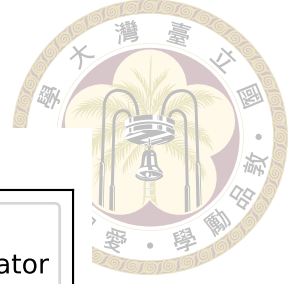


Figure 5.14: (a) The results of ITE on both a simulator and IBMQ real devices, where we approximate the ITE operator as a unitary one, $e^{-iA(\tau)d\tau}$. The black solid line shows the result of the statevector simulation, while the blue circle markers correspond to results obtained from the `ibmq_qasm_simulator`. The upward- and downward-triangle markers indicate results from two different IBMQ quantum real devices. The dashed black line indicates the ground-state energy obtained from the VUMPS algorithm with a virtual bond dimension $D = 2$. (b) The results of the QLanczos algorithm, with the regularization parameter set to $s = 0.95$. The orange triangle markers correspond to results from the real quantum device `ibmq_nairobi` (as seen with markers in (a)). The crosses represent the chosen basis for the Krylov subspace, and the blue square markers denote the energy results from the QLanczos algorithm.

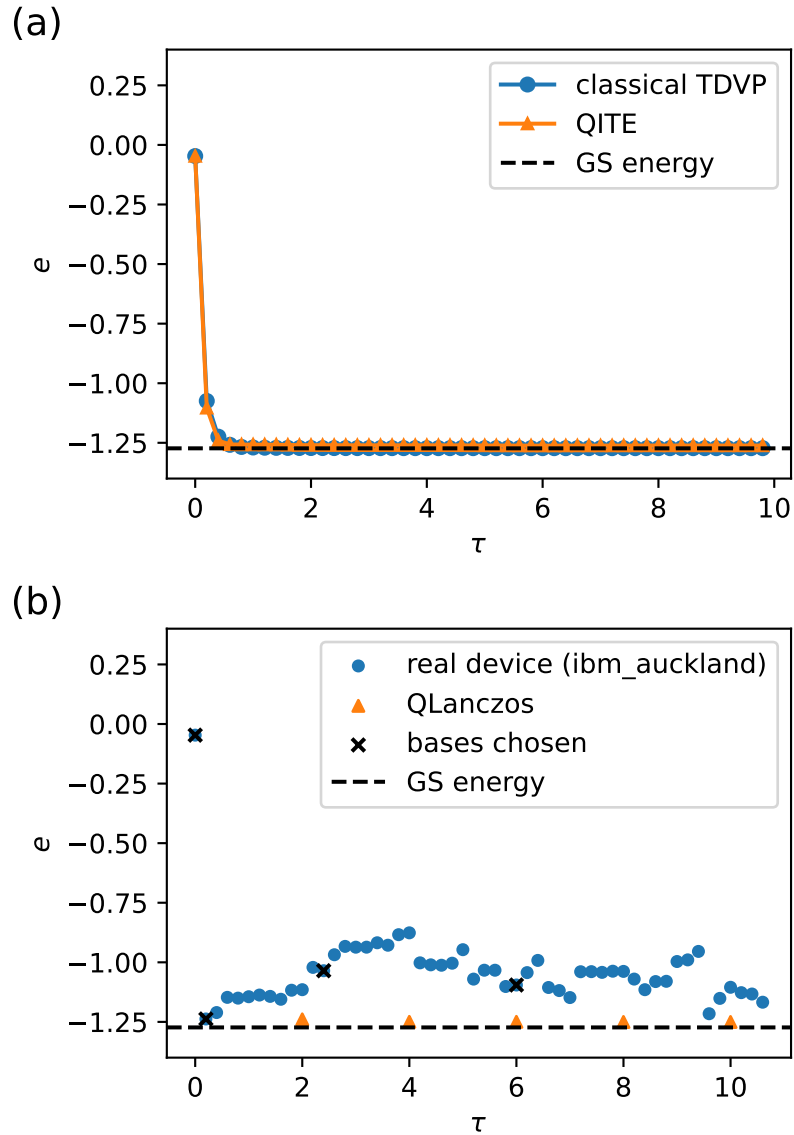


Figure 5.15: (a) The blue dots represent results from the classical TDVP TN algorithm with bond dimension $D = 2$, while the orange markers show results from the ITE method, where we extend the dimension of the ITE operator to a unitary one using an additional ancilla qubit, as described in Sec. 5.7. (b) The results from the real quantum device 'ibmq_auckland', optimized using the QLanczos algorithm.

do not provide the exact cost function value. Despite this noise, we applied the QLanczos algorithm to optimize the results, as shown in Fig. 5.15(b). The QLanczos optimization significantly reduces the noise, bringing the results closer to the ground state energy.

Both methods presented above start from an initial state where all parameters $\{\theta_i\}$ are set to zero, with the ansatz depicted in Fig. 5.8. The cost function, constructed as a quantum circuit, was evaluated using real quantum devices (as presented in Fig. 5.14 and 5.15(b)). However, it is important to note that all optimization processes were performed on a classical computer. For optimization, we utilized the `scipy.optimize.minimize` function from the Python package, specifically choosing the ‘COBYLA’ method, with the tolerance set to around 10^{-3} . Additionally, it is noteworthy that in the first method, where we approximate the ITE operator to a unitary one, the energy tends to increase after some time, which is not an expected outcome. In contrast, the second method, which incorporates an extra ancilla qubit, does not exhibit this issue and converges well to the exact energy. This suggests that the unexpected energy increase observed in the first method is likely due to the errors introduced when approximating the non-unitary ITE operator as a unitary operator.

5.5.2 Results for 1D Finite-size

To benchmark our simulation results for the 1D finite-size system, we first performed exact diagonalization (ED) to find the ground state energy. Following this, we constructed the TN as depicted in Eq. (5.20) to carry out the ITE. The simulation results are presented in Fig. 5.16(a), where we observe that the energies converge close to the ground state energy but are slightly lower. This error is caused by the Trotter-Suzuki decom-

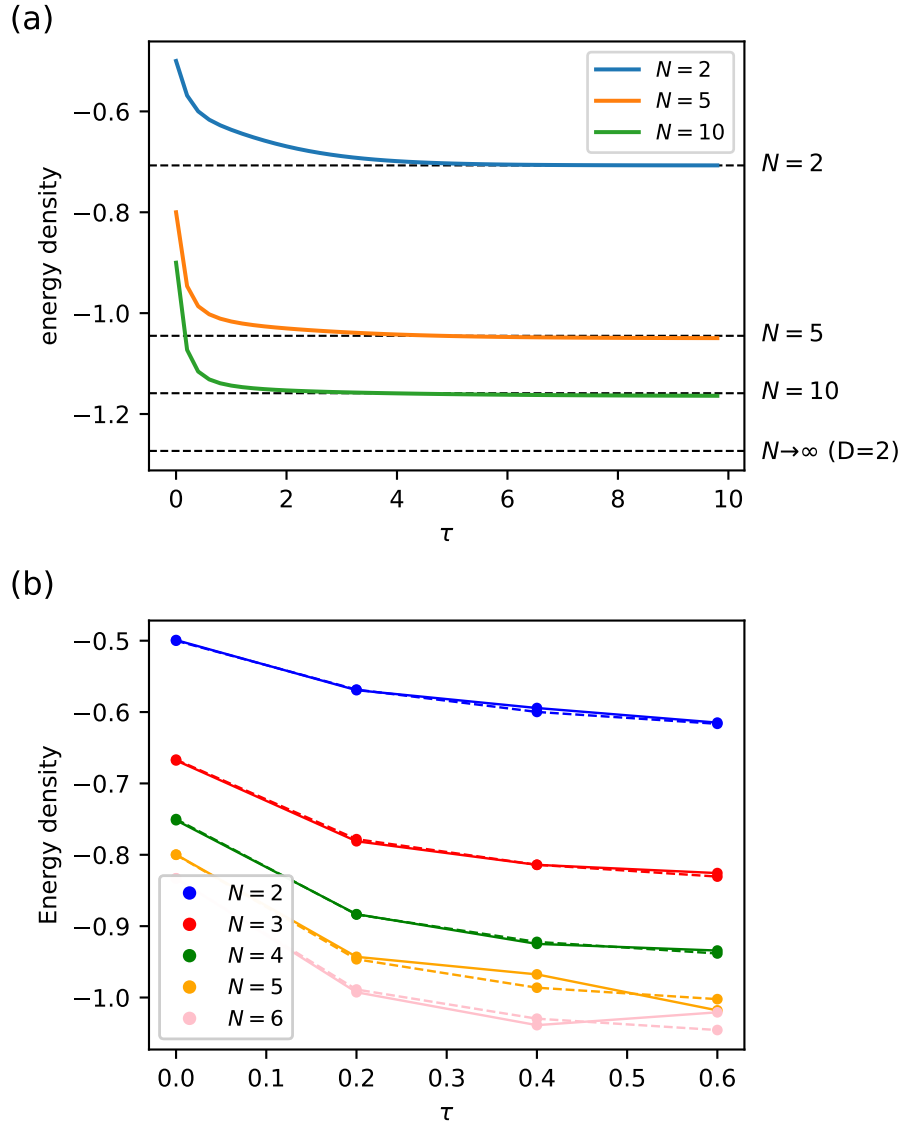


Figure 5.16: The solid lines show the classical TN simulation results from the tensor diagram 5.20. The dashed lines for finite-size systems represent the ED, and the dashed line for the infinite-size system corresponds to the VUMPS results with bond dimension $D = 2$. (b) The dashed lines represent the statevector results from the quantum circuit, as described in Eq. (5.21), which have been confirmed to match the TN results from (a). The solid lines in (b) represent results from the quantum circuit simulator using reused ancilla qubits, as shown in Fig. 5.13, with a shot number of 8192.

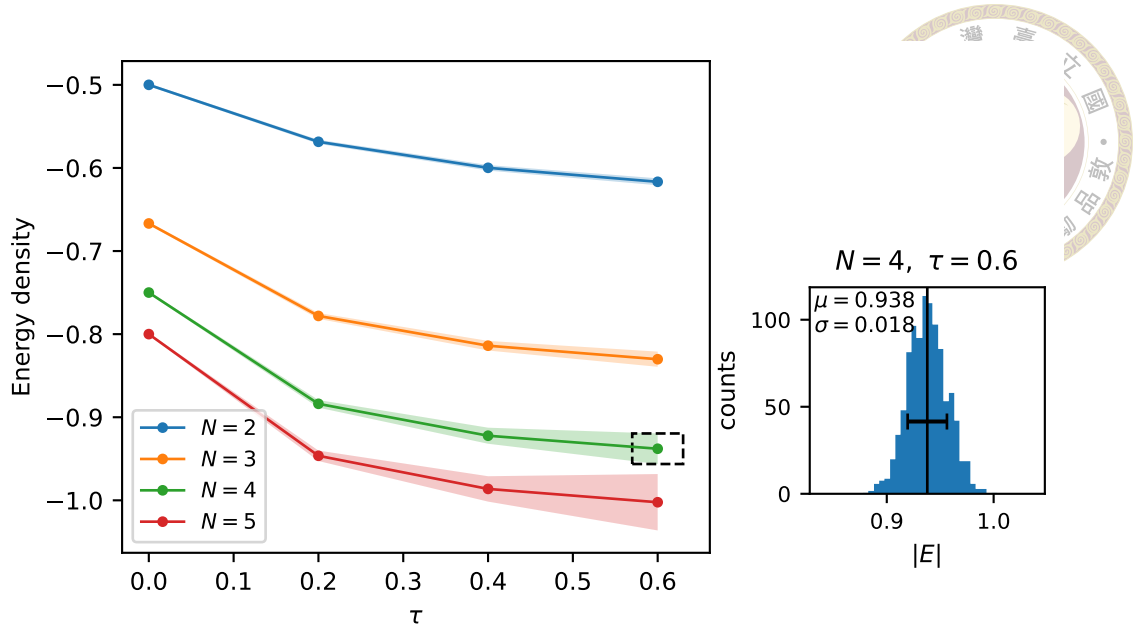


Figure 5.17: Results from the quantum circuit simulator using reused ancilla qubits, as shown in Fig. 5.13, obtained from 1000 independent simulation runs. The points in the main panel represent the mean energy densities, and the colored shaded regions indicate the corresponding standard deviations. The dashed rectangle marks the data point for $N = 4$ at $\tau = 0.6$, from which the inset is constructed. The inset shows the distribution of the energy density for $N = 4$ at $\tau = 0.6$. The histogram displays the results of 1000 simulation runs with a shot number of 8192; the vertical black line indicates the mean value, and the error bar denotes the standard deviation.

position applied to the $e^{-2\tau H}$ term. In fact, if we construct the TN diagram to simulate $\langle \psi_0 | e^{-\tau H} H e^{-\tau H} | \psi_0 \rangle$, where the Hamiltonian is placed in the middle of the TN instead of at the right end, it would converge exactly to the ground state energy without this slight error.

Additionally, we implemented this algorithm on a quantum circuit by expanding the ITE operator to a unitary form with an extra ancilla qubit, reusing the qubit as shown in the quantum circuit in Fig. 5.13. The results are presented in Fig. 5.16(b), where the dashed line represents the statevector simulation results, and the solid line represents the results through perform the measurement with 8192 shots. We also repeated this experiment 1000 times, and compute the mean value and standard deviation, the results are shown in

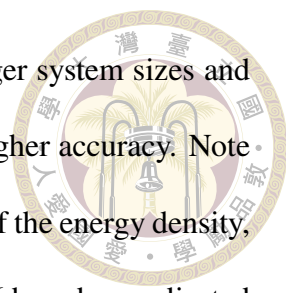


Fig. 5.17. It is evident that the standard deviation increases with larger system sizes and longer time evolution, indicating that more shots are necessary for higher accuracy. Note that the quantum circuit in Fig. 5.13 only provides the absolute value of the energy density, as discussed in Sec. 5.4. Therefore, the simulation results in Figs. 5.16 have been adjusted by taking the negative of the measured results, so $e = -|e|$.

5.6 Statistical Effects of Quantum Measurements

In the previous section, we presented simulation results obtained on quantum devices (See Fig. 5.15(b)). We observe that the energy density does not converge to the ground-state value during ITE unless further optimization is performed using the QLanczos method. Despite the fact that current quantum devices are still affected by significant noise, the primary limitation in this case arises from the estimation of the cost value via measurement outcomes from quantum circuits, as shown in Fig. 5.7. (The ITE requires an additional ancilla qubit used for the cost evaluation.) In this section, we will have a further discussion on how obtaining the cost value from quantum measurements affects the convergence of the energy during ITE.

As shown in Fig. 5.7, the cost value is obtained from a quantum circuit as the overlap with the all-zero state. On a real quantum device, this quantity can only be estimated by performing repeated measurements of the output state, yielding the probability P_0 of obtaining the all-zero bitstring, which serves as the cost value. This procedure corresponds to a Bernoulli process, where a “success” is defined as a measurement outcome in the all-zero state and a “failure” otherwise. In such a process, the standard deviation of the

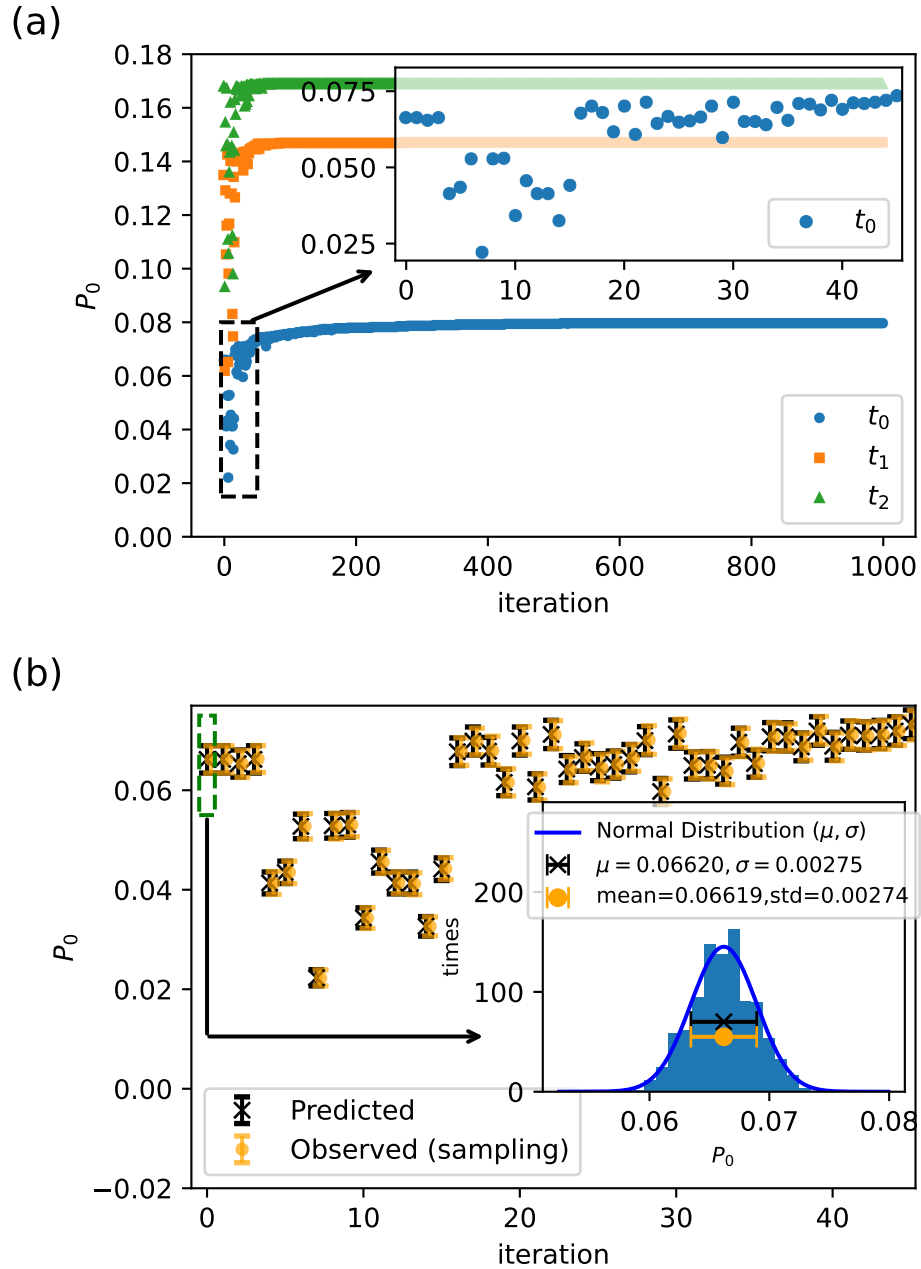
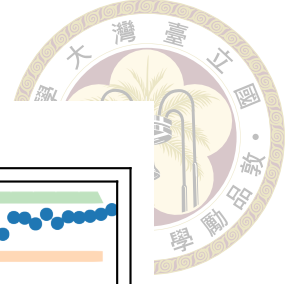


Figure 5.18: (a) Cost value during the iterations of the maximization process, obtained directly from the statevector, for the first three time steps of QITE in Fig. 5.15(a). The inset shows a zoom-in of the dashed-rectangle region. (b) Comparison of measured mean values and standard deviations with predictions from the statevector probability P_0 and Eq. (5.24) for the first time step of QITE and the first few iterations of the maximization process, using $N_{\text{shot}} = 8192$ and 1000 independent repetitions. The inset shows the histogram obtained from 1000 repetitions for the first iteration. In the inset, the orange error bar with a center dot represents the result from measurement statistics, while the black error bar with a cross marker denotes the prediction from P_0 and Eq. (5.24).

estimator for P_0 is given by

$$\sigma = \sqrt{\frac{P_0(1 - P_0)}{N_{\text{shots}}}}, \quad (5.24)$$

where N_{shots} denotes the number of measurement shots. To illustrate this effect, we extract data from our simulation as shown in Fig. 5.18. Figure 5.18(a) displays the cost value during the iterations of the maximization process, where the values are obtained directly from the statevector, corresponding to the first three time steps of QITE in Fig. 5.15(a). Focusing on the first time step of QITE and the first few iterations in the maximization process, we perform measurements (on a simulator) to estimate the cost value with $N_{\text{shots}} = 8192$, and repeat this procedure 1000 times to evaluate the mean value and standard deviation. The results in Fig. 5.18(b) show that the measured mean values and standard deviations (orange error bars) agree closely with the range predicted from the statevector probability P_0 and the standard deviation given by Eq. (5.24) (blue error bars). The inset in Fig. 5.18(b) presents an example distribution for the first iteration shown in Fig. 5.18(b), demonstrating that it follows a normal distribution with a mean consistent with P_0 and a standard deviation in agreement with Eq. (5.24).

