Dynamics and Simulation of Open Quantum Systems

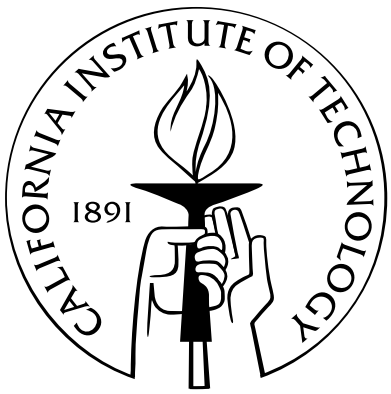
Thesis by

Michael Philip Zwolak

In Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy



California Institute of Technology

Pasadena, California

2008

(Defended June 12, 2007)

© 2008
Michael Philip Zwolak
All Rights Reserved

To my family and friends

Acknowledgements

I would like to thank John Preskill, Guifre Vidal, Massimiliano Di Ventra, Gil Refael, and Y. A. Liu for support, guidance, and/or collaboration. Also, I am appreciative of all of my collaborators over the years for fruitful scientific discourse. And of course, I am ever grateful to my family and friends.

Abstract

All systems are open to an essentially uncontrollable environment that acts as a source of decoherence and dissipation. In some cases the environment's only effect is to add a weak relaxation mechanism and thus can be ignored for short timescales. In others, however, the presence of the environment can fundamentally alter the behavior of the system. Such is the case in mesoscopic superconductors where the environment can stabilize superconductivity and in spin-boson systems where the environment induces a localization transition. Likewise, in technological applications we are often interested in systems operating far from equilibrium. Here the environment might act as a particle reservoir or strong driving force.

In all these examples, we need accurate methods to describe the influence of the environment on the system and to solve for the resulting dynamics or equilibrium states. In this thesis, we develop computational and conceptual approaches to efficiently simulate quantum systems in contact with an environment. Our starting point is the use of numerical renormalization techniques. Thus, we restrict our attention to one-dimensional lattices or small quantum systems coupled to an environment. We have developed several complementary algorithms: a superoperator renormalization algorithm for simulating real-time Markovian dynamics and for calculating states in thermal equilibrium; a blocking algorithm for simulating integro-differential equations with long-time memory; and a tensor network algorithm for branched lattices, which can be used to simulate strongly dissipative systems. Further, we provide support for an idea that to generically and accurately simulate the real-time dynamics of strongly dissipative systems, one has to include all or part of the environment within the simulation. In addition, we discuss applications and open questions.

Contents

Chapter 1

Introduction

Understanding the dynamics of quantum systems open to an environment can be a formidable challenge in many-body physics. On the one hand, even in cases where a system has trivial dynamics, coupling the system to an environment can immediately result in an unsolvable model. On the other hand, the system dynamics alone may be unsolvable, and the environment will add further complexity. Therefore, computational methods, in addition to being useful and enlightening in their own right, are crucial in assessing analytical approximations and making close contact with experiment. One of the main goals of this thesis is to develop and examine computational methods to efficiently simulate many-body systems that are in contact with an environment.

Generally, we designate systems in contact with an environment as *open systems*. These systems, for instance, can exchange energy or particles with the surrounding medium. In any particular situation, though, we have to define what we mean by *open* by choosing a partitioning of a collective set of degrees of freedom into two parts, a system and an environment. Typically one thinks of the environment as being a heat bath (e.g., a collection of phonons) or a reservoir of fermions, or the modes of the electromagnetic field. From a practical point of view, however, it is not always clear how one should choose a partitioning of the degrees of freedom. When a small collection of modes is weakly coupled to a large number of other degrees of freedom, then it is natural to make the division along this line. In this situation, the environment likely gives rise only to weak relaxation of the modes into thermal equilibrium or slow decoherence. Frequently, such a situation allows for generic approximations to be made.

In other cases, however, the most direct way of making the partition may not be adequate from the point of view of accurately calculating and understanding the influence of the environment. There is, though, a good conceptual reason for still making this division. What we want to call the system is the "interesting portion" of the complete degrees of freedom. For instance, if we have a pair of electrodes and we want to understand the response of what is in the junction between the electrodes, it is natural to designate the electrodes as the environment and the contents of the junction as the system. This is because we might want to vary the contents of the junction, modify

its parameters, or examine different realizations of disorder, while the degrees of freedom of the electrodes do not change. Thus, conceptually it categorizes the degrees of freedom into ones that we want to modify and systematically understand, and into ones that we are, most of the time, uninterested in.

Conceptual benefits aside, there are practical drawbacks to such a partitioning. Although formally the degrees of freedom of the environment might not change, approximate treatments of the environment might be invalidated under certain variations of the system. This is exactly what happens in many areas of research, where certain approximate treatments of the environment (and systems as well) can not be indiscriminately used. Thus, it would be of tremendous value to have controllable, and therefore generic, approximations to treat the influence of the environment. This means, in some situations, that to get an accurate description of the dynamics of the "interesting" degrees of freedom we might have to diligently keep tract of additional degrees of freedom that represent all or part of the environment.

This is really the purpose of this thesis: to develop a collection of computational and conceptual tools that allow for generic, tractable, and accurate ways to treat the influence of the environment on the interesting degrees of freedom. These tools should find use in many disciplines, including the investigation of more fundamental issues such as the effect of dissipation on quantum phase transitions, but also in the applied disciplines that often deal with quite complex situations and, thus, have to resort to the concept of *open systems*. In the remainder of this introductory chapter, we introduce the basic concepts of dissipative quantum systems and to the ideas behind the numerical renormalization techniques that form the basis of our approach.

1.1 Some examples

From the above discussion, we see that the study of open systems can be very complex. However, it is worthwhile endeavor as, since every system is essentially an open system, there are many applications which require a detailed examination of the influence of the environment. Two areas of particular interest are strongly dissipative quantum systems and driven quantum systems, which we now briefly introduce and mention their relevance to this thesis.

1.1.1 Strongly dissipative quantum systems

The preeminent example of a dissipative quantum system is the spin-boson model. The model is that of a spin, or two-level system, interacting with a bosonic bath. In the absence of the bath, the two-level system undergoes coherent oscillations between the two levels, which might represent the lowest energy states in two quantum wells. Thinking in terms of a particle tunneling between the two wells, the presence of a bath exerts a drag on the particle: as the particle tries to tunnel

from one well to the other, it has to release itself from the bosons relaxed around it. Under certain conditions and when the coupling strength, e.g., the dissipation strength, to the bath gets too strong, the particle becomes localized. That is, it is no longer able to oscillate between the wells. This is known as a *localization transition*. It is an example of dissipation fundamentally altering the physics of the system, and it is not limited to just localization. The bath can cause a variety of crossovers in the behavior of the system taking it from oscillatory, to exponential relaxation, to power-law decay, to completely localized regimes.

One of the original motivations for studying the spin-boson model was to examine the effect of dissipation on "macroscopic" quantum tunneling of the flux trapped in a SQUID [1,2]. Since the flux is a relatively macroscopic variable, it can couple to many modes of the environment, and it was questioned under what conditions this coupling destroys the quantum behavior of the system (e.g., how the dissipation effects the tunneling rate). However, dissipative effects of the type displayed by the spin-boson model appear in many places. In addition to macroscopic quantum tunneling, the spin-boson model is relevant to charge transport in organic and biological molecules. In this scenario, there are typically charge traps along a molecule for which electrons or holes can tunnel between. In the process, though, they interact with the vibrations of the molecule and also modes of the surrounding medium (e.g., ionic and dipole fluctuations in the surrounding solution). Thus, how well the charges can tunnel will depend on how strong the charges interact with this environment and on the exact nature of the environment.

Other very interesting phenomena come from the drastic effect of dissipation on quantum phase transitions. One such effect is the the stabilization of superconductivity (see below). This is the situation in ultrasmall superconducting elements where charging effects can become very important. In the absence of dissipation and when the charging energy is large, there will be large fluctuations in the phase of the superconducting order parameter. These fluctuations destroy phase coherence between neighboring elements (connected via a Josephson junction), bringing the total system into an insulating state. However, dissipation can very efficiently damp these phase fluctuation and restore superconductivity, which is also a realization of a localization transition. This, at first glance, is very counterintuitive, as one normally associates superconductivity with the absence of dissipation.

In this thesis we develop a method that can be used to numerically study one-dimensional lattices where each lattice site is connected to an independent bath. This is exactly the scenario that describes a one-dimensional array of low-capacitance ($C \sim 1$ fF) Josephson junctions connected to shunt resistors or in the presence of quasiparticle dissipation. Thus, the method can be used to study these intricate quantum systems and the interplay between the local dissipative physics and the long-range quantum fluctuations in the quantum critical regime of the pure lattice.

1.1.2 Driven quantum systems

In essentially every technological application, a system is driven out of equilibrium. This is the case from microscopic examples, such as the transistors running your computer, to examples where the element is macroscopic, such as the engine that powers your car to the beach. As we strive ever harder to develop new technologies, we have continued to miniaturize the active elements in devices. For instance, we now envision using nanoscale objects, such as individual molecules, in electronics applications.

In these novel situations, a fundamental challenge is to understand the response of some very small object to the presence of a probe or reservoir. This can be, for example, a pair of electrodes at different chemical potentials which drive electrons through a system. Or, the reservoirs could simply be materials maintained at different temperatures, which then cause energy (or particles) to flow through the system. Typically, though, we are uninterested in (or just completely unable to solve for) the degrees of freedom that maintain the non-equilibrium state of the system, and therefore they constitute what we want to call an environment. However, the way we describe the influence of the environment can be critical to our understanding of the system. Therefore, we should not take the task of modeling the environment lightly, especially if we want a generic method capable of examining a range of systems and parameters.

Of particular interest to us is the understanding the conductance of nanoscale systems. We will discuss later in this thesis some drawbacks in the current state-of-the-art calculations. We will also, though, introduce a methodology using some of the simulation techniques developed in this thesis. This methodology may serve as a testing ground for developing better approximations to calculating properties of these non-equilibrium systems.

1.2 Dissipative quantum systems

We briefly discussed above dissipative systems and some interesting examples. But we did not go into any detail about what actually constitutes the environment in those examples and how we can represent this environment. Obviously, in some cases the environment's Hamiltonian will be true to the actual physical system. Some examples are when the environment is composed of phonons in a crystal, the electromagnetic field, a nuclear spin bath, or a band of fermions. In other cases, there may be an indirect connection, for instance, where the environment is a set of fermionic modes but represented as a bosonic bath (via bosonization). However, in the vast majority of cases, the microscopic details of the couplings and the modes will not be known, and we will be reduced to a phenomenological representation of the environment. In the next subsection we describe a generic way to represent the environment, which encompasses many of these cases, and discuss how to phenomenologically make a connection with real systems. In the subsequent subsections we will

describe how to use this representation, and we give a specific example of a dissipatively driven phase transition.

1.2.1 Representing the environment

A standard approach to representing an environment is to take a system-bath Hamiltonian where the bath is represented as a collection of harmonic oscillators. For example, a single particle in contact with a bath is given the Hamiltonian

$$H = \frac{P^2}{2M} + V(Q,t) + \sum_{i=1}^{N} \left\{ \frac{p_i^2}{2m_i} + \frac{m_i \omega_i^2}{2} \left(x_i - \frac{c_i}{m_i \omega_i^2} Q \right)^2 \right\}, \tag{1.1}$$

where the first two terms are for the particle of mass M moving in a possibly time-dependent potential V. The last term is a sum over N environment modes, with a linear couping $(-c_i x_i Q)$ to the position coordinate of the particle. The term proportional to Q^2 is simply to remove the potential renormalization due to the having the particle in contact with the environment. This is a rather *ad hoc* prescription, and may or may not correspond to the physical situation at hand. If one wants a full treatment of the effect of the environment, and not just the dissipation, the potential renormalization has to be treated with more care.

The system-bath model has been studied for a long time. However, Caldeira and Leggett [1,3,4] gave the model new life by using it for strong dissipation in condensed matter systems (e.g., dissipative tunneling of a "macroscopic" variable mentioned above), and it has taken on the name *Caldeira-Leggett model*. Part of the justification in using this model is that it is assumed the environment is so large that the effect of the system on any single mode of the environment need only be considered in linear response. If this is the case, the bath can be represented as just a collection of harmonic oscillators with a suitably chosen set of frequencies and coupling constants. In particular, we can define a collective coordinate

$$X = \sum_i c_i x_i \tag{1.2}$$

as the coordinate of the bath that is coupled to the system. The Fourier transform of the two-point correlation function of this operator at zero temperature is called the *spectral function* and is just

$$J(\omega) = \frac{1}{2} \int dt e^{i\omega t} \langle X(t) X(0) \rangle . \tag{1.3}$$

The idea is either to have a microscopic model which gives this spectral function or to find it by requiring that the classical equations of motion of the system are reproduced.

For instance, if the system is a Josephson junction shunted by a resistor, the classical equation of

motion is provided by the resistively and capacitively shunted junction model (RCSJ) [5], which gives

$$\frac{\hbar C}{2e}\frac{d^2\gamma}{dt^2}+\frac{\hbar}{2eR_S}\frac{d\gamma}{dt}+I_c\sin\gamma=I \tag{1.4}$$

in the presence of an external current I, and $\gamma=\Delta\varphi-(2\pi/\Phi_0)\int\mathbf{A}\cdot d\mathbf{s}$ is the gauge invariant phase difference across the junction. This equation of motion has dissipation proportional to the "velocity" of the phase, with a friction coefficient proportional to $1/R_S$, e.g., ohmic dissipation due to the shunt resistance R_S. In general the choice of the set of harmonic oscillators to give ohmic dissipation in the classical limit is not unique (see, e.g., Ref. 6 for examples).

More generally for ohmic dissipation, Caldeira and Leggett choose the bosonic bath model to reproduce the equation of motion

$$M\ddot{Q}+\eta\dot{Q}+\frac{\partial V}{\partial Q}=F_{ext}(t) \tag{1.5}$$

when the Hamiltonian 1.1 is treated classically. $F_{ext}(t)$ represents some fluctuating external force and η is the classical friction coefficient. The energy dissipated by a particle obeying this equation of motion is $\eta\dot{Q}^2$. Specifically, we now show that one has to choose the bath spectral function $J(\omega)=\eta\omega$ to reproduce the equation of motion 1.5. To show this, we work in Fourier space, where the equation of motion 1.5 is

$$-M\omega^2Q(\omega)-\imath\eta\omega Q(\omega)+\left(\frac{\partial V}{\partial Q}\right)(\omega)=F_{ext}(\omega)\ . \tag{1.6}$$

Using the classical Lagrangian (corresponding to the Hamiltonian 1.1),

$$L=\frac{M}{2}\dot{Q}^2-V(Q)+\frac{1}{2}\sum_i m_i\left(\dot{x}_i^2-\omega_i^2x_i^2\right)-\sum_i c_ix_iQ-\sum_i\frac{c_i^2Q^2}{2m_i\omega_i^2}\,, \tag{1.7}$$

we can get the equations of motion

$$M\ddot{Q}(t)=-\frac{\partial V}{\partial Q}+\sum_i\frac{c_i}{m_i\omega_i^2}Q(t)+F_{ext}(t)-\sum_i c_ix_i\,, \tag{1.8}$$

and

$$m_i\ddot{x}_i(t)=-m_i\omega_i^2x_i(t)-c_iQ(t)\ . \tag{1.9}$$

From the Fourier transform of the latter equations, we find

$$x_i(\omega)=\frac{c_iQ(\omega)}{m_i\left(\omega_i^2-\omega^2\right)}\,. \tag{1.10}$$

We can use this to eliminate the bath degrees of freedom in the equation of motion for Q. Putting

the $x_i(\omega)$ into the Fourier transform of the equation of motion for Q we get

$$-M\omega^2 Q(\omega) = -\left(\frac{\partial V}{\partial Q}\right)(\omega) + F_{ext}(\omega) + K(\omega)\, Q(\omega)\ . \tag{1.11}$$

The new function K is

$$K(\omega) = \sum_i \frac{c_i^2 \omega^2}{m_i \omega_i^2 \left(\omega_i^2 - \omega^2\right)}\,. \tag{1.12}$$

We can write this quantity in terms of the bath spectral function, defined in equation 1.3,

$$J(\omega) = \frac{\pi}{2}\sum_i \frac{c_i^2}{m_i \omega_i}\delta(\omega - \omega_i)\ . \tag{1.13}$$

For the equations of motion for the particle to give the classical equation of motion 1.6, we need to match the function $K(\omega)$ with $\imath\eta\omega$. Going to the continuum limit, K becomes

$$K(\omega) = \frac{2\omega^2}{\pi}\int_0^\infty \frac{J(\omega')\, d\omega'}{\omega'\left(\omega'^2 - \omega^2\right)} \equiv \imath\eta\omega\,. \tag{1.14}$$

This gives the requirement[†]

$$J(\omega) = \eta\omega\,. \tag{1.15}$$

The equation 1.15 is the spectral function for ohmic dissipation.

Thus, we now have a physical motivation for choosing a bosonic bath with linear coupling to the system and for choosing a specific spectral function. Obviously, this procedure can be generalized to other cases, either by similarly requiring that the spectral function be such that it reproduces a classical equation of motion or by having a microscopic derivation (e.g., which can be done in certain cases, see Ref. 7,8) or requirements that need to be satisfied by the spectral function.

1.2.2 Generic path integral treatment

One of the nice properties of the representation of the environment given above is that it gives a generic starting point to treat dissipation within the path integral formalism. By generic, we simply mean that the bath degrees of freedom can be integrated out analytically (because they are quadratic), without regard to the system Hamiltonian, resulting in a general damping term in the effective action. This effective action can be used with whatever system one desires, although one may have to resort to possibly system-specific approximations to calculate physical properties. In this subsection we show some of the main results of the path integral formulation of the Caldeira-Leggett model.

[†]We can do this integral by taking $\omega \to \omega + \imath\epsilon$ and doing the contour integration. The result for $J(\omega)$ is consistent with assumptions made in doing the contour integration.

We start with the partition function in the path integral representation

$$Z_\beta = \int dQ \int_{Q(0)=Q}^{Q(\beta)=Q} \mathcal{D}Q \int dx \int_{x(0)=x}^{x(\beta)=x} e^{-S^E[Q,x]}\,, \tag{1.16}$$

where the Euclidean action is a sum over the different contributions

$$S^E\left[Q,x\right] = S_S^E + S_I^E + S_R^E\,, \tag{1.17}$$

and x represents the coordinates of the bath. The contributions are

$$S_S^E = \int_0^\beta d\tau \left\{ \frac{M}{2}\dot{Q}^2 + V\left(Q\right) \right\}, \tag{1.18}$$

$$S_I^E = \sum_i \int_0^\beta d\tau \left\{ -c_i x_i Q + \frac{1}{2}\frac{c_i^2 Q^2}{m_i \omega_i^2} \right\}, \tag{1.19}$$

and

$$S_S^E = \sum_i \int_0^\beta d\tau \left\{ \frac{m_i}{2}\dot{x}_i^2 + \frac{m_i\omega_i^2}{2}x_i^2 \right\}. \tag{1.20}$$

The idea is then to define an effective action that involves only the system degrees of freedom. That is, we integrate out the bath coordinates. The effective action is defined by

$$e^{-S_{eff}[Q]} \equiv \int dx \int_{x(0)=x}^{x(\beta)=x} \mathcal{D}x e^{-S^E[Q,x]}\,. \tag{1.21}$$

For ohmic dissipation, the effective action for the Caldeira-Leggett model is (see Refs. 9,10)

$$S_{eff}\left[Q\right] = \int_0^\beta d\tau \left\{ \frac{M}{2}\dot{Q}\left(\tau\right)^2 + V\left(Q\left(\tau\right)\right) \right\} + \frac{\eta}{4\pi}\int_0^\beta \int_0^\beta d\tau d\tau' \left(\frac{\pi}{\beta}\right)^2 \left(\frac{Q\left(\tau\right) - Q\left(\tau'\right)}{\sin\left(\frac{\pi}{\beta}\left|\tau - \tau'\right|\right)} \right)^2, \tag{1.22}$$

which is just the action of the isolated system plus an additional term that adds a long-range interaction in (imaginary) time. At zero temperature, this long-range interaction drops off as $\left(\tau - \tau'\right)^{-2}$. Thus, for instance, our particle could be the phase difference across a Josephson junction. Due to this long-range interaction in time, even the single junction can display a phase transition. The interaction in time is also commonly referred to as a *memory*, because it represents the system leaving an impression on the environment and having this impression interact back with the system at a later time. Ohmic dissipation gives a long-time memory, but for other bath spectral functions, the memory can be shorter range.

1.2.3 Dissipative quantum phase transitions

There has been a tremendous amount of interest in quantum phase transitions in recent years. These are continuous transitions that occur at zero temperature as one varies some control parameter of the Hamiltonian. For instance, in the Ising model as one varies the transverse magnetic field, the ground state of the system goes from an ordered state with broken Z_2 symmetry to a "disordered" state with that symmetry restored. These two states have qualitatively different properties.

Of more relevance to this thesis, is the the superconductor-insulator transition in a one-dimensional array of Josephson junctions. The array might be formed by a set of superconducting grains or a nicely lithographed set of junctions. Of course, this system also will have a finite temperature transition of the superconducting elements within the array. Therefore, we can focus instead on the energy regime for which $E_J \ll T_c$, where T_c is the critical temperature for superconductivity on a grain or element and E_J is the Josephson coupling. Further, we focus on $T \lesssim E_J$, e.g., at temperatures far below the critical temperature of the superconducting elements. Thus, all cooper pairs will be formed and we can ignore fluctuations in the magnitude of the order parameter, e.g., only the phase φ of the order parameter (on each superconducting element) is dynamical. Given that the phase is the only dynamical part of the order parameter, we can write down the Hamiltonian as the quantum phase model

$$H = \frac{1}{2}\sum_{i,j} 4e^2 n_i \left(C^{-1}\right)_{ij} n_j + \sum_{\langle ij\rangle} V_{ij}\left[1-\cos\left(\varphi_i-\varphi_j\right)\right] . \tag{1.23}$$

The number and phase operators are conjugate variables satisfying the commutation relation

$$[\varphi_i, n_j] = \imath\delta_{ij} , \tag{1.24}$$

and thus the number operator n_l is

$$n_l = -i\frac{\partial}{\partial\varphi_l} . \tag{1.25}$$

The capacitance matrix, C, gives the interactions of cooper pairs on the grains with those on the same grain and other grains. We assume $\left(C^{-1}\right)_{ij} = C^{-1}\delta_{ij}$, e.g., we will have only on-site interactions. One could, for instance, take into account long-range Coulomb interactions or just nearest-neighbor interactions. Given a pure (not disordered) one-dimensional system and only on-site Coulomb interaction, the quantum phase model is

$$H_{QPM} = 4E_C\sum_l n_l^2 - E_J\sum_l \cos\left(\Delta\varphi_l\right) , \tag{1.26}$$

where E_C is the charging energy for a single electron $e^2/2C$ and $\Delta\varphi_l = \varphi_l - \varphi_{l+1}$ is the phase difference across the junction.

Qualitatively, at large E_C the ground state of the system is in a definite number state with zero

net charge ($n_l = 0$) on each superconducting element. This state will have large fluctuations of the phase and, thus, the system will be in an insulating state. On the other extreme, when E_J is large, it is energetically favorable for the phase difference across each junction to be zero (or integral multiples of 2π). Thus, the state will be superconducting. A variational approach similar to the one below gives a phase transition at

$$E_J/E_C \sim 4 \tag{1.27}$$

for a one-dimensional lattice[11,12].

As mentioned above, dissipation can also drive a phase transition. Quantum phase transitions, such as the ones just described, are the result of quantum fluctuations. For instance, the quantum phase model can be thought of as a collection of particles (the phases of each superconductor) with a kinetic energy ($\sim E_C$) and a nearest neighbor interaction potential ($\sim E_J$). When the kinetic energy is much larger than the potential energy, the particle wants to fluctuate around. This prevents the system from having the necessary order to be superconducting. However, in the presence of dissipation, this interplay between kinetic and potential energy is fundamentally changed.

One particular case that is really interesting is when the Josephson junction or Josephson junction array (with the quantum phase Hamiltonian) is in the presence of ohmic dissipation. Ohmic dissipation can be provided either by a resistor in parallel with the junction or by quasiparticle tunneling across the junction. These two situations are not equivalent. A shunt resistor allows for a continuous transfer of charge across the junction. In this case, the phase variable will no longer be periodic and will be able to take on all values[13]. Whereas, for quasiparticle dissipation, charge can only be discretely transferred across the junction. Although not equivalent, both cases can drive a dissipative phase transition.

For the case of a shunt resistor, the Hamiltonian for a single junction is

$$H = 4E_C n^2 - E_J \cos(\varphi) + \varphi X + H_B\,, \tag{1.28}$$

where we have the quantum phase model with φ being the phase difference across the junction and n the number of cooper pairs transferred across the junction (also, E_C is the nearest-neighbor capacitive interaction, rather than on-site). The coupling of the junction to its bath is through the phase difference across the junction and the collective bath coordinate X (see equation 1.2). The Hamiltonians H_B for the bath has the remaining terms of the Caldeira-Leggett form. For a one-dimensional array, the Hamiltonian would be similar but with an independent bath at each junction. For these systems, the dissipation favors superconductivity. When the shunt resistance $R_S < R_Q = h/(2e)^2 \approx 6.5\ \mathrm{k}\Omega$, e.g., for large dissipation, the junction will be in the superconducting state. Whereas, for larger R_S the fluctuations of the phase can cause the state to be insulating.

We can get insight into this behavior by using the self-consistent harmonic approximation

(SCHA)[14,15,13,16].‡ Working with the Hamiltonian 1.28, we can write down the effective action using equation 1.22,

$$S_{eff} = \int_0^\beta d\tau \left[\frac{1}{16E_C} \left(\frac{\partial \varphi}{\partial \tau} \right)^2 - E_J \cos \varphi(\tau) \right] + \frac{\alpha}{8} \int_0^\beta \int_0^\beta d\tau d\tau' \left(\frac{1}{\beta} \right)^2 \left(\frac{\varphi(\tau) - \varphi(\tau')}{\sin\left(\frac{\pi}{\beta} |\tau - \tau'| \right)} \right)^2 , \tag{1.29}$$

where we have replaced the friction coefficient η with $\alpha = R_Q/R_S$. Now the idea with the SCHA is to bound the free energy of the system by using a trial action, S_{tr}, that is quadratic and the Gibbs-Helmholtz inequality[9]

$$F' \equiv F_{tr} + \frac{1}{\beta} \langle S - S_{tr} \rangle_{tr} \geq F \, . \tag{1.30}$$

Choosing the trial action to have the quadratic form

$$S_{tr} = \int_0^\beta d\tau \left[\frac{1}{16E_C} \left(\frac{\partial \varphi}{\partial \tau} \right)^2 + \frac{D}{2} (\varphi(\tau))^2 \right] + \frac{\alpha}{8} \int_0^\beta \int_0^\beta d\tau d\tau' \left(\frac{1}{\beta} \right)^2 \left(\frac{\varphi(\tau) - \varphi(\tau')}{\sin\left(\frac{\pi}{\beta} |\tau - \tau'| \right)} \right)^2 , \tag{1.31}$$

we can bound the free energy by calculating F' and minimizing it by varying D. The variational parameter D plays the role of a stiffness. If $D = 0$, there is no potential constraining the fluctuations of the phase. Thus, there will be large fluctuations and no superconductivity. However, if D takes on a finite value, this indicates that the phase is at least partially pinned around the origin. Therefore, the important quantity to look at is how large are the phase fluctuations in the state that puts the minimum bound on the free energy.

Minimizing the equation 1.30 with respect to D, one obtains[13]

$$D = E_J e^{-\langle \varphi^2 \rangle / 2} , \tag{1.32}$$

with

$$\langle \varphi^2 \rangle = \frac{1}{\beta} \sum_\nu \left[C \left(\frac{\omega_\nu}{2e} \right)^2 + \alpha \frac{|\omega_\nu|}{2\pi} + D \right]^{-1} , \tag{1.33}$$

where $\omega_\nu = 2\pi\nu/\beta$ are the Matsubara frequencies. This equation has to be solved self-consistently to find D. At zero temperature and small E_J,

$$D = \begin{cases} E_J \left[E_J/E_C 8\alpha^2 \right]^{1/(\alpha-1)} & \alpha > 1 \\ 0 & \alpha < 1 \end{cases} . \tag{1.34}$$

‡One can also do a similar calculation for quasiparticle dissipation, see Ref. 17,13.

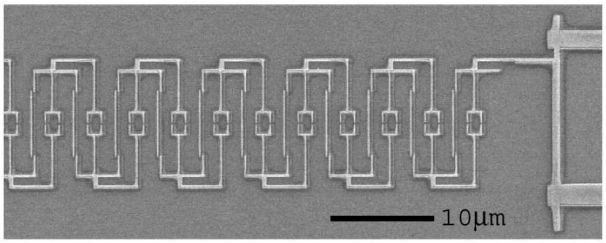


Figure 1.1: One-dimensional Josephson junction array from Ref. 19. Each junction is actually two junctions forming a SQUID and shunted by a resistor.

This gives infinite fluctuations of the phase for $\alpha < 1$ and

$$\langle \varphi^2 \rangle \approx \frac{2}{\alpha - 1} \ln \frac{8\alpha^2 E_C}{E_J} \tag{1.35}$$

for $\alpha > 1$. Thus, the variational calculation predicts a transition in the behavior of the junction at $\alpha = 1$. This result has been found by other more detailed calculations [16,6].