However, the statistical analysis discussed above corresponds to the case where the cost value is obtained from the statevector, with the parameterized quantum circuit fixed, and measurements are then performed. We now consider the scenario in which the circuit parameters are updated using the measurement-based method. The orange points in Fig. 5.19(a) show the cost value when the updates are performed from measurement outcomes (on a simulator) with $N_{\text{shots}} = 8192$, repeating the process 1000 times. For comparison, the statevector results are shown as the blue curve, with error bars given by Eq. (5.24), and the statistical uncertainty of the measurement-based update is indicated by



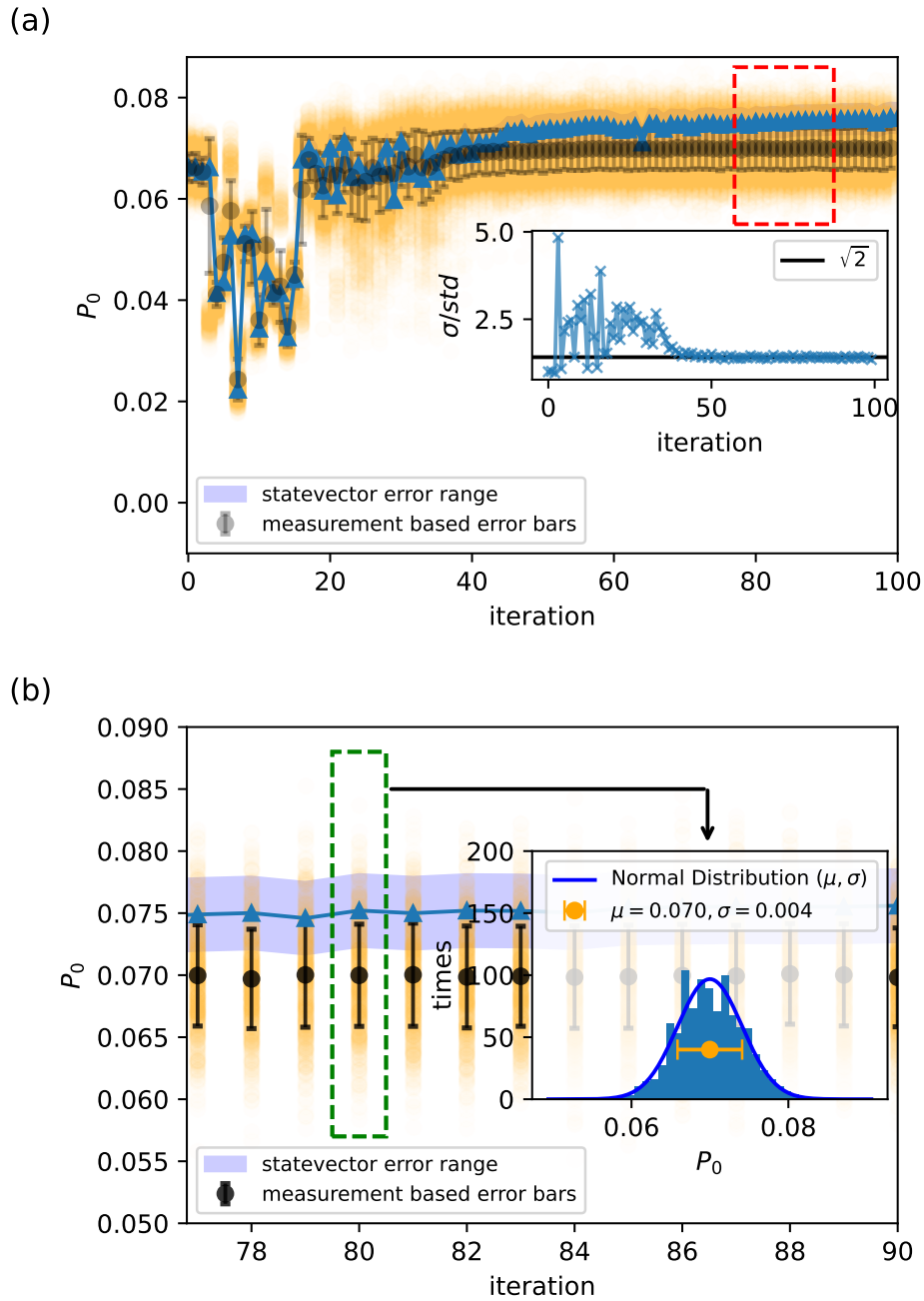


Figure 5.19: (a) Cost value during the iterations when the circuit parameters are updated using the measurement-based method (orange points with black error bars), repeated 1000 times with $N_{\text{shots}} = 8192$. For comparison, the statevector results are shown as a blue curve with error bars estimated from Eq. (5.24), also using $N_{\text{shots}} = 8192$. The inset shows the ratio between the standard deviation from the measurement-based update and that from the statevector case, which converges toward $\sqrt{2}$ in the saturation regime. (b) Zoom-in of the red dashed-rectangle region in panel (a). The inset in (b) shows the distribution of the cost value at the 80th iteration, obtained from 1000 repetitions. The blue line indicates a normal distribution with mean μ and standard deviation σ , where μ is the sample mean of the distribution and σ is the standard deviation estimated from the 1000 samples.

black error bars. We observe that at the beginning of the iteration, the cost value exhibits larger fluctuations and is not fully stable, but it becomes stable and saturates after several iterations. Nevertheless, the standard deviation in the measurement-based update remains larger than that in the statevector case. Interestingly, the ratio between the standard deviation from the measurement-based update and that from the statevector case approaches $\sqrt{2}$ in the saturation regime.

Recall that in the quantum real-device simulation shown in Fig. 5.15(b), the energy density does not converge to the ground-state energy during the time evolution. To further investigate this behavior, we perform simulations on a classical simulator, where the quantum circuit is updated using the measurement-based method, and execute the full QITE procedure up to 100 times, all starting from the same initial state. This yields 100 energy-evolution curves, as shown in Fig. 5.20(a).

In these simulations, the energy approaches the ground-state value and remains bounded below it. Furthermore, we perform statistical analysis using data from the last 10 time steps, corresponding to the stable region of the evolution, resulting in a total of 1000 data points. The resulting energy distribution, shown in Fig. 5.20(b), is surprisingly well approximated by a Maxwell–Boltzmann-like form:

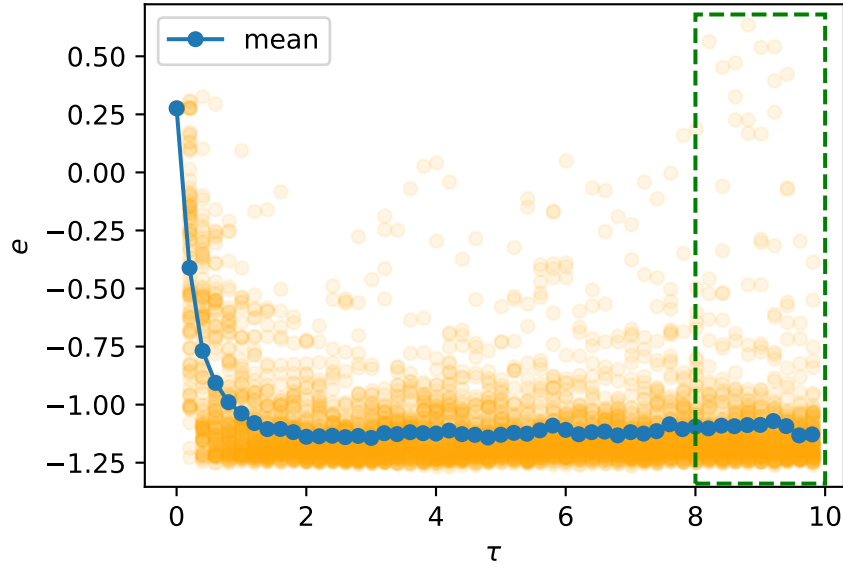
$$\rho_{MB}(E) = 2\sqrt{\frac{E - E_0}{\pi}} \left(\frac{1}{a}\right)^{3/2} \exp\left(-\frac{E - E_0}{a}\right), \quad (5.25)$$

with $a = 0.08$.

To characterize the empirical distribution of the sampled energy densities, we compared the numerical data with several candidate probability distributions. First, we con-



(a)



(b)

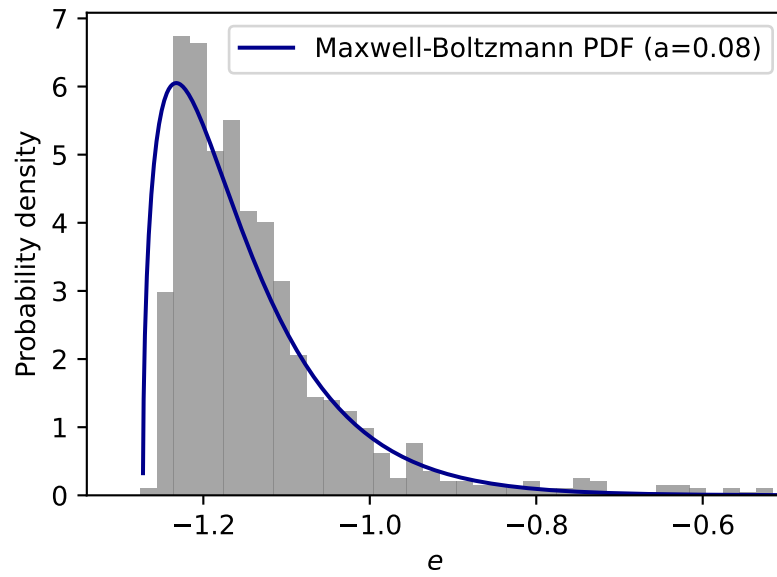


Figure 5.20: (a) Energy evolution curves from 100 independent QITE simulations on a noiseless simulator, where the circuit parameters are updated using the measurement-based method with $N_{\text{shots}} = 8192$. (b) Energy distribution obtained from the last 10 time steps of each simulation (1000 data points in total). The histogram is normalized to unit area, allowing for a direct comparison with the fitted probability density functions.

sider the Fréchet distribution, implemented as the inverse Weibull distribution, $p_F(E) \propto \left(\frac{E}{\lambda}\right)^{-1-c} \exp\left[-\left(\frac{E}{\lambda}\right)^{-c}\right]$, where the shape parameter c controls the asymptotic tail behavior, while the scale parameter λ sets the characteristic energy scale of the distribution. We also examine a shifted lognormal distribution, $p_{LN}(E) = \frac{1}{E s \sqrt{2\pi}} \exp\left[-\frac{(\ln(E/\lambda))^2}{2s^2}\right]$, in which the parameter s determines the width of the logarithmic fluctuations and the scale parameter λ fixes the typical magnitude of the energy density.

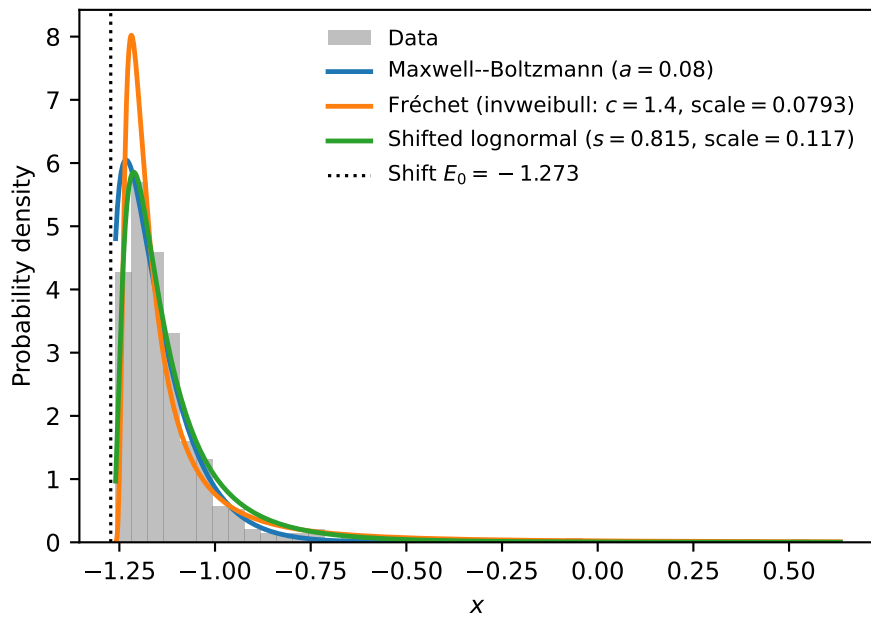


Figure 5.21: Empirical distribution of sampled energy densities (histogram) compared with fits to the Fréchet (inverse Weibull) and shifted lognormal distributions. All distributions are evaluated for the shifted variable $E - E_0$ to ensure support on positive energies, where the reference shift E_0 is indicated by the vertical dotted line. In both cases, the scale parameter λ sets the characteristic energy scale of the distribution, while the shape parameters control the width and tail behavior. The fitted parameters are reported in the legend. The histogram is normalized to unit area, allowing for a direct comparison with the fitted probability density functions.

In all cases, the distributions are evaluated for the shifted variable $E - E_0$ to ensure support on positive energies, where the reference shift E_0 is indicated in Fig. 5.21. The corresponding fit parameters are extracted from the numerical data and reported in the figure legend. While the distributions provide reasonable fits over part of the sampled

range, the microscopic origin of the observed distribution remains unclear and calls for further investigation.

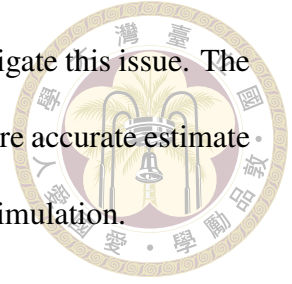


5.7 Conclusion

In this study, we propose algorithms based on quantum circuits to simulate the 1D infinite-size systems. We employ a variational ansatz, constructed using quantum circuits, to represent the projected state. Since the ITE operator is non-unitary and thus not directly implementable as a quantum gate, we use two methods to handle this. The first method approximates the non-unitary ITE operator as a unitary one without adding any additional dimensions. However, this approach leads to an unexpected energy increase during the ITE, likely due to the errors introduced when approximating the non-unitary operator with a unitary one. The second method involves extending the ITE operator by adding an extra ancilla qubit, which preserves the operator's properties while ensuring unitarity. This method doubles the operator's dimension and provides more accurate results, especially when compared with classical TN methods TDVP.

Although the method of using an extra ancilla qubit to extend the ITE operator for simulating the 1D infinite-size system performs well in statevector simulations, the measurement process needed to estimate the cost function value through sampling on quantum devices introduces inaccuracies. This sampling, which is necessary for quantum hardware, results in noise in the ITE simulation. In particular, the quantum measurements give rise to a distribution of the cost values, which follows a normal distribution, while the corresponding energy distribution we observe takes on a Maxwell–Boltzmann-like form. However,

we can apply the QLanczos algorithm as a post-processing step to mitigate this issue. The QLanczos algorithm helps to correct the noisy results, leading to a more accurate estimate of the ground-state energy, thus improving the reliability of the ITE simulation.



In summary, we have introduced methods to perform ITE on quantum circuits for 1D infinite-size quantum systems. In our study, we demonstrated the results for the 1D infinite-size system with a bond dimension $D = 2$ in the uMPS framework, which required only 7 qubits to perform the ITE. Although we only focus on $D = 2$ case, recent research has explored the mapping between MPS (including uMPS) and quantum gates for larger bond dimensions [103, 106, 110]. This suggests that the method we propose can, in theory, be generalized to larger bond dimensions, which would allow for the simulation of more complex and entangled quantum systems. Extending our approach to higher bond dimensions is a potential direction for future work.





Chapter 6 E_8 Symmetry of the Ising

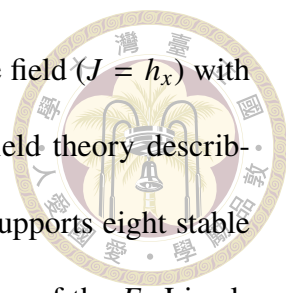
Model: Using Quantum

Circuits Simulation

The one dimensional transverse field Ising model (TFIM) with an additional longitudinal field is a central model in quantum many-body physics, offering a minimal setting to study quantum phase transitions and emergent particle-like excitations [146]. The corresponding Hamiltonian can be represented as

$$H = - \left(J \sum_i \sigma_i^z \sigma_{i+1}^z + h_x \sum_i \sigma_i^x + h_z \sum_i \sigma_i^z \right), \quad (6.1)$$

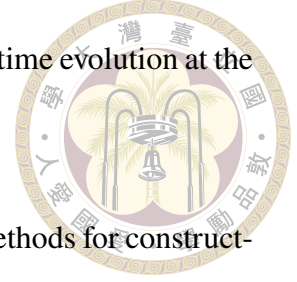
where J is the coupling term, and h_x and h_z are transverse and longitudinal magnetic fields, respectively. In the absence of a longitudinal field ($h_z = 0$), the model maps to free fermions, and for small transverse field h_x , its low-energy excitations consist of pairs of kinks (domain walls) separating regions of opposite magnetization. These domain walls can propagate freely without a longitudinal field. In contrast, introducing a finite longitudinal field ($h_z > 0$) generates a linear confining potential between domain walls, resulting in bound states that exhibit meson-like behavior [146–148].



A particularly remarkable regime arises at the critical transverse field ($J = h_x$) with a finite longitudinal field h_z . In the scaling limit, the continuum field theory describing this regime maps to the integrable E_8 model [149, 150], which supports eight stable particles (mesons) whose mass ratios are exactly fixed by the structure of the E_8 Lie algebra, and the ratios of the eight meson masses take universal values. This E_8 spectrum has been investigated extensively using tensor network (TN) methods [146, 151–155], and has even been observed experimentally in quasi-one-dimensional Ising-like magnets, including the ferromagnetic material CoNb_2O_6 [156] and the antiferromagnetic material $\text{BaCo}_2\text{V}_2\text{O}_8$ [157, 158]. However, simulations of this regime on quantum hardware remain scarce [159], due to the challenges posed by noise and circuit depth limitations in current devices. Moreover, accurately extracting the meson spectrum from real-time dynamics is particularly demanding, as it requires high-frequency resolution in the Fourier-transformed data.

The goal of this work is to simulate the TFIM with an longitudinal field under these critical conditions using a gate-based quantum computer, extract the meson masses from real-time dynamics, and compare their ratios with the E_8 predictions, at least for the first few spectral peaks, to assess whether the symmetry is faithfully reproduced. Specifically, we perform simulations on IBM's superconducting quantum device `ibm_torino` using the Qiskit framework [160]. However, current quantum hardware still suffers from considerable noise and decoherence, especially for deep circuits. To mitigate these limitations, several approaches have been proposed to compress or optimize quantum circuits using classical techniques such as tensor network (TN) [106, 110, 148], which aim to reduce circuit depth while maintaining accuracy. While several studies have explored TFIM dy-

namics using quantum hardware [111, 159, 161–166], simulating real-time evolution at the critical point remains particularly challenging.



To carry out the quantum simulation, we explore two different methods for constructing the time-evolution operator. The first approach uses first-order Trotter decomposition and construct the real-time evolution operator directly using native gate operations. The second approach employs Riemannian optimization [167], a TN-based technique that approximates the evolution operator using a fixed-depth variational circuit. We compare the performance of both approaches on real quantum hardware. Since current quantum devices operate in the Noisy Intermediate-Scale Quantum (NISQ) [168], noise mitigation is essential. We propose and implement an error mitigation strategy which allows us to correct for decay in the measured observables. After applying this mitigation, we extract frequency-domain spectra via Fourier transformation. Few main peaks show good agreement with the mass excitations predicted by E_8 symmetry.

The remainder of this chapter is organized as follows. Sec. 6.1 introduces the E_8 symmetry in the 1D Ising model with transverse and longitudinal fields. Sec. 6.2 explains quantum circuit construction and error mitigation. Sec. 6.3 describes the use of Riemannian optimization for low-depth circuit compression. Sec. 6.4 presents real-device simulation results in the frequency domain and compares them with E_8 theoretical predictions.

6.1 E_8 Symmetry in Quantum Ising Model

The model we consider is the one-dimensional TFIM with an additional longitudinal field, as defined in Eq. (6.1). At the quantum critical point ($h_x = J, h_z = 0$), the system

can be described by the Ising conformal field theory (CFT) with central charge $c = 1/2$. When the critical theory is perturbed by a small longitudinal field ($h_z \neq 0$), it relates to E_8 Lie algebra symmetry in the scaling limit [149, 150]. We list a few of the mass values in Table 6.1.

In this work, we investigate the emergent E_8 symmetry in the TFIM with a longitudinal field using a quantum simulation approach. We prepare an initial state with two domain walls, such that the spin configuration is up at the boundaries and down in the center (For example, $|\uparrow\uparrow\downarrow\downarrow\uparrow\rangle$ in an 5-site chain.) The dynamical structure factor, $S^y(k, \omega)$, is a good parameter for extracting meson masses from spin systems, which can be found by a two-point correlation function, $C^y(i, t) = \langle \psi_0 | \sigma_i^y(t) \sigma_i^y(0) | \psi_0 \rangle$ [151, 154, 156, 169]. However, it is too complicated to simulate this function on IBM Hardware. Instead, several studies have shown that the meson spectrum can be extracted from the entanglement spectrum [153], or the dynamics of the magnetization [146, 155, 170, 171], $\langle \sigma^z(t) \rangle = \frac{1}{L} \sum_{i=1}^L \langle \sigma_i^z(t) \rangle$. To further simplify the measurement, we propose using only the central-site magnetization, $\langle \sigma_{\text{cen}}^z(t) \rangle$, and extracting the spectrum from its frequency-domain representation, $\langle \sigma_{\text{cen}}^z(\omega) \rangle$. These spectral peaks are expected to match the mass ratios predicted by the E_8 in quantum field theory. We aim to use quantum devices provided from IBMQ to carry out the real-time evolution with initial states prepared from the two domain walls and extract the mass spectrum of the E_8 symmetry.

To verify our approach, we do the simulation on classical computer to perform exact real-time evolution via exact denationalization (ED) and compare the results with the expected E_8 spectrum. As shown in Fig. 6.1, we simulate a chain of 11 sites with the initial state prepared as $|\uparrow\uparrow\downarrow\downarrow\downarrow\downarrow\downarrow\uparrow\uparrow\rangle$. The figure presents both the time-domain dynamics of

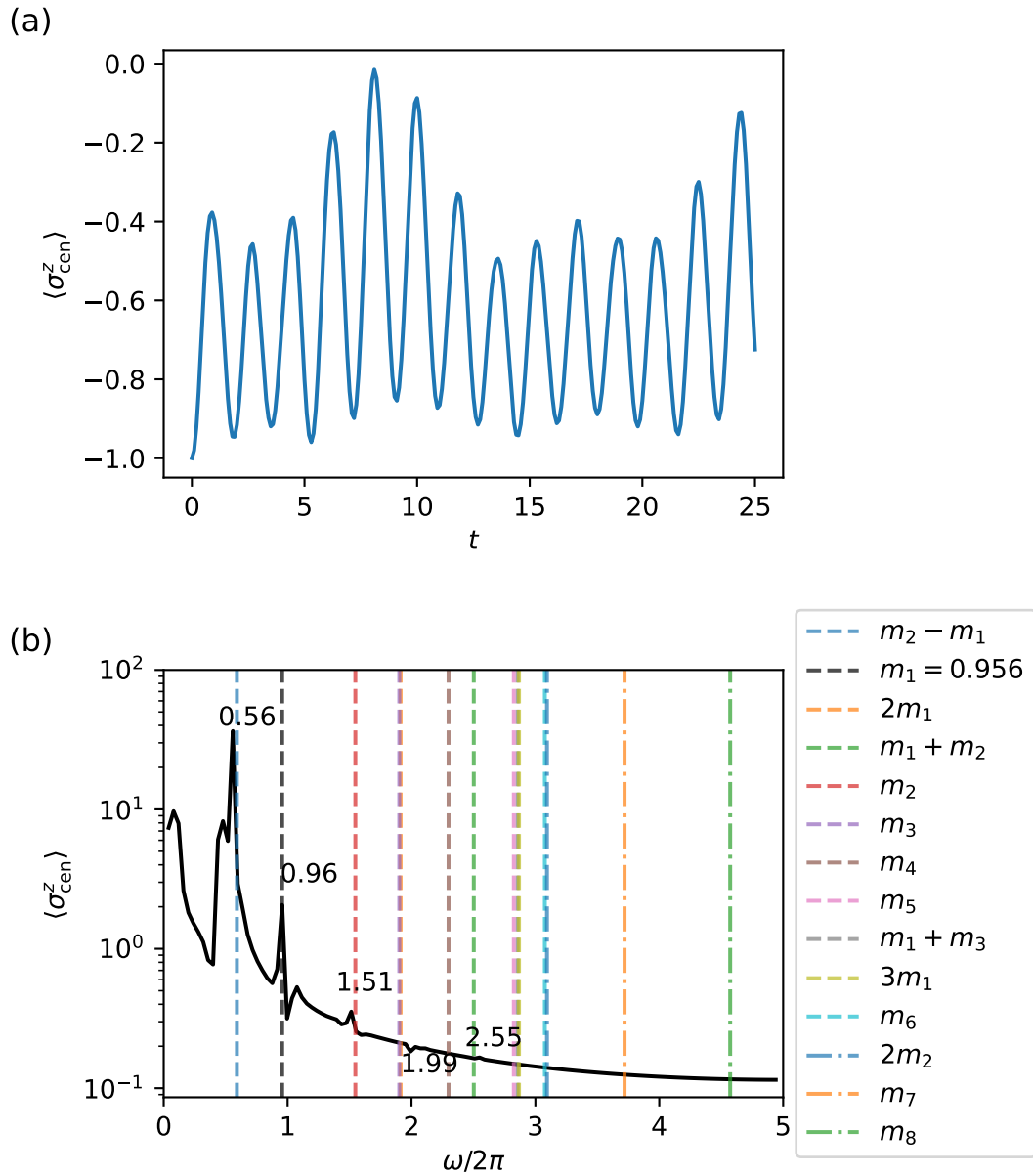
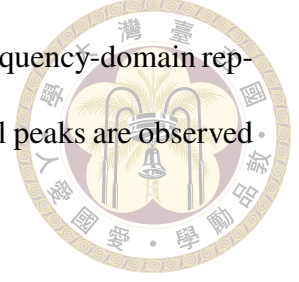


Figure 6.1: Simulation results obtained via exact diagonalization on a classical computer. The system evolves from the initial state $|\uparrow\uparrow\downarrow\downarrow\downarrow\downarrow\downarrow\uparrow\uparrow\rangle$ under the TFIM with a longitudinal field, using parameters $J = h_x = 1$ and $h_z = 3$. The observable is the central spin expectation value $\langle \sigma_{\text{cen}}^z(t) \rangle$. (a) Time-domain dynamics of the observable. (b) Frequency-domain spectrum obtained via Fourier transformation plotted on a semi-log scale. Colored dashed lines indicate several expected peak positions predicted by the E_8 mass ratios. The value of m_1 is selected manually, and the positions of the other expected peaks are computed by applying the known E_8 mass ratios. The peak values obtained from ED are denoted directly on the plot.

$\langle \sigma^z(t) \rangle$ at the center of the chain (Fig. 6.1(a)) and its corresponding frequency-domain representation obtained via Fourier transform (Fig. 6.1(b)). Clear spectral peaks are observed at frequencies that align with the theoretical E_8 mass ratios.



In addition to exact real-time evolution, we simulate the dynamics using the time-dependent variational principle (TDVP) [1, 101] within the TN framework, and also apply a first-order Trotter decomposition to construct the real-time evolution operator using TN methods, to mimic the statevector simulation used in quantum circuits. All simulations are implemented with the TN library Cytntx [145]. Additionally, we benchmarked simulations across different system sizes $L = 5, 8, 27$, and found that $L = 8$ is sufficient to capture the essential spectral features relevant to our analysis.

To validate the quantum simulation results presented, we also perform classical simulations of the real-time dynamics using multiple methods:

- (1) Exact Diagonalization (ED, as shown in Fig.6.1).
- (2) Time-Dependent Variational Principle (TDVP).
- (3) First-Order Trotter Decomposition via TN. This approach constructs the time-evolution operator as a product of local gates using first-order Trotterization and evolves the state using TN techniques. It mimics the statevector dynamics of quantum circuits on classical hardware.

The simulations are implemented using the Cytntx library [145]. All simulations are performed for the Hamiltonian defined in Eq. (6.1), with parameters set to $J = h_x = 1$, $h_z = 3$, and system size $L = 11$. The initial state is chosen as $|\uparrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\uparrow\rangle$, and we

track the time evolution of the central-site magnetization $\langle \sigma_{\text{cen}}^z(t) \rangle$. Fig. 6.2(a) shows the

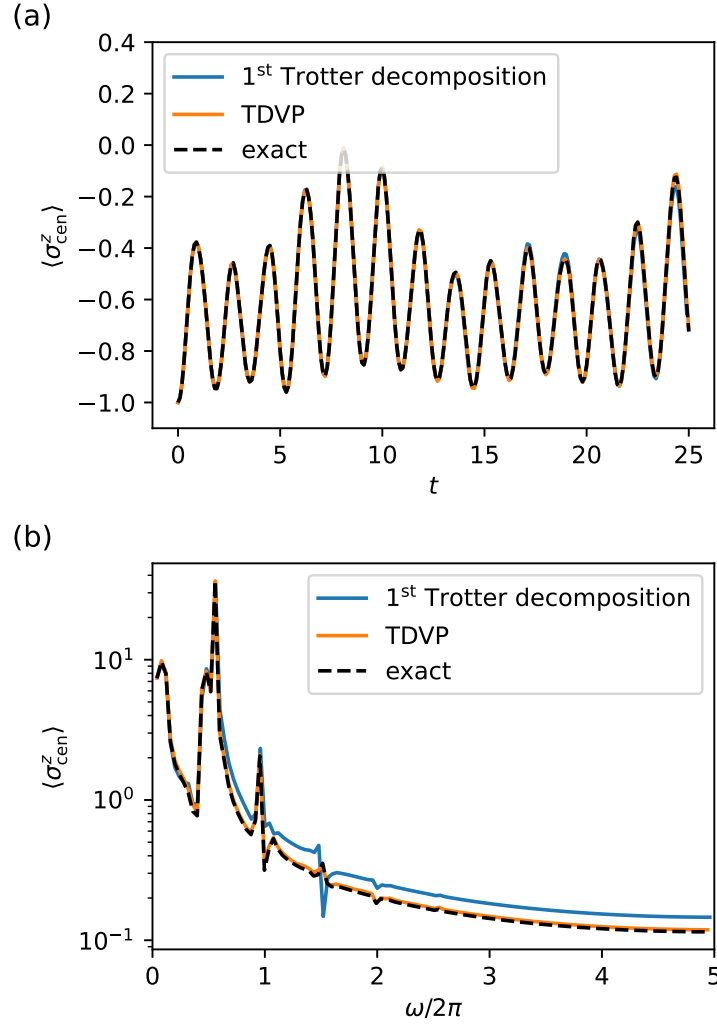


Figure 6.2: Classical simulation of the real-time evolution of the central-site magnetization $\langle \sigma_{\text{cen}}^z(t) \rangle$ using three methods: exact diagonalization (ED), time-dependent variational principle (TDVP), and first-order Trotter decomposition. Parameters are set to $J = h_x = 1$, $h_z = 3$, with system size $L = 11$ and initial state $|\uparrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\uparrow\rangle$. The bond dimension of TDVP is $D = 8$. (a) shows the results in the time domain, and (b) shows the corresponding frequency-domain spectrum obtained via Fourier transform.

time-domain results from all three methods. The TDVP simulation closely agrees with the ED with bond dimension only set as $D = 8$. The first-order Trotter decomposition also performs well for short to intermediate times, but small deviations emerge at longer times due to accumulated Trotter errors. While these discrepancies slightly distort the frequency-domain spectrum, the dominant peak positions remain consistent, as seen in

the Fourier-transformed data $\langle \sigma_{\text{cen}}^z(\omega) \rangle$ (see Fig. 6.2(b)). Importantly, these small Trotter errors are negligible compared to the level of noise encountered in quantum real-device simulations.



To explore finite-size effects, we further conduct simulations for different system sizes $L = 5, 8$, and 27 . The time evolution is simulated up to $t = 25$ with a time step $dt = 0.1$. The resulting spectrum in the frequency domain is shown in Fig. 6.3. The results indicate that $L = 8$ is already sufficient to capture the key spectral features of interest. This suggests that relatively small system sizes are sufficient to extract meaningful information about the E_8 mass spectrum in this model.