Indeed the transition discussed above has been observed in experiment [18,19,20,21,22]. We now show some results for the experiment of Ref. 19. In these experiments, one-dimensional Josephson junctions arrays (of 46 sites) were fabricated with electron-beam lithography. An example array is shown in Figure 1.1. Each junction is actually two junctions in a SQUID geometry so that the Josephson coupling can be tuned. In addition, each junction is shunted by a resistor. In the experiment, several such arrays were fabricated with different R_S. Among other quantities, they measured the zero-bias resistance as a function of temperature. For arrays with $R_S < R_Q$ (large dissipation), they find that the resistance goes to zero as the temperature is decreased, see Figure 1.2. Whereas for arrays with $R_S > R_Q$ the resistance increases as temperature is decreased. This behavior is consistent with that predicted by the dissipatively driven phase transition discussed above.

In this section we have went through some of the basic physics of dissipative quantum systems. In particular, we have introduced a generic system-bath model Hamiltonian that allows one to treat strong dissipation present in condensed matter systems. This model is quite tractable since it is composed of noninteracting bosons connected linearly to the system. One of the goals of this thesis is to give a generic and tractable computational approach to problems of this type. We have also seen a specific example of a Josephson junction (and Josephson junction arrays) with ohmic dissipation, where a novel dissipatively driven phase transition occurs.

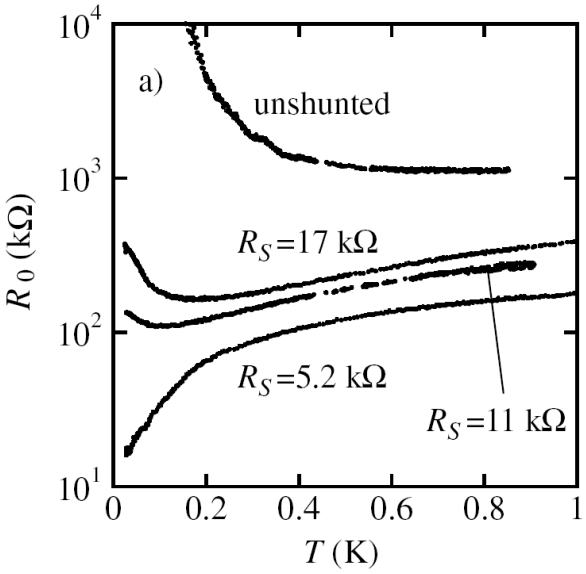


Figure 1.2: Zero-bias resistance of four Josephson junction arrays each with a different shunt resistance (labeled on the plot), from Ref. 19.

1.3 Matrix product state simulations

When one tries to directly simulate a quantum system, one encounters an insurmountable obstacle for large systems: the exponentially large size of the underlying Hilbert space. This makes it all but impossible to classically simulate a quantum system, or even store a quantum state. Thus, one has to resort to techniques that circumvent the direct simulation of the system. One could, for instance, use quantum Monte Carlo simulations[23] to calculate path integrals like the ones that appear above[24]. Alternatively, one can use numerical renormalization techniques like the density matrix renormalization group (DMRG)[25,26] or the numerical renormalization group (NRG)[27,28].

Both these renormalization techniques are systematic and controlled ways to truncate the Hilbert space of a long, one-dimensional lattice system. That is, if we take a pure state, $|\psi\rangle \in \mathbb{C}^{\otimes d^L}$, of a length L lattice of d dimensional sites, the state can be written in the lattice basis as

$$|\psi\rangle = \sum_{i_1 \cdots i_L}^{d} c_{i_1 \cdots i_L} |i_1 \cdots i_L\rangle. \tag{1.36}$$

This state would take d^L complex parameters to describe, which would only be possible to store on a typical computer for spin half lattices of size $L \sim 15$. If we are interested only in local Hamiltonians, e.g., ones with on-site and nearest neighbor interactions, we might be tempted to alleviate this exponential growth by writing down a description of our state 1.36 based on local coefficients instead of the global coefficients $c_{i_1 \cdots i_L}$.

It turns out that there is a representation of states that gives this local description. These states

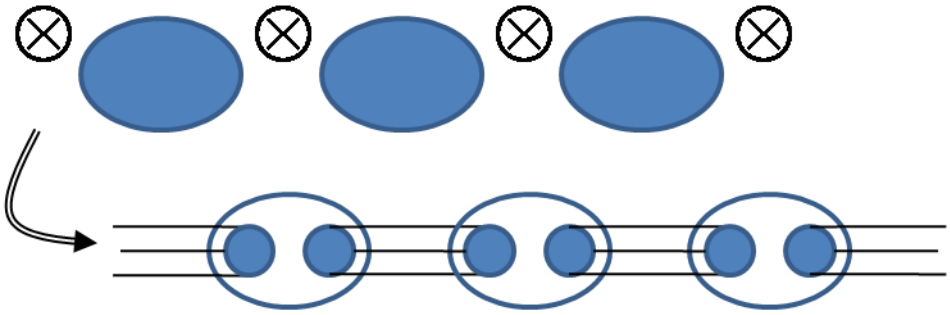


Figure 1.3: Matrix product states replace the tensor product structure of the original lattice (shown top) with a finite dimensional bonded structure (shown bottom).

are called *matrix product states*. The idea behind matrix product states is to abandon the tensor product structure of the Hilbert space of the lattice, and replace it with a local structure as shown in Figure 1.3. The local structure considers each lattice site as bonded to the rest of the lattice on each side through a bond of finite dimension χ. More specifically, the replacement of the lattice structure can be understood by defining two auxiliary sites for each lattice site. Then, pairing nearest neighbor auxiliary sites (not on the same lattice site) one can form an entangled state

$$|I\rangle = \sum_{\alpha_1 \cdots \alpha_L = 1}^{\chi} |\alpha_1 \alpha_1 \alpha_2 \alpha_2 \cdots \alpha_L \alpha_L\rangle \tag{1.37}$$

where the auxiliary sites have been paired together as shown in Figure 1.3. Then, we can define a map from this state to the original state in equation 1.36 by [29]

$$\mathcal{A} = \prod_l \mathcal{A}_l = \prod_l \left(\sum_{i_l, \beta_{2l}, \beta_{2l+1}} A^{[l] i_l}_{\beta_{2l} \beta_{2l+1}} |i_l\rangle\langle \beta_{2l} \beta_{2l+1}| \right) . \tag{1.38}$$

This gives the state

$$|\psi\rangle = \mathcal{A}|I\rangle = \sum_{i_1 \cdots i_L} \mathrm{tr}\left[A^{[1] i_1} A^{[2] i_2} \cdots A^{[L] i_L} \right] |i_1 i_2 \cdots i_L\rangle \, , \tag{1.39}$$

hence the name *matrix product state*. The original coefficients $c_{i_1 \cdots i_L}$ have been replaced by a product of matrices A. As we will see below, any arbitrary state can be represented in this way, except that the bond dimension will grow exponentially in the system size L. Matrix product states are the basis of DMRG [30] and NRG [31].

1.3.1 Representation

Recently, there has been a tremendous number of advancements in our ability to simulate quantum lattice systems. These advancements have mainly stemmed from the constant flow of ideas from

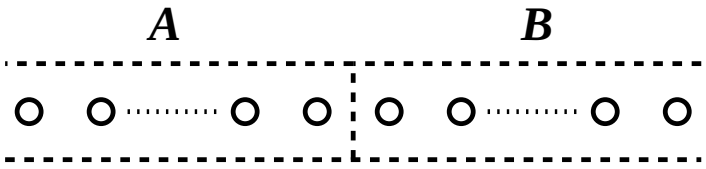


Figure 1.4: Lattice partitioned in to two parts, A and B.

quantum information into computational physics. The extent of this exchange is so large that we can not possibly cover it. We refer the reader to Refs. 32 and 29, which capture some of the main ideas. Basically, though, these advancements have involved developing new ways to manipulate matrix product states, generalizing the concepts to higher dimension, and overall, showing the deep connections between entanglement and classical simulation of quantum systems. Here, we will largely limit ourselves to the formulation of matrix product state algorithms due to Vidal[33,34].

Following Vidal, we will use a slightly different representation of matrix product states than the one in equation 1.39. The representation we use comes from the quantum state 1.36 by performing a series of Schmidt decompositions successively through the lattice. Consider the state 1.36 and a partition $A : B$ of the system into two subsystems A and B, as shown in Figure 1.4. We can decompose this state via a Schmidt decomposition (SD)[35,36,37] with respect to the partition as

$$|\psi\rangle = \sum_{\alpha=1}^{\chi_A} \lambda_\alpha |\Phi_\alpha^{[A]}\rangle \otimes |\Phi_\alpha^{[B]}\rangle \,, \tag{1.40}$$

where $\left\{|\Phi_\alpha^{[A]}\rangle, |\Phi_\alpha^{[B]}\rangle\right\}$ are a set of orthonormal vectors, called the *Schmidt vectors* (SV), which span the support of the reduced density matrices $\rho^{[A]}$ and $\rho^{[B]}$. The λ_α are the square roots of the eigenvalues of the reduced density matrix of subsystem A (or subsystem B, which has identical eigenvalues). Since λ_α^2 are the eigenvalues of a density matrix, they will be real numbers such that $\lambda_\alpha^2 > 0$ and $\sum_\alpha \lambda_\alpha^2 = 1$. χ_A is the Schmidt number with respect to the partition $A : B$ and is a measure of entanglement between the two subsystems A and B. We use a general value of χ defined as

$$\chi = \max_A \chi_A \,, \tag{1.41}$$

where the maximum is over all possible bipartite divisions of the lattice with a single boundary.

We proceed as follows: SDs are performed successively through the lattice and, at each partition, we expand the new SVs in terms of the ones from the previous partition[33,34]. The first partitioning is into the first site on one side and the remaining sites on the other side, e.g., $A : B = 1 : [2 \cdots L]$.

The SD for this partition is

$$\begin{aligned} |\psi\rangle &= \sum_{\alpha_1} \lambda^{[1]}_{\alpha_1} |\Phi^{[1]}_{\alpha_1}\rangle |\Phi^{[2\cdots n]}_{\alpha_1}\rangle \\ &= \sum_{i_1,\alpha_1} \Gamma^{[1]i_1}_{\alpha_1} \lambda^{[1]}_{\alpha_1} |i_1\rangle |\Phi^{[2\cdots n]}_{\alpha_1}\rangle \,, \end{aligned}$$

where the $\Gamma^{[1]i_1}_{\alpha_1}$ are the expansion coefficients which give the SVs in terms of the lattice basis

$$|\Phi^{[1]}_{\alpha_1}\rangle = \sum_{i_1} \Gamma^{[1]i_1}_{\alpha_1} |i_1\rangle. \tag{1.42}$$

The SVs for the right partition B are expanded into the local basis for the second site

$$|\Phi^{[2\cdots n]}_{\alpha_1}\rangle = \sum_{i_2} |i_2\rangle |\tau^{[3\cdots n]}_{\alpha_1 i_2}\rangle \,. \tag{1.43}$$

To find the states $|\tau\rangle$, we need to perform another SD with the partition $A : B = [1 \cdot 2] : [3 \cdots L]$. The $|\tau\rangle$ are then given as

$$|\tau^{[3\cdots n]}_{\alpha_1 i_2}\rangle = \sum_{\alpha_2} \Gamma^{[2]i_2}_{\alpha_1\alpha_2} \lambda^{[2]}_{\alpha_2} |\Phi^{[3\cdots n]}_{\alpha_2}\rangle. \tag{1.44}$$

With this set of transformations, the original state is

$$|\psi\rangle = \sum_{i_1,\alpha_1,i_2,\alpha_2} \Gamma^{[1]i_1}_{\alpha_1} \lambda^{[1]}_{\alpha_1} \Gamma^{[2]i_2}_{\alpha_1\alpha_2} \lambda^{[2]}_{\alpha_2} |i_1\rangle |i_2\rangle |\Phi^{[3\cdots n]}_{\alpha_2}\rangle. \tag{1.45}$$

Successively performing a similar set of transformations gives us the coefficients c in equation 1.36 as

$$c_{i_1\cdots i_L} = \sum_{\alpha_1,\cdots,\alpha_{L-1}} \Gamma^{[1]i_1}_{\alpha_1} \lambda^{[1]}_{\alpha_1} \Gamma^{[2]i_2}_{\alpha_1\alpha_2} \lambda^{[2]}_{\alpha_2} \Gamma^{[3]i_3}_{\alpha_2\alpha_3} \cdots \Gamma^{[L]i_L}_{\alpha_{L-1}} \,. \tag{1.46}$$

This is basically the same as the matrix product state expression 1.39 (for open-boundary conditions). However, the matrices $A^{[l]i_l}$ have been divided into a set of transformation matrices $\Gamma^{[l]i_l}$ and Schmidt coefficients $\lambda^{[l]}$. With this slightly different representation, the SD of the system split between sites l and $l+1$ is easily obtained to be

$$|\psi\rangle = \sum_{\alpha_1} \lambda^{[l]}_{\alpha_1} |\Phi^{[1\cdots l]}_{\alpha_l}\rangle |\Phi^{[l+1\cdots L]}_{\alpha_l}\rangle \,, \tag{1.47}$$

with

$$
\begin{aligned}
|\Phi_{\alpha_l}^{[1\cdots l]}\rangle &= \sum_{\alpha_1,\cdots,\alpha_{l-1}} \Gamma_{\alpha_1}^{[1]i_1}\lambda_{\alpha_1}^{[1]}\cdots\Gamma_{\alpha_{l-1}\alpha_l}^{[l]i_l}|i_1\cdots i_l\rangle\,, \\
|\Phi_{\alpha_l}^{[l+1\cdots L]}\rangle &= \sum_{\alpha_{l+1},\cdots,\alpha_{L-1}} \Gamma_{\alpha_l\alpha_{l+1}}^{[l+1]i_{l+1}}\cdots\lambda_{\alpha_{L-1}}^{[L-1]}\Gamma_{\alpha_{L-1}}^{[L]i_L}|i_{l+1}\cdots i_L\rangle.
\end{aligned}
$$

In addition, the procedure shows that any state can be represented in matrix product form.

This gives a potentially compact representation of the state of the system. Instead of the d^L coefficients c, we have the state written in terms of (at most) Ld, $\chi\times\chi$ transformation matrices $\left\{\Gamma^{[l]i_l}\right\}$ (and also the coefficients $\left\{\lambda^{[l]}\right\}$, which we will ignore because there are only $L\chi$ of them). This gives us $Ld\chi^2$ complex parameters total. However, this can represent an arbitrary state, but the bond dimension χ has to grow exponentially large in the size of the system. In fact, from the properties of the SD, we can bound χ by $d^{L/2}$, e.g., this is the maximum number of SV one can have when one divides the system in half. Thus, the representation 1.46 (or 1.39) has to rely on properties of the state (or system) to be compact. The property one needs is that there has to be a limited amount of entanglement between every partition of the system (see below).

1.3.2 Simulation

In addition to a representation of the state of the system, we need a way to perform simulations, either to find the ground state or evolve some state in real-time. In other words, we need a way to manipulate matrix product states efficiently. In addition to the prescriptions given by DMRG and NRG, there have been more recent methods developed such as those in Refs. 33 and 34. We will not go into any detail here. Instead, we just highlight some of the main ideas of a particular simulation method. One way to perform simulations is to Trotter decompose the evolution operator $U = e^{-\imath Ht}$ into a set of local transformations, which can be then be applied successively to evolve some initial state. However, for this to work efficiently, one has to place restrictions on the types of Hamiltonians under consideration. In particular, Hamiltonians with on-site and nearest neighbor interactions can be efficiently simulated in this way.[§] The computational cost of performing such a simulation is $NLd^3\chi^3$, where N is the number of time (imaginary time) steps to perform a real-time evolution (find the ground state). That is, the computational cost only scales linearly in the system size, as opposed to exponentially, so long as the amount of entanglement (the magnitude of χ) does not grow with the system size.

Thus, we can ask the question, in what way do we expect χ to behave? Numerically it is known from DMRG that χ does not have to be too large to represent ground states accurately, even for Hamiltonians that are in the critical regime (but on a finite lattice). What is found is that the

[§]Some of the techniques, though, can be generalized beyond just nearest-neighbor interactions, see Chapter 4.

spectrum of coefficients, $\lambda_\alpha^{[l]}$, for each partition, decays very rapidly. One can then find a good approximation to the state by truncating the SD, incurring some small and controllable error, and otherwise obtaining a very compact representation of the state. This is how DMRG and NRG systematically truncate the Hilbert space.

We can get a crude intuition of how efficiently an MPS can describe the state of a system by using the fact that for critical systems in one-dimension with finite range interactions the block entropy grows logarithmically in the system size. For a lattice of size L and a contiguous block of sites within the lattice of size $B \ll L$, the von Neumann entropy of the block grows as

$$S \approx \frac{c}{3} \log_2 B \,, \tag{1.48}$$

where c is the central charge. This entropy growth is much smaller than the maximum possible entropy growth, which is linear in the block size B. And more specifically, for a lattice with open boundary conditions, the maximum entropy across any partition in half grows as

$$S \approx \frac{c}{6} \log_2 L \,, \tag{1.49}$$

e.g., at half the rate of the block because the block has two boundaries. Immediately then, one suspects that ground states obeying such a slow entropy growth may satisfy the requirement of limited entanglement in the system. For instance, consider the maximum amount of entanglement that a χ-dimensional MPS can hold

$$S_{max}(\chi) = \sum_{\alpha=1}^{\chi} -\frac{1}{\chi} \log_2 \frac{1}{\chi} = \log_2 \chi. \tag{1.50}$$

This just comes from being in an equal superposition of χ states in the SD. Intuitively, since polynomial decay of correlations is related to the logarithmic entropy growth in equation 1.48, one would expect if the entropy of the MPS does not grow at the same rate that one will eventually not be able to accurately represent correlations in the system. Thus, any state that does not have

$$\frac{dS_{max}\left[\chi(L)\right]}{dL} = \frac{dS\left(B_{L/2}\right)}{dL} \tag{1.51}$$

will not be adequate to represent the state. This leads to the condition

$$\chi \propto L^{c/6} \,. \tag{1.52}$$

This is a very crude estimate, as it relies on the maximum entropy an MPS of dimension χ can hold, rather than the actual fidelity of the states. However, it gives intuition that the efficiency of the

MPS representation and simulation will grow as a small polynomial in the system size. Actually, one can give a similar argument that is rigorous, as was shown in Ref. 38. The result, though, is not quite as a generous as equation 1.52, but nevertheless polynomial in the system size.

1.4 Summary

Before we embark on this the bulk of this thesis, let us summarize some of the main goals and developments. In this introduction, we have seen some very interesting applications of methods to understand the effects of the environment. Mainly we focused on the effects of strong dissipation on quantum systems and the importance of understanding driven systems. In addition, we examined a very powerful approach to dissipative systems based on the path integral formalism and the Caldeira-Leggett model for the environment. This gives a very generic and tractable analytic approach to study of the effects of strong dissipation.[¶] Our goal is somewhat analogous: to develop a generic and tractable computational method. Specifically, we are interested in developing methods to simulate open quantum systems via the use of numerical renormalization techniques.

We discuss, in Chapter 2, a method to simulate open systems with Markovian dissipation. The idea behind this method is a superoperator renormalization procedure based on a matrix product decomposition of the density matrix of a system. In addition to being able to simulate Markovian dissipation, this method can be used to calculate states in thermal equilibrium or unequal-time correlation functions. In Chapter 3, we will give a real-time procedure for solving certain classes of integro-differential equations that show up in systems with strong dissipation. In Chapter 4, we describe some tools that allow one to perform real-time simulations with MPS that can be used to simulate systems with slightly longer range interactions (beyond nearest neighbor) or certain restricted classes of non-Markovian evolution. The method can also be used to simulate a finite lattice interacting with a common bath. We then give an argument in Chapter 5 that to simulate non-Markovian evolution generically, one has to include at least part of the environment explicitly within the simulation. Based on this, we developed a tensor network simulation for branched lattices in Chapter 6, which can be used to simulate strongly dissipative systems in one-dimension when each lattice site is connected to an independent bath. In Chapter 7 we discuss a particular application of using these simulation techniques to understanding non-equilibrium transport in nanoscale systems. Finally, in Chapter 8, we examine how the entanglement entropy behaves in branched lattices and how this relates to efficient simulation of these lattices. We conclude in Chapter 9.

[¶]And in some cases, numerical, such as Monte Carlo simulations to calculate the path integrals.

Chapter 2

Mixed state quantum dynamics in one dimension

We present an algorithm to study mixed-state dynamics in one-dimensional quantum lattice systems. The algorithm can be used, e.g., to construct thermal states or to simulate real-time evolution given by a generic master equation. Its two main ingredients are (*i*) a *superoperator* renormalization scheme to efficiently describe the state of the system and (*ii*) the time evolving block decimation technique to efficiently update the state during a time evolution. The computational cost of a simulation increases significantly with the amount of correlations between subsystems but it otherwise depends only linearly on the system size. We present simulations involving quantum spins and fermions in one spatial dimension.

2.1 Introduction

The most interesting quantum phenomena involve strongly correlated many-body systems, but studying such systems — a central task in the areas of condensed matter physics, quantum field theory and, since recent years, also quantum information science[35,39]— has too often proven a formidable challenge. Indeed, in quantum many-body theory only a few exact solutions are available, while most analytical approximations remain uncontrolled. As a consequence, numerical calculations are of great importance. But even these suffer from a severe computational obstacle: an exponential growth of degrees of freedom with the system size that renders the direct simulation of most quantum systems prohibitively inefficient.

And yet, ingenious methods such as quantum Monte Carlo techniques[23] can be used to approximately evaluate, e.g., certain ground state properties in quantum lattice models. In one-dimensional lattices, strikingly accurate results for quantities such as ground state energies and two-point correlators can be obtained by using White's *density matrix renormalization group* (DMRG)[25,26,40], a technique that has dominated most numerical research in the field since its invention more than

a decade ago. Generalizations of the DMRG have also yielded accurate low energy spectra [41] or allowed for the simulation of real-time evolution for small times [42,43,44].

Recently, the *time evolving block decimation* (TEBD) algorithm [33,34] has been proposed to simulate real-time evolution in one-dimensional quantum lattice systems. This technique can be easily adapted to standard DMRG implementations [45,46] and seems to be very efficient [45,46,47,48]. As in DMRG, a decisive factor in the performance of the TEBD method is that not a lot of entanglement is present in the system, a condition that is ordinarily met in one-dimensional lattices at low energies [33,34].

In this chapter we extend the TEBD algorithm to handle mixed states. We describe how to efficiently simulate, in one-dimensional quantum lattice systems, real-time Markovian dynamics as given by a (possibly time-dependent) master equation made of arbitrary nearest neighbor couplings. By considering evolution in imaginary time, the present extension can also be used to construct thermal states for any given temperature. Thus, we show how to numerically explore non-equilibrium many-body dynamics under realistic conditions, including the effects of finite-temperature and decoherence.

A key observation for the success of the algorithm is that in one spatial dimension many states of interest, including thermal states and local perturbations thereof, contain only a restricted amount of correlations between subsystems, in a sense to be further specified. This fact, parallel to the restricted amount of entanglement observed in the pure-state case, allows us to introduce an efficient decomposition for the state of the system, referred to as *matrix product decomposition* (MPD). The MPD is nothing but a mixed-state version of a matrix product state [49,30], and as such, we can use the TEBD to update it during a time evolution. It also follows that our scheme can again be fully incorporated into standard DMRG implementations without much programming effort [45,46].

2.2 Superoperator renormalization

We consider a generic one-dimensional quantum lattice made of n sites, labeled by index l, $l \in \{1, \cdots, n\}$, each one described by a local Hilbert space $\mathbb{H}^{[l]} \cong \mathbb{C}_d$ of finite dimension d. We assume the evolution of the n sites, in a global state ρ, is given by a master equation [39]

$$\begin{aligned} \dot{\rho} &= \mathcal{L}[\rho] \\ &= -i[H,\rho] + \sum_{\mu} \left(L_\mu \rho L_\mu^\dagger - \frac{1}{2} L_\mu L_\mu^\dagger \rho - \frac{1}{2} \rho L_\mu L_\mu^\dagger \right), \end{aligned} \tag{2.1}$$

where H and L_μ are the Hamiltonian and Lindblad operators, and where we require that the (possibly time-dependent) Lindbladian superoperator $\mathcal{L}$ further decompose into terms involving at

most two contiguous sites,

$$\mathcal{L}[\rho] = \sum_l \mathcal{L}_{l,l+1}[\rho]. \tag{2.2}$$

2.2.1 Reduced superoperators

A pure state evolution is described by a vector $|\Psi\rangle$ in the n-fold tensor product of $\mathbb{C}_d$. Let us divide the n sites into two blocks, denoted L (left) and R (right). Then DMRG and TEBD consider reduced density matrices, e.g., that of block L,

$$|\Psi\rangle \in \mathbb{C}_d^{\otimes n} \quad \longrightarrow \quad \rho^{[L]} \equiv \mathrm{tr}_R(|\Psi\rangle\langle\Psi|) \ \in \mathbb{L}(\mathbb{H}^{[L]}), \tag{2.3}$$

where $\mathbb{L}(\mathbb{H})$ denotes the set of linear mappings on $\mathbb{H}$ or, equivalently, the complex vector space of $\dim(\mathbb{H})\times\dim(\mathbb{H})$ matrices. Here we are concerned with the evolution of a mixed state, which requires more notation. For each site l, let $\mathbb{K}^{[l]} \cong \mathbb{L}(\mathbb{H}^{[l]}) \cong \mathbb{C}_{d^2}$ denote the vector space of $d \times d$ complex matrices. We switch into representing a density matrix $\sigma \in \mathbb{L}(\mathbb{H})$ as a "superket" $|\sigma\rangle_\sharp \in \mathbb{K} \cong \mathbb{L}(\mathbb{H})$, while a superoperator $\mathcal{Q} \in \mathbb{L}(\mathbb{L}(\mathbb{H}))$ is regarded as a linear mapping $\mathcal{Q}_\sharp \in \mathbb{L}(\mathbb{K})$ [†],

$$\left.\begin{array}{rcl} |\Phi\rangle & \in & \mathbb{H} \\ \sigma & \in & \mathbb{L}(\mathbb{H}) \\ \mathcal{Q} & \in & \mathbb{L}(\mathbb{L}(\mathbb{H})) \end{array}\right\} \rightarrow \left\{\begin{array}{rcl} |\Phi\rangle_\sharp & \in & \mathbb{K} \\ |\sigma\rangle_\sharp & \in & \mathbb{K} \\ \mathcal{Q}_\sharp & \in & \mathbb{L}(\mathbb{K}), \end{array}\right. \tag{2.4}$$

where $|\Phi\rangle_\sharp \equiv ||\Phi\rangle\langle\Phi|\rangle_\sharp$. For $d \times d$ matrices A and B, the scalar product ${}_\sharp\langle|\rangle_\sharp$ between superkets $|A\rangle_\sharp$ and $|B\rangle_\sharp$, and the action of $\mathcal{Q}_\sharp$ on $|A\rangle_\sharp$, are defined through

$${}_\sharp\langle A|B\rangle_\sharp \equiv \frac{1}{d}\mathrm{tr}(A^\dagger B), \qquad \mathcal{Q}_\sharp|A\rangle_\sharp \equiv |\mathcal{Q}[A]\rangle_\sharp. \tag{2.5}$$

Also, if $\mathcal{Q}$ is a superoperator on a bipartite space $\mathbb{H}^{[L]} \otimes \mathbb{H}^{[R]}$ and $\{|M_\mu\rangle_\sharp\}$ is an orthonormal basis in $\mathbb{K}^{[R]} \cong \mathbb{L}(\mathbb{H}^{[R]})$, we define the partial trace of $\mathcal{Q}_\sharp$ over block R as

$$\mathrm{tr}_{\sharp R}(\mathcal{Q}_\sharp) \equiv \sum_\mu {}_\sharp\langle M_\mu|\mathcal{Q}_\sharp|M_\mu\rangle_\sharp. \tag{2.6}$$

Finally, let $\rho \in \mathbb{L}(\mathbb{C}_d^{\otimes n})$ be the state of the n-site lattice and $|\rho\rangle_\sharp$ its superket. We define the *reduced superoperator* for a block of sites, say for block L, as

$$|\rho\rangle_\sharp \in (\mathbb{C}_{d^2})^{\otimes n} \ \longrightarrow \ \mathcal{Q}_\sharp{}^{[L]} \equiv \mathrm{tr}_{\sharp R}(|\rho\rangle_\sharp\langle\rho|) \ \in \mathbb{L}(\mathbb{K}^{[L]}), \tag{2.7}$$

[†] We use the subscript ♯ "sharp" to denote operators [superoperators] when represented as superkets [respectively mappings between superkets].

in analogy with (2.3), and rewrite equation (2.1) as

$$|\dot{\rho}\rangle_\sharp = \mathcal{L}_\sharp |\rho\rangle_\sharp, \tag{2.8}$$

which parallels the Schrödinger equation $|\dot{\Psi}\rangle = -iH|\Psi\rangle$.