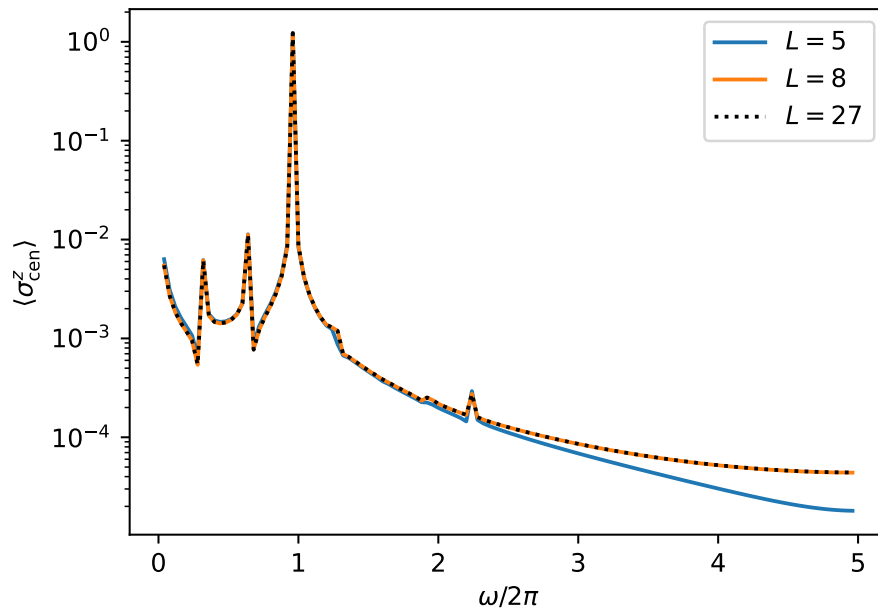


Figure 6.3: Frequency-domain spectrum $\langle \sigma_{\text{cen}}^z(\omega) \rangle$ obtained via exact ED for different system sizes: $L = 5, 7$, and 27 . Initial states are prepared such that the two central sites are in the down state while all other sites are in the up state.



6.2 Using IBMQ Quantum Real Device

6.2.1 Evolution Gate

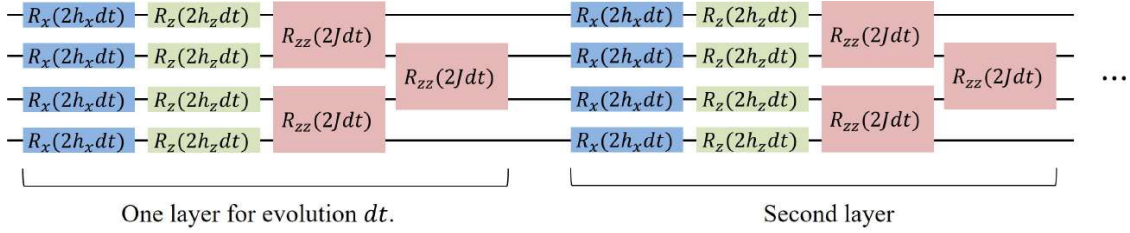


Figure 6.4: Quantum circuit construction of the real-time evolution operator for the TFIM using first-order Trotter decomposition, as described in Eq. (6.2). The circuit consists of sequential layers implementing the longitudinal field (RZ gates), transverse field (RX gates), and Ising interaction (RZZ gates) terms.

To perform real-time evolution on a quantum device, the first step is to construct the real-time evolution operator corresponding to the Hamiltonian. In the case of the TFIM with a longitudinal field (see Eq. (6.1)), we use first-order Trotter decomposition. The time evolution operator over a short time interval dt can be approximated as

$$U(dt) = e^{-idtH} \approx \left(\prod_i e^{ih_z dt \sigma_i^z} \right) \left(\prod_i e^{ih_x dt \sigma_i^x} \right) \times \left(\prod_{i \in \text{odd}} e^{iJ dt \sigma_i^z \sigma_{i+1}^z} \right) \left(\prod_{i \in \text{even}} e^{iJ dt \sigma_i^z \sigma_{i+1}^z} \right). \quad (6.2)$$

Each exponential term corresponds to unitary gates that can be directly implemented using native quantum gates: RZZ for the interaction term and RX/RZ for the transverse and longitudinal fields. The quantum circuit representing Trotter step is shown in Fig. 6.4.

In our quantum simulations, we utilize the Estimator primitive provided from IBM Quantum to directly evaluate the time-dependent observable $\langle \sigma_{\text{cen}}^z(t) \rangle$, rather than constructing both the evolution operator U and $U^\dagger$ around a measurement circuit. This ap-

proach allows us to access the expectation value of the central spin observable directly.



6.2.2 Error Mitigation

Current quantum hardware remains highly noisy due to gate errors, decoherence, and readout imperfections. As a result, applying error mitigation techniques are needed. One strategy discussed in [172] is to estimate gate-induced errors by preparing a reference circuit in which all gate parameters are set to zero. In this case, the time evolution circuit becomes an identity operation, allowing one to measure the accumulated noise during the time evolution. In our implementation, however, we further refine this approach by exploiting the following commutation relation:

$$\left[\sum_i \sigma_i^z \sigma_{i+1}^z, \sigma_{\text{cen}}^z \right] = 0 \quad (6.3)$$

which implies that the observable $\langle \sigma_{\text{cen}}^z(t) \rangle$ is a conserved quantity. Based on this, we construct a reference circuit in which all RX and RZ gate angles are set to zero, while the RZZ gates remain unchanged. Fig. 6.5(a) shows the results of our error mitigation procedure performed on the IBM Quantum device `ibm_torino`, and also original strategy. As seen in the figure, the refined method ($\square$ markers) captures the decay envelope of the dynamics more accurately, whereas the original method ($\triangle$ markers) fails to do so. Fig. 6.5(b) shows the final mitigated result, where the raw data ($\bullet$ markers) is divided by the refined reference ($\square$ markers); the corrected signal is shown as the $\blacktriangle$ markers.

To closely track the evolving state of the quantum hardware, we construct the physical

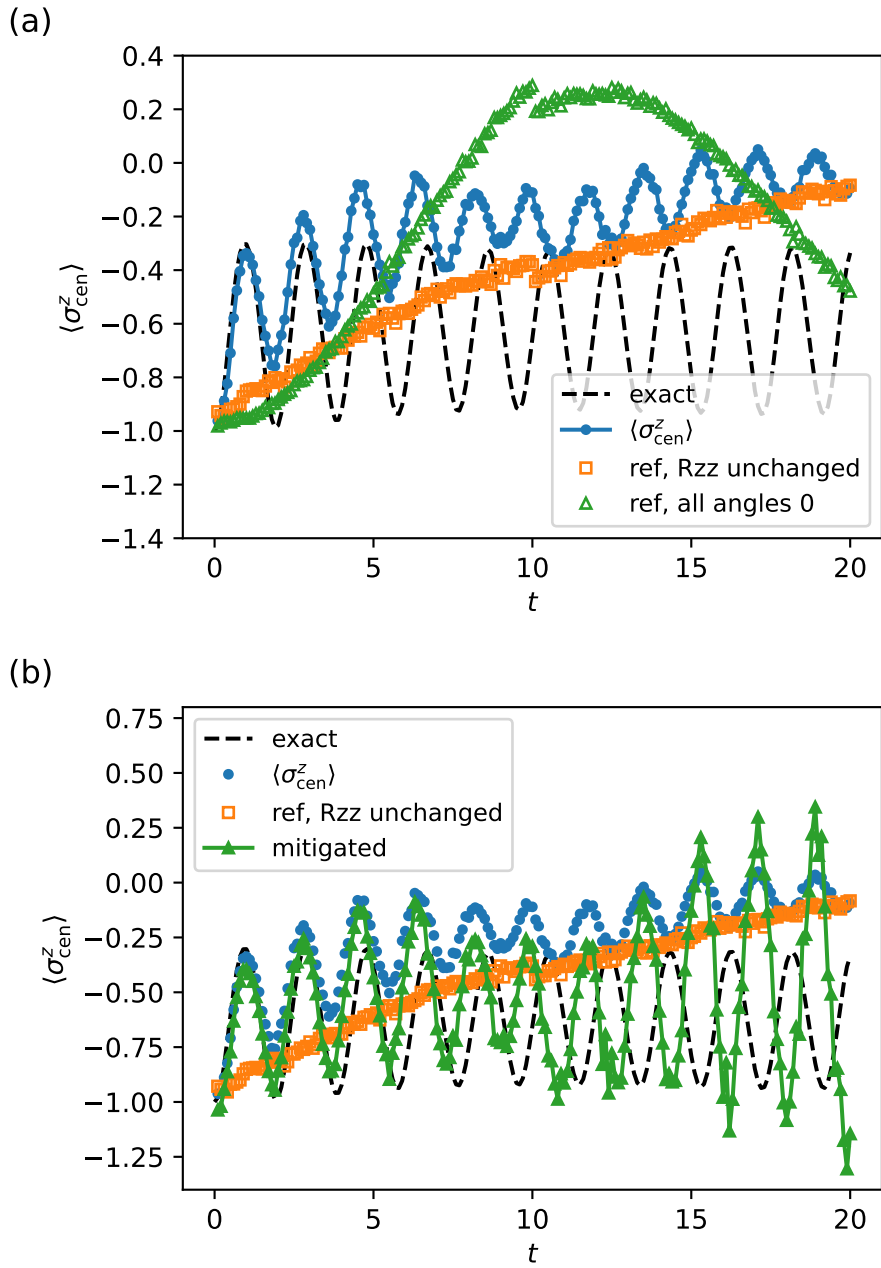


Figure 6.5: Real-time evolution results on the IBM Quantum device `ibm_torino` for an 8-site transverse-field Ising model with parameters $J = h_x = 1$, $h_z = 3$, initial state $|\uparrow\uparrow\downarrow\downarrow\uparrow\uparrow\rangle$, and 8192 shots per circuit. The observable $\langle \sigma^z(t) \rangle$ is measured at site 4 (the central site). (a) Comparison of two reference circuit strategies for error mitigation: $\bullet$ markers correspond to the raw measured signal, $\square$ markers correspond to setting all RX and RZ angles to zero while leaving RZZ gates unchanged, and $\triangle$ markers correspond to setting all gate parameters to zero. A small discontinuity is visible around $t = 10$, which may arise from splitting the job into two separate submissions. (b) Post-processed results: $\bullet$ markers show the original raw data, $\square$ markers are the refined reference circuit results, and $\blacktriangle$ markers represent the final corrected signal obtained by dividing the raw data by the refined reference.

and reference circuits alternatively as the following order:

$$[U(t_0), U_{\text{ref}}(t_0), U(t_1), U_{\text{ref}}(t_1), \dots,]$$



where $U(t_i)$ denotes the circuit implementing the time evolution at step t_i , and $U_{\text{ref}}(t_i)$ is its corresponding reference circuit. The time steps are defined as $t_i = i \, dt$, with $dt = 0.1$. This alternative structure ensures that each reference circuit runs under more similar conditions to its corresponding evolution circuit. In our implementation, we manually set the number of circuits per job to 200, which results in the simulations for $t \leq 10$ and $t > 10$ being split across two separate jobs. As seen in Fig. 6.5, a visible discontinuity appears around $t = 10$, which likely arises from different submission of the two jobs.

6.3 Riemannian Quantum Circuit Optimization

Current quantum hardware remains limited by significant noise and decoherence, particularly for deep circuits. Therefore, reducing circuit depth is essential for improving simulation fidelity. To this end, we explore a compression technique called Riemannian quantum circuit optimization [167]. This approach constructs a low-depth quantum circuit to approximate a target unitary by treating the target unitary (the real-time evolution operator e^{-iHdt}) as a matrix product operator (MPO) and optimizing the quantum gates (please refer to Fig. 6.6). The initial circuit is prepared as a second-order Trotterized evolution operator, and then TN techniques with Riemannian optimization are applied to search for the optimal gate sequence using a fixed number of layers. The optimization is performed on a classical computer, and once the optimal gates are obtained, the resulting circuit can

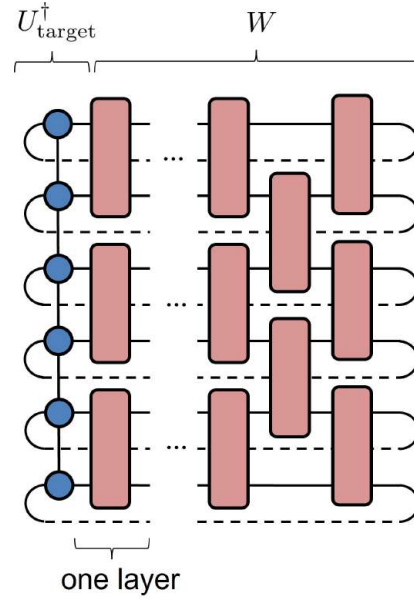


Figure 6.6: TN representation of the overlap between the target unitary operator and the quantum gates. The blue tensors MPO form of the real-time evolution operator, while the pink tensors correspond to the quantum circuit with a fixed number of layers, initialized using second-order Trotter decomposition. This overlap corresponds to the trace term in the cost function defined in Eq. 6.4. The goal of the Riemannian optimization is to optimize the quantum gates such that this overlap is maximized, i.e., the cost function is minimized.

be implemented on a quantum device. We can define the cost function

$$C_F(U_{\text{target}}, W) = 1 - \frac{1}{d} \text{Re} \left[\text{Tr} \left(U_{\text{target}}^\dagger W \right) \right], \quad (6.4)$$

which corresponds to the normalized Frobenius norm [167]. Here, U_{target} is the target unitary represented as a MPO, and W is the quantum circuit composed of a fixed sequence of gates. The factor $d = 2^N$, with N the number of qubits. The trace term can be computed using TN contraction, as illustrated in Fig. 6.6.

To evaluate the performance of the Riemannian optimization approach, we simulate real-time dynamics using classical simulator based on the optimized quantum gates prepared. The results are presented in Fig. 6.7. The quantum circuits are initialized using the

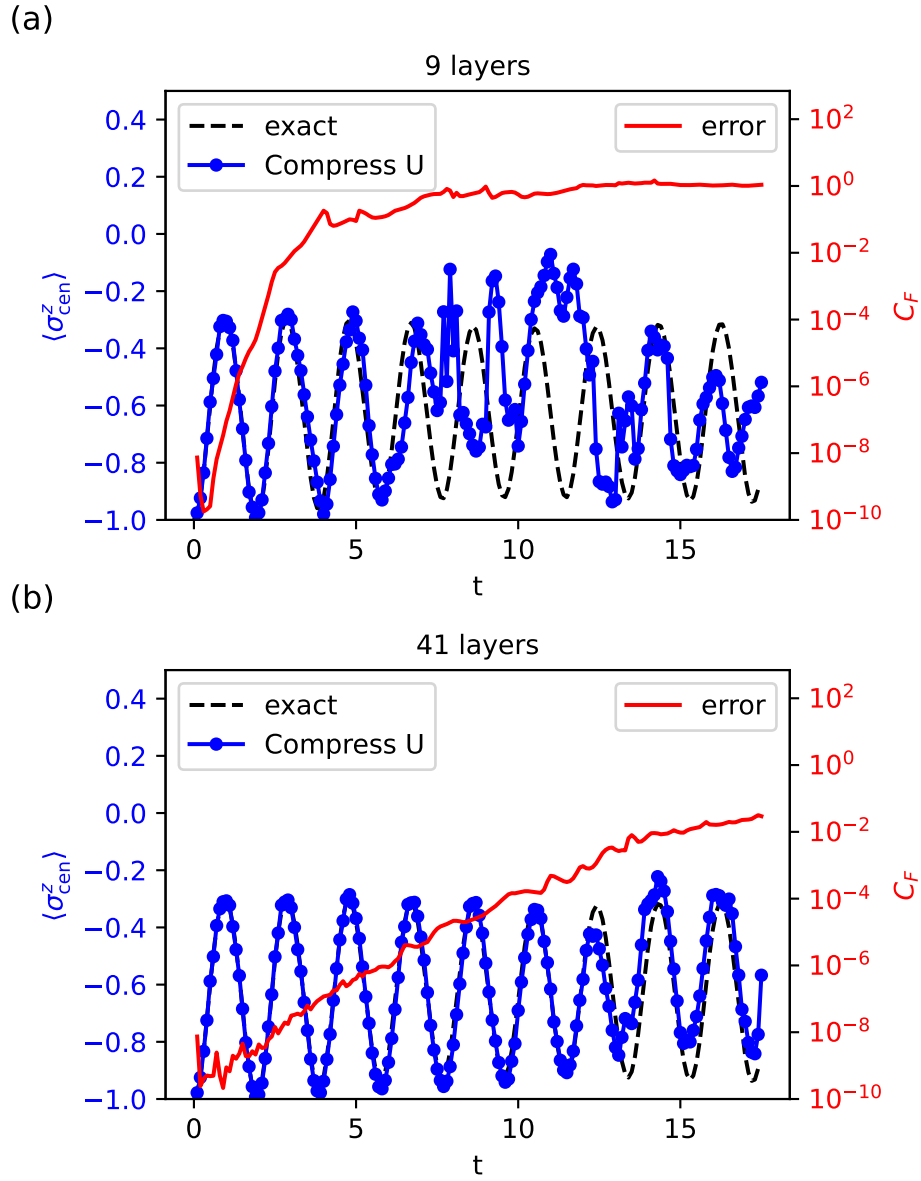


Figure 6.7: Simulation results using Riemannian optimization to construct time-evolution circuits with different numbers of layers, executed on a classical simulator. The initial state is $|\uparrow\uparrow\downarrow\downarrow\downarrow\uparrow\uparrow\rangle$, and $\langle\sigma_z^z(t)\rangle$ is measured at site 4. The model parameters are $J = h_x = 1$, $h_z = 3$, and each simulation uses 8192 shots. (a) Results using 9 circuit layers; (b) results using 41 layers. The black dashed line represents the exact evolution, the blue line with marker shows the simulator results with the optimized quantum gates, and the red solid line indicates the cost function defined in Eq. 6.4 In both cases, the fidelity of the simulation decreases over time, and longer-time evolution requires more circuit layers to maintain accuracy.

optimized gates, and the resulting dynamics of $\langle \sigma_{\text{cen}}^z(t) \rangle$ are compared with ED results. Additionally, we simulate the cost function defined in Eq. 6.4 as an indicator of circuit accuracy. As expected, the approximation error accumulates over time, and cause the deviation from the exact evolution. The rate of this error growth depends on the depth of the circuit: circuits with more layers exhibit slower error accumulation and remain accurate for longer times. This observation suggests a practical strategy for optimizing circuit resources: *adapt the number of layers based on the total desired evolution time*. That is, shallow circuits are adequate for short-time dynamics, while longer evolutions require deeper circuits to preserve fidelity. This layer-adaptive strategy helps reduce the number of quantum gates deployed on real hardware.

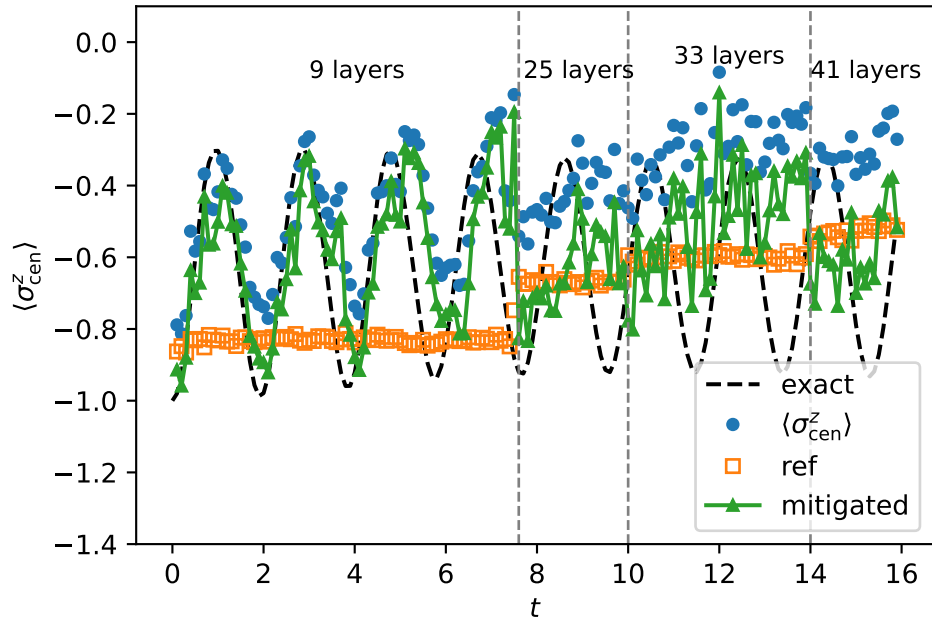
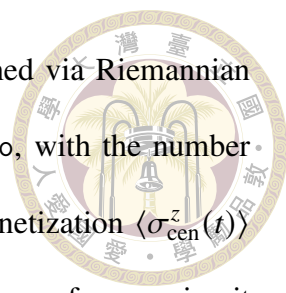


Figure 6.8: Real-device simulation results using the IBM Quantum device `ibm_torino`, where the time-evolution gates are prepared via Riemannian optimization. The model parameters and number of shots are the same as in Fig. 6.7. The black dashed line shows the exact evolution of $\langle \sigma^z(t) \rangle$ at site 4. • markers represent the raw measurement results from the real device, □ markers correspond to the reference circuit in which all native gate parameters are set to zero after transpilation except for the RZZ gates, which remain unchanged, and ▲ markers show the final error-mitigated result obtained by dividing the raw data by the reference. Vertical gray dashed lines indicate the time points where the number of circuit layers increases.



In Fig. 6.8, we present results where the quantum gates obtained via Riemannian optimization are executed on the IBM Quantum device `ibm_torino`, with the number of layers adjusted according to the evolution time. The central magnetization $\langle \sigma_{\text{cen}}^z(t) \rangle$ are obtained from the device. To mitigate hardware noise, we also run a reference circuit in which all parameters are set to zero except for the RZZ gates (To run on real quantum hardware, the unitary gates must be transpiled into native gates. In the reference circuit, all gate parameters are set to zero except for the RZZ gates, which remain unchanged.) The result of this reference measurement is also shown the figure. We observe that the signal from the reference circuit exhibits a step-like decay, with boundaries corresponding to points where the number of layers increases. Finally, we apply error mitigation by dividing the raw signal by the reference.

The simulation results on the real quantum device become very noisy when the number of layers exceeds 25 as shown in Fig. 6.8. Compared to our earlier simulations that the evolution operator can be directly constructed using native RZ, RX, and RZZ gates in Fig. 6.5, this approach introduces more noise. This is likely due to the need for transpilation on real hardware, where even a fixed-layer circuit must be decomposed into a deeper sequence of native gates. Nonetheless, this method remains valuable: for more complicated Hamiltonians, it may not be possible to construct the evolution operator directly from native gates, and compression via Riemannian optimization offers an alternative. Although the compressed gates do not match the fidelity of directly constructed circuits, we still observe a clear main spectral peak at the correct position after applying a Fourier transform, an aspect we examine further in the next section.

6.4 Masses in Frequency Domain



In this section, we analyze the dynamics in the frequency domain by examining the spectrum of $\langle \sigma_{\text{cen}}^z(\omega) \rangle$ by applying a Fourier transform. One challenge in this analysis is that, as seen in Fig. 6.1, only a few dominant peaks (typically one or two) appear in the frequency spectrum and the other are very weak. When simulating on real quantum devices, the signal becomes increasingly noisy during the real-time evolution and make it difficult to extract those weaker signal after Fourier transformation. To address this, we explore the strategy of starting from different initial states, each potentially exciting different main frequencies in the spectrum. Fig 6.9(a). shows the exact real-time evolution computed on a classical simulator for several different initial states. Each initial configuration leads to a distinct dominant frequency. Recognizing that long-time quantum simulations are unreliable due to accumulated noise, we restrict the Fourier transform to data up to $t = 10$. The resulting frequency-domain spectra are shown in Fig. 6.9(b). Even within this limited time window, distinct main frequency components emerge from different initial states. This approach makes it possible to access a broader range of the E_8 mass spectrum by combining the frequency responses from multiple chosen initial states.

We also perform real-time evolution on a real quantum device, starting from various initial states to explore their frequency responses as we done in simulator. As shown in Fig. 6.10, we consider two approaches we discussed in previous sections: (1) direct construction of quantum circuits using first-order Trotterization with native gates, and (2) Riemannian optimization to generate compressed evolution circuits with fixed depth, which are then transpiled into native gates for execution. For both methods, error mitigation is

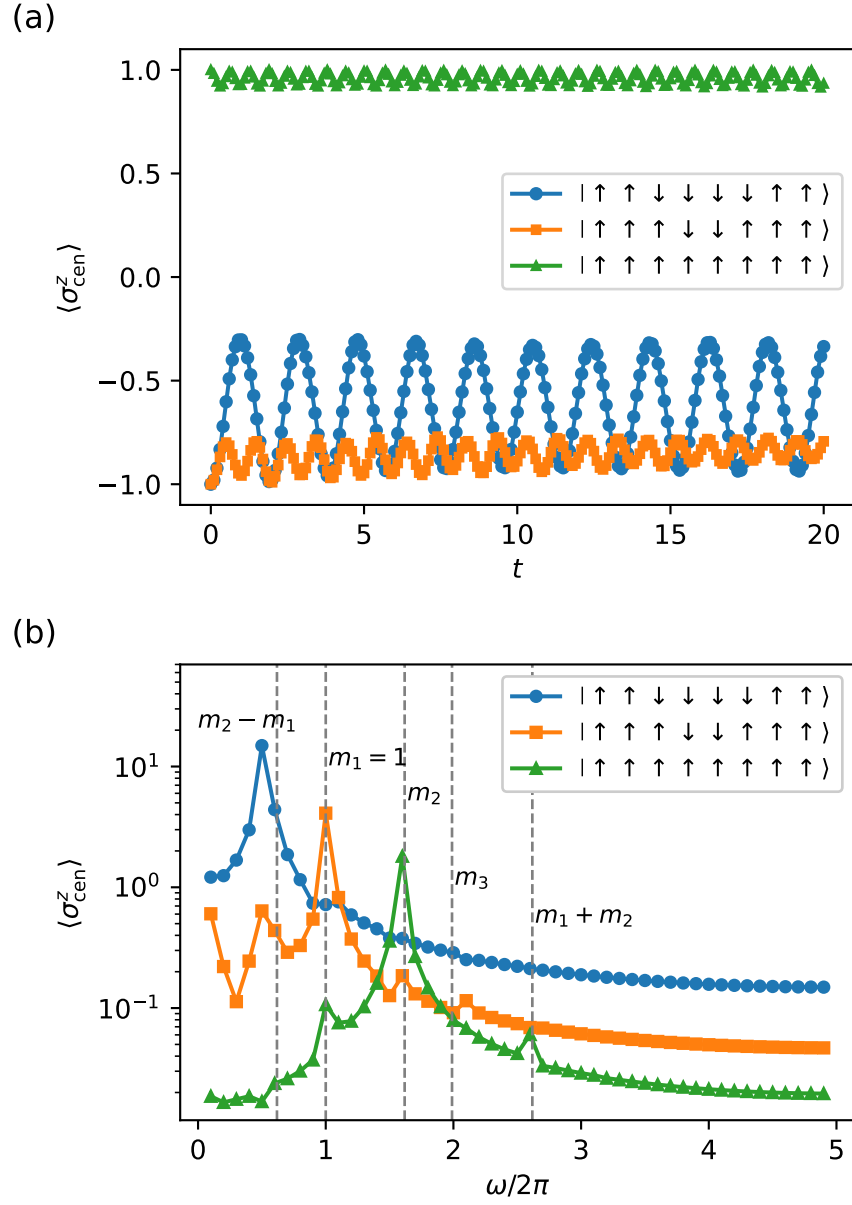
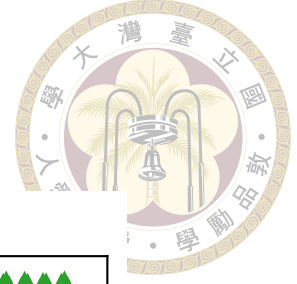


Figure 6.9: Exact real-time evolution starting from different initial states for the transverse-field Ising model with parameters $J = h_x = 1$, $h_z = 3$. (a) Time-domain results showing $\langle \sigma_{\text{cen}}^z(t) \rangle$. (b) corresponding frequency-domain spectra obtained by Fourier transforming the data in (a), restricted to $t \leq 10$. The gray dashed line indicate the expected position of peaks predict from Table 6.1. $m_1 = 1$ is chosen manually. Different initial states produce distinct dominant frequency peaks, all of which align with the predicted E_8 mass spectrum.

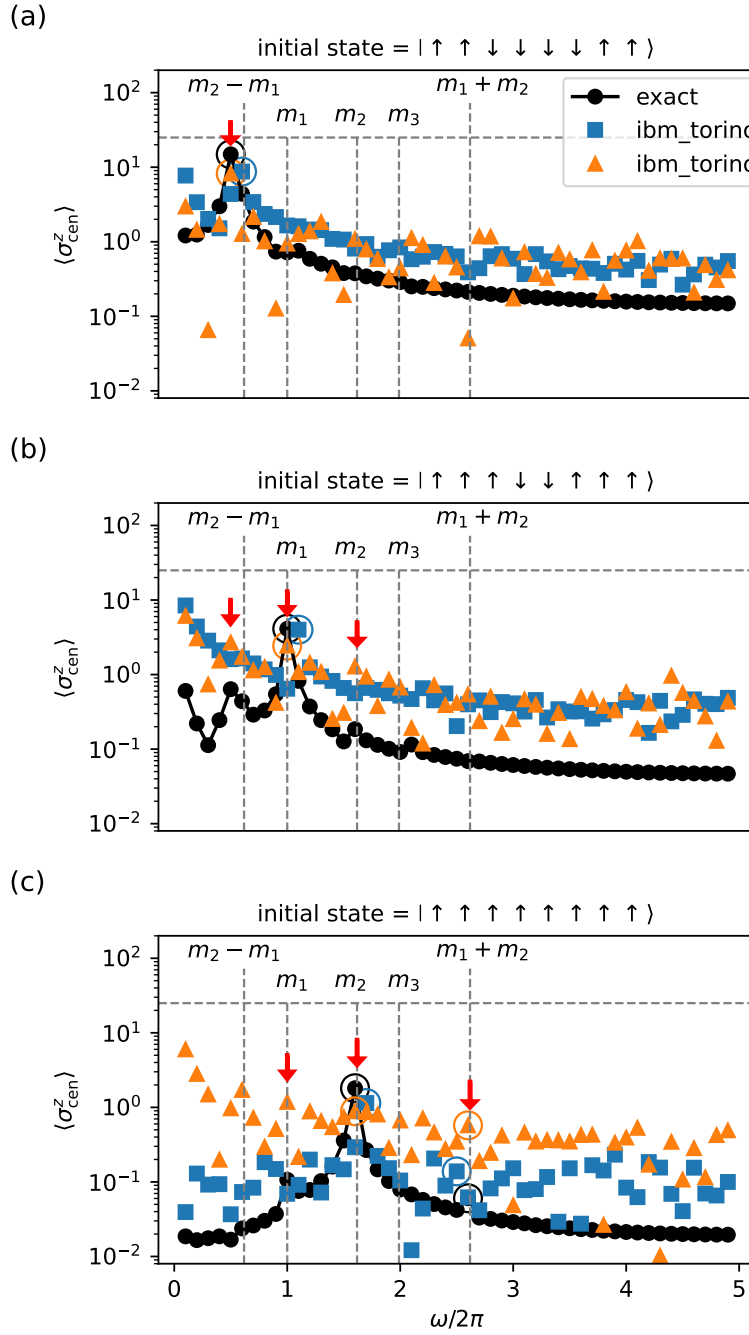


Figure 6.10: Frequency-domain simulation results on the IBM Quantum device `ibm_torino`, with model parameters and initial states matching those in Fig. 6.9, and using 8192 shots per circuit. $\bullet$ markers show the exact result from the classical computer, $\blacksquare$ markers correspond to results obtained by constructing the evolution gates using first-order Trotterization, and $\blacktriangle$ markers represent results using gates prepared via Riemannian optimization. Vertical gray dashed lines indicate the expected peak positions, as shown in Fig. 6.9, which are predicted by the E_8 spectrum. Both quantum-device methods include error mitigation before applying the Fourier transform to extract the spectra. All subfigures use data up to $t = 10$.

Label	E_8 prediction	ED		Trotter		Riemannian	
		Value	Deviation	Value	Deviation	Value	Deviation
$m_2 - m_1$	$0.618 m_1$	$0.5 m_1$	$-0.118 m_1$	$0.6 m_1$	$-0.018 m_1$	$0.5 m_1$	$-0.118 m_1$
m_1	m_1	$1.0 m_1$	0	$1.1 m_1$	$+0.10 m_1$	$1.0 m_1$	0
m_2	$1.618 m_1$	$1.6 m_1$	$-0.018 m_1$	$1.7 m_1$	$+0.082 m_1$	$1.6 m_1$	$-0.018 m_1$
$m_1 + m_2$	$2.618 m_1$	$2.6 m_1$	$-0.018 m_1$	$2.5 m_1$	$-0.118 m_1$	$2.6 m_1$	$-0.018 m_1$

Table 6.1: The first two columns list the first few mass spectrum predicted by the E_8 field theory. Each mass values of m_i is expressed in terms of the lightest mass m_1 . The next column pair reports the masses obtained from ED, while the following two column pairs present the masses extracted from the IBM Quantum device `ibm_torino` using first-order Trotter decomposition and Riemannian-optimized circuits, respectively. Deviations are computed as the difference between the measured value and the E_8 prediction. The corresponding peak positions in the frequency spectra are those highlighted with circled markers in Fig. 6.10. See text for more discussion.

applied by setting all native gate parameters to zero except for the RZZ gates. After collecting measurement data from the real device and do the error mitigation, we apply a Fourier transform to obtain the frequency-domain signal. We find that each initial state yields a clear dominant peak, and in some cases, even a secondary peak is visible. From these results, we can observe at least the peaks corresponding to $m_2 - m_1$, m_1 , and m_2 across different initial states. These findings demonstrate that parts of the E_8 mass spectrum can be extracted from quantum real-device data using both circuit construction strategies.

6.5 Detail of Setting on Real Device

In this section, we summarize the technical details of how the quantum simulations were executed on IBM Quantum hardware, specifically on the `ibm_torino` backend. This device is part of IBM's Heron-class QPUs, which support fractional gates.

One critical setting in our experiments is enabling native fractional gates during backend selection. This allows us to access the RZZ gate directly as a native hardware-supported

operation, which effectively reduces the circuit depth by avoiding additional gate decomposition. In Qiskit, this is done by initializing the backend with the `use_fractional_gates=True` option. The use of fractional gates is currently supported only on Heron processors, such as `ibm_torino`, and is essential for expressing two-qubit gate like RZZ natively. Additionally, when transpiling our circuits to match the native gate set of the backend, we set the transpiler `optimization_level=0` using Qiskit's `generate_preset_pass_manager`. We find that if we set `optimization_level=2` leads to nonsense result, the measured observable becomes nearly constant during the entire evolution.

In our simulations, we manually select the specifically qubits to ensure better connectivity and reduced noise. The layout of the quantum processor used in `ibm_torino`, including the chosen qubits highlighted in red, is shown in Fig. 6.11.

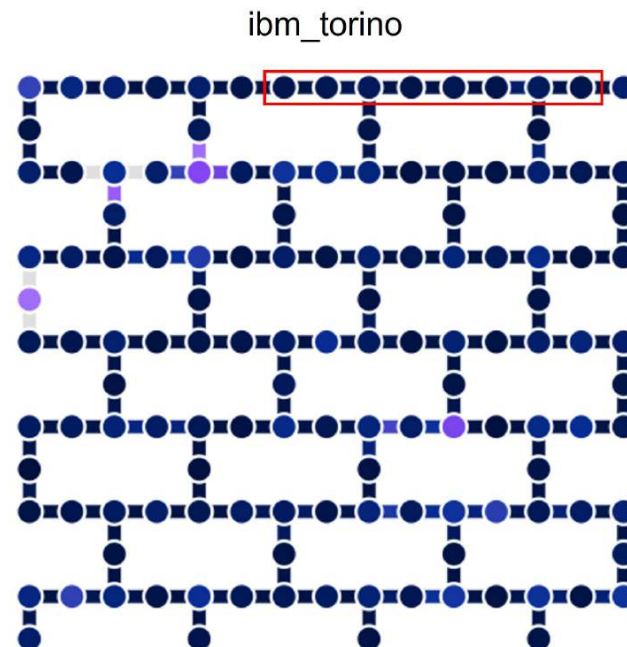


Figure 6.11: Qubit layout of the IBM Quantum device `ibm_torino`, a Heron-class processor used in our experiments. The red rectangle contains the QPUs selected for our simulations.