2.2.2 Renormalization of reduced superoperators

Given blocks L and R, the Schmidt decomposition of $|\rho\rangle_\sharp$ reads

$$|\rho\rangle_\sharp = \sum_{\alpha=1}^{\chi_\sharp} \lambda_{\sharp\alpha} |M_\alpha^{[L]}\rangle_\sharp \otimes |M_\alpha^{[R]}\rangle_\sharp, \quad \lambda_{\sharp\alpha} \geq \lambda_{\sharp\alpha+1} \geq 0, \tag{2.9}$$

where the Schmidt superkets $\{|M_\alpha^{[L,R]}\rangle_\sharp\}$ fulfill

$$\mathcal{Q}_\sharp{}^{[L]} |M_\alpha^{[L]}\rangle_\sharp = (\lambda_{\sharp\alpha})^2 |M_\alpha^{[L]}\rangle_\sharp, \quad \mathcal{Q}_\sharp{}^{[L]} \equiv \mathrm{tr}_{\sharp R}(|\rho\rangle_\sharp\langle\rho|), \tag{2.10}$$

$$\mathcal{Q}_\sharp{}^{[R]} |M_\alpha^{[R]}\rangle_\sharp = (\lambda_{\sharp\alpha})^2 |M_\alpha^{[R]}\rangle_\sharp, \quad \mathcal{Q}_\sharp{}^{[R]} \equiv \mathrm{tr}_{\sharp L}(|\rho\rangle_\sharp\langle\rho|). \tag{2.11}$$

The rank $\chi_\sharp$ of the reduced superoperators $\mathcal{Q}_\sharp{}^{[L]}$ and $\mathcal{Q}_\sharp{}^{[R]}$ measures the amount of correlations between blocks L and R. In principle its value is only bounded above by the dimensions of $\mathbb{K}^{[L]}$ and $\mathbb{K}^{[R]}$, which grow exponentially in the number of sites. However, as the examples below illustrate, many situations of interest involving one-dimensional mixed-state dynamics are only *slightly correlated*, in that the coefficients $\{\lambda_{\sharp\alpha}\}$ *decay very fast with* α. That is, a good approximation to $|\rho\rangle_\sharp$ can be obtained by truncating (2.9) so that only a relatively small number of terms are considered.

Thus, whereas DMRG and TEBD are based on decimating the block space $\mathbb{H}^{[L]}$ supporting the reduced density matrix $\rho^{[L]}$ of pure state $|\Psi\rangle$, equation (2.3), here we propose to decimate the block space $\mathbb{K}^{[L]} \cong \mathbb{L}(\mathbb{H}^{[L]})$ supporting the reduced superoperator $\mathcal{Q}^{[L]}$ in (2.7) [‡].

2.2.3 Matrix product decomposition and TEBD

We regard ρ as a vector $|\rho\rangle_\sharp$ in the n-fold tensor product of $\mathbb{C}_{d^2}$, while the master equation (2.8) is formally equivalent to the Schrödinger equation. Mixed-state dynamics can therefore be simulated by adapting the pure-state techniques of Refs. 33 and 34. Given an orthonormal basis $\{|i_l\rangle_\sharp\}$ of $\mathbb{K}^{[l]}$

[‡] Given a bipartite pure state $|\Psi^{[LR]}\rangle$ with χ Schmidt coefficients λ_α, the decomposition (2.9) for superket $|\Psi^{[LR]}\rangle_\sharp$ will have $\chi_\sharp = \chi^2$ coefficients given by $\lambda_{\sharp[\alpha\alpha']} = \lambda_\alpha \lambda_{\alpha'}$, see below.

for site l, we expand $|\rho\rangle_\sharp$ as [§]

$$|\rho\rangle_\sharp = \sum_{i_1=0}^{d^2-1} \cdots \sum_{i_n=0}^{d^2-1} c_{i_1 \cdots i_n} \; |i_1\rangle_\sharp \otimes \cdots \otimes |i_n\rangle_\sharp . \tag{2.12}$$

We choose $|0_l\rangle_\sharp = |I/d\rangle_\sharp$ to be proportional to the identity in $\mathbb{H}^{[l]}$, so that physical normalization of ρ, $\mathrm{tr}(\rho) = 1$, corresponds to $c_{0\cdots 0} = 1$. Then we use a MPD,

$$c_{i_1 i_2 \cdots i_n} = \sum_{\alpha_1, \cdots, \alpha_{n-1}} \Gamma^{[1]i_1}_{\alpha_1} \lambda^{[1]}_{\sharp\,\alpha_1} \Gamma^{[2]i_2}_{\alpha_1 \alpha_2} \lambda^{[2]}_{\sharp\,\alpha_2} \Gamma^{[3]i_3}_{\alpha_2 \alpha_3} \cdots \Gamma^{[n]i_n}_{\alpha_{n-1}}, \tag{2.13}$$

which can be built through a succession of Schmidt decompositions of $|\rho\rangle_\sharp$ (see Refs. 33 and 34 for details). Finally, we update tensors $\{\Gamma^{[l]}\}$ and $\{\lambda_\sharp{}^{[l]}\}$ during an evolution of the form (2.1)-(2.2) by using the TEBD algorithm [¶].

2.3 Examples

2.3.1 Thermal states

Given a nearest neighbor Hamiltonian H and an inverse temperature $\beta \equiv 1/kT$, a mixed state of interest is the thermal state

$$\rho_\beta \equiv \frac{e^{-\beta H}}{Z(\beta)} = \frac{1}{Z(\beta)} \sum_s e^{-\beta E_s} |E_s\rangle\langle E_s|, \tag{2.14}$$

where $Z(\beta) \equiv tr(e^{-\beta H})$ is the partition function. One can numerically simulate ρ_β by attempting to compute all relevant energy eigenstates $|E_s\rangle$ and averaging them with weights $e^{-\beta E_s}/Z(\beta)$. A very simple and efficient alternative is to build a MPD for $|\rho_\beta\rangle_\sharp$ by simulating an imaginary time evolution from the completely mixed state,

$$|e^{-\beta H}\rangle_\sharp = \exp(-\beta \mathcal{T}_\sharp)|I\rangle_\sharp, \tag{2.15}$$

where superket $|I\rangle_\sharp$ and superoperator $\mathcal{T}_\sharp$ correspond to

$$|I\rangle_\sharp = |I_1\rangle_\sharp \otimes \cdots \otimes |I_n\rangle_\sharp, \quad \mathcal{T}[A] \equiv \frac{1}{2}(HA + AH). \tag{2.16}$$

[§] Coefficients $c_{i_1 \cdots i_n}$ in equation (2.12) can be made real by choosing $|i_l\rangle_\sharp$ to be the superket of an Hermitian matrix in $\mathbb{L}(\mathbb{H}^{[l]})$. However, in some occasions (e.g., when computing two-point time-dependent expectation values in a state out of equilibrium) the use of complex coefficients seems unavoidable.

[¶] The evolution resulting from (2.8) is in general non-unitary in the space $(\mathbb{C}_{d^2})^{\otimes n}$ of superkets $|\rho\rangle_\sharp$. The TEBD algorithm can still be used.

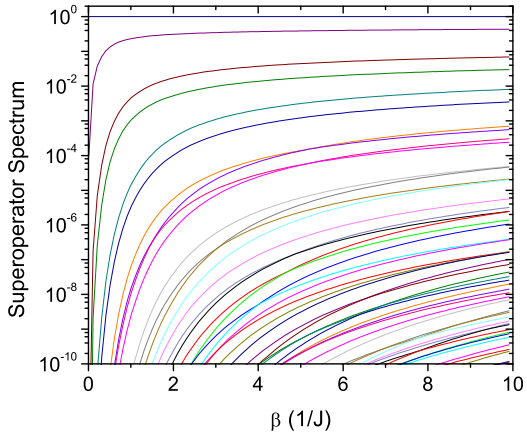


Figure 2.1: Quantum Ising chain with transverse magnetic field, equation (2.17), at finite temperature. Local dimension $d = 2$, $n = 100$ sites and effective $\chi_\sharp = 80$. At zero temperature, $\beta \to \infty$, this model corresponds to a quantum critical point. The spectrum $\{\lambda_{\sharp\alpha}^2\}$ of the reduced superoperator $\mathcal{Q}_\sharp{}^{[L]}$ for the left half chain is plotted as a function of $\beta \in [0, 10/J]$ (only the 52 largest eigenvalues are shown). For any inverse temperature β, a fast decay of $\lambda_{\sharp\alpha}^2$ in α ensures that the state can be accurately approximated by a MPD with small effective $\chi_\sharp$.

Indeed, $\exp(-\beta\mathcal{T}_\sharp)$ can be Trotter expanded into transformations involving only two adjacent sites, and the MPD can be therefore updated using the TEBD [‖]. Notice that a single run of the simulation builds the thermal state $\rho_{\beta'}$ for any intermediate value of $\beta' \in [0, \beta]$. Figure (2.1) corresponds to thermal states for a quantum Ising model with transverse magnetic field,

$$H = \sum_{l=1}^{n-1} \sigma_l^x \otimes \sigma_{l+1}^x + \sum_{l=1}^{n} \sigma_l^z. \tag{2.17}$$

2.3.2 Time-dependent Markovian master equation

We consider a lattice of $n = 100$ sites loaded with $n/2$ fermions that evolve according to a Lindbladian

$$\mathcal{L}[\rho] = -i[H, \rho] + \gamma \sum_{l=1}^{n} (n_l \rho n_l - \frac{1}{2}\rho n_l^2 - \frac{1}{2} n_l^2 \rho), \tag{2.18}$$

$n_l \equiv a_l^\dagger a_l$, where the last term accounts for phase damping and the Hamiltonian part corresponds to hoping between adjacent sites and a time-dependent on-site energy,

$$H = -J \sum_{l=1}^{n-1} (a_l^\dagger a_{l+1} + h.c.) - \mu(t)(\sum_{l=1}^{n/2} n_l - \sum_{l=n/2+1}^{n} n_l), \tag{2.19}$$

[‖]It is advisable to renormalize the state $|e^{-\beta H}\rangle_\sharp$ as $|e^{-\beta H}/Z(\beta)\rangle_\sharp$ during the evolution by requiring $c_{0\cdots 0} = 1$ in equations (2.12)-(2.13) at all intermediate values of β.

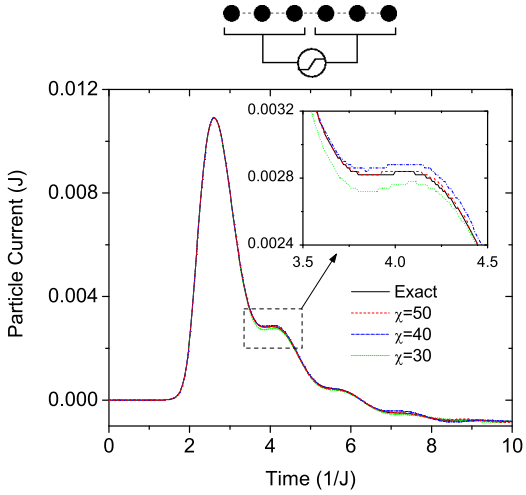


Figure 2.2: Fermionic lattice of equation (2.18) at finite temperature $\beta = 1/J$, dephasing $\gamma = 0.4J$, and with $d = 2$, $n = 100$. The particle current, see equation (2.20), is due to a time-dependent applied bias $\mu(t)$ with turn-on time $t_0 = 2/J$ and rise time $t_s = 0.1/J$. The exact solution is obtained by numerically integrating the (Gaussian) time evolution for two-point correlators. Instead, the numerical simulations are achieved after mapping the fermion lattice into a spin lattice with nearest neighbor couplings by means of a Jordan-Wigner transformation. The time evolution is then broken into small gates for pairs of nearest neighbor spins and implemented using the TEBD algorithm. The simulations with an effective $\chi_\sharp = 30, 40, 50$ show rapid convergence to the exact solution. Figure (2.3) justifies this convergence, which will also be addressed in more detail.

where $\mu(t) \equiv \mu_0[e^{-\frac{t-t_0}{t_s}} + 1]^{-1}$ introduces a bias μ_0 between the left and right halves of the lattice at $t = t_0$. Figure (2.2) shows the particle current

$$-2\,\mathrm{Im}\,\langle a_{50}^\dagger(t)a_{51}(t)\rangle \tag{2.20}$$

as a result of switching on the bias.

2.3.3 Unequal-time correlators

For the above fermion system with no bias, $\mu(t) = 0$, and finite temperature, we finally consider the expectation value

$$\langle a_l^\dagger(t)a_1(0)\rangle = \mathrm{tr}\left(a_l^\dagger \mathcal{E}_t[a_1\rho]\right), \tag{2.21}$$

where $\mathcal{E}_t$ is the time evolution operator resulting from the master equation. Since the Lindbladian $\mathcal{L}$ is time-independent, we can integrate the master equation (2.8) to obtain $\mathcal{E}_{t_\sharp} = \exp(\mathcal{L}_\sharp t)$. The simulation (see Figure (2.4)) is achieved as follows: (i) the initial state of the system, a thermal state with $\beta = 1/J$, is obtained by evolution in imaginary time as explained in Example 1; (ii) the annihilation operator a_1 is applied to the initial state ρ to obtain $a_1\rho$; (iii) $a_1\rho$ is evolved in time

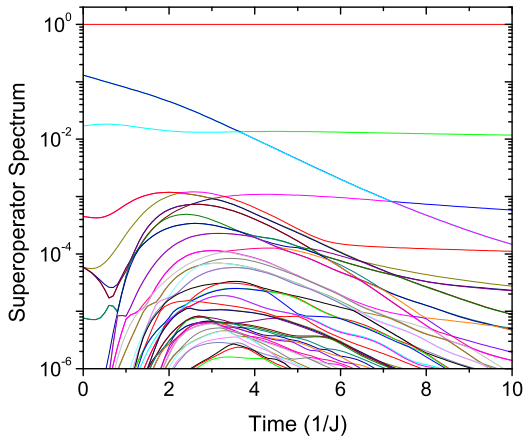


Figure 2.3: Same system as in Figure (2.2). The spectrum $\{\lambda_\sharp^2{}_\alpha\}$ of the reduced superoperator $\mathcal{Q}_\sharp{}^{[L]}$ for the left $n/2$ sites is plotted as a function of time. The number of relevant eigenvalues $\lambda_\sharp^2{}_\alpha$, say above 10^{-6}, increases as the applied bias is turned on, but remains small throughout the evolution and it even decreases for long times.

according to $\mathcal{E}_t$; (iv) the creation operator $a_l^\dagger$ is applied on $\mathcal{E}_t[a_1\rho]$; and (v) the trace of the resulting operator $a_l^\dagger\mathcal{E}_t[a_1\rho]$ is computed. Each of these steps can be performed efficiently by using a MPD and the update techniques of Refs. 33 and 34.

2.3.4 Pure state density matrices

Now we turn our attention to a problem with using a density matrix to represent a pure state of the system. Consider the Schmidt decomposition of a pure state

$$|\Psi\rangle = \sum_{\alpha=1}^{\chi_\Psi} \lambda_\alpha^{[L]} |\Phi_\alpha^{[L]}\rangle \otimes |\Phi_\alpha^{[R]}\rangle\,. \tag{2.22}$$

The density matrix for this state is

$$|\rho_\Psi\rangle_\sharp = \sum_{\alpha,\alpha'=1}^{\chi_\Psi} \lambda_\alpha\lambda_{\alpha'} |\Phi_\alpha^{[L]}\rangle|\Phi_\alpha^{[R]}\rangle\langle\Phi_{\alpha'}^{[R]}|\langle\Phi_{\alpha'}^{[L]}|. \tag{2.23}$$

Defining an orthonormal basis $|\tilde{M}_{\alpha\alpha'}^{[L]}\rangle_\sharp = |\Phi_\alpha^{[L]}\rangle\langle\Phi_{\alpha'}^{[L]}|$ and similarly for partition R, we can immediately write down the Schmidt decomposition of the density matrix as

$$|\rho_\Psi\rangle_\sharp = \sum_{\alpha,\alpha'=1}^{\chi_\Psi} \lambda_\alpha\lambda_{\alpha'} |\tilde{M}_{\alpha\alpha'}^{[L]}\rangle_\sharp \otimes |\tilde{M}_{\alpha\alpha'}^{[R]}\rangle_\sharp. \tag{2.24}$$

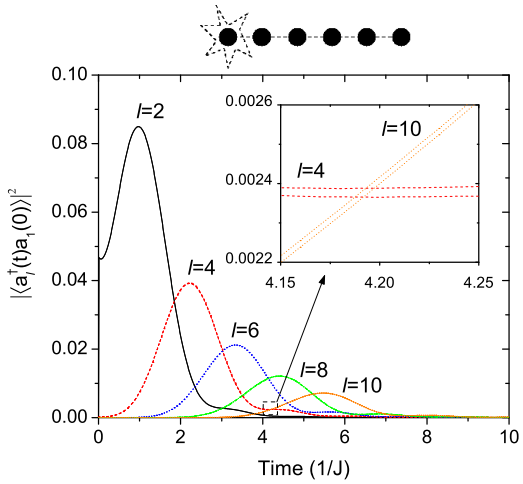


Figure 2.4: Fermionic lattice of equation (2.18) at finite temperature $\beta = 1/J$, dephasing $\gamma = 0.4J$, and with $d = 2$, $n = 100$ and no applied bias, $\mu(t) = 0$. Unequal time, two-point correlator (2.21) for $l = 2, 4, 6, 8, 10$ and $t \in [0, 10/J]$. The results corresponding to an effective $\chi_\sharp = 40$ and 50 practically overlap at all times, as the inset shows.

Since this is the basis that diagonalizes the reduced superoperator **. Due to the double sum over α, α', $\chi_\sharp = \chi_\Psi^2$. Thus, there is an extreme loss in efficiency when using the density matrix to describe a pure state ††. We show example spectrums in Figure 2.5 of a Heisenberg spin chain simulated with a wavefunction and with a pure-state density matrix. Obviously, the density matrix is not the most compact description of the state. However, there are other implications which we now discuss.

If the system is in only a mildly mixed state, stochastic integration techniques [50] within the pure state description will be a more efficient way to simulate the system. Also, even when the state is highly mixed, if there is some important pure component of the state, using the density matrix description will be relatively inefficient. This corresponds to the state $|\rho\rangle_\sharp = \mathrm{p}_1 |I/d^n\rangle_\sharp + \mathrm{p}_2 |\Psi\rangle\langle\Psi|$, which is called a pseudo-pure state and is the type of state present in NMR [51,52].

**Since we have degenerate eigenvalues in the Schmidt decomposition, we have freedom in the basis we choose; thus we can choose a basis for the left partition (likewise for right) that is a local orthonormal Hermitian basis. The basis is similar to the Pauli matrices: $|X^{[L]}_{\alpha\alpha'}\rangle_\sharp = \frac{1}{\sqrt{2}}(|\Phi^{[L]}_\alpha\rangle\langle\Phi^{[L]}_{\alpha'}| + |\Phi^{[L]}_{\alpha'}\rangle\langle\Phi^{[L]}_\alpha|)$, $|Y^{[L]}_{\alpha\alpha'}\rangle_\sharp = \frac{\imath}{\sqrt{2}}(|\Phi^{[L]}_\alpha\rangle\langle\Phi^{[L]}_{\alpha'}| - |\Phi^{[L]}_{\alpha'}\rangle\langle\Phi^{[L]}_\alpha|)$, and $|Z^{[L]}_{\alpha\alpha}\rangle_\sharp = |\Phi^{[L]}_\alpha\rangle\langle\Phi^{[L]}_\alpha|$ (For R, we take the same matrices, except $-|Y^{[R]}_{\alpha\alpha'}\rangle_\sharp$ instead of $|Y^{[R]}_{\alpha\alpha'}\rangle_\sharp$). This basis also diagonalizes the reduced superoperator, and now gives the Schmidt decomposition in terms of a local Hermitian basis of each subsystem. Since this is just a rotation of the previous basis, we still have the same set of Schmidt coefficients.

††The Schmidt coefficients are given as $\lambda_{\sharp[\alpha\alpha']} = \lambda_\alpha \lambda_{\alpha'}$. If there is only a few Schmidt coefficients for the wavefunction, this will still give many, densely packed, Schmidt coefficients for the density matrix. If the pure state Schmidt coefficients drop off exponentially, as has been observed [34], the Schmidt coefficients of the density matrix will show an exponential spread (with some degeneracy). And an effective $\chi_\sharp$ can be taken as $\sim \chi\Psi^2/2$ which will give similar precision. However, this is still a severe loss in efficiency.

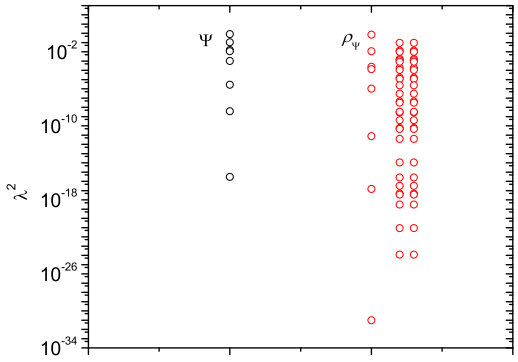


Figure 2.5: The eigenvalues of the reduced density matrix (left) and the reduced superoperator (right, with degenerate values shown in pairs in the rightmost two columns).

2.4 Conclusions

We have presented an extension of the TEBD algorithm to mixed states. With specific examples involving spins and non-interacting fermions, we have shown how to (*i*) construct thermal states; (*ii*) evolve a state in time according to a time-dependent master equation; and (*iii*) compute unequal time correlation functions. The algorithm can be used for generic one-dimensional lattice systems, including interacting fermions and bosons.

Chapter 3

Numerical ansatz for integro-differential equations

We present an efficient and stable numerical ansatz for solving a class of integro-differential equations. We define the class as integro-differential equations with *increasingly smooth* memory kernels. The resulting algorithm reduces the computational cost from the usual T^2 to $TC(T)$, where T is the total simulation time and $C(T)$ is some function. For instance, $C(T)$ is equal to $\ln T$ for polynomially decaying memory kernels. Due to the common occurrence of increasingly smooth memory kernels in physical, chemical, and biological systems, the algorithm can be applied in quite a wide variety of situations. We demonstrate the performance of the algorithm by examining two cases. First, we compare the algorithm to a typical numerical procedure for a simple integro-differential equation. Second, we solve the NIBA equations for the spin-boson model in real time.

3.1 Introduction

A major problem in the study of quantum systems is the influence of the environment on its dynamics. A variety of techniques, such as the influence functional[53], non-equilibrium perturbation theory[54,55], and the weak-coupling approximation[56], have been developed to address this issue. The general approach is to derive equations of motion which only involve the degrees of freedom of the system. When working with systems weakly coupled to a rapidly relaxing environment, one can make the Markov approximation, which results in a local in time, first-order differential equation. This Markovian master equation is relatively easy to solve either analytically or numerically. As soon as the restrictions of weak coupling or fast environmental relaxation are removed, however, integro-differential equations appear, which reflects the fact that the environment retains a memory of the system at past times.

In particular, a number of interesting physical systems have integro-differential equations of motion with polynomially decaying memory kernels. For instance, the spin-boson model[2] and

Josephson Junctions with quasi-particle dissipation or resistive shunts[13] have $1/t^2$ asymptotic kernels when going through a dissipatively driven phase transition. The occurrence of polynomially decaying kernels is common in physics and chemistry because many open systems are connected to an ohmic environment. Polynomially decaying kernels appear in biological applications as well.[57] Due to the frequent appearance of this type of kernel and also others which satisfy a requirement we call *increasingly smooth* below, it would be beneficial to have efficient numerical methods to handle the corresponding integro-differential equations. Such a method will be useful in applications in many disciplines. In addition, the method will enable the simulation of memory-dependent dissipation when used in conjunction with other computational methodologies, such as matrix product state algorithms for open systems.[58,59,33,34]

In this chapter, we give an efficient and stable numerical ansatz for solving integro-differential equations with increasingly smooth memory kernels. Using typical techniques, one incurs a computational cost of $AT^2 \propto N^2$, where A is some constant, T is the total simulation time, and N is the number of time steps in the simulation. Thus, there is a considerable advantage in using a high-order approach to reduce the required value of N as much as possible. However, the computational cost of the algorithm described herein scales as $TC\left(T\right)$, where $C\left(T\right)$ is some function that depends on the form of the memory kernel. For example, $C\left(T\right)$ is equal to $\ln T$ for polynomially decaying kernels. Such a reduction represents a substantial advancement in numerically solving integro-differential equations since it allows for the efficient calculation of long-time behavior.

This chapter is organized as follows: In 3.2, we introduce the types of equations under consideration, define *increasingly smooth*, and present the numerical ansatz with a discussion of errors. We demonstrate the performance of the algorithm using two example integro-differential equations in 3.3. First, we consider an integro-differential equation composed of a polynomially decaying kernel and an oscillating function. Comparing directly to a two-stage Runge-Kutta method, we see a large improvement in the scaling of the error as a function of computational cost. Second, we solve the noninteracting blip approximation (NIBA) equations for the spin-boson model in real time. We show that one can rapidly obtain the full real time solution. In 3.5.1, we discuss how to extend the algorithm to higher orders. In 3.5.2, we outline the Runge-Kutta method we use as a comparison to the numerical ansatz. In 3.5.3, we give a simple derivation of the NIBA equations, showing the physical situation behind the appearance of an increasingly smooth memory kernel.

3.2 Algorithm

We want to be able to numerically solve linear integro-differential equations of the form

$$\frac{\partial \rho(t)}{\partial t} = -\mathcal{K}\int_0^t dt' \alpha(t-t')\, e^{-\mathcal{L}(t-t')}\mathcal{K}'\rho(t') \tag{3.1}$$

$$= -\mathcal{K}\int_0^t d\tau \alpha(\tau)\, e^{-\mathcal{L}(\tau)}\mathcal{K}'\rho(t-\tau)\ , \tag{3.2}$$

where $\rho(t)$ is some quantity of interest (like a density matrix) and $\alpha(\tau)$ is the memory kernel. $\mathcal{K}$, $\mathcal{K}'$, and $\mathcal{L}$ are time-independent operators. An equation of the form 3.2 appears in open quantum mechanical systems in the Born approximation to the full master equation[56], some exact non-Markovian master equations[60], and in phenomenological memory kernel master equations.[61,62,63] For an arbitrary form of the memory kernel, it is necessary to compute the integral on the right-hand side of 3.2 at each time step in the numerical simulation. Thus, ignoring the error of the simulation, the computational cost scales as T^2, where T is the total simulation time. This is prohibitively expensive in all but the shortest time simulations.

On the opposite extreme is when the memory kernel has an exponential form

$$\alpha(t-t') = \gamma e^{-\gamma(t-t')}\,. \tag{3.3}$$

In this case, both functions responsible for evolving the integrand $(\alpha(t-t'), e^{\mathcal{L}(t-t')})$ obey simple differential equations in t. This allows us to define a "history"

$$H(t) = \int_0^t dt' \alpha(t-t') e^{\mathcal{L}(t-t')}\mathcal{K}'\rho(t')\,, \tag{3.4}$$

which obeys the differential equation

$$\dot{H}(t) = \gamma\mathcal{K}'\rho(t) - \gamma H(t) - \mathcal{L}H(t)\,. \tag{3.5}$$

By solving this local in time differential equation together with

$$\dot{\rho}(t) = -\mathcal{K}H(t)\,, \tag{3.6}$$

we can solve the integro-differential equation with a computational cost scaling as T. Cases in between these two extremes have a spectrum of computational costs scaling from T to T^2.

3.2.1 Increasing Smoothness

We are interested in integro-differential equations of the form 3.2 with memory kernels which are *increasingly smooth*

$$\frac{\partial}{\partial t}\left|\frac{\alpha'(t)}{\alpha(t)}\right| < 0\,. \tag{3.7}$$

The idea behind increasing smoothness will become clear below. More specifically in this paper, we will look at examples where the memory kernel decays polynomially outside some cut-off time

$$\alpha(t) = \begin{cases} \alpha_{\Delta T}(t) & t < \Delta T \\ \propto \frac{1}{t^p} & t \geq \Delta T \end{cases}, \tag{3.8}$$

where $\alpha_{\Delta T}(t)$ is a bounded but otherwise arbitrary function and p is some positive real number.* ΔT is a cut-off which allows us to ignore any transient behavior in the memory kernel which does not satisfy the increasingly smooth condition 3.7. There will generally be some natural cut-off to the integral. For the overall scaling of the computational cost for large simulation times, we can ignore the resulting problem-dependent, constant computational cost due to the integral at times $t < \Delta T$.

Below we will give an algorithm that is specific to the polynomial kernel 3.8. Similar algorithms are possible for any kernel which gets smoother as the time argument gets larger. The computational speedup will depend on how the function gets smoother. For instance, the polynomially decaying kernel gets smoother as

$$\left|\frac{\frac{\partial 1/t^p}{\partial t}}{1/t^p}\right| = \frac{p}{t}\,, \tag{3.9}$$

which allows one to take larger integration blocks as t gets larger. In this case, one can take a grid with spacing $\delta \propto t$ to cover the integration region. We call the partition around a grid point a *block*. In this case, the number of blocks scales logarithmically with the total time of the simulation. Consider another form of the kernel, $\beta(t)$, that gets smoother even faster than polynomial, for example

$$\left|\frac{\beta'(t)}{\beta(t)}\right| = be^{-ct}\,. \tag{3.10}$$

This will give an even smaller number of blocks (that have exponential size in t) needed to cover a given range of integration and thus a faster speedup.† In this case, though, a simple truncation of the integral is sufficient, which is not the case for a polynomially decaying kernel.

*For the algorithm, p need not be greater than one. However, physical equations of motion, such as the NIBA equations at $\alpha = 1/2$, have an additional factor in the kernel to ensure its integral is bounded.

†3.10 has two solutions. Only the one with bounded integral would be physically relevant.

3.2.2 Blocking algorithm for a polynomially decaying kernel

The algorithm is designed to take advantage of the fact that at each new time step in solving the integro-differential equation 3.2, we almost already have the integral on the right hand side. Thus, if we have $\rho(t)$ at all the previous time steps and we group them together in a particular way, we should be able to do away with both storing all these previous values and the need for a full integration of the right hand side. In more concrete terms, we want to be able to evaluate the history integral

$$I(T, \Delta T) = \int_{\Delta T}^{T} d\tau \alpha(\tau) F(\tau, T) , \tag{3.11}$$

with

$$F(\tau, T) = e^{\mathcal{L}\tau} \mathcal{K}' \rho(T - \tau) , \tag{3.12}$$

in such a way that when $T \to T + h$ in the integro-differential equation 3.2, all the pieces used to compute the integral can be recycled by evolving for a time step h and then added back together to get the new integral. This is just what is achieved for the exponential memory kernel 3.4, but in that case it is very extreme, the old integral simply has to be multiplied by a factor and added to a new piece. Below we show how to evaluate the integral with a blocking scheme that requires only $\ln T$ number of blocks to cover the integration region, can achieve any desired accuracy, and has blocks that can all be recycled when $T \to T + h$. Again, we emphasize that the scheme we give now is specific for polynomially decaying memory kernels. The basic scheme can be constructed for other memory kernels as well simply by choosing a block size such that the block size times the smoothness stays constant.

Lets first rewrite the integral 3.11 as a sum over K blocks of width δ_i and Taylor expand the memory kernel:

$$\begin{aligned} I(T, \Delta T) &= \sum_{i=1}^{K} \int_{-\delta_i/2}^{+\delta_i/2} d\epsilon F(\tau_i + \epsilon, T) \alpha(\tau_i + \epsilon) && (3.13) \\ &\approx \sum_{i=1}^{K} \int_{-\delta_i/2}^{+\delta_i/2} d\epsilon F(\tau_i + \epsilon, T) \\ &\quad \times \left\{ \alpha(\tau_i) + \alpha'(\tau_i) \epsilon + \mathcal{O}\left(\epsilon^2\right) \right\} . && (3.14) \end{aligned}$$

The lowest order approximation to this integral is

$$I^0(T, \Delta T) = \sum_{i=1}^{K} \alpha(\tau_i) \int_{-\delta_i/2}^{+\delta_i/2} d\epsilon F(\tau_i + \epsilon, T) . \tag{3.15}$$

This equation represents the fact that we will take some step size, h, to compute the integral of F over the block, but use some other variable step size to calculate the integral of the product of α

and F. For the polynomial kernel, the whole procedure is based on choosing a block size that grows with increasing distance from the current time,

$$\delta_i = b\tau_i \,, \tag{3.16}$$

or close to it, as shown in Figure 3.1. We call b the *block parameter*. This function for the block size gives a constant value when multiplied by the smoothness 3.9 of the kernel. Assuming the function F is bounded by 1, the highest-order error on each block is bounded by

$$\left| \alpha'(\tau_i) \int_{-\delta_i/2}^{+\delta_i/2} d\epsilon F(\tau_i + \epsilon, T)\, \epsilon \right| \leq \frac{p}{4} \frac{b^2}{\tau_i^{p-1}} \,. \tag{3.17}$$

Thus, when choosing the blocks in this way, there is a constant ratio, $pb/4$, of the bound on the error to the bound on the integral. The bound on the highest-order error of the whole integral is

$$\left| \sum_{i=1}^{K} \alpha'(\tau_i) \int_{-\delta_i/2}^{+\delta_i/2} d\epsilon F(\tau_i + \epsilon, T)\, \epsilon \right| \leq \frac{pb}{4(p-1)} \frac{1}{\Delta T^{p-1}} \,. \tag{3.18}$$

Roughly, we then expect the error to decrease at least linearly in the block parameter b. We discuss the errors in more detail below.

If we perfectly choose the block positions, we can calculate the number of blocks required to cover the integration region. The first block has position (its midpoint)

$$t_1 = \Delta T + \frac{b}{2} t_1 = \frac{\Delta T}{1 - \frac{b}{2}} \,, \tag{3.19}$$

and given the position of block i, block $i+1$ is at

$$t_{i+1} = t_i + \frac{b}{2} t_i + \frac{b}{2} t_{i+1} = \frac{1 + \frac{b}{2}}{1 - \frac{b}{2}} t_i \,. \tag{3.20}$$

The n^{th} block is then at

$$t_n = \frac{\left(1 + \frac{b}{2}\right)^{n-1}}{\left(1 - \frac{b}{2}\right)^{n}} \Delta T \,. \tag{3.21}$$

Since T is arbitrary, the n^{th} block does not necessarily cover the remaining area. But approximately,

$$t_n \;\approx\; T - \frac{b}{2} T + \mathcal{O}\left(b^2\right) \,, \tag{3.22}$$

and the total number of blocks for the integral is

$$K = \lceil n \rceil = \left\lceil \frac{\ln\left(\frac{\Delta T}{T}\right)}{\ln\left(1 - \frac{b}{2}\right) - \ln\left(1 + \frac{b}{2}\right)} \right\rceil \,. \tag{3.23}$$

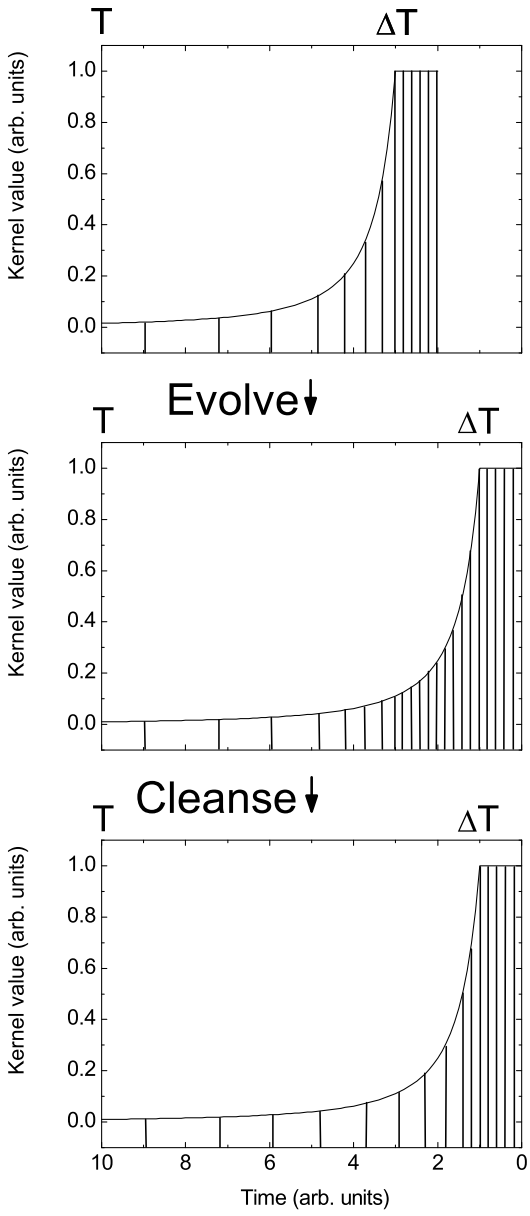


Figure 3.1: Procedure for evolving the integral 3.11 in time. We start with the integral represented as a sum over variable size blocks, with short times having constant block sizes and longer times having larger block sizes. We evolve for some amount of time (here shown as a large amount of time, but in practice this whole procedure is done discretely in the small time steps of the differential equation). Then we cleanse the blocks, grouping them together so long as their size is not too big for their location in time.

In practice, K will always be bigger than 3.23 because of two factors. One is the finite step size, h, which forces the blocks to be an integral multiple of h. Two is that we are not able to take the complete integral and divide it up perfectly. The division into blocks has to happen as we solve the integro-differential equation. We will see in 3.2.3 that neither of these factors poses a serious problem and the true value of K will be of the same order as 3.23. For small b (or large K), we can simplify to

$$K \approx \frac{1}{b} \ln\left(\frac{T}{\Delta T}\right) . \tag{3.24}$$

Within the algorithm, then, we need to keep track of the K integrals

$$I_i = \int_{-\delta_i/2}^{+\delta_i/2} d\epsilon F\left(\tau_i + \epsilon, T\right) \ , \ \delta_i = b\tau_i \, , \tag{3.25}$$

which can be summed up with the weight $\alpha\left(\tau_i\right)$ to get the full integral. Putting in the explicit form of F, the K integrals are

$$\int_{-\delta_i/2}^{+\delta_i/2} d\epsilon e^{-\mathcal{L}(\tau_i+\epsilon)} \mathcal{K}' \rho(T - \tau_i - \epsilon) \, . \tag{3.26}$$

When we increment $T \to T + h$, we first need to fix the block size, $\delta_i \to B_i$. Thus, the blocks will no longer be exactly $b\tau_i$ in width. As $T \to T + h$, $\tau_i \to \tau_i + h$ and the integrals are easily updated by

$$\begin{aligned} &\int_{-B_i/2}^{+B_i/2} d\epsilon F\left(\tau_i + \epsilon, T\right) \\ &\to e^{-\mathcal{L}(h)} \left\{ \int_{-B_i/2}^{+B_i/2} d\epsilon F\left(\tau_i + \epsilon, T\right) \right\} \quad . \end{aligned} \tag{3.27}$$

After evolving, $B_i < b\tau_i$, which is acceptable since the smaller the blocks the smaller the error. The block sizes have to grow eventually or else we will not get the logarithmic coverage. Each time we increment T we can check whether nearest neighbor blocks can be grouped. We can group them so long as

$$B^{new} \leq b\tau^{new} \tag{3.28}$$

where τ^{new} is the midpoint of the new block. This is the "cleansing" stage of Figure 3.1, and is discussed in more detail in 3.2.3. When the condition on the new block size is met, we can simply add the blocks together

$$\begin{aligned} &\int_{-B^{new}/2}^{+B^{new}/2} d\epsilon F\left(\tau^{new} + \epsilon, T\right) \\ &= \int_{-B_i/2}^{+B_i/2} d\epsilon F\left(\tau_i + \epsilon, T\right) + \int_{-B_{i+1}/2}^{+B_{i+1}/2} d\epsilon F\left(\tau_{i+1} + \epsilon, T\right) \, . \end{aligned} \tag{3.29}$$

To summarize, we defined a logarithmic covering of the integration region in terms of growing

blocks. As we solve the integro-differential equation, we can evolve the blocks and perform further groupings to ensure approximately the same covering. This covering allows for a logarithmic speedup in computing the history and a logarithmic reduction in the memory necessary to store the history. At the same time, due to the increasing smoothness of the polynomial memory kernel, it has a controllable error. The algorithm can be extended to higher orders as shown in 3.5.1.

3.2.3 Growing blocks in real time

The above description of the blocking algorithm presented the idea of using a logarithmic covering of the history integral. The formula 3.23 gives the number of blocks needed for the covering if we could perfectly choose their positions and widths. However, when solving the integro-differential equation, growing pains exist: the blocks have to be formed from the smaller blocks of the integration step size h. The block covering will thus be suboptimal.

To determine how close to optimal the number of blocks will be, we perform a simulation. Here we choose a polynomially decaying function but with a more natural cut-off, i.e.,

$$\alpha(t) = \frac{1}{(t+1)^p}\,. \tag{3.30}$$

This will be fully equivalent to $1/(t')^p$ with a cut-off $\Delta T = 1$ and a shifted time $t' = t+1$. Because of this, the optimal formula 3.23 will only hold for large T (or with $\Delta T = 1$ and T replaced with $T+1$). Within the simulation, we start at $T = 0$ and create blocks of size h. At each step we check if neighboring blocks can be grouped by satisfying the condition 3.28. If they satisfy that condition, we group them and check the again with the next block. We perform this grouping from the smallest to the largest blocks, but the directionality of growth does not matter. Figure 3.2 shows how the number of blocks grow in practice. Although suboptimal, it is still on the same order as the optimal K. How close it is to optimal is dependent on the step size and block parameter.

3.2.4 Errors

Given the blocking construction above, we can analyze how we expect the errors to behave versus both the number of blocks K and the computational cost. The first question that is natural to ask is what error do we make by the replacement of the integral 3.11 with the approximation 3.15. This gives us an idea of the error incurred by keeping K blocks of history in a logarithmic covering.

To answer this first we consider just the integral

$$\int_0^{T_f} \frac{e^{\imath\omega t}}{(t+1)^p} dt = \int_{\Delta T=1}^{T_f+1} \frac{e^{\imath\omega(t-1)}}{t^p} dt\,, \tag{3.31}$$

where we have some frequency of oscillation ω and some power of the polynomial decay p. In the

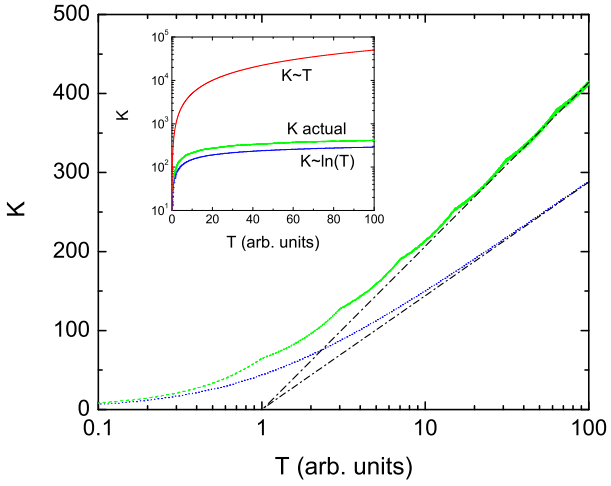


Figure 3.2: The number of blocks K versus T. The dotted (blue) curve shows the optimal case given by 3.23. The dashed (green) curve shows the number of blocks in practice. The dot-dashed curves represent the asymptotic behavior of the number of blocks versus time. These curves intersect the axis at $T = 1$ because of the cut-off $\Delta T = 1$. The inset shows the two latter quantities plotted together with the linear growth of K for a typical numerical solution. The parameters in the evolution are the following: $\Delta T = 1$, $d = 0.016$, and $h = 0.002$.

integration below we take $p = 2$. It is important to note that in demonstrating the algorithm with just the integral 3.31, and not an integro-differential equation, it is not possible to show the computational speedup. However, it can show the accuracy of the integration as a function of the logarithmic coverage of the integration region when using the algorithm. We can also use it to get an idea of how the error changes when we vary ω or p without having to solve the more complicated integro-differential equations. The form of this integral was chosen because of its relevance to the real-time simulation of quantum systems, where the integral of the integro-differential equation will be a sum over oscillating terms times a polynomial memory kernel.

We gave a simple error bound above. To get a better idea of errors, we can examine the behavior of the integral 3.31 as a function of the blocking parameter, frequency, and the integration range (e.g., simulation time). If one takes the highest order error of the expansion 3.14, the worse case error in a block will occur when there is approximately one oscillation in the block,

$$b\tau_i = \frac{2\pi}{\omega}\,, \tag{3.32}$$

as this is when the asymmetry contributes most. However, this contribution will only occur in some finite region of the integral because the block sizes vary. When one changes the block parameter, this error will be shifted further back in time, e.g., to the left in Figure 3.1. The memory kernel has a smaller value at this past time, and thus the error will be decreased. If one has many frequencies contributing to the integrand, then each of their error contributions will be shifted to a later time

as one decreases the block parameter. Thus, the error will be systematically reduced. The same conclusion can be drawn if one expands both the functions in the integrand around their midpoint for each block. The fractional error (the highest-order error term as a fraction of the zeroth order term) in this case is

$$\frac{b^2}{24}\left(p(p+1) - 2\imath\omega p\tau_i - \omega^2\tau_i^2\right) . \tag{3.33}$$

The expansion will be invalid at approximately

$$\frac{\omega^2 b^2 \tau_i^2}{24} \approx 1 , \tag{3.34}$$

which gives $b\tau_i \sim 5/\omega$.

We compare the logarithmic covering to the midpoint method with step sizes $h = 10^{-3} * 2^j$ with $j = 1, \ldots, 9$. This gives a number of blocks $K_{mid} = T_f/h$. For the algorithm, we use a step size $h = 10^{-3}$ to compute the integrals 3.25 and then use block parameters $b = 10^{-3} * 2^j$ with $j = 1, \ldots, 8$ to compute 3.15. The number of blocks K_{alg} is then given approximately by 3.23. It is only approximate because each block width has to be rounded down to a multiple of h; this gives a larger number of blocks. We use this larger number of blocks as the value for K_{alg}.

If we are just looking at the integral 3.31, we expect the error due to the algorithm to be

$$E_{alg} \propto K_{alg} b^3 \approx \frac{1}{K_{alg}^2}\left(\ln\frac{T_f}{\Delta T}\right)^3 , \tag{3.35}$$

so long as $bT \lessapprox 2\pi/\omega$. This can be compared to the error of the midpoint method,

$$E_{mid} \propto K_{mid} h^3 \approx \frac{T_f^3}{K_{mid}^2} . \tag{3.36}$$

Thus, if we want a fixed error, E, and end time, we get a difference in the number of blocks of

$$K_{mid} \propto \left(\frac{T_f^3}{E}\right)^{1/2} \tag{3.37}$$

compared to

$$K_{alg} \propto \left(\frac{\left(\ln\frac{T_f}{\Delta T}\right)^3}{E}\right)^{1/2} , \tag{3.38}$$

which is quite large for long times.

Figure 3.3 shows how the error behaves versus the number of blocks used to compute the integral. The fast decrease of error as a function of K_{alg} shows that we can block together integrals as we evolve the integro-differential equation and have a controlled error. Further, the longer the simulation

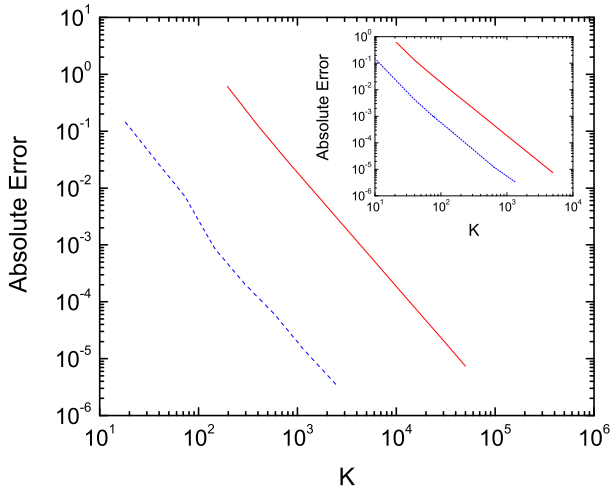


Figure 3.3: Absolute error versus number of blocks for the two methods of calculating the integral 3.31 with $\omega = 2\pi$ and $p = 2$ (similar curves were found for other ω). The solid (red) curves are for the midpoint method and the dashed (blue) curves are for the first order blocking algorithm. The main figure shows the error for $T_f = 100$ and the inset shows it for $T_f = 10$. Since the midpoint method is second order, we find what is expected, $E \propto K^{-2}$. The blocking algorithm has the same dependence on K. The difference in error for the same number of blocks with the two methods is dependent on $T/\ln T$, reflecting the fact that the algorithm chooses blocks at optimal places and only covers the integration area with $\ln T$ blocks. We take as the exact value the one computed with a step size $h = 10^{-4}$.

time, T_f, the better performing the algorithm should be when compared to the midpoint method. In addition, the overall performance of the algorithm should not be significantly affected by the frequency of integrand oscillation ω or the power p. One interesting feature of the error versus K_{alg} figure is the slight dip in the error at approximately $bT \sim 2\pi/\omega$, which represents the error discussed above being pushed out of the integration region.

The second question we can ask is how the error depends on the computational cost compared to more traditional methods. If we use the method for integro-differential equations outlined in 3.5.2, we have an error at each step of h^3 and we have N steps. Thus

$$E_{mid} \propto Nh^3 \propto \frac{1}{C_{pu}}\,, \tag{3.39}$$

where $C_{pu} \propto N^2$ is the computational cost and we hold the simulation time fixed. For the algorithm

$$E_{alg} \propto \frac{1}{C_{pu}^2} \tag{3.40}$$

where we hold the step size h fixed and $C_{pu} = NK$. That is, the error goes down with a higher power in the computational cost.

Of course, using the algorithm is not the only way to get the error to scale better with the

computational cost. One can also just block the history with constant blocks larger than h. Although the error does scale better, the error can never be lower than just using the brute force method, e.g., the error versus cost curve will never cross the brute force curve. For the algorithm these two curves do cross, as we will see below.

3.3 Examples

We consider two example problems to demonstrate the performance of the algorithm: *(1)* solving an integro differential equation that has both an oscillating function and a polynomially decaying function in the integral and *(2)* the NIBA equations for the spin-boson model.

3.3.1 Oscillating integro-differential equation

The oscillating integro-differential equation we use is

$$\dot{n}(t) = \imath\omega n(t) - \int_0^t dt' \alpha(t-t') n(t') \tag{3.41}$$

with

$$\alpha(t) = \frac{1}{(t+1)^2} . \tag{3.42}$$

We get rid of the oscillation in the differential equation by setting

$$P(t) = e^{-\imath\omega t} n(t) , \tag{3.43}$$

which gives

$$\dot{P}(t) = -\int_0^t dt' \alpha(t-t') e^{-\imath\omega(t-t')} P(t') . \tag{3.44}$$

This integro-differential equation involves the computation of an integral similar to 3.31, with $p = 2$, at each time step. With the algorithm, we will be able to recycle the integral with the logarithmic covering at each step, rather than computing it from scratch. Note that we choose blocks by $b(t+1)$ here because we have the cut-off included in the polynomial decay. The simulation results for one set of parameters are shown in Figure 3.4.

Now we examine the error versus computational cost. Figure 3.5 is one of the main results of this paper. It shows three key results: (1) The algorithm has significantly better error for the same computational cost compared to a brute force method in a large range of parameter space. (2) As the simulation time is increased, the efficiency of the algorithm drastically increases compared to the brute force method. (3) The figure also suggests how to use the whole ansatz. Rather than have two parameters b and h, one should set $b \propto h$ (for polynomial decay). This will give a similar

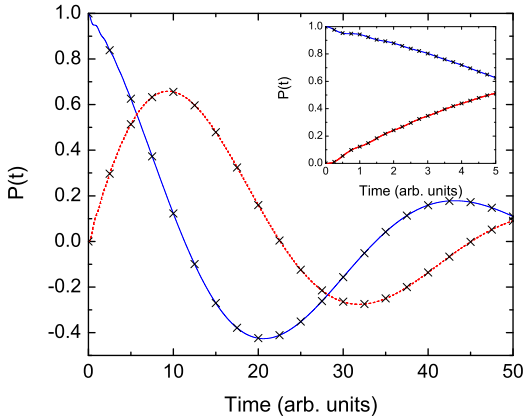


Figure 3.4: Real time evolution of 3.44 with $\omega = 2\pi$. The solid (blue) curve shows the real part of $P(t)$ and the dashed (red) curve the imaginary part. These curves were calculate with the algorithm with $h = 0.002$ and $d = 0.016$. The crosses represent the solution using the midpoint procedure of 3.5.2 with a step size of $h = 0.001$. The inset shows a blow up of the initial time where there is some transient structure in the solution.

convergence to the exact answer as the second order method, but a smaller factor of proportionality.

As mentioned in 3.2.4, one can also consider another method that just uses a constant grouping to reduce the number of blocks in the history. However, this method can not achieve better results as determined by the error versus computational cost. For a given step size, the highest accuracy will be achieved by using the brute procedure. A lower computational cost can be achieved by grouping the history into constant blocks, but the reduction in the computational cost will yield an even stronger loss in accuracy because $E \propto C^{-2}$, when one step size is fixed. Thus, a similar figure to Figure 3.5 would show curves moving only upward from the brute force line and not crossing it.

3.3.2 NIBA equations

The NIBA for the spin-boson model gives a good physical example for a system with a polynomially decaying kernel. The NIBA equations are derived by Leggett *et al.*[2] (see also 3.5.3 for a simple derivation) from which we can get the equation of motion

$$\dot{P}(t) = -\int_0^t dt' f(t-t')P(t')\,, \tag{3.45}$$

where

$$f(t) = \Delta^2 \cos\left[2\alpha \tan^{-1} t\right] \left[\frac{(\pi t/\beta\omega_c)\,\mathrm{csch}\,(\pi t/\beta\omega_c)}{(1+t^2)^{1/2}}\right]^{2\alpha}\,. \tag{3.46}$$

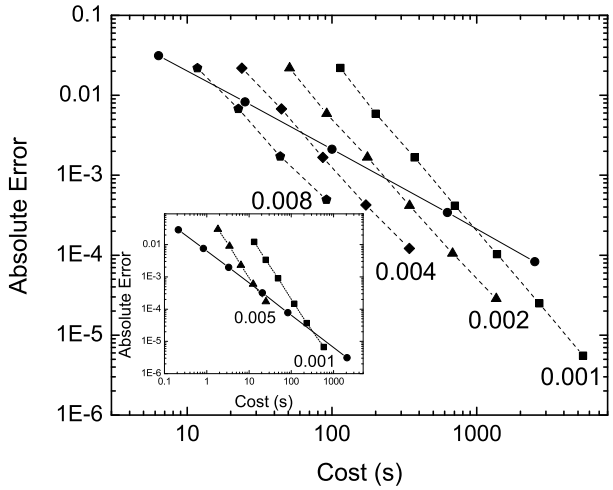


Figure 3.5: Absolute error versus computational cost. The error is computed by comparing the simulation results with those the brute force method with $h = 0.001$ for $T_f = 50$ (in the inset, the "exact" solution is taken as the brute method with $h = 0.0002$ and $T_f = 10$). Circles represent the brute force method for step sizes 0.1, 0.05, 0.025, 0.01, and 0.005 (in the inset, 0.1, 0.05, 0.025, 0.01, 0.005, and 0.001). The other data represents using the algorithm with the step size as shown and each data point has a different block parameter starting with $2h$ and increasing by factors of 2. For the longer simulation time of $T_f = 50$, the algorithm has quite a substantial decrease in error for the same computational cost (for the shorter time in the inset, one can see that there is barely a gain. This is due to the overhead cost in implementing the algorithm). Also, the curves have the behavior discussed in 3.2.4. We perform all computations on a P4, 3 GHz processor.

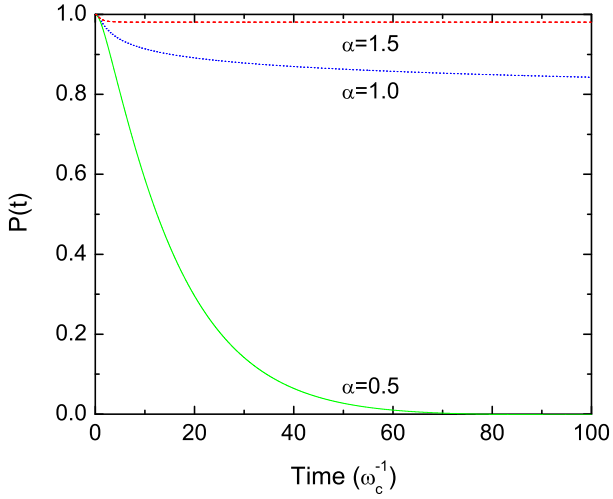


Figure 3.6: NIBA equations simulated in real time for $\alpha = 0.5$, 1.0, and 1.5. The simulations were done with $h = 0.008$ and $d = 0.016$. They took approximately two minutes of CPU time on a P4, 3 GHz processor. This is orders of magnitude faster than a brute force approach would take to get the same error. From the figure we can clearly observe the physics of the different regimes. For weak dissipation, one gets decay to $P = 0$. For the transition, $\alpha = 1$, one gets algebraic decay of $P(t)$. Above the transition, $P(t)$ gets locked into a nonzero value.

Time is in units of the cut-off frequency ω_c^{-1}. At zero temperature ($\beta \to \infty$) the kernel becomes

$$f(t) = \frac{\Delta^2 \cos\left[2\alpha \tan^{-1} t\right]}{(1+t^2)^{\alpha}} \,. \tag{3.47}$$

The $1 + t^2$ has a physical cut-off to the low time behavior of the polynomial decay. There are only two parameters of interest to us. We set $\Delta = 0.2$ as it is required to be small (see 3.5.3) and we vary the dissipation strength α to see the behavior of $P(t)$ on different sides of the dissipative transition. Varying α also shows the algorithm at work for different powers of polynomial decay.

Depending on the value of the dissipation strength α, we have to use a different cut-off ΔT. For $\alpha = 0.5$, 1.0, and 1.5 we use $\Delta T = 1$, 3, and 2, respectively. After these cut-offs, $f(t)$ is increasingly smooth (it is also smoother than bare polynomial decay). We want to point out that due to the form of the polynomial decay, the most efficient way to choose block sizes is by $b\left(1+t^2\right)/t$, which just comes from the inverse of the increasing smoothness. Despite this, we still use the less efficient bt.

The simulation results for the NIBA equations are shown in Figure 3.6. Using the algorithm allows one to have long simulation times for negligible computational cost. Further, simulations of NIBA-like equations on a lattice or with time dependent Hamiltonians can be done efficiently with the method.

We want to emphasize that the simulation of the NIBA equations at finite temperature 3.46

can also use the algorithm presented in this paper. The finite temperature, however, introduces a long-time cut-off into the kernel, beyond which the kernel may not be increasingly smooth. The contribution beyond this cut-off, though, is negligible. Thus, the algorithm can be used for times less than $\sim \beta$ exactly how it is used for the zero temperature case, and there can be a truncation of the kernel at times beyond times of $\sim \beta$.

3.4 Conclusions

We have given an efficient and stable numerical algorithm for solving integro-differential equations with increasingly smooth memory kernels. The general computational speedup is $T^2 \to TC(T)$, where $C(T)$ depends on how fast the kernel gets smooth. For example, the computational cost of the algorithm for polynomially decaying kernels is $T \ln T$ rather than the usual T^2.

Using a simple integro-differential equation, we demonstrated how well the algorithm performs compared to a second-order, constant step size method. For long times, there is quite a substantial speedup in the computation cost to achieve a given error. The solution to the NIBA equations for the spin-boson model in real-time showed that one can get results and analyze the situation quite rapidly. Similar procedures can be applied to other forms of memory kernels which satisfy the increasingly smooth condition 3.7. In these other cases, the computational speedup can be better or worse than the case presented here.