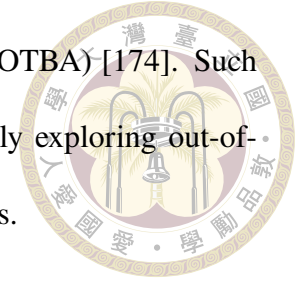
6.6 Conclusion



We demonstrate the state-of-the-art simulations of E_8 mass spectra using the IBM Quantum device `ibm_torino`, which supports native RX , RZ and RZZ gates. By benchmarking two approaches of the real-time evolution, we find direct construction of the Trotterized evolution unitary using native gates and the Riemannian compression of the evolution operator into a fixed-layer quantum circuit gives similar performance. However, we expect the Riemannian optimization should perform better when the time-evolution operators can not be expressed in native gates. Although simulating real-time evolution on a gate-based digital quantum computer suffers from the Trotterization error, our results are comparable to those obtained through the analog Rydberg atom system [166]. This demonstrates that, despite the current noise limitations in quantum hardware, essential physical features, such as low-lying mass peaks in the E_8 spectrum, can be successfully observed and resolved. These results show that current quantum devices, when combined with classical techniques like circuit compression and error mitigation, can be used to explore nontrivial quantum field theory phenomena, and with limited circuit resources, physically meaningful information such as mass spectra can still be extracted.

Looking forward, an exciting direction is to extend these techniques to the quantum quench dynamics of the XXZ spin chain. Although preparing its arbitrary ground state is challenging on current hardware [173], simple product states such as the Néel state or dimer product state can be readily initialized, enabling quench protocols similar to those demonstrated in [164]. With optimized low-depth circuits and improved error mitigation, real devices may access sufficiently long evolution times to allow comparison with the-

oretical predictions from the overlap-thermodynamic Bethe ansatz (OTBA) [174]. Such an approach would provide a promising route toward experimentally exploring out-of-equilibrium dynamics in integrable models using quantum processors.



A further potential direction is the massive Thirring model, whose lattice discretization can be mapped to the XXZ spin chain coupled to both uniform and staggered magnetic fields via the Jordan-Wigner transformation [42]. Recent numerical studies have explored its nonequilibrium dynamics under quantum quenches [175]. Implementing such quenches on gate-based quantum hardware would open a complementary route to studying real-time dynamics in quantum field theories. Extending these methods to both spin-chain and field-theory models would expand the out-of-equilibrium physics accessible on near-term devices.





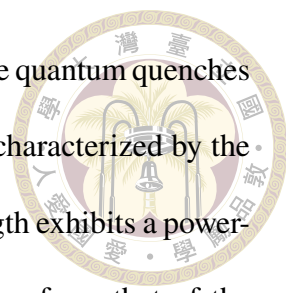
Chapter 7 Summary and Outlook

In this dissertation, we explored a range of topics in quantum simulation and numerical computation, all unified by the use of tensor network (TN) algorithms—notably VUMPS, DMRG, and TDVP—for solving time-dependent and static problems. Although each chapter addresses a different physical or computational system, they share strong methodological connections and are linked by the common theme of time evolution in either real or imaginary time.

Chapters 2–4: Tensor Network Simulations

In Chapters 2 to 4, we applied classical tensor network methods to a variety of problems.

In **Chapter 2**, we used VUMPS and infinite-size TDVP to simulate dynamical quantum phase transitions (DQPTs) in both the Thirring model and the XXZ spin chain. Our results show that DQPTs consistently occur when the post-quench energy density exceeds a certain threshold. In the XXZ model, we further performed complex-time evolution and demonstrated that DQPTs are closely related to the locations of Fisher zeros in the complex time plane.



In **Chapter 3**, we applied DMRG and finite-size TDVP to simulate quantum quenches in the XXZ chain. We observed signatures of coarsening dynamics, characterized by the formation and growth of domain walls. The characteristic domain length exhibits a power-law-like growth in time, although the precise scaling behavior differs from that of the classical two-dimensional Ising model and remains an open question.

In **Chapter 4**, we extended tensor network techniques to the numerical solution of partial differential equations. By discretizing functions in Cartesian or polar coordinates and representing them in the QTT format, we solved both static and time-dependent PDEs using DMRG and TDVP. In particular, we introduced a regularization scheme to treat the singularity of the Laplacian at the origin in polar coordinates, which integrates naturally into the MPO framework.

Chapters 5–6: Tensor Networks and Quantum Circuits

In the final two chapters, we combined tensor network ideas with quantum circuits, enabling simulations on real quantum hardware.

In **Chapter 5**, we mapped uMPS and MPO structures to parameterized quantum circuits and developed TDVP-inspired algorithms for imaginary-time evolution of the Ising model. Cost functions were evaluated on IBM quantum devices, and quantum Lanczos (QLanczos) post-processing was employed to improve convergence. The optimized energies showed good agreement with ground-state values, highlighting the potential of variational quantum algorithms informed by tensor network structure.

In **Chapter 6**, we performed real-time quantum simulations of the transverse-field

Ising model with a longitudinal field on IBM quantum hardware to investigate the emergence of E_8 symmetry. By combining real-device simulations with tensor-network-based circuit compression and error mitigation techniques, we were able to reduce circuit depth and improve fidelity, ultimately observing characteristic E_8 spectral peaks on NISQ devices.

Outlook

The scope and interconnections of the completed work, as well as several future research directions, are summarized schematically in Fig. 7.1. The gray arrows in the figure represent the research directions that have been realized in this dissertation, starting from TDVP within tensor networks and extending to both classical simulations and quantum-circuit-based implementations.

The black arrows indicate potential extensions beyond the present work. One particularly challenging direction, shown as a dashed arrow in Fig. 7.1, is the simulation of infinite-system real-time dynamics and DQPTs using quantum circuits. While we attempted to pursue this direction, we found that real-time evolution on quantum hardware is extremely sensitive to statistical noise. In particular, the cost functions required for TDVP-based updates are estimated from measurement outcomes and therefore correspond to probability distributions with intrinsic statistical uncertainty. This uncertainty propagates through the iterative time-evolution procedure, leading to unstable dynamics and preventing reliable simulations. In contrast, imaginary-time evolution targets a fixed point—the ground state—so that small errors at intermediate steps can still be corrected in subsequent iterations. This fundamental difference explains why imaginary-time algorithms

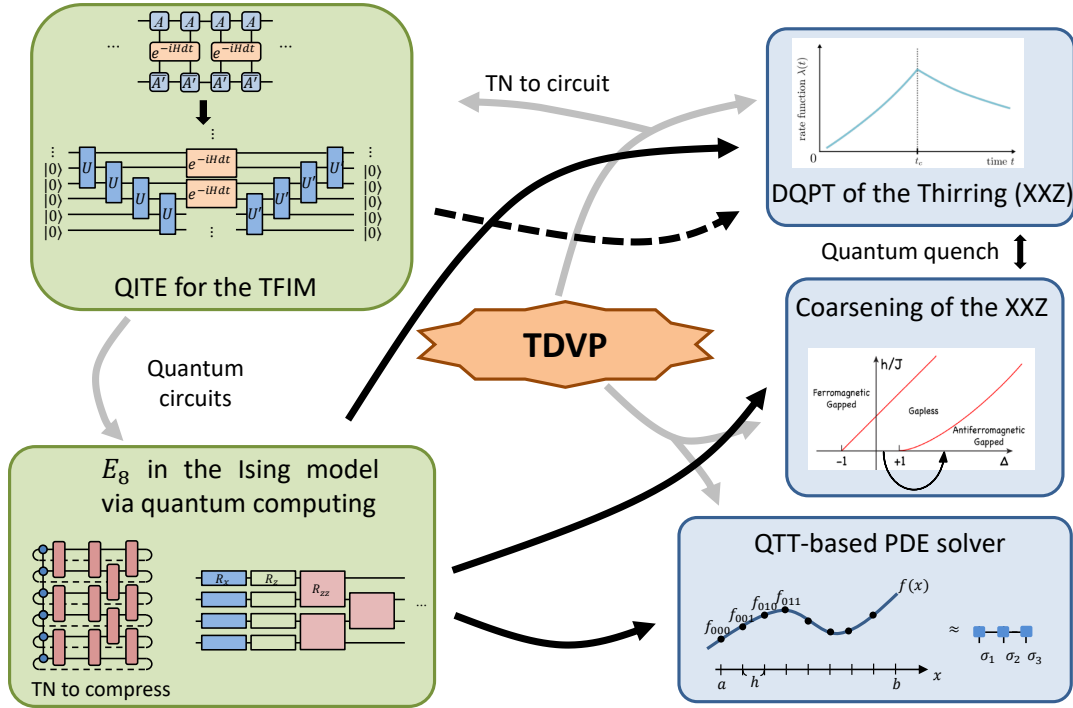


Figure 7.1: Overview of completed work and potential future directions discussed in this dissertation. Gray arrows indicate research directions realized in this thesis, starting from TDVP within tensor networks and extending to classical simulations and quantum-circuit-based implementations. Black arrows represent potential extensions, including the use of quantum circuits to simulate real-time dynamics, coarsening phenomena, and partial differential equations. The dashed arrow highlights the challenge of simulating infinite-system real-time dynamics and DQPTs on quantum hardware, where measurement-induced statistical uncertainty strongly affects iterative time evolution.

are currently more robust on NISQ devices.

Beyond this challenge, Fig. 7.1 also highlights several promising future directions. One possibility is the use of quantum circuits to solve partial differential equations, extending the QTT-based tensor network methods developed here to quantum hardware; related ideas have already been explored in Ref. [123]. Another direction is the investigation of coarsening dynamics using quantum circuits, as well as the study of the quantum quench dynamics in finite-size systems on quantum devices, which has begun to be explored in Ref. [176]. Together, these directions suggest that the combination of tensor networks, variational principles, and quantum circuits provides a flexible and powerful framework for future studies of nonequilibrium quantum dynamics.

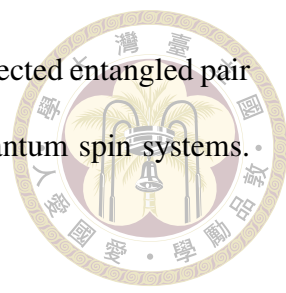


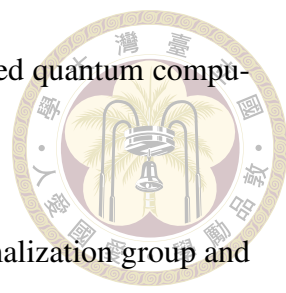




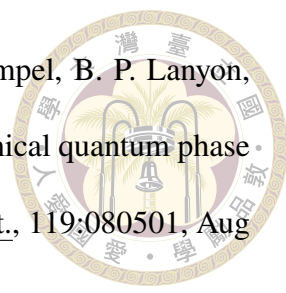
References

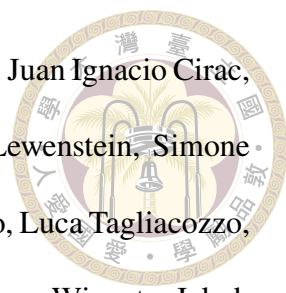
- [1] Jutho Haegeman, J. Ignacio Cirac, Tobias J. Osborne, Iztok Pižorn, Henri Verschelde, and Frank Verstraete. Time-dependent variational principle for quantum lattices. Phys. Rev. Lett., 107:070601, Aug 2011.
- [2] Aditi Mitra. Quantum quench dynamics. Annu. Rev. Condens. Matter Phys., 9:245, Mar 2018.
- [3] M. Heyl, A. Polkovnikov, and S. Kehrein. Dynamical quantum phase transitions in the transverse-field ising model. Phys. Rev. Lett., 110:135704, Mar 2013.
- [4] Markus Heyl. Dynamical quantum phase transitions: a review. Reports on Progress in Physics, 81(5):054001, apr 2018.
- [5] G Torlai, L Tagliacozzo, and G De Chiara. Dynamics of the entanglement spectrum in spin chains. Journal of Statistical Mechanics: Theory and Experiment, 2014(6):P06001, jun 2014.
- [6] Philipp Uhrich, Nicolò Defenu, Rouhollah Jafari, and Jad C. Halimeh. Out-of-equilibrium phase diagram of long-range superconductors. Phys. Rev. B, 101:245148, Jun 2020.

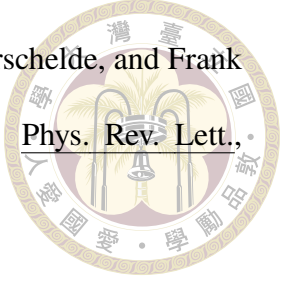
- 
- [7] V. Murg F. Verstraete and J.I. Cirac. Matrix product states, projected entangled pair states, and variational renormalization group methods for quantum spin systems. Advances in Physics, 57(2):143–224, 2008.
- [8] Ulrich Schollwöck. The density-matrix renormalization group in the age of matrix product states. Ann. Phys., 326(1):96 – 192, 2011.
- [9] Román Orús. A practical introduction to tensor networks: Matrix product states and projected entangled pair states. Annals Phys., 349:117–158, 2014.
- [10] Pietro Silvi, Ferdinand Tschirsich, Matthias Gerster, Johannes Jünemann, Daniel Jaschke, Matteo Rizzi, and Simone Montangero. The Tensor Networks Anthology: Simulation techniques for many-body quantum lattice systems. SciPost Phys. Lect. Notes, page 8, 2019.
- [11] Kouichi Okunishi, Tomotoshi Nishino, and Hiroshi Ueda. Developments in the Tensor Network — from Statistical Mechanics to Quantum Entanglement. J. Phys. Soc. Jpn, 91(6):062001, 2022.
- [12] Mari Carmen Bañuls. Tensor Network Algorithms: A Route Map. Annu. Rev. Condens. Matter Phys, 14(1), 2023.
- [13] M. Fannes, B. Nachtergaele, and R. F. Werner. Finitely correlated states on quantum spin chains. Commun. Math. Phys., 144:443–490, 1992.
- [14] Stellan Östlund and Stefan Rommer. Thermodynamic limit of density matrix renormalization. Phys. Rev. Lett., 75:3537–3540, Nov 1995.

- 
- [15] Guifré Vidal. Efficient classical simulation of slightly entangled quantum computations. Phys. Rev. Lett., 91:147902, Oct 2003.
- [16] F. Verstraete, D. Porras, and J. I. Cirac. Density matrix renormalization group and periodic boundary conditions: A quantum information perspective. Phys. Rev. Lett., 93:227205, Nov 2004.
- [17] David Perez-Garcia, Frank Verstraete, Michael M. Wolf, and J. Ignacio Cirac. Matrix product state representations. Quantum Inf. Comput., 7:401–430, 2007.
- [18] F. Andraschko and J. Sirker. Dynamical quantum phase transitions and the loschmidt echo: A transfer matrix approach. Phys. Rev. B, 89:125120, Mar 2014.
- [19] Pasquale Calabrese and John L. Cardy. Entanglement entropy and quantum field theory. J. Stat. Mech., 0406:P06002, 2004.
- [20] Tobias J. Osborne. Efficient approximation of the dynamics of one-dimensional quantum spin systems. Phys. Rev. Lett., 97:157202, Oct 2006.
- [21] N Schuch, M M Wolf, K G H Vollbrecht, and J I Cirac. On entropy growth and the hardness of simulating time evolution. New J. Phys., 10(3):033032, 2008.
- [22] Jutho Haegeman, Christian Lubich, Ivan Oseledets, Bart Vandereycken, and Frank Verstraete. Unifying time evolution and optimization with matrix product states. Phys. Rev. B, 94:165116, Oct 2016.
- [23] Laurens Vanderstraeten, Jutho Haegeman, Frank Verstraete, and Didier Poilblanc. Quasiparticle interactions in frustrated Heisenberg chains. Phys. Rev. B, 93:235108, Jun 2016.

- 
- [24] Sebastian Paeckel, Thomas Köhler, Andreas Swoboda, Salvatore R. Manmana, Ulrich Schollwöck, and Claudius Hubig. Time-evolution methods for matrix-product states. Annals Phys., 411:167998, 2019.
- [25] C. Karrasch and D. Schuricht. Dynamical quantum phase transitions in the quantum potts chain. Phys. Rev. B, 95:075143, Feb 2017.
- [26] Valentin Zauner-Stauber and Jad C. Halimeh. Probing the anomalous dynamical phase in long-range quantum spin chains through fisher-zero lines. Phys. Rev. E, 96:062118, Dec 2017.
- [27] Stefano De Nicola, Alexios A. Michailidis, and Maksym Serbyn. Entanglement view of dynamical quantum phase transitions. Phys. Rev. Lett., 126:040602, Jan 2021.
- [28] Jad C. Halimeh, Daniele Trapin, Maarten Van Damme, and Markus Heyl. Local measures of dynamical quantum phase transitions. Phys. Rev. B, 104:075130, Aug 2021.
- [29] Jad C. Halimeh, Fabian Kolley, and Ian P. McCulloch. Chebyshev matrix product state approach for time evolution. Phys. Rev. B, 92:115130, Sep 2015.
- [30] Maarten Van Damme, Jean-Yves Desaulles, Zlatko Papić, and Jad C. Halimeh. Anatomy of dynamical quantum phase transitions. Phys. Rev. Res., 5:033090, Aug 2023.
- [31] Jesse J. Osborne, Johannes Knaute, Ian P. McCulloch, and Jad C. Halimeh. Meson mass sets onset time of anomalous dynamical quantum phase transitions, 2024.