In practice, the usefulness of this algorithm is due to the possibility of its incorporation into other techniques, such as matrix product state algorithms, or its use with time-dependent Hamiltonian terms. Thus, one can simulate lattices or driven systems subject to strong dissipation. The algorithm can also be used for equations less restrictive than 3.2, such as integro-differential equations with memory kernels dependent on both time arguments.

3.5 Subsidiary Calculations

3.5.1 Higher-order blocking algorithms

The algorithm above can be extended to higher orders. For instance, if we want the next order method, we need to be able to evaluate and update

$$\begin{aligned} & I^1(T, \Delta T) \\ & = I^0(T, \Delta T) + \sum_{i=1}^{K} \alpha'(\tau_i) \int_{-\delta_i/2}^{+\delta_i/2} d\epsilon F(\tau_i + \epsilon, T)\, \epsilon\,. \end{aligned} \tag{3.48}$$

The latter term picks out the asymmetric part of F inside of block i and then multiplies it by the derivative of α. Lets define this asymmetric part of the block as

$$A_i(T) \equiv \int_{-\delta_i/2}^{+\delta_i/2} d\epsilon F(\tau_i + \epsilon, T)\,\epsilon\,. \tag{3.49}$$

If we suppose we have these asymmetric integrals at a given time, and we want to increment T, we first need to fix the block size, as before. Then we can update them by

$$A_i(T+h) = e^{-\mathcal{L}(h)} A_i(T)\ . \tag{3.50}$$

We also need to be able to add these blocks together. We can do this by

$$\begin{aligned} A^{new}(T) &= \int_{-B^{new}/2}^{+B^{new}/2} d\epsilon F(\tau^{new} + \epsilon, T)\,\epsilon \\ &= A_i(T) - B_{i+1} I_i(T) + A_{i+1}(T) + B_i I_{i+1}(T)\ , \end{aligned} \tag{3.51}$$

where we use the asymmetric integrals from before we did the blocking and also the first-order integrals 3.25 for the two blocks. The error for an order z algorithm will have a bound proportional to b^z.

3.5.2 Two-stage Runge-Kutta method

In this article, we compare the proposed numerical ansatz to a two-stage Runge-Kutta method. Since we are dealing with integro-differential equations, we give the details of the second order technique we use. In the main text, we discuss how the errors scale with total simulation time, step size, and computational cost.

For all the examples in this work, the basic integro-differential equation 3.2, is reduced to

$$\frac{\partial \rho(t)}{\partial t} = \int_0^t dt' \alpha(t-t')\, e^{-\mathcal{L}(t-t')} \rho(t')\ , \tag{3.52}$$

e.g., $\mathcal{K}, \mathcal{K}' = \mathrm{I}$. In a generic form we can write

$$\frac{\partial \rho(t)}{\partial t} = f\left[t, \rho(t), \int_0^t dt' \mathcal{F}(t-t')\,\rho(t')\right]\ . \tag{3.53}$$

Discretizing time as

$$t_n = t_0 + nh\,, \tag{3.54}$$

we can write a general two-stage Runge-Kutta integration scheme

$$\rho_{n+1} = \rho_n + h\, d\rho_n \,, \tag{3.55}$$

$$d\rho_n = f\left[t, P_n, \tilde{z}_n\right] \,, \tag{3.56}$$

$$P_n = \rho_n + dP_n \,, \tag{3.57}$$

$$dP_n = \frac{h}{2} f\left[t, \rho_n, z_n\right] \,, \tag{3.58}$$

$$z_n = h \sum_{m=1}^{n-1} \bar{\mathcal{F}}_{nm} \left(\rho_m + \rho_{m+1}\right)/2 \,, \tag{3.59}$$

and

$$\begin{aligned} \tilde{z}_n &= h \sum_{m=1}^{n-1} \bar{\mathcal{F}}_{nm} \left(\rho_m + \rho_{m+1}\right)/2 \\ &\quad + \frac{h}{2} \bar{\mathcal{F}}_0 \left(\rho_n + P_n\right)/2 \,, \end{aligned} \tag{3.60}$$

where

$$\bar{\mathcal{F}}_{nm} = \frac{1}{h} \langle \mathcal{F}\left(t_n - t'\right) \rangle_{t_m}^{t_m+h} = \frac{1}{h} \int_{t_m}^{t_m+h} dt' \mathcal{F}\left(t_n - t'\right) \,, \tag{3.61}$$

and

$$\bar{\mathcal{F}}_0 = \frac{2}{h} \int_{t_n - h/2}^{t_n} dt' \mathcal{F}\left(t_n - t'\right) \,. \tag{3.62}$$

Although using the average $\bar{\mathcal{F}}_{nm}$ over an interval does not increase the order of the method, it does preserve important properties of the kernel such as its total integration. This is very important in cases where the kernel integrates to zero and thus the transient behavior completely determines the steady state. We use the average of the kernel over each block with the algorithm as well. The Runge-Kutta scheme can of course be generalized to higher stages and to kernels with two time arguments.[64,65,66,67]

To test the basic two-stage method, we solve the integro-differential equation

$$\dot{P}\left(t\right) = -\int_0^t dt' \alpha\left(t - t'\right) e^{-\imath\omega\left(t-t'\right)} P\left(t'\right) \,, \tag{3.63}$$

with

$$\alpha\left(t\right) = \gamma e^{-\gamma t} \,. \tag{3.64}$$

We can write this as

$$\dot{P}(t) = -\int_0^t dt' \gamma e^{-t'(\gamma+\imath\omega)} P(t-t') \tag{3.65}$$

$$= -\gamma e^{-t(\gamma+\imath\omega)} \star P(t) \ , \tag{3.66}$$

where $\star$ stands for the convolution. We can solve this analytically by Laplace transform.[‡] Transforming 3.66, one gets

$$s\tilde{P}(s) - P(0) = -\gamma L\left[e^{-t(\gamma+\imath\omega)}\right]\tilde{P}(s) \tag{3.67}$$

$$= \frac{-\gamma}{s+\gamma+\imath\omega}\tilde{P}(s) \ , \tag{3.68}$$

when $\mathrm{Re}s > -\mathrm{Re}(\gamma+\imath\omega)$. This gives

$$\tilde{P}(s) = \frac{s+\gamma+\imath\omega}{s^2+(\gamma+\imath\omega)s+\gamma}P(0) \ , \tag{3.69}$$

which can take the inverse Laplace transform

$$P_{ex}(t) = \frac{P(0)}{2\pi\imath}\int_{R-i\infty}^{R+i\infty} \frac{s+\imath\omega+\gamma}{(s-s_+)(s-s_-)}e^{st}ds \, , \tag{3.70}$$

with $R > \mathrm{Re}s_\pm$, to get

$$P_{ex}(t) = P(0)\left[\frac{s_+ +\gamma+\imath\omega}{s_+ - s_-}e^{s_+ t} - \frac{s_- +\gamma+\imath\omega}{s_+ - s_-}e^{s_- t}\right] \ , \tag{3.71}$$

where

$$s_\pm = \frac{1}{2}\left[-\gamma-\imath\omega \pm \sqrt{(\imath\omega+\gamma)^2 - 4\gamma}\right] \ . \tag{3.72}$$

In Figure 3.7, we plot the real and imaginary parts of the numeric solution of 3.66 versus the analytical solution 3.71. We expect the overall error to be proportional to h^2 as with the two-state Runge-Kutta for differential equations. In the inset to Figure 3.7, we plot the error of the numerical solution versus h^2.

3.5.3 Simple derivation of the NIBA equations

In this section we give a simple derivation of the NIBA equations for the spin-boson model to show how polynomially decaying memory kernels can arise physically in the case of strong dissipation. The derivation of the NIBA equations is based on the observation that they come from a Born approximation of a transformed Hamiltonian.[68]

[‡]The equation can also be solved easily numerically as shown in equations 3.5 and 3.6.

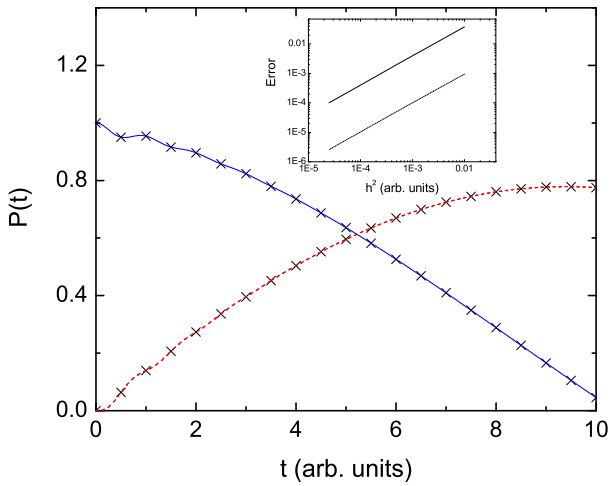


Figure 3.7: Real and imaginary parts of $P(t)$. The solid (blue) curve represents the real part and the dashed (red) curve represents the imaginary part. The crosses represent the real and imaginary parts of the exact solution. The parameters of the simulation are $\omega = 2\pi$, $\gamma = 1$, and $h = 0.005$. The inset shows the absolute error $|P(T_f) - P_{ex}(T_f)|$ versus step size h^2. The dashed error curve is for including the oscillation as part of the kernel, thus using its average over each step rather than its midpoint, which substantially reduces the error but does not change the order of the method.

The spin-boson Hamiltonian is

$$H_{SB} = -\frac{1}{2}\Delta\sigma_x + \frac{1}{2}\epsilon\sigma_z + \sum_k \omega_k a_k^\dagger a_k + \gamma\sigma_z \sum_k g_k \left(a_k^\dagger + a_k\right) , \tag{3.73}$$

where we have a two level system with internal coupling constant Δ and bias ϵ. The two level system is coupled to an collection of bosons of frequencies $\{\omega_k\}$ with a coupling constant $g_k = c_k/\sqrt{2m_k\omega_k}$ and overall coupling factor γ. The spectral density of the bath is given by

$$J(\omega) = \pi \sum_k g_k^2 \delta(\omega - \omega_k) . \tag{3.74}$$

The physical scenario we want to explore is one in which the two level system is initially held fixed in an eigenstate of σ_z while the bath equilibrates around this state. That is, the bath will equilibrate around the Hamiltonian

$$H_R = \sum_k \omega_k a_k^\dagger a_k + \sum_k \gamma g_k \left(a_k^\dagger + a_k\right) , \tag{3.75}$$

if we hold the two level system in the $+1$ eigenstate of σ_z. This gives the thermal starting state for the bath

$$R_0 = \frac{e^{-\beta H_R}}{Z_R} . \tag{3.76}$$

Then, we want to release the two level system and follow its dynamics in real time. In particular,

we look at the expectation value of σ_z

$$P(t) \equiv \langle \sigma_z \rangle_t \tag{3.77}$$

$$= \mathrm{tr}\left\{ e^{\imath H_{SB} t} \sigma_z e^{-\imath H_{SB} t} |1\rangle\langle 1| \otimes R_0 \right\} . \tag{3.78}$$

Since we are interested in strong dissipation, we can perform a canonical transformation on this Hamiltonian to incorporate all orders of the system-bath interaction. With

$$S = -\sum_k \frac{\gamma g_k}{\omega_k} \left(a_k - a_k^\dagger \right) \sigma_z \,, \tag{3.79}$$

we get

$$H = e^S H_{SB} e^{-S} \tag{3.80}$$

$$= -\frac{1}{2}\Delta \left(\sigma_+ B_- + \sigma_- B_+ \right) + \frac{1}{2}\epsilon \sigma_z + \sum_k \omega_k a_k^\dagger a_k \,, \tag{3.81}$$

where

$$B_\pm = \exp\left\{ \pm 2 \sum_k \frac{\gamma g_k}{\omega_k} \left(a_k - a_k^\dagger \right) \right\} . \tag{3.82}$$

For the unbiased case, $\epsilon = 0$, this gives the interaction picture Hamiltonian

$$H^I(t) = -\frac{1}{2}\Delta \left(\sigma_+ B_-^I(t) + \sigma_- B_+^I(t) \right) , \tag{3.83}$$

with

$$B_\pm^I(t) = \exp\left\{ \pm 2 \sum_k \frac{\gamma g_k}{\omega_k} \left(a_k e^{-\imath \omega_k t} - a_k^\dagger e^{\imath \omega_k t} \right) \right\} . \tag{3.84}$$

We can then transform the equation for $P(t)$ to get

$$P(t) = \mathrm{tr}\left\{ e^{\imath H t} \sigma_z e^{-\imath H t} |1\rangle\langle 1| \otimes R_0' \right\} , \tag{3.85}$$

where

$$R_0' = e^{-\beta \sum_k \omega_k a_k^\dagger a_k} / Z_R' \,. \tag{3.86}$$

We can find the master equation in the Born approximation for $P(t)$, also known as the non-interacting blip approximation, by performing perturbation theory on 3.85. To second order in Δ,

$$\dot{P}(t) = -\int_0^t dt' f(t-t') P(t') \,, \tag{3.87}$$

with

$$\begin{aligned} f(t) &= \frac{\Delta^2}{4}\left\{\langle\left[B_+^I(t), B_-^I(0)\right]\rangle_{R_0'}\right. \\ &\quad \left.+\langle\left[B_+^I(-t), B_-^I(0)\right]\rangle_{R_0'}\right\} \qquad (3.88) \\ &= \frac{\Delta^2}{2}\langle\left[B_+^I(t), B_-^I(0)\right]\rangle_{R_0'}, \qquad (3.89) \end{aligned}$$

where we have used that the correlation functions are equal. To compute $f(t)$ we can use the Feynman disentangling of operators[69] to get, in the notation of Leggett *et al.*[2],

$$f(t) = \Delta^2 \cos\left\{\frac{4\gamma^2}{\pi}Q_1(t)\right\} \exp\left\{-\frac{4\gamma^2}{\pi}Q_2(t)\right\}, \qquad (3.90)$$

with

$$Q_1(t) = \int_0^\infty d\omega \frac{J(\omega)}{\omega^2}\sin(\omega t), \qquad (3.91)$$

and

$$Q_2(t) = \int_0^\infty d\omega \frac{J(\omega)}{\omega^2}(1-\cos\omega t)\coth\left(\frac{\beta\omega}{2}\right). \qquad (3.92)$$

For ohmic dissipation, $J(\omega) = \eta\omega\exp(-\omega/\omega_c)$, these quantities become

$$Q_1(t) = \eta\tan^{-1}\omega_c t, \qquad (3.93)$$

and

$$Q_2(t) = \frac{\eta}{2}\ln\left(1+\omega_c^2 t^2\right) + \eta\ln\left(\frac{\beta}{\pi t}\sinh\frac{\pi t}{\beta}\right). \qquad (3.94)$$

With $\alpha \equiv 2\eta\gamma^2/\pi$, this gives 3.46 for the NIBA memory kernel.

Chapter 4

Untrotterized: adding matrix product states

We present a method to perform matrix product simulations based on adding matrix product states. This procedure can be used to simulate interactions beyond nearest neighbor and also certain types of non-Markovian master equations. The method is based on a variational procedure to find the optimal overlap of a single MPS with a state that is the sum of many MPS. We examine the form of a non-Markovian master equation that can be simulated with this procedure and discuss how the integral in the integro-differential equation can be calculated. For the particular case introduced, the computational cost is $NL^2d^6\chi^3$ for simulating a lattice where each site interacts with an independent reservoir with an exponentially decaying correlation function. The additional factor of L indicates that we have to keep track of and evolve a separate matrix product state for each reservoir which represents a history of the system. For a similar case but with a common reservoir, the computational cost of the simulation is $NLd^6\chi^3$, which is the same as Markovian dissipation. For a reservoir correlation function that is increasingly smooth, we can use the method of the previous chapter. The simulation in this case will have a computational cost $NC(N)L^2d^6\chi^3$, where the function $C(N)$ can be quite small.

4.1 Introduction

There have been an extraordinary number of advances in algorithms for simulating one-dimensional quantum lattice systems. One such advance was the time-evolving block decimation scheme which can be used to simulate real-time dynamics on one-dimensional lattice systems with local Hamiltonians. In Chapter 2, we extended this algorithm to mixed states. This extension can be used to calculate thermal states and evolve states in time according to a generic Markovian master equation. Thus, it can be used to study finite temperature crossover regions in systems containing a quantum critical point, or can be used to study the interplay between a external driving force, dissipation, and

Hamiltonian evolution. However, in many interesting open systems, in particular, in most condensed matter systems, the Markovian approximation for the influence of the environment is not justified or there are interactions beyond nearest neighbor. If one looks at equations of motion that involve only the system degrees of freedom, the resulting non-Markovian behavior produces a nonlocal-in-time portion of the master equation. That is, an equation with history dependence (memory effects). Our goal is to analyze some particular cases to see how efficiently we can simulate these equations.

We first give a variational procedure that can be used to add together many matrix product states, and thus it can be used to calculate integrals involving matrix product state integrands or be used to simulate systems with interactions beyond nearest neighbor. We then examine the types of non-Markovian master equations that we want to consider. Basically, we want master equations which have memory dependence, but that maintain some sense of locality (see below). Master equations within the Born approximation satisfy our requirements, but also, various phenomenological equations will also do so. We discuss the efficiency of this method for several cases and its usefulness.

4.2 Addition of matrix product states

In order to simulate systems with interactions beyond nearest neighbor[†] or with memory-dependent terms, we need a way to add together matrix product states. In the next subsection, we give a variational method to find the optimal MPS that overlaps some other MPS of higher dimension. On the basis of this method, in the subsequent subsection we show how to add together many MPS.

4.2.1 Variational method for finding an optimal MPS

We now describe a procedure to find the optimal MPS that describes some other MPS of higher dimension.[59] Suppose we have a state $|\phi\rangle$ as an MPS of dimension χ_ϕ and we want the best MPS $|\psi\rangle$ of dimension $\chi_\psi < \chi_\phi$ that approximates $|\phi\rangle$. The initial state has the MPS form

$$|\phi\rangle = \sum_{i_1,\ldots,i_L} \mathrm{tr}\left[A_1^{i_1}\cdots A_L^{i_L}\right]|i_1,\ldots,i_L\rangle\,, \tag{4.1}$$

where the A's are matrices of dimension $\chi_\phi \times \chi_\phi$. The optimal state is the one that minimizes

$$\||\phi\rangle - |\psi\rangle\|^2 = 1 + \langle\psi|\psi\rangle - \langle\psi|\phi\rangle - \langle\phi|\psi\rangle\,, \tag{4.2}$$

[†]Actually, there are other method to simulate systems with interactions beyond nearest neighbor, some of which, though, modify the computational cost fairly drastically.

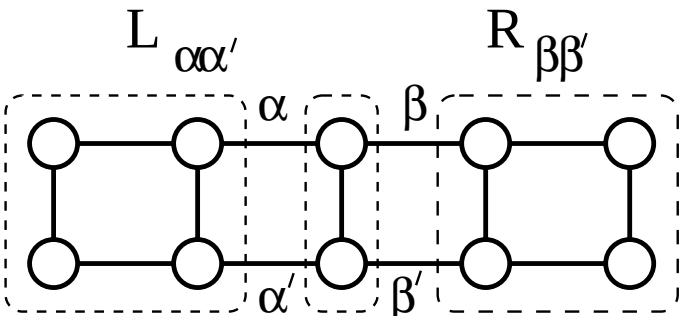


Figure 4.1: Inner product with a site l singled out. The top lattice is $\langle\psi|$ and the bottom lattice is $|\phi\rangle$.

where we have written this unconstrained to the normalization $\langle\psi|\psi\rangle$.[‡] That is, we want to maximize the overlap $\langle\psi|\phi\rangle$. The state $|\psi\rangle$ is also assumed to be an MPS

$$|\psi\rangle = \sum_{i_1,\ldots,i_L} \mathrm{tr}\left[\tilde{A}_1^{i_1}\cdots\tilde{A}_L^{i_L}\right]|i_1,\ldots,i_L\rangle\,, \tag{4.3}$$

where the $\tilde{A}$'s are matrices of dimension $\chi_\psi \times \chi_\psi$. A global minimization of equation 4.2 is not feasible. Instead, since the MPS has a local structure, we can perform a series of local minimizations, sweeping through the lattice site by site. At each site, we hold all the tensors of the state constant except for the one at the site.[§]

We start by singling out a single site, as shown in Figure 4.1. In vector notation, the equation 4.2 becomes, dropping the one,

$$\min_x \left\{x^\dagger h x - x^\dagger y - y^\dagger x\right\}\,. \tag{4.4}$$

We want the minimum with respect to x, which contains just the tensor $\Gamma^{[l]}$ of a single site l of the state $|\psi\rangle$. Expanding x about its minimum, $x = x_0 + \delta x$, we get

$$\begin{aligned}&\min_{\delta x}\left\{\left(x_0^\dagger + \delta x^\dagger\right)h\left(x_0+\delta x\right) - \left(x_0^\dagger+\delta x^\dagger\right)y - y^\dagger\left(x_0+\delta x\right)\right\}\\ &= \min_{\delta x}\left\{\left(x_0^\dagger h - y^\dagger\right)\delta x + \delta x^\dagger\left(h x_0 - y\right)\right\}\,.\end{aligned}$$

This is minimized when $hx_0 = y$. From this, we can calculate x_0, which contains the optimal tensor of the site l in the state $|\psi\rangle$ given that the rest of the state $|\psi\rangle$ is held constant. The matrix h contains the effects of the rest of the state $|\psi\rangle$. The vector y contains information on the overlap of $|\phi\rangle$ and $|\psi\rangle$ not including site l in $|\psi\rangle$. We want to perform the minimization, however, with the actual tensors and overlaps.

To find the form of the matrix h and vector y in terms of the original states, we can set the

[‡]In the case of mixed states, we minimize $\mathrm{tr}\left\{(\rho-\rho_0)^2\right\}$ as this leads to the same equations below.

[§]This is not guaranteed to find the global optimum, but we ignore this difficulty because in practice the procedure seems to work quite well.

overlap of the two states to be equal

$$\langle\psi|\psi\rangle = \langle\psi|\phi\rangle\,, \tag{4.5}$$

and extract out their form. We expand the state $|\psi\rangle$ around site l as

$$|\psi\rangle \quad = \quad \sum_{i_l,\alpha,\beta} \tilde{\Gamma}^{[l]i_l}_{\alpha\beta}|\alpha\rangle|i_l\rangle|\beta\rangle\,,$$

with $|\alpha\rangle = \tilde{\lambda}^{[l-1]}_{\alpha}|\Phi^{[1,\ldots,l-1]}_{\alpha}\rangle$ and $|\beta\rangle = \tilde{\lambda}^{[l]}_{\beta}|\Phi^{[l+1,\ldots,L]}_{\beta}\rangle$, using the notation in the introduction to this thesis. Therefore, when computing $\langle\psi|\psi\rangle$ with site l singled out, we get

$$\langle\psi|\psi\rangle \quad = \quad \sum_{i'_l\alpha',\beta'} \left(\tilde{\Gamma}^{[l]i'_l}_{\alpha'\beta'}\right)^{\star} \tilde{\lambda}^{[l-1]\star}_{\alpha'}\tilde{\lambda}^{[l]\star}_{\beta'} \left\{ \sum_{i_l,\alpha,\beta} \tilde{\Gamma}^{[l]i_l}_{\alpha\beta}\delta_{i'_l i_l}\delta_{\alpha'\alpha}\delta_{\beta'\beta}\tilde{\lambda}^{[l-1]}_{\alpha}\tilde{\lambda}^{[l]}_{\beta} \right\}, \tag{4.6}$$

where we have supposed that we have orthonormal eigenvectors describing the sites to the left and right of the isolated site. This latter fact has to be ensured during the minimization procedure (see below). Likewise, for $|\phi\rangle$ we have

$$|\phi\rangle = \sum_{i_l,\alpha_\phi,\beta_\phi} \Gamma^{[l]i_l}_{\alpha_\phi\beta_\phi}|\alpha_\phi\rangle|i_l\rangle|\beta_\phi\rangle\,, \tag{4.7}$$

where the subscript ϕ just reminds us that the indices get summed over different values. We can compute the inner product,

$$\langle\psi|\phi\rangle \quad = \quad \sum_{i'_l,\alpha',\beta'} \left(\tilde{\Gamma}^{[l]i'_l}_{\alpha'\beta'}\right)^{\star} \tilde{\lambda}^{[l-1]\star}_{\alpha'}\tilde{\lambda}^{[l]\star}_{\beta'} \left\{ \sum_{i_l,\alpha_\phi,\beta_\phi} \Gamma^{[l]i_l}_{\alpha_\phi\beta_\phi}\langle i'_l|i_l\rangle\langle\Phi^{[1,\ldots,l-1]}_{\alpha'}|\alpha_\phi\rangle\langle\Phi^{[l+1,\ldots,L]}_{\beta'}|\beta_\phi\rangle \right\} \tag{4.8}$$

and then equating 4.6 with 4.8,

$$\sum_{i'_l\alpha',\beta'} \left(\tilde{\Gamma}^{[l]i'_l}_{\alpha'\beta'}\right)^{\star} \tilde{\lambda}^{[l-1]\star}_{\alpha'}\tilde{\lambda}^{[l]\star}_{\beta'} \left\{ \sum_{i_l,\alpha,\beta} \tilde{\Gamma}^{[l]i_l}_{\alpha\beta}\delta_{i'_l i_l}\delta_{\alpha'\alpha}\delta_{\beta'\beta}\tilde{\lambda}^{[l-1]}_{\alpha}\tilde{\lambda}^{[l]}_{\beta} \right\}$$

$$= \sum_{i'_l,\alpha',\beta'} \left(\tilde{\Gamma}^{[l]i'_l}_{\alpha'\beta'}\right)^{\star} \tilde{\lambda}^{[l-1]\star}_{\alpha'}\tilde{\lambda}^{[l]\star}_{\beta'} \left\{ \sum_{i_l,\alpha_\phi,\beta_\phi} \Gamma^{[l]i_l}_{\alpha_\phi\beta_\phi}\langle i'_l|i_l\rangle\langle\Phi^{[1,\ldots,l-1]}_{\alpha'}|\alpha_\phi\rangle\langle\Phi^{[l+1,\ldots,L]}_{\beta'}|\beta_\phi\rangle \right\}.$$

Each term in the outer summations (over the same variables) are independent, so we get

$$\sum_{i_l,\alpha,\beta} \tilde{\Gamma}^{[l]i_l}_{\alpha\beta}\delta_{i'_l i_l}\delta_{\alpha'\alpha}\delta_{\beta'\beta}\tilde{\lambda}^{[l-1]}_{\alpha}\tilde{\lambda}^{[l]}_{\beta} = \sum_{i_l,\alpha_\phi,\beta_\phi} \Gamma^{[l]i_l}_{\alpha_\phi\beta_\phi}\langle i'_l|i_l\rangle\langle\Phi^{[1,\ldots,l-1]}_{\alpha'}|\alpha_\phi\rangle\langle\Phi^{[l+1,\ldots,L]}_{\beta'}|\beta_\phi\rangle\,. \tag{4.9}$$

This finally gives

$$\tilde{\Gamma}^{[l]i_l'}_{\alpha'\beta'}\tilde{\lambda}^{[l-1]}_{\alpha'}\tilde{\lambda}^{[l]}_{\beta'} = \sum_{\alpha_\phi,\beta_\phi}\Gamma^{[l]i_l'}_{\alpha_\phi\beta_\phi}\langle\Phi^{[1,\ldots,l-1]}_{\alpha'}|\alpha_\phi\rangle\langle\Phi^{[l+1,\ldots,L]}_{\beta'}|\beta_\phi\rangle\,, \tag{4.10}$$

or, for the optimal tensor at site l,

$$\tilde{\Gamma}^{[l]i_l}_{\alpha\beta} = \frac{1}{\tilde{\lambda}^{[l-1]}_{\alpha}\tilde{\lambda}^{[l]}_{\beta}}\left\{\sum_{\alpha_\phi}L^{[l-1]}_{\alpha\alpha_\phi}\left\{\sum_{\beta_\phi}\Gamma^{[l]i_l}_{\alpha_\phi\beta_\phi}R^{[l+1]}_{\beta\beta_\phi}\right\}\Bigg|^{\chi_\phi}_{d\chi_\psi\chi_\phi}\right\}\Bigg|^{\chi_\phi}_{d\chi_\psi^2}\,, \tag{4.11}$$

where we have defined

$$L^{[l-1]}_{\alpha\alpha_\phi} = \langle\Phi^{[1,\ldots,l-1]}_{\alpha}|\alpha_\phi\rangle\,, \tag{4.12}$$

and

$$R^{[l+1]}_{\beta\beta_\phi} = \langle\Phi^{[l+1,\ldots,L]}_{\beta}|\beta_\phi\rangle\,, \tag{4.13}$$

as the left and right overlap of the state $|\psi\rangle$ with the state $|\phi\rangle$. Also indicated in equation 4.11 is the best way to contract the tensors. Thus, we have the form of the optimal $\tilde{\Gamma}^{[l]}$ of the state $|\psi\rangle$ while holding the rest of the state constant.

To find the best approximation to the complete state, a sweep is used as shown in Figure 4.2. Going from site $l = 1, \ldots, L$, the $\tilde{\Gamma}$'s are replaced by equation 4.11. Doing it the naive way, it would have a computational cost proportional to L^2, because there are $\mathcal{O}(L)$ overlaps on the RHS and they would have to be computed $\mathcal{O}(L)$ times. Thus, we have to recycle the overlaps. It saves a factor of L because we do not have to recompute the overlaps on the RHS when recycling the overlap from site $l-1$ (or $l+1$). Thus, given the tensors L and R, the computational cost for computing the new $\tilde{\Gamma}^{[l]}$ is $d\chi^3$. We have to do this for the whole lattice, so the total cost is $Ld\chi^3$. Assuming that only a few sweeps are necessary (independent of L or χ) to get convergence, then there are no extra factors. However, there is one more step.

After we calculate the new $\tilde{\Gamma}$, we just replace the old one in the description of $|\psi\rangle$ but with one caveat. We need an additional step to orthogonalize the new $\tilde{\Gamma}$'s. This orthonormalization costs us $d^3\chi^3$ to diagonalize the reduced density matrix at a partition at site l. To ensure that the $\tilde{\Gamma}$'s are properly othogonalized and that the coefficients $\tilde{\lambda}$'s take on the right values, we can perform two Schmidt decompositions. The states $|\Phi^{[1,\ldots,l-1]}_{\alpha}\rangle$ and $|\Phi^{[l+1,\ldots,L]}_{\beta}\rangle$ (contained in $|\psi\rangle$) are assumed to already be orthonormal as they do not change in the minimization at site l. We can write the new

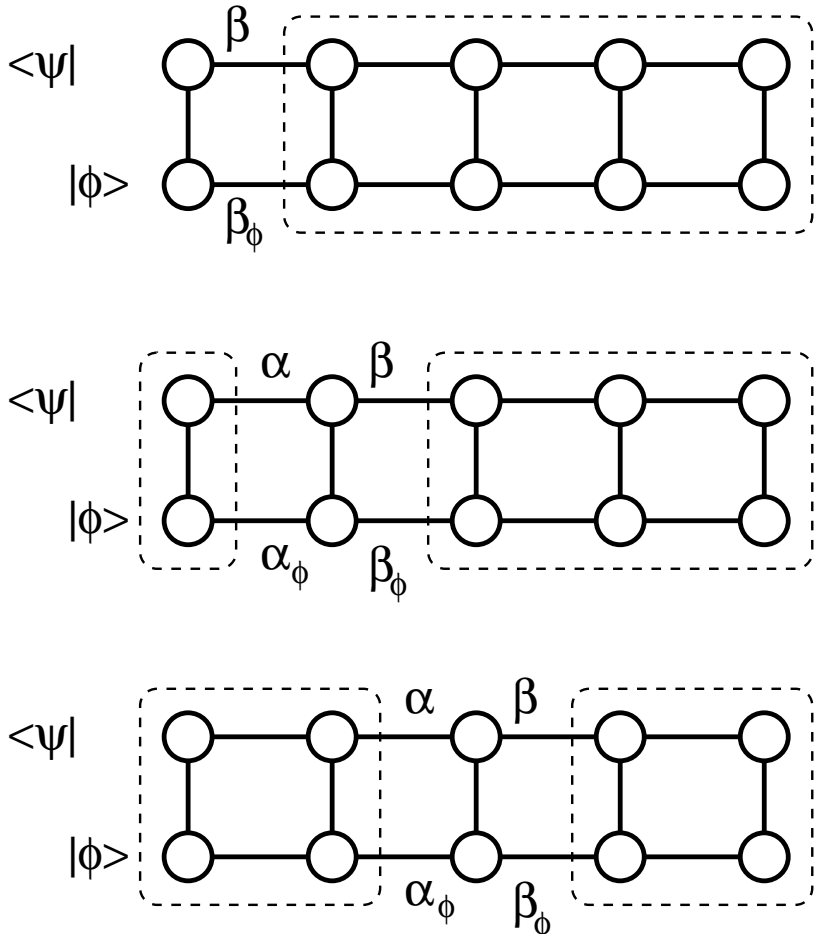


Figure 4.2: Sweep for finding the optimal state $|\psi\rangle$ to represent $|\phi\rangle$.

state of the system as

$$\begin{aligned} |\psi\rangle &= \sum_{i_l,\alpha,\beta} \tilde{\Gamma}'^{[l]i_l}_{\alpha\beta} |\alpha\rangle|i_l\rangle|\beta\rangle \\ &= \sum_{i_l,\alpha,\beta} \tilde{\Gamma}'^{[l]i_l}_{\alpha\beta} \tilde{\lambda}^{[l-1]}_{\alpha} |\Phi^{[1,\ldots,l-1]}_{\alpha}\rangle|i_l\rangle|\beta\rangle\,, \end{aligned}$$

where the prime indicates the new unnormalized tensor. The density matrix for this state is

$$\begin{aligned} \rho &= |\psi\rangle\langle\psi| \\ &= \sum_{i_l,\alpha,\beta,i'_l,\alpha',\beta'} \tilde{\Gamma}'^{[l]i_l}_{\alpha\beta} \tilde{\Gamma}'^{[l]i'_l\star}_{\alpha'\beta'} |\alpha\rangle|i_l\rangle|\beta\rangle\langle\beta'|\langle i'_l|\langle\alpha'|\,. \end{aligned}$$

The reduced density matrix for the left half, including site l is

$$\rho_L = \sum_{i_l,\alpha,i'_l,\alpha'} \left(\sum_{\beta} \tilde{\Gamma}'^{[l]i_l}_{\alpha\beta} \tilde{\Gamma}'^{[l]i'_l\star}_{\alpha'\beta} \left|\tilde{\lambda}^{[l]}_{\beta}\right|^2 \right) |\alpha\rangle|i_l\rangle\langle i'_l|\langle\alpha'|\,. \tag{4.14}$$

This can be diagonalized, and the orthonormalized $\tilde{\Gamma}$'s can be taken directly from the eigenvectors

$$|\Phi^{[1,\ldots,l]}_{\beta}\rangle = \sum_{i_l,\alpha} \tilde{\Gamma}^{[l]i_l}_{\alpha\beta} |i_l\rangle|\alpha\rangle\,. \tag{4.15}$$

The new $\tilde{\Gamma}$ should be normalized according to

$$\sum_{i_l,\alpha} \tilde{\Gamma}^{[l]i_l\star}_{\alpha\beta'} \tilde{\Gamma}^{[l]i_l}_{\alpha\beta} \left|\tilde{\lambda}^{[l-1]}_{\alpha}\right|^2 = \delta_{\beta'\beta}\,, \tag{4.16}$$

so the eigenvectors extracted from the diagonalization of ρ_L need to be divided by $\left|\tilde{\lambda}^{[l-1]}_{\alpha}\right|$. By doing this diagonalization, we have guaranteed that

$$\langle\Phi^{[1,\ldots,l]}_{\beta'}|\Phi^{[1,\ldots,l]}_{\beta}\rangle = \sum_{i_l,\alpha} \tilde{\Gamma}^{[l]i_l\star}_{\alpha\beta'} \tilde{\Gamma}^{[l]i_l}_{\alpha\beta} \left|\lambda^{[l-1]}_{\alpha}\right|^2 = \delta_{\beta'\beta}\,. \tag{4.17}$$

But we have not guaranteed that

$$\langle\Phi^{[l,\ldots,L]}_{\alpha'}|\Phi^{[l,\ldots,L]}_{\alpha}\rangle = \sum_{i_l,\beta} \tilde{\Gamma}^{[l]i_l\star}_{\alpha'\beta} \tilde{\Gamma}^{[l]i_l}_{\alpha\beta} \left|\lambda^{[l]}_{\beta}\right|^2 = \delta_{\alpha'\alpha}\,. \tag{4.18}$$

Actually, the latter does not need to be guaranteed in the sweeping procedure, so long as the sweeping procedure goes from left to right than back to the left, etc. Otherwise, we should perform a second SD.

We want to outline how to use the recycling method now. Starting with site 1, we compute the

sequence in Figure 4.2. When we are at site 1, we have to compute

$$R^{[2]}_{\beta\beta_\phi} \equiv \langle \Phi^{[2,\ldots,L]}_{\beta} | \beta_\phi \rangle \,, \tag{4.19}$$

which will cost us $Ld\chi^3$ (assuming $\chi_\psi \approx \chi_\phi = \chi$). But along the way we calculated all the other R,

$$\begin{aligned}
\langle \Phi^{[2,\ldots,L]}_{\beta} | \beta_\phi \rangle &= \left\{ \sum_{i_2,\beta_2} \tilde{\Gamma}^{[2]i_2\star}_{\beta\beta_2} \lambda^{[2]\star}_{\beta_2} \langle \Phi^{[3,\ldots,L]}_{\beta_2} | \langle i_2 | \right\} \left\{ \lambda^{[1]}_{\beta_\phi} \sum_{i'_2,\beta_{2\phi}} \Gamma^{[2]i'_2}_{\beta_\phi\beta_{2\phi}} | i'_2 \rangle | \beta_{2\phi} \rangle \right\} \\
&= \left\{ \sum_{i_2,\beta_2} \tilde{\Gamma}^{[2]i_2\star}_{\beta\beta_2} \lambda^{[2]\star}_{\beta_2} \lambda^{[1]}_{\beta_\phi} \left\{ \sum_{\beta_{2\phi}} \Gamma^{[2]i_2}_{\beta_\phi\beta_{2\phi}} R^{[3]}_{\beta_2\beta_{2\phi}} \right\} \Bigg|^{\chi_\phi}_{d\chi_\psi\chi_\phi} \right\} \Bigg|^{d\chi_\psi}_{\chi_\psi\chi_\phi} .
\end{aligned}$$

As we move through the lattice, we can save all the tensors R that we calculated along the way, for a memory of $L\chi^2$.

Moving on to site 2, we already have $R^{[3]}_{\beta\beta_\phi}$ and we need to compute $L^{[1]}_{\alpha\alpha_\phi}$:

$$\begin{aligned}
L^{[1]}_{\alpha\alpha_\phi} &= \langle \Phi^{[1]}_{\alpha} | \alpha_\phi \rangle = \left\{ \sum_{i_1} \tilde{\Gamma}^{[1]i_1\star}_{\alpha} \langle i_1 | \right\} \left\{ \sum_{i'_1} \Gamma^{[1]i'_1}_{\alpha_\phi} | i'_1 \rangle \right\} \\
&= \left\{ \sum_{i_1} \tilde{\Gamma}'^{[1]i_1\star}_{\alpha} \Gamma^{[1]i_1}_{\alpha_\phi} \right\} \Bigg|^{d}_{\chi_\psi\chi_\phi} .
\end{aligned}$$

Using the new $\tilde{\Gamma}$ computed from the previous step. The cost here is $d\chi^2$. Then we need to store the L we just computed, replacing $R^{[2]}$ in memory.

Moving on to site 3, we already have $R^{[4]}_{\beta\beta_\phi}$ and we need to compute $L^{[2]}_{\alpha\alpha_\phi}$:

$$\begin{aligned}
L^{[2]}_{\alpha\alpha_\phi} &= \langle \Phi^{[2]}_{\alpha} | \alpha_\phi \rangle = \left\{ \sum_{i_2,\alpha_1} \tilde{\Gamma}^{[2]i_2\star}_{\alpha_1\alpha} \lambda^{[1]}_{\alpha_1} \langle \Phi^{[1]}_{\alpha_1} | \langle i_2 | \right\} \left\{ \sum_{i'_2,\alpha_{1\phi}} \Gamma^{[2]i'_2}_{\alpha_{1\phi}\alpha_\phi} | i'_2 \rangle | \alpha_{1\phi} \rangle \right\} \\
&= \left\{ \sum_{i_2,\alpha_1} \tilde{\Gamma}^{[2]i_2\star}_{\alpha_1\alpha} \lambda^{[1]}_{\alpha_1} \left\{ \sum_{\alpha_{1\phi}} \Gamma^{[2]i_2}_{\alpha_{1\phi}\alpha_\phi} L^{[1]}_{\alpha_1\alpha_{1\phi}} \right\} \Bigg|^{\chi_\phi}_{d\chi_\phi\chi_\psi} \right\} \Bigg|^{d\chi_\psi}_{\chi_\phi\chi_\psi} .
\end{aligned}$$

So again, this is just a cost of $d\chi^3$, and then we need to store it, replacing $R^{[3]}$ in memory.

From here on it is the same. Each step requires $d\chi^3$ operations to compute the new L (or R depending on which direction we are sweeping) and we need order L steps. Thus, our computational cost is $Ld\chi^3$ for initialization and $Ld\chi^3$ for computing the new partial overlaps in the sweeps, and $Ld^3\chi^3$ for computing the new tensors in the sweeps. The memory (in addition to the normal memory storing the states) is $L\chi^2$.

4.2.2 Adding together many MPS

We can directly use the above procedure to add together many MPS. The minimization will now be over

$$\left\| \sum_i a_i |\phi_i\rangle - |\psi\rangle \right\|^2 \rightarrow \langle\psi|\psi\rangle - \sum_i a_i \left\{ \langle\psi|\phi_i\rangle + \langle\phi_i|\psi\rangle \right\} , \tag{4.20}$$

where i is summed over K terms and we dropped all the constant terms. In vector notation

$$\min_x \left\{ x^\dagger x - \sum_i a_i x^\dagger y_i - \sum_i a_i^\star y_i^\dagger x \right\} , \tag{4.21}$$

which we want the minimum of with respect to x. Expanding x about its minimum, $x = x_0 + \delta x$, we get

$$\begin{aligned} &\min_{\delta x} \left\{ \left(x_0^\dagger + \delta x^\dagger \right) (x_0 + \delta x) - \sum_i a_i \left(x_0^\dagger + \delta x^\dagger \right) y_i - \sum_i a_i^\star y_i^\dagger \left(x_0 + \delta x \right) \right\} \\ &= \min_{\delta x} \left\{ \left(x_0^\dagger - \sum_i a_i^\star y_i^\dagger \right) \delta x + \delta x^\dagger \left(x_0 - \sum_i a_i y_i \right) \right\} . \end{aligned}$$

This is minimized when $x_0 = \sum_i a_i y_i$. Again, then, taking the inner product with $\langle\psi|$:

$$\langle\psi|\psi\rangle = \sum_i a_i \langle\psi|\phi_i\rangle . \tag{4.22}$$

Thus, we get the same equation as before for the optimal $\tilde{\Gamma}$'s, but we have an additional index which goes over K values. In cases where no other simplification can be made, the computational cost will be $KLd^3\chi^3$ for the sweeps. The memory (in addition to the normal memory storing the states) is $KL\chi^2$. For cases where the MPS represents a history of the system, no simplification can generally be made, as the $|\phi_i\rangle$ are not related simply to $|\psi\rangle$. However, in cases where we want to go beyond nearest neighbor interactions (or even just integrate $H|\psi_0\rangle$ with nearest neighbor interactions), all the $|\phi_i\rangle = H_i|\psi_0\rangle$, thus, they differ from the original state only by a local operation around a group of lattice sites. Thus, one does not have to compute the overlap of each $|\phi_i\rangle$ separately, but can instead group the process together. This brings the computational cost back down to $Ld^3\chi^3$ regardless of the range of the interactions, so long as that range is independent of L (obviously, though, there is an extra factor in the cost, it just does not scale with the system parameters).

4.3 Integrating with MPS addition

Lets first derive a form of a non-Markovian master equation that may be of interest. Consider a general system plus reservoir Hamiltonian:

$$H = H_S + H_R + H_I \,, \tag{4.23}$$

where H_S is the system Hamiltonian, H_R is the reservoir Hamiltonian, and H_I represents the interaction between the system and reservoir and is responsible for the dissipation. The equation of motion for the density matrix for the system plus reservoir is

$$\frac{\partial \chi(t)}{\partial t} = -\imath \left[H, \chi(t)\right] . \tag{4.24}$$

Now transforming to the interaction representation

$$\chi^I(t) = e^{\imath(H_S+H_R)t} \chi(t)\, e^{-\imath(H_S+H_R)t} \tag{4.25}$$

for which the equation of motion is

$$\frac{\partial \chi^I(t)}{\partial t} = -\imath \left[H_I^I(t), \chi^I(t)\right] . \tag{4.26}$$

Now we can integrate to get

$$\chi^I(t) - \chi^I(0) = -\imath \int_0^t dt' \left[H_I^I(t'), \chi^I(t')\right] , \tag{4.27}$$

which we can put back into the equation of motion for $\chi^I(t)$ to get

$$\dot{\chi}^I(t) = -\imath \left[H_I^I(t), \chi^I(0)\right] - \int_0^t dt' \left[H_I^I(t), \left[\left[H_I^I(t'), \chi^I(t')\right]\right] . \tag{4.28}$$

We make the following standard assumption for simplicity: we assume that we turn on the interaction at $t = 0$ and that no correlations exist between the system and the reservoir. Thus, the total state is given by the tensor product state

$$\chi(0) = \rho(0) \otimes R_0 \,, \tag{4.29}$$

where R_0 is the density matrix for the reservoir, which can be, for instance, a state in thermal equilibrium:

$$R_0 = \frac{1}{Z_R} e^{-\beta H_R} . \tag{4.30}$$

$\rho(t)$ is the density matrix for the system:

$$\rho(t) = \mathrm{tr}_R\{\chi(t)\} \tag{4.31}$$

and the same for the interaction representation:

$$\rho^I(t) = e^{\imath H_S t}\rho(t)\, e^{-\imath H_S t} = \mathrm{tr}_R\left\{\chi^I(t)\right\}. \tag{4.32}$$

In equation 4.28, we can trace over the reservoir to get the equation of motion for the reduced density matrix of our system:

$$\begin{aligned}\dot{\rho}^I(t) &= \mathrm{tr}_R\left\{\left[H_I^I(t), \chi^I(0)\right]\right\} - \int_0^t dt' \mathrm{tr}_R\left\{\left[H_I^I(t), \left[H_I^I(t'), \chi^I(t')\right]\right]\right\} &(4.33)\\ &= -\int_0^t dt' \mathrm{tr}_R\left\{\left[H_I^I(t), \left[H_I^I(t'), \chi^I(t')\right]\right]\right\}, &(4.34)\end{aligned}$$

where we have assumed $\mathrm{tr}_R\left\{\left[H_I^I(t), \chi^I(0)\right]\right\} = 0$.

If the reservoir is very large compared to the system, interactions between the two will not change the state of the reservoir appreciably. Also, if the interaction between the two is weak, correlations will take a long time to develop. Thus we are interested in time scales of $\tau \ll \tau_c$, where τ_c is the timescale correlations develop. We can then approximate our state for all times as an uncorrelated state, this is the *Born approximation*:

$$\chi^I(t) = \rho^I(t) \otimes R_0 + \mathcal{O}(H_I). \tag{4.35}$$

Thus, equation 4.34 becomes

$$\dot{\rho}^I(t) = -\int_0^t dt' \mathrm{tr}_R\left\{\left[H_I^I(t), \left[H_I^I(t'), \rho^I(t') \otimes R_0\right]\right]\right\}. \tag{4.36}$$

Let the interaction have the form

$$H_I = \sum_\mu L_\mu \Gamma_\mu, \tag{4.37}$$

where $\{L_\mu\}$ is a set of operators acting on the system and $\{\Gamma_\mu\}$ is a set of operators acting on the reservoir (the set has to be such that H_I is Hermitian). Putting the interaction representation $e^{\imath(H_s+H_R)t} H_I e^{-\imath(H_s+H_R)t}$ into equation 4.36, we get the equation of motion for the density matrix of the system

$$\begin{aligned}\dot{\rho}^I(t) &= -\sum_{\mu\nu}\int_0^t dt' \left\{\left[L_\mu^I(t) L_\nu^I(t') \rho^I(t') - L_\nu^I(t') \rho^I(t') L_\mu^I(t)\right] \alpha_{\mu\nu}(t'-t) + \right.\\ &\quad \left.\left[\rho^I(t') L_\nu^I(t') L_\mu^I(t) - L_\mu^I(t) \rho^I(t') L_\nu^I(t')\right] \alpha_{\nu\mu}(t-t')\right\},\end{aligned}$$

where the cyclic property of the trace was used and

$$\alpha_{\mu\nu}(t'-t) = \mathrm{tr}_R\left\{R_0\Gamma^I_\mu(t)\,\Gamma^I_\nu(t')\right\}, \tag{4.38}$$

$$\alpha_{\nu\mu}(t-t') = \mathrm{tr}_R\left\{R_0\Gamma^I_\nu(t')\,\Gamma^I_\mu(t)\right\}. \tag{4.39}$$

We now have a non-Markovian master equation where we have made a Born approximation. The reservoir correlation functions 4.39 are only dependent on the time difference if the reservoir is in a thermal state.¶

Lets now convert the equation of motion back to the Schrodinger picture and put in the time evolution operator:

$$U(t-t') = e^{-iH_s\left(t-t'\right)}, \tag{4.40}$$

to get

$$\begin{aligned}\dot\rho(t) &= -i\left[H_s,\rho(t)\right]\\ &\quad -\sum_{\mu\nu}\int_0^t dt'\left\{\left[L_\mu U(t-t')\,L_\nu\rho(t')\,U(t'-t) - U(t-t')\,L_\nu\rho(t')\,U(t'-t)\,L_\mu\right]\alpha_{\mu\nu}(t'-t) + \right.\\ &\quad \left.\left[U(t-t')\,\rho(t')\,L_\nu U(t'-t)\,L_\mu - L_\mu U(t-t')\,\rho(t')\,L_\nu U(t'-t)\right]\alpha_{\nu\mu}(t-t')\right\}.\end{aligned} \tag{4.41}$$

We can now say what we meant by "some sense of locality." The operators L_μ appearing in this equation need not be local operators. However, their spread in the lattice due to the non-locality in time is given by the evolution operator, which has to have only local terms. Second, the operators L_u can not have any evolution in them that makes them a complicated object. We can generalize the form of this master equation so that the we do not use the unitary evolution operator U, but instead use some non-unitary evolution operator like $e^{\mathcal{L}\left(t-t'\right)}$. This is what appears in equation 3.2 and also in the phenomenological equation discussed in section 5.3.

In chapter 2, we also made the *Markov approximation*. For this approximation, we assume that changes induced in the reservoir due to H_{int} quickly dissipate away from the system. Thus, this change will not interact back with the system. Then the derivative of the system density matrix will not depend on its history, so we can replace $\rho^I(t')$ with $\rho^I(t)$ and $L^I_\mu(t')$ with $L^I_\mu(t)$ in equation 4.36:

$$\dot\rho^I(t) = -\int_0^t dt'\mathrm{tr}_R\left\{\left[H^I_I(t),\left[\sum_\mu L^I_\mu(t)\,\Gamma^I_\mu(t'),\rho^I(t)\otimes R_0\right]\right]\right\}. \tag{4.42}$$

This approximation is justified for weak coupling and so long as the $\alpha_{\mu\nu}(t-t')$ are proportional to $\delta(t-t')$.

¶To see this, just write the operators back in the Schrodinger representation.

4.3.1 Integration

Now, we are interested in solving equation 4.41 and we will not be able to use the exponential of a Lindbladian, because the derivative is not just proportional to the density matrix at that particular instant. Thus, we will not be able to apply the Trotter decomposition to evolve the density matrix in time. Therefore, we have to use MPS addition and normal integration techniques.

Suppose we have the MPS for our density matrix, $\rho_{MPS}(t)$ at time t. If we are able to compute the time derivative of our density matrix in MPS form, $\dot{\rho}_{MPS}(t)$, then we can evolve in time according to

$$\rho_{MPS}(t+\delta t) = \mathrm{Var}\left\{\rho_{MPS}(t) + \delta t\, \dot{\rho}_{MPS}(t)\right\}, \tag{4.43}$$

where the Var indicates that we find $\rho_{MPS}(t+\delta t)$ from the variational procedure above to get the state that best overlaps the desired state. If we want to go to second order we use the standard techniques. Compute first the density matrix at a half time step:

$$\rho_{MPS}(t+\delta t/2) = \mathrm{Var}\left\{\rho_{MPS}(t) + \delta t\, \dot{\rho}_{MPS}(t)/2\right\}, \tag{4.44}$$

then compute the density matrix at a full time step using the derivative from the half time step:

$$\rho_{MPS}(t+\delta t) = \mathrm{Var}\left\{\rho_{MPS}(t) + \delta t\, \dot{\rho}_{MPS}(t+\delta t/2)\right\}. \tag{4.45}$$

If we want to go to higher order, then we can use Rung-Kutta methods. This is the basic idea of the following but to compute the integrals we need to do some further work.

Since all the terms in the nonlocal piece (the piece that spreads in time) of equation 4.41 are identical in form, lets just consider one part of it:

$$I_{\mu\nu,L}(t) \equiv \int_0^t dt' U(t-t')\, L_\nu \rho(t')\, U(t'-t)\, \alpha_{\mu\nu}(t'-t), \tag{4.46}$$

where the subscript "L" stands for "Left" (because $L_\nu\rho$). This integral represents the interactions with bath in the past propagating up to the current time. In terms of these quantities, the master equation becomes

$$\dot{\rho}(t) = -i\left[H_s, \rho(t)\right] - \sum_{\mu\nu}\left(L_\mu I_{\mu\nu,L}(t) - I_{\mu\nu,L}(t)\, L_\mu - L_\mu I_{\mu\nu,R}(t) + I_{\mu\nu,R}(t)\, L_\mu\right). \tag{4.47}$$

This is still general. However, we want to go from the easiest to the hardest examples of this equation.

4.3.2 Locally updatable integrals

Suppose we have the integrals, $I_{\mu\nu,L}(t), I_{\mu\nu,R}(t)$, and want them at time $t+\delta t$:

$$\begin{aligned} I_{\mu\nu,L}(t+\delta t) &= \int_0^{t+\delta t} dt' U(t+\delta t-t') L_\nu \rho(t') U(t'-t-\delta t) \alpha_{\mu\nu}(t'-t-\delta t) \\ &\approx U(\delta t)\left\{\int_0^t dt' U(t-t') L_\nu \rho(t') U(t'-t) \alpha_{\mu\nu}(t'-t-\delta t)\right\} U(-\delta t) \\ &\quad +\delta t\, U\left(\frac{\delta t}{2}\right) L_\nu \rho\left(t+\frac{-\delta t}{2}\right) U\left(\frac{-\delta t}{2}\right) \alpha_{\mu\nu}\left(\frac{-\delta t}{2}\right) + O\left(\delta t^3\right) . \end{aligned}$$

If we want to relate $I_{\mu\nu,\sigma}(t+\delta t)$ to $I_{\mu\nu,\sigma}(t)$ locally in time, then we must have an equation of the form

$$I_{\mu\nu,\sigma}(t+\delta t) = \sum_i T_i^\dagger(t)\left\{I_{\mu\nu,\sigma}(t) c(t)\right\} T_i(t) + \delta I_{\mu\nu,\sigma}(t) , \tag{4.48}$$

where $c(t)$ is a constant which is only dependent on the time and we have a set of transformations $\{T_i\}$[‖], which only depend locally on time. Indeed, this looks almost like a master equation for a new quantity except for the additive constant that depends on time.

But, doing the simplest possible thing, we can identify the terms in equation 4.48 with those in the equation above. Thus, we first get,

$$I_{\mu\nu,\sigma}(t+\delta t) = U(\delta t)\left\{I_{\mu\nu,\sigma}(t) c(t)\right\} U(\delta t) + \delta I_{\mu\nu,\sigma}(t) , \tag{4.49}$$

and we can set

$$\delta I_{\mu\nu,\sigma}(t) \approx \delta t\, U\left(\frac{\delta t}{2}\right) L_\nu \rho\left(t+\frac{-\delta t}{2}\right) U\left(\frac{-\delta t}{2}\right) \alpha_{\mu\nu}\left(\frac{-\delta t}{2}\right) + O\left(\delta t^3\right) . \tag{4.50}$$

As for the constant $c(t)$, it is really what makes the approximation:

$$\begin{aligned} &\int_0^t dt' U(t-t') L_\nu \rho(t') U(t'-t) \alpha_{\mu\nu}(t'-t-\delta t) \\ &\doteq \int_0^t dt' U(t-t') L_\nu \rho(t') U(t'-t) \alpha_{\mu\nu}(t'-t) c(t) . \end{aligned}$$

One solution is to just suppose every term in the integral multiplying α is independent, which will result in the condition

$$\alpha_{\mu\nu}(t'-t-\delta t) = \alpha_{\mu\nu}(t'-t) c(t) \ \forall t' . \tag{4.51}$$

[‖] Actually, this is not the most general, we would have to allow for general transformations instead of just similarity transformations.

We can derive the differential equation for α:

$$\alpha_{\mu\nu}\left(t'-t-\delta t\right)=\alpha_{\mu\nu}\left(t'-t\right)-\delta t\left.\frac{\partial\alpha_{\mu\nu}\left(\tau\right)}{\partial\tau}\right|_{\tau=t'-t}=\alpha_{\mu\nu}\left(t'-t\right)c\left(t\right)\,, \tag{4.52}$$

which gives

$$\frac{\partial\alpha_{\mu\nu}\left(\tau\right)}{\partial\tau}=\lim_{\delta t\to0}\alpha_{\mu\nu}\left(\tau\right)\frac{\left[1-c\left(t'-\tau\right)\right]}{\delta t}\,\forall t'\,. \tag{4.53}$$

Since this has to hold for all t', and we cannot separate the t' dependence from the τ dependence, c is not a function of anything. Thus, let us just set

$$\lim_{\delta t\to0}\frac{\left[1-c\right]}{\delta t}\equiv-\tau_{R,\mu\nu}^{-1}\,, \tag{4.54}$$

which could be done by just letting $c=\kappa^{-\delta t}=e^{-\delta t\ln\kappa}$,

$$\lim_{\delta t\to0}\frac{\left[1-c\right]}{\delta t}=\lim_{\delta t\to0}\frac{\left[1-e^{-\delta t\ln\kappa}\right]}{\delta t}=\lim_{\delta t\to0}\frac{\delta t\ln\kappa}{\delta t}=\ln\kappa\,. \tag{4.55}$$

Therefore, our differential equation is

$$\frac{\partial\alpha_{\mu\nu}\left(\tau\right)}{\partial\tau}=-\tau_{R,\mu\nu}\alpha_{\mu\nu}\left(\tau\right)\,, \tag{4.56}$$

which just gives an exponential as a solution:

$$\alpha_{\mu\nu}\left(\tau\right)=\alpha_{\mu\nu}\left(0\right)e^{-\left|\tau\right|/\tau_{R,\mu\nu}}\,. \tag{4.57}$$

We can define a class of two parameter reservoir correlation functions by enforcing an overall dissipative strength:

$$\int_0^\infty d\tau\alpha_{\mu\nu}\left(0\right)e^{-\left|\tau\right|/\tau_{R,\mu\nu}}=\alpha_{\mu\nu}\left(0\right)\tau_{R,\mu\nu}\equiv\gamma_{\mu\nu}\,. \tag{4.58}$$

This gives, then,

$$\alpha_{\mu\nu}\left(\tau\right)=\frac{\gamma_{\mu\nu}}{\tau_{R,\mu\nu}}e^{-\left|\tau\right|/\tau_{R,\mu\nu}}\,. \tag{4.59}$$

It turns out this will be the class of reservoir correlation functions which we can simulate the most efficiently.