- 
- [32] P. Jurcevic, H. Shen, P. Hauke, C. Maier, T. Brydges, C. Hempel, B. P. Lanyon, M. Heyl, R. Blatt, and C. F. Roos. Direct observation of dynamical quantum phase transitions in an interacting many-body system. Phys. Rev. Lett., 119:080501, Aug 2017.
- [33] Xue-Yi Guo, Chao Yang, Yu Zeng, Yi Peng, He-Kang Li, Hui Deng, Yi-Rong Jin, Shu Chen, Dongning Zheng, and Heng Fan. Observation of a dynamical quantum phase transition by a superconducting qubit simulation. Phys. Rev. Appl., 11:044080, Apr 2019.
- [34] Szabolcs Vajna and Balázs Dóra. Disentangling dynamical phase transitions from equilibrium phase transitions. Phys. Rev. B, 89:161105, Apr 2014.
- [35] Jad C. Halimeh and Valentin Zauner-Stauber. Dynamical phase diagram of quantum spin chains with long-range interactions. Phys. Rev. B, 96:134427, Oct 2017.
- [36] Ingo Homrighausen, Nils O. Abeling, Valentin Zauner-Stauber, and Jad C. Halimeh. Anomalous dynamical phase in quantum spin chains with long-range interactions. Phys. Rev. B, 96:104436, Sep 2017.
- [37] Ángel L. Corps and Armando Relaño. Dynamical and excited-state quantum phase transitions in collective systems. Phys. Rev. B, 106:024311, Jul 2022.
- [38] David Pérez-García, Leonardo Santilli, and Miguel Tierz. Dynamical quantum phase transitions from random matrix theory. Quantum, 8:1271, February 2024.
- [39] Mari Carmen Bañuls and Krzysztof Cichy. Review on novel methods for lattice gauge theories. Reports on Progress in Physics, 83(2):024401, jan 2020.

- 
- [40] Mari Carmen Bañuls, Rainer Blatt, Jacopo Catani, Alessio Celi, Juan Ignacio Cirac, Marcello Dalmonte, Leonardo Fallani, Karl Jansen, Maciej Lewenstein, Simone Montangero, Christine A. Muschik, Benni Reznik, Enrique Rico, Luca Tagliacozzo, Karel Van Acoleyen, Frank Verstraete, Uwe-Jens Wiese, Matthew Wingate, Jakub Zakrzewski, and Peter Zoller. Simulating lattice gauge theories within quantum technologies. Eur. Phys. J. D, 74(8):165, 2020.
- [41] Steven D. Bass and Erez Zohar. Quantum technologies in particle physics. Phil. Trans. R. Soc. A, 380:20210072, 2022.
- [42] Mari Carmen Bañuls, Krzysztof Cichy, Ying-Jer Kao, C. J. David Lin, Yu-Ping Lin, and David T. L. Tan. Phase structure of the $(1+1)$ -dimensional massive Thirring model from matrix product states. Phys. Rev. D, 100(9):094504, 2019.
- [43] Markus Hauru, Glen Evenbly, Wen Wei Ho, Davide Gaiotto, and Guifre Vidal. Topological conformal defects with tensor networks. Phys. Rev. B, 94:115125, Sep 2016.
- [44] Q. Hu and G. Vidal. Spacetime symmetries and conformal data in the continuous multiscale entanglement renormalization ansatz. Phys. Rev. Lett., 119:010603, Jul 2017.
- [45] Robijn Vanhove, Matthias Bal, Dominic J. Williamson, Nick Bultinck, Jutho Haegeman, and Frank Verstraete. Mapping topological to conformal field theories through strange correlators. Phys. Rev. Lett., 121:177203, Oct 2018.

- 
- [46] Boye Buyens, Jutho Haegeman, Karel Van Acoleyen, Henri Verschelde, and Frank Verstraete. Matrix product states for gauge field theories. Phys. Rev. Lett., 113:091601, 2014.
- [47] Stefan Kühn, Erez Zohar, J. Ignacio Cirac, and Mari Carmen Bañuls. Non-Abelian string breaking phenomena with matrix product states. Journal of High Energy Physics, 2015(7):130, 2015.
- [48] Boye Buyens, Jutho Haegeman, Henri Verschelde, Frank Verstraete, and Karel Van Acoleyen. Confinement and string breaking for QED₂ in the Hamiltonian picture. Phys. Rev., X6(4):041040, 2016.
- [49] T. Pichler, M. Dalmonte, E. Rico, P. Zoller, and S. Montangero. Real-time dynamics in U(1) lattice gauge theories with tensor networks. Phys. Rev., X6(1):011023, 2016.
- [50] Philipp Schmoll, Jan Naumann, Alexander Nietner, Jens Eisert, and Spyros Sotiriadis. Hamiltonian truncation tensor networks for quantum field theories, 2023.
- [51] Mari Carmen Bañuls, Krzysztof Cichy, J. Ignacio Cirac, Karl Jansen, and Hana Saito. Matrix product states for lattice field theories. PoS(LATTICE 2013), 332, 2014.
- [52] Sebastiano Peotta, Fredrik Brange, Aydin Deger, Teemu Ojanen, and Christian Flindt. Determination of dynamical quantum phase transitions in strongly correlated many-body systems using loschmidt cumulants. Phys. Rev. X, 11:041018, Oct 2021.

[53] Ian McCulloch. Detecting and characterising dynamical quantum phase transitions through higher moments. In APS March Meeting Abstracts, volume 2023, page B33.001, New York, January 2023. APS.



[54] Ashok K. Das and Vishnu S. Mathur. Path Integral Solubility of Two-Dimensional Models. Phys. Rev., D33:489, 1986.

[55] Julian S. Schwinger. Gauge Invariance and Mass. 2. Phys. Rev., 128:2425–2429, 1962.

[56] CR Hagen. New solutions of the Thirring model. Il Nuovo Cimento, B51(1):169–186, 1967.

[57] A. Luther. Eigenvalue spectrum of interacting massive fermions in one-dimension. Phys. Rev., B14:2153–2159, 1976.

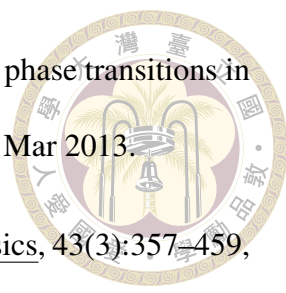
[58] Ho N. Phien, Guifré Vidal, and Ian P. McCulloch. Infinite boundary conditions for matrix product state calculations. Phys. Rev. B, 86:245107, Dec 2012.

[59] V. Zauner-Stauber, L. Vanderstraeten, M. T. Fishman, F. Verstraete, and J. Haegeman. Variational optimization algorithms for uniform matrix product states. Phys. Rev. B, 97:045145, Jan 2018.

[60] M. C. Bañuls, K. Cichy, J. I. Cirac, and K. Jansen. The mass spectrum of the Schwinger model with matrix product states. J. High Energy Phys., 2013(11):158, Nov 2013.

- 
- [61] Jutho Haegeman, J. Ignacio Cirac, Tobias J. Osborne, Iztok Pižorn, Henri Verschelde, and Frank Verstraete. Time-dependent variational principle for quantum lattices. Phys. Rev. Lett., 107:070601, Aug 2011.
- [62] D J Amit, Y Y Goldschmidt, and S Grinstein. Renormalisation group analysis of the phase transition in the 2d Coulomb gas, sine-Gordon theory and XY-model. Journal of Physics A: Mathematical and General, 13(2):585, feb 1980.
- [63] David B. Kaplan, Jong-Wan Lee, Dam T. Son, and Mikhail A. Stephanov. Conformality lost. Phys. Rev. D, 80:125005, Dec 2009.
- [64] J. Zinn-Justin. Quantum field theory at finite temperature: An introduction, 2000.
- [65] J. M. Kosterlitz and D. J. Thouless. Ordering, metastability and phase transitions in two-dimensional systems. J. Phys., C6:1181–1203, 1973.
- [66] J M Kosterlitz. The critical properties of the two-dimensional xy model. Journal of Physics C: Solid State Physics, 7(6):1046, mar 1974.
- [67] A. LeClair, G. Mussardo, H. Saleur, and S. Skorik. Boundary energy and boundary states in integrable quantum field theories. Nuclear Physics B, 453(3):581–618, 1995.
- [68] W van Saarloos and D A Kurtze. Location of zeros in the complex temperature plane: Absence of Lee-Yang theorem. Journal of Physics A: Mathematical and General, 17(6):1301, apr 1984.
- [69] C. Karrasch and D. Schuricht. Dynamical phase transitions after quenches in non-integrable models. Phys. Rev. B, 87:195104, May 2013.

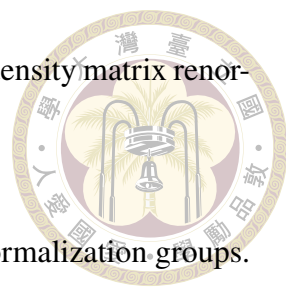
- 
- [70] Souvik Bandyopadhyay, Anatoli Polkovnikov, and Amit Dutta. Observing dynamical quantum phase transitions through quasilocal string operators. Phys. Rev. Lett., 126:200602, May 2021.
- [71] Souvik Bandyopadhyay, Anatoli Polkovnikov, and Amit Dutta. Late-time critical behavior of local stringlike observables under quantum quenches. Phys. Rev. B, 107:064105, Feb 2023.
- [72] Ruchira V. Bhat and Soumya Bera. Distinguishing dynamical quantum criticality through local fidelity distances. Phys. Rev. B, 109:214313, Jun 2024.
- [73] Pengju Zhao, Jingxin Sun, Shengjie Jin, Zhongshu Hu, Dingping Li, Xiong-Jun Liu, Jörg Schmiedmayer, and Xuzong Chen. Determination of dynamical quantum phase transitions for boson systems using the Loschmidt cumulants method. Phys. Rev. A, 109:013309, Jan 2024.
- [74] Ryusuke Hamazaki. Exceptional dynamical quantum phase transitions in periodically driven systems. Nature Communications, 12(1):5108, Sep 2021.
- [75] Román Orús. Advances on tensor network theory: symmetries, fermions, entanglement, and holography. Eur. Phys. J. B, 87(11):280, Nov 2014.
- [76] M. B. Hastings. Solving gapped Hamiltonians locally. Phys. Rev. B, 73:085115, Feb 2006.
- [77] Andras Molnar, Norbert Schuch, Frank Verstraete, and J. Ignacio Cirac. Approximating Gibbs states of local Hamiltonians efficiently with projected entangled pair states. Phys. Rev. B, 91:045138, Jan 2015.

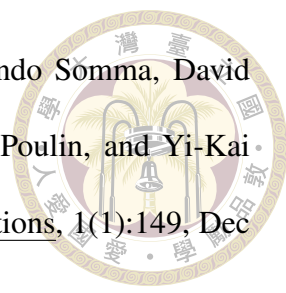
- 
- [78] M. Heyl, A. Polkovnikov, and S. Kehrein. Dynamical quantum phase transitions in the transverse-field ising model. Phys. Rev. Lett., 110:135704, Mar 2013.
- [79] A.J. Bray. Theory of phase-ordering kinetics. Advances in Physics, 43(3):357–459, 1994.
- [80] Jiahao Yang, Xiao Wang, and Jianda Wu. Magnetic excitations in the one-dimensional heisenberg – ising model with external fields and their experimental realizations. Journal of Physics A: Mathematical and Theoretical, 56(1):013001, jan 2023.
- [81] Qian-Can Chen, I-Kang Liu, Jheng-Wei Li, and Chia-Min Chung. Solving the gross-pitaevskii equation with quantic tensor trains: Ground states and nonlinear dynamics, 2025.
- [82] Marcel Niedermeier, Adrien Moulinas, Thibaud Louvet, Jose L. Lado, and Xavier Waintal. Solving the gross-pitaevskii equation on multiple different scales using the quantics tensor train representation, 2025.
- [83] Aleix Bou-Comas, Marcin Płodzień, Luca Tagliacozzo, and Juan José García-Ripoll. Quantics tensor train for solving gross-pitaevskii equation, 2025.
- [84] Yuriel Núñez Fernández, Marc K. Ritter, Matthieu Jeannin, Jheng-Wei Li, Thomas Kloss, Thibaud Louvet, Satoshi Terasaki, Olivier Parcollet, Jan von Delft, Hiroshi Shinaoka, and Xavier Waintal. Learning tensor networks with tensor cross interpolation: New algorithms and libraries. SciPost Phys., 18:104, 2025.
- [85] Hiroshi Shinaoka, Markus Wallerberger, Yuta Murakami, Kosuke Nogaki, Rihito Sakurai, Philipp Werner, and Anna Kauch. Multiscale space-time ansatz for corre-

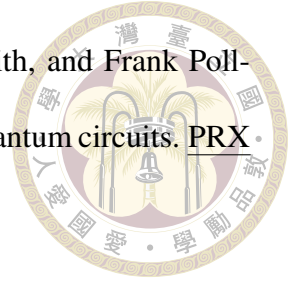
lation functions of quantum systems based on quantics tensor trains. *Phys. Rev. X*, 13:021015, Apr 2023.



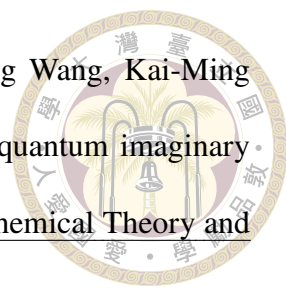
- [86] Vladimir A. Kazeev and Boris N. Khoromskij. Low-rank explicit qtt representation of the laplace operator and its inverse. *SIAM Journal on Matrix Analysis and Applications*, 33(3):742–758, 2012.
- [87] Luigi Galati and Shijun Zheng. Nonlinear schrödinger equations for bose-einstein condensates. *AIP Conference Proceedings*, 1562(1):50–64, 10 2013.
- [88] Nis-Luca van Hülst, Pia Siegl, Paul Over, Sergio Bengoechea, Tomohiro Hashizume, Mario Guillaume Cecile, Thomas Rung, and Dieter Jaksch. Quantum-inspired tensor-network fractional-step method for incompressible flow in curvilinear coordinates, 2025.
- [89] Jutho Haegeman, Christian Lubich, Ivan Oseledets, Bart Vandereycken, and Frank Verstraete. Unifying time evolution and optimization with matrix product states. *arXiv:1408.5056*, 2014.
- [90] E. Karimi, R. W. Boyd, P. de la Hoz, H. de Guise, J. Řeháček, Z. Hradil, A. Aiello, G. Leuchs, and L. L. Sánchez-Soto. Radial quantum number of laguerre-gauss modes. *Phys. Rev. A*, 89:063813, Jun 2014.
- [91] John Preskill. Quantum Computing in the NISQ era and beyond. *Quantum*, 2:79, August 2018.
- [92] M. Fannes, B. Nachtergaele, and R. F. Werner. Finitely correlated states on quantum spin chains. *Communications in Mathematical Physics*, 144(3):443 – 490, 1992.

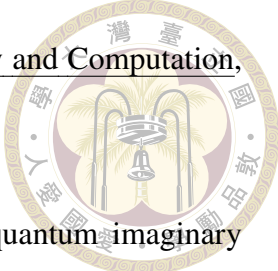
- 
- [93] Stellan Östlund and Stefan Rommer. Thermodynamic limit of density matrix renormalization. Phys. Rev. Lett., 75:3537–3540, Nov 1995.
- [94] Steven R. White. Density matrix formulation for quantum renormalization groups. Phys. Rev. Lett., 69:2863–2866, Nov 1992.
- [95] Steven R. White. Density-matrix algorithms for quantum renormalization groups. Phys. Rev. B, 48:10345–10356, Oct 1993.
- [96] Ian P McCulloch. From density-matrix renormalization group to matrix product states. Journal of Statistical Mechanics: Theory and Experiment, 2007(10):P10014, oct 2007.
- [97] Ulrich Schollwöck. The density-matrix renormalization group in the age of matrix product states. Annals of Physics, 326(1):96–192, 2011. January 2011 Special Issue.
- [98] Guifré Vidal. Efficient classical simulation of slightly entangled quantum computations. Phys. Rev. Lett., 91:147902, Oct 2003.
- [99] Guifré Vidal. Efficient simulation of one-dimensional quantum many-body systems. Phys. Rev. Lett., 93:040502, Jul 2004.
- [100] G. Vidal. Classical simulation of infinite-size quantum lattice systems in one spatial dimension. Phys. Rev. Lett., 98:070201, Feb 2007.
- [101] Jutho Haegeman, Christian Lubich, Ivan Oseledets, Bart Vandereycken, and Frank Verstraete. Unifying time evolution and optimization with matrix product states. Phys. Rev. B, 94:165116, Oct 2016.

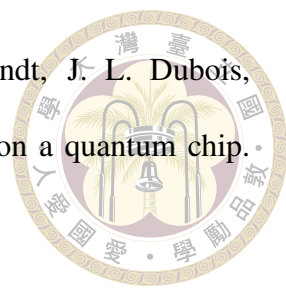
- 
- [102] Marcus Cramer, Martin B. Plenio, Steven T. Flammia, Rolando Somma, David Gross, Stephen D. Bartlett, Olivier Landon-Cardinal, David Poulin, and Yi-Kai Liu. Efficient quantum state tomography. Nature Communications, 1(1):149, Dec 2010.
- [103] Shi-Ju Ran. Encoding of matrix product states into quantum circuits of one- and two-qubit gates. Phys. Rev. A, 101:032310, Mar 2020.
- [104] F. Barratt, James Dborin, Matthias Bal, Vid Stojevic, Frank Pollmann, and A. G. Green. Parallel quantum simulation of large systems on small nisq computers. npj Quantum Information, 7(1):79, May 2021.
- [105] Adam Smith, Bernhard Jobst, Andrew G. Green, and Frank Pollmann. Crossing a topological phase transition with a quantum computer. Phys. Rev. Res., 4:L022020, Apr 2022.
- [106] Nikita Astrakhantsev, Sheng-Hsuan Lin, Frank Pollmann, and Adam Smith. Time evolution of uniform sequential circuits. Phys. Rev. Res., 5:033187, Sep 2023.
- [107] Francois Jamet, Connor Lenihan, Lachlan P. Lindoy, Abhishek Agarwal, Enrico Fontana, Baptiste Anselme Martin, and Ivan Rungger. Anderson impurity solver integrating tensor network methods with quantum computing, 2023.
- [108] Ian MacCormack, Alexey Galda, and Adam L. Lyon. Simulating large peps tensor networks on small quantum devices, 2021.
- [109] Adam Smith, M. S. Kim, Frank Pollmann, and Johannes Knolle. Simulating quantum many-body dynamics on a current digital quantum computer. npj Quantum Information, 5(1):106, Nov 2019.