With the exponential form of the reservoir correlation function, we can now use the following property,

$$\alpha_{\mu\nu}\left(\tau-\delta t\right)=\alpha_{\mu\nu}\left(\tau\right)e^{-\left|-\delta t\right|/\tau_{R,\mu\nu}}=\alpha_{\mu\nu}\left(\tau\right)\kappa^{\delta t}\equiv\alpha_{\mu\nu}\left(\tau\right)c\left(t\right)\,, \tag{4.60}$$

to our advantage. Putting equation 4.60 and equation 4.50 into equation 4.49, we get

$$I_{\mu\nu,\sigma}\left(t+\delta t\right)\approx\kappa^{\delta t}U\left(\delta t\right)\left\{I_{\mu\nu,\sigma}\left(t\right)\right\}U\left(-\delta t\right)+\delta t\,U\left(\frac{\delta t}{2}\right)L_{\nu}\rho\left(t+\frac{-\delta t}{2}\right)U\left(\frac{-\delta t}{2}\right)\alpha_{\mu\nu}\left(\frac{-\delta t}{2}\right). \tag{4.61}$$

This will become a Markovian equation if we take the limit $\tau_R \to 0$. One can compare this equation with equation 3.5.

4.3.3 Computational cost

4.3.3.1 Independent reservoirs

If we have an independent reservoirs for each site, or pair of nearest neighbor sites, then our interaction Hamiltonian looks like

$$H_I=\sum_{\mu=1}^{\mathcal{O}(L)}\left(L_{\mu}\Omega_{\mu}+L_{\mu}^{\dagger}\Omega_{\mu}^{\dagger}\right). \tag{4.62}$$

Then, the α are

$$\alpha_{\mu\nu}\left(t'-t\right)=\mathrm{tr}\left(\otimes_{\sigma}R_{\sigma,0}\Omega_{\mu}^{I}\left(t'\right)\Omega_{\nu}^{I\dagger}\left(t\right)\right)=\delta_{\mu\nu}\alpha_{\nu}\left(t'-t\right). \tag{4.63}$$

Therefore we have $\mathcal{O}\left(L\right)$ integrals $I_{\mu\nu,\sigma}=\delta_{\mu\nu}I_{\nu,\sigma}$ because μ and ν have to be indices on the same site (pair of sites) for the reservoir correlation function to be non-zero.

We want to store L MPSs, which means a memory of $L^2d\chi^2$, where we let d be the local dimension (thus, d is really d^2 when d represents the local Hilbert space dimension). For the variational procedure, we get an additive memory cost of $L^2\chi^2$. So overall the memory is $L^2d\chi^2$. Further, each time step we need to update the integrals. To update one integral, there is a cost of $Ld^3\chi^3$ to apply the unitary operator $U\left(\delta t\right)$ and a cost of $Ld\chi^3$ to add on the additional term δI. Thus, we get a cost of $L^2d\chi^3$ to update all the integrals. Then, to update the density matrix, we have to add $\mathcal{O}(L)$ MPS, thus the computational cost is $L^2d\chi^3$. For the whole time evolution it is then $NL^2d\chi^3$ where N is the number of time steps.

4.3.3.2 Single reservoir

An alternate reservoir is just a single reservoir interaction:

$$H_{int}=\left(\sum_{\mu=1}^{\mathcal{O}(L)}L_{\mu}\right)\Omega+\left(\sum_{\mu=1}^{\mathcal{O}(L)}L_{\mu}^{\dagger}\right)\Omega^{\dagger}\equiv\bar{L}\Omega+\bar{L}^{\dagger}\Omega^{\dagger}. \tag{4.64}$$

In this case, we can do better, as we can take just $\mathcal{O}(1)$ histories, which are obtained from summing up the action of the $\bar{L}$ on the density matrix. Thus, we have only $\mathcal{O}(1)$ additional MPS to keep track of. The additional memory is just $Ld\chi^2$, and the variational procedure only involves memory

$L\chi^2$. We also need the cost to update the single quantity:

$$I_\sigma(t+\delta t) \approx \kappa^{\delta t} U(\delta t)\{I_\sigma(t)\} U(-\delta t) + \delta t\, U\left(\frac{\delta t}{2}\right) \bar{L}\rho\left(t+\frac{-\delta t}{2}\right) U\left(\frac{-\delta t}{2}\right) \alpha\left(\frac{-\delta t}{2}\right). \quad (4.65)$$

The first part we can update with a cost $Ld^3\chi^3$. How its written, the second term would require $L^2d^3\chi^3$ to update (applying L operators contained within $\bar{L}$, then adding them together with a cost of L^2). As mentioned earlier, these $\mathcal{O}(L)$ MPS are simply related and thus we can add them with cost scaling with L. However, we could also use the trick

$$\delta t \bar{L}_\nu \approx e^{\delta t \bar{L}_\nu} - \mathrm{I} + O\left(\delta t^2\right), \quad (4.66)$$

which would cost $Ld^3\chi^3$ to apply the exponential operator and then order $Ld\chi^3$ to add the two MPSs together. Thus, our overall cost would be $NLd^3\chi^3$, the same as a Markovian master equation. It is OK to have the the integrals only to first order in δt because when we compute ρ at the next time step, we multiply the integrals by another factor of δt. So overall it will be second order.

This type of reservoir introduces effective long range interactions that are independent of the distance between two sites. This may destroy the MPS form of our states (i.e., make it so the coefficients do not decay exponentially anymore). However, the formal scaling is nice.

4.3.3.3 Arbitrary reservoir

The update of the density matrix will be the same as above: $L^2d^3\chi^3$. Thus, it is only the update of the integrals that will change. If we have the integrals

$$I_{\mu\nu,L}(t) \equiv \int_0^t dt' U(t-t') L_\nu \rho(t') U(t'-t) \alpha_{\mu\nu}(t'-t), \quad (4.67)$$

we can write them approximately as a sum with $N_t = \mathrm{int}(t/\delta t)$

$$I_{\mu\nu,L}(N_t) \approx \delta t \sum_{m=0}^{N_t-1} U\left(t-\frac{\delta t}{2}-m\delta t\right) L_\nu \rho\left(\frac{\delta t}{2}+m\delta t\right) U\left(\frac{\delta t}{2}+m\delta t-t\right) \alpha_{\mu\nu}\left(\frac{\delta t}{2}+m\delta t-t\right). \quad (4.68)$$

And then we just have to keep track of each individual term in the summation, with $N_t \sim O(N)$, so memory it will be order $NL^2d^3\chi^2$ and the computational cost will be $N^2L^2d^3\chi^3$, which is not a very friendly expression. If the reservoir correlation function is increasingly smooth, we can use directly the method in Chapter 3 to calculate the long-time memory. Thus, the computational cost will have a $NC(N)$ factor in front rather than N^2.

4.4 Conclusions

In this chapter we described a method that can simulate real-time evolution by adding together matrix product states. It can be used to solve certain types of non-Markovian master equations or systems with interactions beyond nearest neighbor. The usefulness of this method for non-Markovian dynamics is unclear as the physical approximations are still fairly strong, e.g., one has to make Born approximation or use a phenomenological non-Markovian master equation. One may be able to transform the Hamiltonian in such a way as to get non-perturbative effects of the environment, as was done in Chapter 3.

However, the additive procedure is useful for real-time evolution when the system interacts with a common reservoir or when longer-range interactions are present. The range of the interactions will contribute a multiplicative term to the computational cost, but it will otherwise be very efficient.

The estimates for the computational cost are also useful in comparison to the method we propose in the next two chapters. We will see that incorporating local reservoirs into the simulation is actually more efficient than incorporating the very approximate non-Markovian terms. This gives added weight to the conclusions of the next chapter: that non-Markovian master equations are not particularly useful numerically as generic ways to simulate strong dissipation.

Chapter 5

Generic approach for simulating real-time, non-Markovian dynamics

We investigate the behavior of a simple non-Markovian process in order to understand how to construct a generic approach for real-time, open-system dynamics. The process we consider is amplitude damping of a spin. Approximate non-Markovian master equations for such a process can lose positivity when the system-reservoir interaction gets too strong compared to the reservoir memory time or when additional interactions in the system's Hamiltonian are turned on. To overcome these difficulties, we introduce a methodology which circumvents the construction of non-Markovian master equations entirely. The methodology is based on a sequence of transformations of the reservoir Hamiltonian that bring it into a form amenable to generic treatment. For the case of a spin interacting with a Lorentzian reservoir, the sequence can be terminated after a single transformation and yields an exact Markovian master equation valid in all of parameter space. We discuss this methodology in terms of more complex reservoirs and in the context of simulations performed with matrix product states. In particular we are interested in using matrix product based simulation algorithms for studying strongly dissipative quantum dynamics. This imposes an added constraint on the form of non-Markovian equations: the non-locality arising from the memory dependence in the equations of motion has to be suppressed in a way to give only local operators or to give only a controllable number of "histories," which can be added together to give the non-Markovian evolution. The transformation we introduce circumvents this difficulty as well.

5.1 Introduction

Understanding the dynamics of quantum systems in the presence of an environment is crucial in many fields of physics and chemistry, from investigations into decoherence to charge transport in organic molecules. To help gain this understanding, there are several analytical techniques available[54,53,56,55]. The basic idea with these techniques is to divide the degrees of freedom into two parts,

a system and an environment, and to obtain equations of motion which only explicitly reference the system. Generally, these equations can not be solved exactly and one has to make approximations. One of the most common approximations is to use a Markovian master equation, which is a local-in-time differential equation referencing only the system's density matrix. Such a master equation is physically justified when the environment has a rapid relaxation timescale and is only weakly coupled to the system. This approximation has been used to examine the real-time dynamics of a variety of open systems, ranging from optics [70] to electronic transport (in the form of a wide-band limit reservoir) [71].

On the one hand, the Markovian approximation is useful for building phenomenological, and generic, equations of motion for systems in the presence of dissipation or decoherence. This is because the form of the equations give valid quantum evolution regardless of the strength of the parameters in the Hamiltonian. On the other hand, the approach can miss important physical phenomena if the environment retains a memory of or couples strongly to the system. For instance, a dissipatively driven phase transition appears in Josephson Junctions [13] and in spin-boson systems [2], where in both cases the environment has a long-time memory. If one goes beyond the Markovian approach to include these non-Markovian effects, the resulting equations of motion are, however, not as well behaved. Approximate non-Markovian master equations often do not give valid quantum evolution in a substantial range of parameter space. And one is still restricted to studying only weak coupling and interaction with rapidly relaxing environments. In addition, like what is achieved with Markovian master equations, it would be a tremendous accomplishment to be able to construct generic non-Markovian master equations: equations which do not depend on the details of the system Hamiltonian and the system-reservoir interaction. Such equations would be useful in computational simulations, such as those using matrix product states for open systems [58,59,33,34].

This chapter is organized as follows. In 5.2, we discuss the problems with treating non-Markovian terms in the master equation as generic and we propose a methodology for simulating strongly dissipative or long-time memory processes in open systems without using non-Markovian equations. In 5.3, we present a proposal by other researchers for a general form of a non-Markovian master equation, which they call "post-Markovian." We give a full solution of the model for amplitude damping discussed in 5.2 and also discuss this in the context of the "post-Markovian" master equation. Finally, in 5.4, we take a Keldysh approach to deriving non-Markovian equations, where we show the general form of the master equation for a spin or fermion decaying into an empty reservoir.

5.2 Generic Approach

In this section, we examine the behavior of a non-Markovian equation in a common paradigm: a single spin interacting with a reservoir that has an exponentially decaying memory. It is known

that a non-Markovian master equation describing such a process loses positivity when the interaction strength gets too large compared to the memory time[72] (see also Ref. 62). However, the master equation can also lose positivity when other interactions in the Hamiltonian are modified. We demonstrate in this particular case that treating the non-Markovian term as generic results in an equation of motion that is unstable (with respect to positivity) when one turns on additional interactions in the system's Hamiltonian. Thus, it does not seem likely that one can have a generic non-Markovian term in the master equation, which can be used, even phenomenologically and keeping the system-reservoir interaction constant, to investigate the effects of dissipation as one varies parameters of the system. Having such a generic term, though, would allow one to build reusable tools and approximations for the investigation of non-Markovian effects. Because of this, we suggest an alternative approach for simulating real-time, open-system dynamics that does not rely on non-Markovian master equations. The methodology is based on performing a sequence of transformations on the reservoir Hamiltonian which create auxiliary quantum degrees of freedom that can be included explicitly in the simulation. Their inclusion allows one to simulate highly non-Markovian processes without the drawbacks of non-Markovian master equations. Further, for systems with strong dissipation or long memory times, such as the spin-boson model, the structure of the reservoir is such that one can use matrix product state techniques. Indeed, this is the basis of the numerical renormalization group, but its implications go well beyond renormalization schemes. With such a generic approach, computational and analytic tools can be developed to handle a wide variety of open systems[58,73]. This approach also gives insight into the mechanism by which an environment retains a memory of the system and allows this to interact back at a later time.

Let us start by introducing the types of systems under consideration. We examine system-reservoir[†] models which have the form

$$H = H_S + H_R + H_I \,, \tag{5.1}$$

where H_S is the Hamiltonian of the system, H_R is the Hamiltonian of a reservoir, and H_I is some interaction Hamiltonian between the system and reservoir. The reservoir Hamiltonian is taken to be a collection of non-interacting particles,

$$H_R = \sum_k \omega_k b_k^\dagger b_k \,, \tag{5.2}$$

where the operators b_k are for either bosonic or fermionic modes of the reservoir. The interaction

[†]We use the terms *reservoir*, *bath*, and *environment* interchangeably.

Hamiltonian will be chosen to have the form

$$H_I = L \sum_k g_k^\star b_k^\dagger + L^\dagger \sum_k g_k b_k \,, \tag{5.3}$$

or a suitable generalization to cases with many independent reservoirs, where L is some operator on the system. Quite generally, we want the system Hamiltonian to be arbitrary: this is a consequence of the fact that we want a procedure for treating the effects of the reservoir that does not rely on the properties of the system.

We illustrate the poor behavior of non-Markovian terms by examining a specific case with only one spin (or lattice site) connected to the reservoir. Further, we consider an interaction Hamiltonian that has the quadratic form

$$H_I = \sigma_0^+ B + B^\dagger \sigma_0^- \,, \tag{5.4}$$

with $B = \sum_k g_k b_k$, where g_k is the coupling constant to the k^{th} mode of the reservoir and the "0" labels the spin connected to the reservoir. We also only consider a system Hamiltonian to take on the quadratic form

$$H_S = \frac{\omega_0}{2} \left(\sigma_0^z + \sigma_1^z\right) + h \left(\sigma_0^+ \sigma_1^- + \sigma_1^- \sigma_0^+\right) \,, \tag{5.5}$$

where here, for instance, we took two spins with identical on-site energies, ω_0, and a nearest-neighbor interaction of strength h. Although some of the following results can be extended, we want to restrict ourselves to the case of zero-temperature reservoirs where all the frequencies of the reservoir modes, ω_k, are positive. Thus, at zero temperature we will have an empty reservoir. The physical scenario is then a collection of spins (or particles) decaying into an empty reservoir.

Now let us consider the behavior of the phenomenological non-Markovian master equation[‡]

$$\frac{\partial \rho(t)}{\partial t} = -\imath \left[H_S, \rho\left(t\right)\right] + \int_0^t dt' \bar{k}(t-t') \mathcal{L}_\kappa \rho(t') \,, \tag{5.6}$$

where $\mathcal{L}_\kappa$ is the Lindbladian

$$\mathcal{L}_\kappa \circ = \kappa \sigma_0^- \circ \sigma_0^+ - \frac{\kappa}{2} \left[\circ, \sigma_0^+ \sigma_0^-\right]_+ \,. \tag{5.7}$$

The function $\bar{k}\left(t-t'\right)$ is the normalized single-particle correlation function of the reservoir at zero temperature:

[‡]This master equation can be derived by making the Born approximation when the nearest-neighbor interaction h is zero and by removing the dependence on ω_0.

$$\begin{aligned}\bar{k}(t-t') &= \frac{2e^{\imath t\omega_0}}{\kappa}\mathrm{tr}_R\left\{B^I(t)B^{I\dagger}(t')\rho_R\left(\beta=\infty\right)\right\} \\ &= \frac{2}{\kappa}\int_{-\infty}^{\infty}d\omega J(\omega)e^{-\imath(\omega-\omega_0)\left(t-t'\right)}\,, \end{aligned} \tag{5.8}$$

where $\rho_R\left(\beta\right)$ is a thermal state of the reservoir at inverse temperature β and we have assumed a constant density of states. We take the reservoir spectral function to be

$$J(\omega)=\frac{1}{2\pi}\frac{\kappa\gamma^2}{(\omega-\omega_0)^2+\gamma^2}\Theta(\omega)\,. \tag{5.9}$$

This gives the reservoir single particle correlation function as[§]

$$\bar{k}\left(t-t'\right)=\gamma e^{-\gamma\left|t-t'\right|}\,. \tag{5.10}$$

One can solve the non-Markovian equation of motion. Of particular interest is when the spin is started in the up state and is allowed to decay into the reservoir. For $h=0$, it is known that the equation looses positivity when $\kappa>\gamma/4$[72]. In addition, though, as we turn on the interaction parameter h, the equation also loses positivity. In Figure 5.1, we numerically show the violations of positivity, i.e., by showing the magnitude of the negative eigenvalue of ρ as function of h. We evolved from time $t=0$ to $t=1$, and the figure shows the violation of positivity at $t=1$. Even for values of κ/γ well below the threshold of $1/4$, we find numerically that one still loses positivity for any finite h. This shows that when adding non-Markovian terms arbitrarily to a master equation, one can get evolution that is unphysical when turning on additional interactions in the system Hamiltonian. This is exactly the situation that has to be avoided for a generic approach.

In light of the above, we now introduce a transformation that gives a manageable, and completely positive, master equation. We first consider again the model above and show the transformation for this specific case. The idea of the transformation is shown in Figure 5.2(a). In the initial model, the spin (or site) is connected to many modes of the reservoir R. The transformation modifies the reservoir modes so that the site is coupled to only one mode (the auxiliary site A) and that mode is then coupled to a new reservoir R'.

From equation 5.4, how to define the reservoir mode coupled directly to the spin should be clear. Lets define a new set of modes, $\{c_k\}$, d, where the c_k give the modes of the new reservoir and d is

[§]We work in the limit $\gamma\ll\omega_0$ throughout this article. In addition, a spectral function of the form 5.9 allows us to remove the ω_0 dependence of the dynamics when the density matrix is a state in the single-particle sector of the Hamiltonian 5.5.

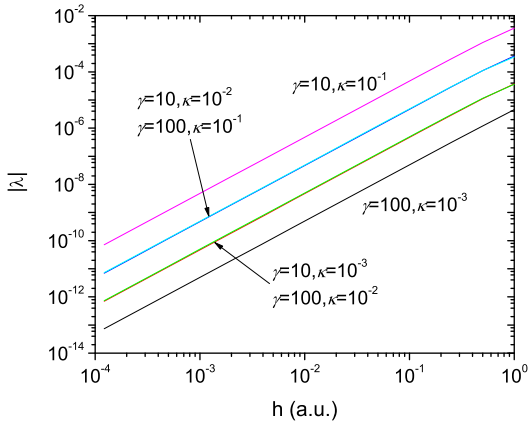


Figure 5.1: The magnitude of the negative eigenvalue λ of ρ as a function of h for several different values of γ and κ at a time $t = 1$. When $h = 0$, the equations are positive for all values of γ and κ shown. Numerically we find that for any finite value of h, positivity is only restored when $\kappa \to 0$ or $\gamma \to \infty$, e.g., in the Markovian limit. In addition, the negative eigenvalue goes as $\sim \kappa h^2/\gamma$ for small κ and large γ.

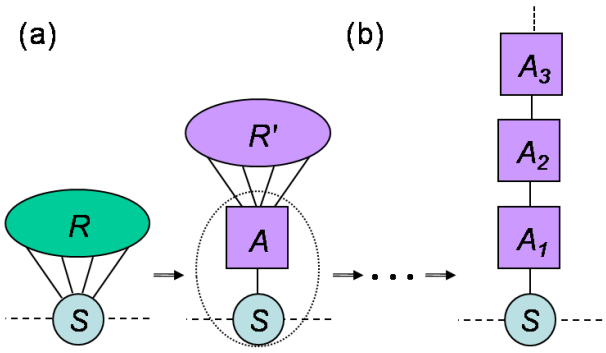


Figure 5.2: (a) Transformation of a local reservoir, R, into local auxiliary site, A and a new reservoir, R', not connected to the system. The auxiliary site is to be included within the simulation. (b) Continued transformation results in a one-dimensional lattice of auxiliary sites, $\{A_i\}$, that can be treated efficiently with simulations using matrix product states.

for the auxiliary site connected to the spin. The mode d is taken as

$$d = \frac{1}{\sqrt{\sum_k |g_k|^2}} \sum_k g_k b_k \,, \tag{5.11}$$

where

$$\sum_k |g_k|^2 = \int J(\omega)\, d\omega = \frac{\kappa\gamma}{2} \,. \tag{5.12}$$

The interaction Hamiltonian then becomes

$$H_I = \sqrt{\frac{\kappa\gamma}{2}} \left(\sigma_0^+ d + d^\dagger \sigma_0^-\right) \equiv \kappa_1 \left(\sigma_0^+ d + d^\dagger \sigma_0^-\right) \,. \tag{5.13}$$

We require that the new reservoir R' and auxiliary site A to have the Hamiltonian

$$H_R = \omega_1 d^\dagger d + \sum_k \left(g_{1k} d^\dagger c_k + g_{1k}^\star c_k^\dagger d\right) + \sum_k \omega_{1k} c_k^\dagger c_k \,. \tag{5.14}$$

The point of this requirement is to ensure that the Hamiltonian of the auxiliary site to be of the same form as a system-reservoir Hamiltonian. Below, this will be used to perform a sequence of transformation.

The Hamiltonian 5.14 can be diagonalized to go back to the original reservoir via the procedure for an impurity in a continuum of modes[69]. The auxiliary site and new reservoir modes can be expanded in the modes of the original reservoir as

$$d = \sum_k \nu_k b_k \,,\; c_k = \sum_{k'} \eta_{kk'} b_k \,, \tag{5.15}$$

where the coefficients just define a unitary transformation between the two sets of modes. The energies ω_{1k} will not change much because each mode k is only weakly connected to the mode d, where in the continuum limit the coupling strength goes to zero. Thus, we can take the set of energies ω_{1k} to be equal to ω_k. The coefficients ν_k can be found to be[69]

$$\nu_k^2 = \frac{|g_{1k}|^2}{\left[\omega_k - \omega - \Sigma(\omega_k)\right]^2 + \left[N |g_{1k}|^2 / 2\right]^2} \,, \tag{5.16}$$

with

$$\Sigma(\omega_k) = \sum_{k' \neq k} \frac{|g'_{k'}|^2}{\omega_k - \omega_{k'}} \,. \tag{5.17}$$

The factor N is the normalization for the continuum limit $\sum_k \frac{2\pi}{N} \to \int d\omega$ and $N\delta_{kk'}/2\pi \to \delta(\omega_k - \omega_{k'})$.

To complete the transformation we match the parameters in equation 5.16 with those in equation

5.11. This can be done by taking $g_{1k} = \sqrt{2\gamma/N}$ and $\omega = \omega_0$. In the continuum limit, this gives the spectral function $J'(\omega) = 2\gamma/2\pi$ for the new reservoir R'[¶]. Since the spectral function is a constant over all frequencies, the single-particle correlation function of the new reservoir is $\gamma'\delta(t)$, with $\gamma' = 2\gamma$, e.g., it is the correlation function of a Markovian reservoir. For the case of the single spin starting in the up state and decaying into the reservoir, the transformation results in an exact Markovian master equation

$$\dot{\chi}(t) = -\imath\left[H_S + H_{I'}, \chi(t)\right] + \mathcal{L}_{\gamma'}\chi(t) \ , \tag{5.18}$$

for the spin (or two spins when $h \neq 0$, but only within the single-particle sector) and the auxiliary site, e.g., χ is a state of the system and auxiliary site A. In this particular case of a single spin decaying into a zero-temperature reservoir, the auxiliary site can be represented as a two-level system. The Lindbladian $\mathcal{L}_{\gamma'}$ is just the same Lindbladian as in equation 5.7 but with $\kappa \to \gamma'$ and $\sigma_0^- \to d$.

The equation 5.18 is exact in the single-particle sector as one does not have to make a weak-coupling approximation because there can not be multiple particles transferred into the reservoir at the same time. With other system Hamiltonians, the equation of motion 5.18 will only be physically approximate, but it will give valid quantum evolution in all of parameter space. Further, if the state of the system has more than one particle, one has to represent the auxiliary site as a higher dimensional object in order to get exact evolution in the $\gamma \to 0$ limit. In addition, if one has a different system-reservoir interaction, for instance with $L = \sigma^x$ in equation 5.3, one still has an approximate master equation 5.18, although to make physical correspondence with the actual systems dynamics, one has to numerically check for convergence in terms of the Hilbert space dimension of the auxiliary site.

For more complicated reservoir structure, one has to define a sequence of transformations that takes the original reservoir into a one-dimensional lattice of auxiliary sites as shown in Figure 5.2(b). Each transformation in the sequence has the same structure, only one performs it on the new reservoir. The transformation is a Householder transformation [74]. We now briefly describe the series of transformations for a general H_R and H_I in equations 5.2 and 5.3.

The approach for general H_R is computational. The first step in the procedure is to choose a finite number, M, of modes to represent the reservoir. Then, we start the transformation process with the matrix

$$T_0 = \begin{bmatrix} 0 & \vec{g}^{\dagger} \\ \vec{g} & \tilde{H}_R \end{bmatrix} , \tag{5.19}$$

where $\vec{g}$ represents a vector containing the coupling constants in equation 5.3, which we will assume

[¶]There is also some freedom in choosing the density of states for this transformation but this choice does not affect the end results.

to be real, and $\tilde{H}_R$ represents the matrix of coefficients, also assumed to be real, of the quadratic form 5.2. The "0" element has no effect on the transformations. The idea, now, is to define a series of orthogonal transformations, P_i, where each one creates a new auxiliary site, A_i, and a new reservoir, R_i. At each step in the series, the orthogonal matrix is defined as

$$P_i = \mathrm{I} - 2\frac{\vec{\mu}_i \cdot \vec{\mu}_i^{\dagger}}{|\vec{\mu}_i|^2}\,, \tag{5.20}$$

where the vector $\vec{\mu}_i$ is

$$\vec{\mu}_i = \vec{x}_i \pm |\vec{x}_i|\,\vec{e}_{i+1}\,. \tag{5.21}$$

The $\vec{e}_i$ is a vector with a one in the i^{th} position and the vector $\vec{x}_i$ is

$$\vec{x}_i = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ (T_{i-1})_{i+1,i} \\ \vdots \\ (T_{i-1})_{M,i} \end{bmatrix}, \tag{5.22}$$

e.g., $\vec{x}_i$ has zeros into the $(i+1)^{th}$ position and then has elements of the corresponding column in T_{i-1}. The matrix P_i defined in equation 5.20 is such that [74]

$$P_i = P_i^{\dagger} = P_i^{-1}\,. \tag{5.23}$$

Thus, the transformation constructs a new T_i as

$$T_i = P_i T_{i-1} P_i\,. \tag{5.24}$$

The matrix T_1, for instance, will have the form

$$T_1 = \begin{bmatrix} 0 & \kappa_1 & 0 \\ \kappa_1 & \omega_1 & \vec{g}_1 \\ 0 & \vec{g}_1 & \tilde{H}_{1R} \end{bmatrix}, \tag{5.25}$$

where κ_1 is the coupling of the site to the first auxiliary site, ω_1 is the energy of the first auxiliary site, $\vec{g}_1$ is the coupling constant of the auxiliary site to the new reservoir, and $\tilde{H}_{1R}$ is the new reservoir. One continues to perform the series of transformations until all modes of the original reservoir are accounted for.

For practical use of this series of transformations, one of two scenarios should occur. Either one can stop the transformation after a small number of steps, e.g., as can be done for the Lorentzian reservoir above, or the series has to be artificially terminated. In the former case, one could directly simulate the system and the auxiliary sites or use numerical renormalization techniques. In the latter case, one would have to use numerical renormalization techniques and the length of the one-dimensional reservoir will limit the timescales reachable by the simulation. There are, of course, families of reservoirs that are exactly, or approximately to a high degree of accuracy, described by a finite number of auxiliary sites connected to Markovian reservoirs. These are basically cases where the reservoir has a structured set of poles. In these cases, the usage of the matrix product state algorithm described in Ref. 58 would be useful.