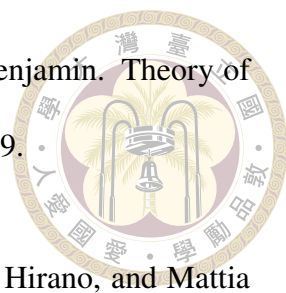


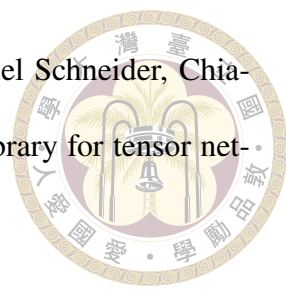
- [110] Sheng-Hsuan Lin, Rohit Dilip, Andrew G. Green, Adam Smith, and Frank Pollmann. Real- and imaginary-time evolution with compressed quantum circuits. PRX Quantum, 2:010342, Mar 2021.
- [111] Niklas Mueller, Joseph A. Carolan, Andrew Connelly, Zohreh Davoudi, Eugene F. Dumitrescu, and Kübra Yeter-Aydeniz. Quantum computation of dynamical quantum phase transitions and entanglement tomography in a lattice gauge theory. PRX Quantum, 4:030323, Aug 2023.
- [112] Taichi Kosugi, Yusuke Nishiya, Hirofumi Nishi, and Yu-ichiro Matsushita. Imaginary-time evolution using forward and backward real-time evolution with a single ancilla: First-quantized eigensolver algorithm for quantum chemistry. Phys. Rev. Res., 4:033121, Aug 2022.
- [113] Chiara Leadbeater, Nathan Fitzpatrick, David Muñoz Ramo, and Alex J W Thom. Non-unitary trotter circuits for imaginary time evolution. Quantum Science and Technology, 9(4):045007, jul 2024.
- [114] Mario Motta, Chong Sun, Adrian T. K. Tan, Matthew J. O’Rourke, Erika Ye, Austin J. Minnich, Fernando G. S. L. Brandão, and Garnet Kin-Lic Chan. Determining eigenstates and thermal states on a quantum computer using quantum imaginary time evolution. Nature Physics, 16(2):205–210, Feb 2020.
- [115] Kübra Yeter-Aydeniz, Raphael C. Pooser, and George Siopsis. Practical quantum computation of chemical and nuclear energy levels using quantum imaginary time evolution and lanczos algorithms. npj Quantum Information, 6(1):63, Jul 2020.

- 
- [116] Niladri Gomes, Feng Zhang, Noah F. Berthussen, Cai-Zhuang Wang, Kai-Ming Ho, Peter P. Orth, and Yongxin Yao. Efficient step-merged quantum imaginary-time evolution algorithm for quantum chemistry. Journal of Chemical Theory and Computation, 16(10):6256–6266, Oct 2020.
- [117] Hirofumi Nishi, Taichi Kosugi, and Yu-ichiro Matsushita. Implementation of quantum imaginary-time evolution method on nisc devices by introducing nonlocal approximation. npj Quantum Information, 7(1):85, Jun 2021.
- [118] Shi-Ning Sun, Mario Motta, Ruslan N. Tazhigulov, Adrian T.K. Tan, Garnet Kin-Lic Chan, and Austin J. Minnich. Quantum computation of finite-temperature static and dynamical properties of spin systems using quantum imaginary time evolution. PRX Quantum, 2:010317, Feb 2021.
- [119] Kübra Yeter-Aydeniz, George Siopsis, and Raphael C Pooser. Scattering in the ising model with the quantum lanczos algorithm*. New Journal of Physics, 23(4):043033, apr 2021.
- [120] Kübra Yeter-Aydeniz, Eleftherios Moschandreou, and George Siopsis. Quantum imaginary-time evolution algorithm for quantum field theories with continuous variables. Phys. Rev. A, 105:012412, Jan 2022.
- [121] Chenfeng Cao, Zheng An, Shi-Yao Hou, D. L. Zhou, and Bei Zeng. Quantum imaginary time evolution steered by reinforcement learning. Communications Physics, 5(1):57, Mar 2022.
- [122] Takashi Tsuchimochi, Yoohee Ryo, Seiichiro L. Ten-no, and Kazuki Sasasako. Improved algorithms of quantum imaginary time evolution for ground and ex-

- 
- cited states of molecular systems. Journal of Chemical Theory and Computation, 19(2):503–513, Jan 2023.
- [123] Swagat Kumar and Colin Michael Wilmott. Generalising quantum imaginary time evolution to solve linear partial differential equations. Scientific Reports, 14(1):20156, Aug 2024.
- [124] Ioannis Kolotouros, David Joseph, and Anand Kumar Narayanan. Accelerating quantum imaginary-time evolution with random measurements, 2024.
- [125] Colin P. Williams. Probabilistic nonunitary quantum computing. In Eric Donkor, Andrew R. Pirich, and Howard E. Brandt, editors, Quantum Information and Computation II, volume 5436 of Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, pages 297–306, August 2004.
- [126] Zixuan Hu, Rongxin Xia, and Sabre Kais. A quantum algorithm for evolving open quantum dynamics on quantum computing devices. Scientific Reports, 10(1):3301, Feb 2020.
- [127] Tong Liu, Jin-Guo Liu, and Heng Fan. Probabilistic nonunitary gate in imaginary time evolution. Quantum Information Processing, 20(6):204, Jun 2021.
- [128] Hirofumi Nishi, Taichi Kosugi, Yusuke Nishiya, and Yu ichiro Matsushita. Acceleration of probabilistic imaginary-time evolution method combined with quantum amplitude amplification, 2022.
- [129] Anthony W. Schlimgen, Kade Head-Marsden, LeeAnn M. Sager-Smith, Prineha Narang, and David A. Mazziotti. Quantum state preparation and nonunitary evolution with diagonal operators. Phys. Rev. A, 106:022414, Aug 2022.

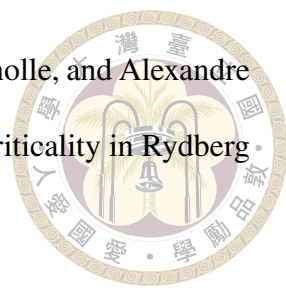
- 
- [130] F. Turro, A. Roggero, V. Amitrano, P. Luchi, K. A. Wendt, J. L. Dubois, S. Quaglioni, and F. Pederiva. Imaginary-time propagation on a quantum chip. Phys. Rev. A, 105:022440, Feb 2022.
- [131] Taichi Kosugi, Hirofumi Nishi, and Yu-ichiro Matsushita. Exhaustive search for optimal molecular geometries using imaginary-time evolution on a quantum computer. npj Quantum Information, 9(1):112, Nov 2023.
- [132] Mariane Mangin-Brinet, Jing Zhang, Denis Lacroix, and Edgar Andres Ruiz Guzman. Efficient solution of the non-unitary time-dependent Schrodinger equation on a quantum computer with complex absorbing potential. Quantum, 8:1311, April 2024.
- [133] Hao-Nan Xie, Shi-Jie Wei, Fan Yang, Zheng-An Wang, Chi-Tong Chen, Heng Fan, and Gui-Lu Long. Probabilistic imaginary-time evolution algorithm based on nonunitary quantum circuits. Phys. Rev. A, 109:052414, May 2024.
- [134] Hirofumi Nishi, Taichi Kosugi, Yusuke Nishiya, and Yu-ichiro Matsushita. Quadratic acceleration of multistep probabilistic algorithms for state preparation. Phys. Rev. Res., 6:L022041, May 2024.
- [135] Tyson Jones, Suguru Endo, Sam McArdle, Xiao Yuan, and Simon C. Benjamin. Variational quantum algorithms for discovering hamiltonian spectra. Phys. Rev. A, 99:062304, Jun 2019.
- [136] Sam McArdle, Tyson Jones, Suguru Endo, Ying Li, Simon C. Benjamin, and Xiao Yuan. Variational ansatz-based quantum simulation of imaginary time evolution. npj Quantum Information, 5(1):75, Sep 2019.

- 
- [137] Xiao Yuan, Suguru Endo, Qi Zhao, Ying Li, and Simon C. Benjamin. Theory of variational quantum simulation. Quantum, 3:191, October 2019.
- [138] David Amaro, Matthias Rosenkranz, Nathan Fitzpatrick, Koji Hirano, and Mattia Fiorentini. A case study of variational quantum algorithms for a job shop scheduling problem. EPJ Quantum Technology, 9(1):5, Feb 2022.
- [139] Marcello Benedetti, Mattia Fiorentini, and Michael Lubasch. Hardware-efficient variational quantum algorithms for time evolution. Phys. Rev. Res., 3:033083, Jul 2021.
- [140] Shi-Xin Zhang and Shuai Yin. Universal imaginary-time critical dynamics on a quantum computer. Phys. Rev. B, 109:134309, Apr 2024.
- [141] Laurens Vanderstraeten, Jutho Haegeman, and Frank Verstraete. Tangent-space methods for uniform matrix product states. SciPost Phys. Lect. Notes, page 7, 2019.
- [142] Markus Heyl. Dynamical quantum phase transitions: a review. Reports on Progress in Physics, 81(5):054001, apr 2018.
- [143] John M. Martyn, Zane M. Rossi, Andrew K. Tan, and Isaac L. Chuang. Grand unification of quantum algorithms. PRX Quantum, 2:040203, Dec 2021.
- [144] V. Zauner-Stauber, L. Vanderstraeten, M. T. Fishman, F. Verstraete, and J. Haegeman. Variational optimization algorithms for uniform matrix product states. Phys. Rev. B, 97:045145, Jan 2018.

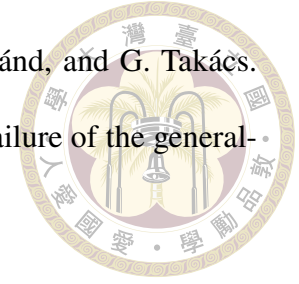
- 
- [145] Kai-Hsin Wu, Chang-Teng Lin, Ke Hsu, Hao-Ti Hung, Manuel Schneider, Chia-Min Chung, Ying-Jer Kao, and Pochung Chen. The Cytntx library for tensor networks. SciPost Phys. Codebases, page 53, 2025.
- [146] Marton Kormos, Mario Collura, Gabor Takács, and Pasquale Calabrese. Real-time confinement following a quantum quench to a non-integrable model. Nature Physics, 13(3):246–249, Mar 2017.
- [147] S. B. Rutkevich. Energy spectrum of bound-spinons in the quantum ising spin-chain ferromagnet. Journal of Statistical Physics, 131(5):917–939, Jun 2008.
- [148] Niall Robertson, Albert Akhriev, Jiri Vala, and Sergiy Zhuk. Approximate quantum compiling for quantum simulation: A tensor network based approach. ACM Transactions on Quantum Computing, 6(3), May 2025.
- [149] A. B. ZAMOLODCHIKOV. Integrals of motion and s-matrix of the (scaled) $t = t_c$ ising model with magnetic field. International Journal of Modern Physics A, 04(16):4235–4248, 1989.
- [150] G. Delfino and G. Mussardo. The spin-spin correlation function in the two-dimensional ising model in a magnetic field at $t = t_c$. Nuclear Physics B, 455(3):724–758, 1995.
- [151] Jonas A. Kjäll, Frank Pollmann, and Joel E. Moore. Bound states and E_8 symmetry effects in perturbed quantum ising chains. Phys. Rev. B, 83:020407, Jan 2011.
- [152] Paolo Pietro Mazza, Gabriele Perfetto, Alessio Lerose, Mario Collura, and Andrea Gambassi. Suppression of transport in nondisordered quantum spin chains due to confined excitations. Phys. Rev. B, 99:180302, May 2019.

- 
- [153] Olalla A. Castro-Alvaredo, Máté Lencsés, István M. Szécsényi, and Jacopo Viti. Entanglement oscillations near a quantum critical point. Phys. Rev. Lett., 124:230601, Jun 2020.
- [154] Flávia B. Ramos, Máté Lencsés, J. C. Xavier, and Rodrigo G. Pereira. Confinement and bound states of bound states in a transverse-field two-leg ising ladder. Phys. Rev. B, 102:014426, Jul 2020.
- [155] Luka Pavešić, Daniel Jaschke, and Simone Montangero. Constrained dynamics and confinement in the two-dimensional quantum ising model. Phys. Rev. B, 111:L140305, Apr 2025.
- [156] R. Coldea, D. A. Tennant, E. M. Wheeler, E. Wawrzynska, D. Prabhakaran, M. Telling, K. Habicht, P. Smeibidl, and K. Kiefer. Quantum criticality in an ising chain: Experimental evidence for emergent E_8 symmetry. Science, 327(5962):177–180, 2010.
- [157] Zhao Zhang, Kirill Amelin, Xiao Wang, Haiyuan Zou, Jiahao Yang, Urmas Nagel, Toomas Rõom, Tusharkanti Dey, Agustinus Agung Nugroho, Thomas Lorenz, Jianda Wu, and Zhe Wang. Observation of E_8 particles in an ising chain antiferromagnet. Phys. Rev. B, 101:220411, Jun 2020.
- [158] Haiyuan Zou, Yi Cui, Xiao Wang, Z. Zhang, J. Yang, G. Xu, A. Okutani, M. Hagiwara, M. Matsuda, G. Wang, Giuseppe Mussardo, K. Hódsági, M. Kormos, Zhangzhen He, S. Kimura, Rong Yu, Weiqiang Yu, Jie Ma, and Jianda Wu. E_8 spectra of quasi-one-dimensional antiferromagnet $\text{BaCo}_2\text{V}_2\text{O}_8$ under transverse field. Phys. Rev. Lett., 127:077201, Aug 2021.

- 
- [159] Joseph Vovrosh and Johannes Knolle. Confinement and entanglement dynamics on a digital quantum computer. Scientific Reports, 11(1):11577, Jun 2021.
- [160] Ali Javadi-Abhari, Matthew Treinish, Kevin Krsulich, Christopher J. Wood, Jake Lishman, Julien Gacon, Simon Martiel, Paul D. Nation, Lev S. Bishop, Andrew W. Cross, Blake R. Johnson, and Jay M. Gambetta. Quantum computing with Qiskit, 2024.
- [161] M. Hebenstreit, D. Alsina, J. I. Latorre, and B. Kraus. Compressed quantum computation using a remote five-qubit quantum computer. Phys. Rev. A, 95:052339, May 2017.
- [162] Alba Cervera-Lierta. Exact Ising model simulation on a quantum computer. Quantum, 2:114, December 2018.
- [163] A. A. Zhukov, S. V. Remizov, W. V. Pogosov, and Yu. E. Lozovik. Algorithmic simulation of far-from-equilibrium dynamics using quantum computer. Quantum Information Processing, 17(9):1–26, September 2018.
- [164] Adam Smith, M. S. Kim, Frank Pollmann, and Johannes Knolle. Simulating quantum many-body dynamics on a current digital quantum computer. npj Quantum Information, 5(1):106, Nov 2019.
- [165] Christopher Lamb, Yicheng Tang, Robert Davis, and Ananda Roy. Ising meson spectroscopy on a noisy digital quantum simulator. Nature Communications, 15(1):5901, Jul 2024.

- 
- [166] Joseph Vovrosh, Julius de Hond, Sergi Julià-Farré, Johannes Knolle, and Alexandre Dauphin. Meson spectroscopy of exotic symmetries of Ising criticality in Rydberg atom arrays, 6 2025.
- [167] Isabel Nha Minh Le, Shuo Sun, and Christian B. Mendl. Riemannian quantum circuit optimization based on matrix product operators, 1 2025.
- [168] John Preskill. Quantum Computing in the NISQ era and beyond. Quantum, 2:79, August 2018.
- [169] Jianda Wu, Márton Kormos, and Qimiao Si. Finite-temperature spin dynamics in a perturbed quantum critical ising chain with an E_8 symmetry. Phys. Rev. Lett., 113:247201, Dec 2014.
- [170] Fangli Liu, Rex Lundgren, Paraj Titum, Guido Pagano, Jiehang Zhang, Christopher Monroe, and Alexey V. Gorshkov. Confined quasiparticle dynamics in long-range interacting quantum spin chains. Phys. Rev. Lett., 122:150601, Apr 2019.
- [171] Joseph Tindall and Dries Sels. Confinement in the transverse field ising model on the heavy hex lattice. Phys. Rev. Lett., 133:180402, Oct 2024.
- [172] Kazuya Shinjo, Kazuhiro Seki, Tomonori Shirakawa, Rong-Yang Sun, and Seiji Yunoki. Unveiling clean two-dimensional discrete time quasicrystals on a digital quantum computer, 3 2024.
- [173] John S Van Dyke, Edwin Barnes, Sophia E Economou, and Rafael I Nepomechie. Preparing exact eigenstates of the open xxz chain on a quantum computer. Journal of Physics A: Mathematical and Theoretical, 55(5):055301, jan 2022.

[174] B. Pozsgay, M. Mestyán, M. A. Werner, M. Kormos, G. Zaránd, and G. Takács. Correlations after quantum quenches in the xxz spin chain: Failure of the generalized gibbs ensemble. Phys. Rev. Lett., 113:117203, Sep 2014.



[175] Mari Carmen Bañuls, Krzysztof Cichy, Hao-Ti Hung, Ying-Jer Kao, C.-J. David Lin, and Amit Singh. Dynamical quantum phase transition and thermal equilibrium in the lattice thirring model. Phys. Rev. Res., 7:023194, May 2025.

[176] Ching-Tai Huang, Yu-Cheng Lin, and Ferenc Igloi. Quench dynamics of the quantum xxz chain with staggered interactions: Exact results and simulations on digital quantum computers, 2025.