Finally, we would like to discuss this procedure in the context of other methods. It was found in Ref. 75 and Ref. 76 that systems with Lorentzian reservoirs can have exact Markovian master equations. However, the correspondence was obtained at the level of the equations of motion, and not by a transformation of the Hamiltonian. The latter allows one not only to see the correspondence between the Markovian master equations and the non-Markovian process, but also to derive approximate, including finite temperature, master equations that can be used generically. Further, the sequence of transformations allows one to get good approximations to reservoirs that are not described by simple poles. The procedure is also used in the Wilson chain construction of the numerical renormalization group[27]. Indeed, one of the main conclusions of this paper is that to numerically approach problems of a system interacting strongly with an environment one should use numerical renormalization-like techniques, unless the reservoir chain can be truncated at a small number of auxiliary sites.

To conclude, we have briefly examined issues arising when trying to use non-Markovian terms generically within a master equation. To alleviate these issues, we have proposed performing a series of transformations on the reservoir Hamiltonian. Each one divides the current reservoir into two parts, a local auxiliary mode that should be included with the system and a new reservoir that is only coupled to the auxiliary mode. In cases where the reservoir is a Lorentzian spectral function, a single transformation leaves one with a Markovian-like new reservoir, which yields an exact Markovian equation in some situations. If the original reservoir is more complex, one has to continue performing similar transformations, which leave the reservoir represented as a one-dimensional lattice, possibly connected to Markovian reservoirs. To study finite time properties, one can always truncate this one-dimensional lattice and have an approximate evolution equation, that nevertheless is always positive. In addition, the form of the new reservoir lattice can be simulated with matrix product states. Of particular use is matrix product techniques to represent density matrices[58,59].

5.3 Another approach to generic non-Markovian dynamics

The approach described in 5.2, although not proven, is probably the best way to tackle systems with non-Markovian dynamics. Although, there have been other proposals for generic approaches/forms of non-Markovian dynamics. Recently, a "post-Markovian" master equation was given by Shabani and Lidar[61]. The post-Markovian equation includes memory effects, like the Born approximation, but retains complete positivity for a wider range of system-environment coupling strengths and memory times.[61] In this section we will both examine in more detail the amplitude damping of single spin and analyze the Shabani-Lidar (SL) equation. We will find that the SL equation does not include physics beyond weak coupling and short memory times, because the equation does not systematically expand in these parameters but rather just cuts off undesirable behavior of the master equation. This gives further weight to our proposal in 5.2.

The SL equation was proposed as an interpolation between Markovian evolution, which can be viewed as a continuous measurement process, and the exact Kraus map, which can be viewed as the full dynamics followed by a measurement at the time for which observables are wanted. With this idea in mind, Shabani and Lidar derived the equation of motion

$$\begin{aligned}\frac{\partial \rho^I(t)}{\partial t} &= \int_0^t dt' \Big[k(t',t)\Lambda(t')\dot{\Lambda}(t-t')\Lambda^{-1}(t-t') \\ &\quad + \frac{\partial k(t',t)}{\partial t}\Lambda(t')\Big]\rho^I(t-t')\,, \end{aligned} \tag{5.26}$$

where $\rho^I(t)$ is the reduced density matrix for the system in the interaction picture, $k(t',t)$ is the memory kernel, and $\Lambda(t)$ is the evolution superoperator.[61] To make further progress they assumed a Markovian form of the evolution superoperator and a kernel which is only dependent on the time difference (i.e., $k(t',t) = k(t'-t)$) to get the tractable non-Markovian master equation

$$\frac{\partial \rho^I(t)}{\partial t} = \mathcal{L}\int_0^t dt' k(t') e^{\mathcal{L}t'}\rho^I(t-t')\,, \tag{5.27}$$

where $\mathcal{L}$ is the time-independent, Markovian Lindbladian. This equation, which we call the Shabani-Lidar (SL) equation, is similar to the Born master equation, except with $\rho^I(t-t')$ replaced with $e^{\mathcal{L}t'}\rho^I(t-t')$. Maniscalco and Petruccione showed that the SL equation preserves positivity in the complete range of parameter space (for all values of the system-reservoir coupling strength and reservoir memory time) for a specific model Hamiltonian, whereas the Born approximation fails to be positive for strong dissipation or long memories.[72]

In their work, Shabani and Lidar also give three requirements for an adequate post-Markovian equation: *(i)* goes beyond the Markovian master equation by including memory effects, *(ii)* analytically and numerically tractable, and *(iii)* completely positive.[61] One additional requirement that we

made explicit above is *(iv)* the post-Markovian equation should also be as generic as possible, i.e., that regardless of the system Hamiltonian and the system operators that interact with the reservoir we can still use the same generic term for the (non-Markovian) dissipation. This is a property held by the Markovian master equation: the validity of the master equation is highly independent of the system Hamiltonian and its form is independent of the system operators that interact with the reservoir.

In an effort to understand generic completely positive non-Markovian master equations, we reexamine the model of Maniscalco and Petruccione from an equation-of-motion perspective. We study the non-Markovian equations of motion for the exact dynamics, the Born approximation, and the SL equation. We first derive using elementary means the equations of motion, from which we can see physically how the two approximate methods come from the exact equations. We then move to discuss the breakdown in positivity of the Born approximation. For this particular model, the Keldysh approach in 5.4 shows that the break down in positivity can be fixed by taking a modified memory kernel, which gives the exact non-Markovian master equation. This indicates why the SL equation restores positivity to the region of parameter space where the Born approximation fails: the SL equation gives a strong, "Markovian" modification of the memory kernel.

5.3.1 Non-Markovian amplitude damping

In this subsection, we focus on the particular model discussed above, which is amenable to analytic treatment. The scenario is a single spin or two-level system interacting with a bosonic reservoir with Hamiltonian 5.1 and with

$$H_S = \frac{\omega_0}{2}\sigma_z \ , \tag{5.28}$$

$$H_R = \sum_k \omega_k b_k^\dagger b_k \ , \tag{5.29}$$

and

$$H_I = \sigma_+ B + \sigma_- B^\dagger \, , \tag{5.30}$$

where $B = \sum_k g_k b_k$ and $\sigma_\pm = (\sigma_x \pm i\sigma_y)/2$.

5.3.1.1 Exact

At zero temperature the above model can be solved exactly by exploiting particle number conservation, which was shown in Ref. 56. The total number operator, $N = \sigma_+\sigma_- + \sum_k b_k^\dagger b_k$, commutes with the Hamiltonian. Thus, when the reservoir is at zero temperature, we can restrict ourselves to

the zero and one particle subspaces which are spanned by the states

$$\begin{aligned} |\psi_0\rangle &= |0\vec{0}\rangle\,, \\ |\psi_1\rangle &= |1\vec{0}\rangle\,, \\ |\psi_k\rangle &= |0\vec{0}/1_k\rangle\,, \end{aligned} \tag{5.31}$$

where $\vec{0}$ indicates an empty vacuum and $\vec{0}/1_k$ indicates an empty vacuum except mode k which has occupation 1. Following Ref. 56, we can then expand the interaction picture wavefunction, $|\psi^I(t)\rangle$, in terms of these states

$$|\psi^I(t)\rangle = c_0(t)|\psi_0\rangle + c_1(t)|\psi_1\rangle + \sum_k c_k(t)|\psi_k\rangle\ . \tag{5.32}$$

Then solve the following equation

$$\frac{\partial|\psi^I(t)\rangle}{\partial t} = -iH_I^I(t)|\psi^I(t)\rangle\,, \tag{5.33}$$

with

$$H_I^I(t) = \sigma_+^I(t)B^I(t) + \sigma_-^I(t)B^{I\dagger}(t)\ . \tag{5.34}$$

The interaction picture operators are

$$\sigma_\pm^I(t) = \sigma_\pm e^{\pm i\omega_0 t}\,, \tag{5.35}$$

and

$$B^I(t) = \sum_k g_k b_k e^{-i\omega_k t}\ . \tag{5.36}$$

The equations of motion for the coefficients are

$$\dot{c}_0(t) = 0\ , \tag{5.37}$$

$$\dot{c}_1(t) = -i\sum_k g_k e^{i(\omega_0-\omega_k)t}c_k(t)\ , \tag{5.38}$$

$$\dot{c}_k(t) = -ig_k^\star e^{-i(\omega_0-\omega_k)t}c_1(t)\ . \tag{5.39}$$

Integrating the equations for the reservoir modes to get

$$c_k(t) = \int_0^t dt'(-ig_k^\star)e^{-i(\omega_0-\omega_k)t'}c_1(t')\,, \tag{5.40}$$

then putting this into the equation for $\dot{c}_1(t)$

$$\begin{aligned}\dot{c}_1(t) &= -\sum_k |g_k|^2 \int_0^t dt' e^{i(\omega_0-\omega_k)(t-t')} c_1(t') \\ &\equiv -\int_0^t dt' \alpha(t-t') e^{i\omega_0(t-t')} c_1(t')\,, \end{aligned} \tag{5.41}$$

where

$$\begin{aligned}\alpha(t-t') &= \mathrm{tr}_R\left\{B^I(t)B^{I\dagger}(t')\rho_R(\beta=\infty)\right\} \\ &= \int_0^\infty d\omega J(\omega) e^{-i\omega(t-t')}\,, \end{aligned} \tag{5.42}$$

is the reservoir correlation function when in a thermal state $\rho_R(\beta) = e^{-\beta H_R}/Z_R(\beta)$ at zero temperature. We will take a specific form for the reservoir correlation function, which is discussed in section 5.3.4.1 in terms of the memory kernel. The full equation of motion for the reduced density matrix can then be obtained by tracing out the reservoir [56]

$$\frac{\partial \rho_S^I(t)}{\partial t} = \frac{\partial}{\partial t}\mathrm{tr}_R\left\{|\psi^I(t)\rangle\langle\psi^I(t)|\right\} \tag{5.43}$$

$$= \begin{pmatrix} \frac{\partial |c_1(t)|^2}{\partial t} & c_0^\star \dot{c}_1(t) \\ c_0 \dot{c}_1^\star(t) & -\frac{\partial |c_1(t)|^2}{\partial t} \end{pmatrix}. \tag{5.44}$$

We now depart from previous investigations to examine the structure of the equations of motion. The diagonal elements of the reduced density matrix can be given in terms of the number operator $n(t) = \left\langle \sigma_+^I(t)\sigma_-^I(t)\right\rangle = |c_1(t)|^2$ where the expectation value is with respect to $\rho_S^I(t)$. The number operator obeys the equation of motion

$$\begin{aligned}\frac{\partial n(t)}{\partial t} &= \dot{c}_1^\star(t)c_1(t) + c_1^\star(t)\dot{c}_1(t) \\ &= -\int_0^t dt' \alpha^\star(t-t') e^{-i\omega_0(t-t')} c_1^\star(t')c_1(t) \\ &\quad -\int_0^t dt' \alpha(t-t') e^{i\omega_0(t-t')} c_1^\star(t)c_1(t')\,. \end{aligned} \tag{5.45}$$

If we then recognize that

$$\begin{aligned}\langle\sigma_+(t)\sigma_-(t')\rangle^H &= \langle\psi^H|\sigma_+(t)\sigma_-(t')|\psi^H\rangle \\ &= \langle\psi^I(t)|\sigma_+^I(t)\mathcal{T}e^{-i\int_{t'}^t dt'' H_I^I(t'')}\sigma_-^I(t')|\psi^I(t')\rangle \\ &= e^{i\omega_0(t-t')}c_1^\star(t)c_1(t')\langle\vec{00}|\mathcal{T}e^{-i\int_{t'}^t dt'' H_I^I(t'')}|\vec{00}\rangle \\ &= e^{i\omega_0(t-t')}c_1^\star(t)c_1(t')\,, \end{aligned} \tag{5.46}$$

where the last line follows from number conservation (or by just expanding the time-ordered exponent). Inserting equation 5.46 into equation 5.45 we get

$$\begin{aligned}\frac{\partial n(t)}{\partial t} &= -\int_0^t dt' \alpha^\star(t-t') \left\langle \sigma_+(t')\sigma_-(t)\right\rangle^H \\ &\quad -\int_0^t dt' \alpha(t-t') \left\langle \sigma_+(t)\sigma_-(t')\right\rangle^H \\ &= -2\mathrm{Re}\left\{\int_0^t dt' \alpha(t-t') \left\langle \sigma_+(t)\sigma_-(t')\right\rangle^H\right\} . \end{aligned} \tag{5.47}$$

Thus, the exact equation of motion for the particle number is related to the product of the bare reservoir correlation function and the exact two-time correlation function $\langle\sigma_+(t)\sigma_-(t')\rangle$ (e.g., the two-time correlation function with evolution according to the total Hamiltonian H). Of course, the particle number as a function of time can be solved by first solving for the coefficient $c_1(t)$ in equation 5.41. This has been done in Ref. 56 and the result is also presented in Ref. 72. Our purpose here, however, is to examine the equations of motion to get insight into the different approximations and the breakdown of positivity.

5.3.1.2 Born Approximation

The Born master equation is the second-order approximation

$$\frac{\partial \rho^I(t)}{\partial t} = \int_0^t dt' \mathrm{tr}_R\left[H_I^I(t), \left[H_I^I(t'), \rho^I(t')\otimes\rho_R\right]\right] . \tag{5.48}$$

At zero temperature the Born master equation for our particular case becomes

$$\begin{aligned}\frac{\partial \rho^I(t)}{\partial t} &= -\int_0^t dt' \alpha(t'-t) e^{-i\omega_0(t-t')} \\ &\quad \times\left[\rho^I(t')\sigma_+\sigma_- - \sigma_-\rho^I(t')\sigma_+\right] \\ &\quad -\int_0^t dt' \alpha(t-t') e^{i\omega_0(t-t')} \\ &\quad \times\left[\sigma_+\sigma_-\rho^I(t') - \sigma_-\rho^I(t')\sigma_+\right] . \end{aligned} \tag{5.49}$$

The two-time correlation function with evolution according to H_0 starting with the state at time t'

$$\begin{aligned}\left\langle \sigma_+(t)\sigma_-(t')\right\rangle^{H_0} &= \mathrm{tr}\left\{\sigma_+^I(t)\sigma_-^I(t')\rho^I(t')\right\} \\ &= e^{i\omega_0(t-t')}\mathrm{tr}\left\{\sigma_+\sigma_-\rho^I(t')\right\} \\ &= e^{i\omega_0(t-t')} n(t') . \end{aligned} \tag{5.50}$$

Using this and $\alpha(t'-t) = \alpha^\star(t-t')$, we can compute the equation of motion for the number operator

$$\begin{aligned}\dot{n}(t) &= \mathrm{tr}\left\{\sigma_+\sigma_-\frac{\partial\rho^I(t)}{\partial t}\right\} \\ &= -2\mathrm{Re}\left\{\int_0^t dt'\alpha(t-t')\left\langle\sigma_+(t)\sigma_-(t')\right\rangle^{H_0}\right\}. \end{aligned} \tag{5.51}$$

Alternatively, we can define the memory kernel

$$k(t-t') = \alpha(t-t')e^{i\omega_0(t-t')}\ , \tag{5.52}$$

which is computed in section 5.3.4.1 for an easily solvable case, and a normalized memory kernel

$$\bar{k}(t-t') = \frac{2}{\kappa}\alpha(t-t')e^{i\omega_0(t-t')}\,, \tag{5.53}$$

which satisfies

$$\int_0^\infty dt\bar{k}(t) = 1\ . \tag{5.54}$$

Assuming that the reservoir correlation function is symmetric in time, $\alpha(t) = \alpha(-t)$, which will be the case for the reservoir in a thermal state, the Born master equation for the particular model above is

$$\frac{\partial\rho^I(t)}{\partial t} = \int_0^t dt'\bar{k}(t-t')\mathcal{L}\rho^I(t')\,, \tag{5.55}$$

where $\mathcal{L}$ is the Markovian Lindbladian in equation 5.7. That the Lindbladian is time independent is not always going to be the case, but here one can be defined because the phase factor $e^{-i\omega_0(t-t')}$ can be pulled out and included with the reservoir correlation function. In section 5.3.4.2 we show this gives an equation of motion for the number operator

$$\dot{n}(t) = -\kappa\int_0^t dt'\bar{k}(t-t')n(t')\ . \tag{5.56}$$

In defining the memory kernel 5.53, we have hidden the fact that equation 5.56 also has an two-time correlation function like the exact equation of motion 5.47. Using 5.50 in 5.56 (and that $\bar{k}(t-t') = 2k(t-t')/\kappa$) gives

$$\begin{aligned}\dot{n}(t) &= -2\int_0^t dt'\alpha(t-t')e^{i\omega_0(t-t')}e^{-i\omega_0(t-t')}\left\langle\sigma_+(t)\sigma_-(t')\right\rangle^{H_0} \\ &= -2\int_0^t dt'\alpha(t-t')\left\langle\sigma_+(t)\sigma_-(t')\right\rangle^{H_0}\ . \end{aligned} \tag{5.57}$$

In this particular case the integrand is already real, so equation 5.57 is equivalent to 5.51.

5.3.1.3 Shabani-Lidar Equation

The SL equation 5.27 can be written for the particular model in a similar way as the Born approximation, see section 5.3.4.2. The equation of motion for the number operator is

$$\dot{n}(t) = -\kappa \int_0^t dt' \bar{k}(t-t') e^{-\kappa(t-t')} n(t') \ . \tag{5.58}$$

We have used the memory kernel in the Born approximation, which need not be done. One could use another kernel but it is not clear this will affect the conclusions of this chapter. As with the exact equation and the Born approximation, we can rewrite this equation of motion using a correlation function. Now, however, we have to define a correlation function that evolves with the bare Hamiltonian and Markovian dissipation[‖]

$$\begin{aligned} \langle \sigma_+(t)\sigma_-(t') \rangle^{H_0,\mathcal{L}} &= \mathrm{tr}\left\{ \sigma_+^I(t) e^{\mathcal{L}(t-t')} \sigma_-^I(t') \rho^I(t') \right\} \\ &= e^{i\omega_0(t-t')} e^{-\kappa(t-t')} \mathrm{tr}\left\{ \sigma_+ \sigma_- \rho^I(t') \right\} \\ &= e^{i\omega_0(t-t')} e^{-\kappa(t-t')} n(t') \ . \end{aligned} \tag{5.59}$$

Putting this into 5.58 (and again using that $\bar{k}(t-t') = 2k(t-t')/\kappa$) gives

$$\dot{n}(t) = -2 \int_0^t dt' \alpha(t-t') \left\langle \sigma_+(t)\sigma_-(t') \right\rangle^{H_0,\mathcal{L}} \ . \tag{5.60}$$

We will see what equation 5.60 means diagrammatically in 5.4.

5.3.2 Solution

Taking the memory kernel to be as in section 5.3.4.1, equations 5.41, 5.56, and 5.58 can all be solved by Laplace transform. Again, the solution was presented in Ref. 72, so we only want to summarize the results here. The number operator as a function of time[72] was found to be

$$n(t) = n(0) e^{-\gamma t} \left| \frac{1}{\sqrt{1-2R}} \sinh\left[\frac{\gamma t}{2} \sqrt{1-2R} \right] + \cosh\left[\frac{\gamma t}{2} \sqrt{1-2R} \right] \right|^2 \quad \text{Exact}\,, \tag{5.61}$$

$$n(t) = n(0) e^{-\gamma t/2} \left\{ \frac{1}{\sqrt{1-4R}} \sinh\left[\frac{\gamma t}{2} \sqrt{1-4R} \right] + \cosh\left[\frac{\gamma t}{2} \sqrt{1-4R} \right] \right\} \quad \text{Born}\,, \tag{5.62}$$

$$n(t) = n(0) \left[\frac{e^{-\kappa t} - R e^{-\gamma t}}{1-R} \right]) \quad \text{SL}\,, \tag{5.63}$$

for the three different treatments. And here $R = \frac{\kappa}{\gamma}$. We can see how the Born approximation will violate positivity. When the parameter R becomes large enough the $\sinh, \cosh$ become $\sin, \cos$ and

[‖] That is, if we take at each time t' in the past the density matrix in the interaction picture and evolve according to the Markovian master equation $\dot{\rho}^I(t) = \mathcal{L}\rho^I(t)$. This equation can be obtained from the Born master equation 5.49 by making the assumption that $\rho^I(t)$ does not change much on the timescale of the memory kernel.

thus oscillate from positive to negative values. And unlike the exact solution, the Born approximation does not have the absolute value squared of these oscillating quantities.

5.3.3 Summary

The main result of this section is comparison of the different non-Markovian equations of motion for the particle number:

$$\begin{aligned}
\frac{\partial n(t)}{\partial t} &= -2\mathrm{Re}\left\{\int_0^t dt'\alpha(t-t')\left\langle\sigma_+(t)\sigma_-(t')\right\rangle^{H}\right\} \quad \text{Exact}\,, \\
\frac{\partial n(t)}{\partial t} &= -2\mathrm{Re}\left\{\int_0^t dt'\alpha(t-t')\left\langle\sigma_+(t)\sigma_-(t')\right\rangle^{H_0}\right\} \quad \text{Born}\,, \\
\frac{\partial n(t)}{\partial t} &= -2\mathrm{Re}\left\{\int_0^t dt'\alpha(t-t')\left\langle\sigma_+(t)\sigma_-(t')\right\rangle^{H_0,\mathcal{L}}\right\} \quad \text{SL}\,,
\end{aligned}$$

Comparing these equations of motion for the three different treatments, we can see what they amount to physically.

The Born equation replaces the exact two-time correlation function with the two-time correlation function due only to H_0 evolution from the state at a past time (i.e., the state in its memory). Such a replacement is valid so long as the memory time is short compared with the interaction time of the system with the reservoir. This is in essence the realm of Markovian dynamics. In fact, the only gain that is obtained from the Born master equation is that the initial transient when the system-reservoir coupling is turned on is more correctly described, but otherwise the equation of motion is not physically justified much beyond the parameter regime of the Markovian master equation.

The SL equation, on the other hand, replaces the exact two-time correlation function with one that takes into account that the system is connected to a reservoir. Thus, *a priori* it seems plausible it could correctly describe the physics beyond the parameter regime of the Markovian master equation and the Born approximation. However, by using the correlation function obtained from Markovian evolution two things are done: *(i)* it is assumed that the reservoir memory time is small (and by small this means compared to the system-reservoir interaction time, $1/\gamma \ll 1/\kappa$). *(ii)* a systematic expansion in the memory is neglected. That is, the Markovian evolution is zeroth order in the reservoir timescale. Thus, the SL equation fails to include memory dependence in a systematic way beyond the Born approximation. But this together with *(i)* $1/\gamma \ll 1/\kappa$, means that the SL equation is not physically valid in any more parameter space than the Born approximation.

We know, though, that the SL equation does something more than the Born approximation, as it restores positivity to all of parameter space. In the next section we will see why this is the case, and show on physical grounds how to restore positivity while retaining the physical justification of the equations.

5.3.4 Some subsidiary calculations

5.3.4.1 Reservoir correlation function

Taking the form for the reservoir spectral density to be a Lorentzian peaked about the system frequency

$$J(\omega) = \frac{1}{2\pi} \frac{\kappa\gamma^2}{(\omega-\omega_0)^2+\gamma^2} \Theta(\omega)\,, \tag{5.64}$$

(where the $\Theta(\omega)$ ensures that we only have positive frequency harmonic oscillators) gives a memory kernel

$$\begin{aligned} k(t) &= \int_{-\infty}^{\infty} d\omega J(\omega) e^{-it(\omega-\omega_0)} \\ &= \int_{0}^{\infty} d\omega \frac{1}{2\pi} \frac{\kappa\gamma^2}{(\omega-\omega_0)^2+\gamma^2} e^{-it(\omega-\omega_0)} \\ &= \int_{-\infty}^{\infty} d\omega \frac{1}{2\pi} \frac{\kappa\gamma^2}{\omega^2+\gamma^2} e^{-i\omega t} \\ &\quad - \int_{-\infty}^{-\omega_0} d\omega \frac{1}{2\pi} \frac{\kappa\gamma^2}{\omega^2+\gamma^2} e^{-i\omega t} \\ &= \frac{\kappa\gamma}{2} e^{-\gamma|t|} + \mathcal{O}\left(\frac{\gamma}{\omega_0}\right), \end{aligned} \tag{5.65}$$

where we obtained the last line by bounding the integral

$$\begin{aligned} \int_{-\infty}^{-\omega_0} d\omega \frac{1}{2\pi} \frac{\kappa\gamma^2}{\omega^2+\gamma^2} e^{-i\omega t} &\leq \int_{-\infty}^{-\omega_0} d\omega \frac{1}{2\pi} \frac{\kappa\gamma^2}{\omega^2+\gamma^2} \\ &= \frac{\kappa\gamma^2}{2\pi} \left. \frac{\tan^{-1}\left(\frac{\omega}{\gamma}\right)}{\gamma} \right|_{-\infty}^{-\omega_0} \\ &= \frac{\kappa\gamma}{2\pi} \left[\tan^{-1}\left(-\frac{\omega_0}{\gamma}\right) + \frac{\pi}{2} \right] \\ &< \frac{\kappa\gamma}{2\pi} \left[-\frac{\pi}{2} + \frac{\gamma}{\omega_0} + \frac{\pi}{2} \right] \\ &= \frac{\kappa\gamma^2}{2\pi\omega_0}\,. \end{aligned} \tag{5.66}$$

This bounds $k(0)$ tightly, which means

$$\frac{\gamma}{\omega_0} \ll 1\,. \tag{5.67}$$

For $t \neq 0$, it is tougher to bound, but we believe the exponential will be a better and better approximation because in the integral $e^{i\omega t}$ oscillates faster and faster. Thus, the main contribution to the integral becomes more and more localized around $\omega = 0$, which is within the original limits. The bound 5.67 just indicates that the reservoir states have to be high in energy.

5.3.4.2 Damping Basis

We can use the damping basis[77] to rewrite the equations of motion for the Born approximation and the SL equation. For the Hamiltonian 5.1 with the reservoir at zero temperature, the right eigenoperators and eigenvalues, $\mathcal{L}\rho_i = \lambda_i \rho_i$, are

$$(\rho_i, \lambda_i) \in \left\{ (\delta, 0), (\sigma_z, -\kappa), \left(\sigma_+, -\frac{\kappa}{2}\right), \left(\sigma_-, -\frac{\kappa}{2}\right) \right\}, \tag{5.68}$$

where $\delta = (I - \sigma_z)/2$. Expanding the density matrix as

$$\rho^I(t) = \sum_i d_i(t)\rho_i, \tag{5.69}$$

the equations of motion for the Born and SL equations become

$$\begin{aligned}
\dot{d}_0(t) &= 0, && (5.70)\\
\dot{d}_1(t) &= -\kappa \int_0^t dt' \bar{k}(t-t') d_1(t'), && (5.71)\\
\dot{d}_2(t) &= -\frac{\kappa}{2} \int_0^t dt' \bar{k}(t-t') d_2(t'), && (5.72)\\
\dot{d}_3(t) &= -\frac{\kappa}{2} \int_0^t dt' \bar{k}(t-t') d_3(t'), && (5.73)
\end{aligned}$$

for the Born approximation and

$$\begin{aligned}
\dot{d}_0(t) &= 0, && (5.74)\\
\dot{d}_1(t) &= -\kappa \int_0^t dt' \bar{k}(t-t') e^{-\kappa t'} d_1(t'), && (5.75)\\
\dot{d}_2(t) &= -\frac{\kappa}{2} \int_0^t dt' \bar{k}(t-t') e^{-\kappa t'/2} d_2(t'), && (5.76)\\
\dot{d}_3(t) &= -\frac{\kappa}{2} \int_0^t dt' \bar{k}(t-t') e^{-\kappa t'/2} d_3(t'), && (5.77)
\end{aligned}$$

for the SL equation. The number operator is

$$n(t) = \mathrm{tr}\left\{\sigma_+^I(t)\sigma_-^I(t)\rho^I(t)\right\} = d_1(t)\ . \tag{5.78}$$

The equation of motions for the number operator are then

$$\dot{n}(t) = -\kappa \int_0^t dt' \bar{k}(t-t') n(t'), \tag{5.79}$$

for the Born approximation and

$$\dot{n}(t) = -\kappa \int_0^t dt' \bar{k}(t-t') e^{-\kappa t'} n(t') \,, \tag{5.80}$$

for the SL equation.

5.4 Keldysh approach for non-Markovian equations

In this section, we use the Keldysh formalism[54,78] to derive the exact form of the non-Markovian master equation for a set of bosonic and fermionic environments. We find that the form of the exact equation does not allow one to construct a valid master equation if the system dynamics are not *a priori* known. We come to two overall conclusions for the construction and use of non-Markovian master equations: *(1)* If the physics desired is somehow contained only within the memory kernel, then it is not worthwhile to construct non-Markovian master equations because the memory kernel itself is just as complicated of an object as the final solution. *(2)* It is unlikely a generic non-Markovian equation can be obtained by partitioning into a system and reservoir, which is the typical approach to non-Markovian dynamics and open systems in general. The observations leading to the latter conjecture lend weight to the proposal in 5.2. The Keldysh approach also enables us to get better insight to the breakdown of positivity and what is missing in the SL equation above.

5.4.1 Equations of motion

In particular, here we use the Keldysh formalism to get the equation of motion for the particle number. We will see that the exact equation of motion 5.47 and Born equation 5.51 come out of this formalism. But further we will see that to derive a proper non-Markovian equation (for the particle number), it is necessary to sum an infinite order of diagrams. Doing this sum to get a good memory kernel, however, also gives you the final solution. In this particular model it turns out that for the reservoir at zero temperature this also gives you the exact non-Markovian equation of motion.

First we define the lesser Green's function as

$$G^<(t,t') = i \langle \sigma_+(t') \sigma_-(t) \rangle \,, \tag{5.81}$$

which gives us the number operator

$$n(t) = \langle \sigma_+(t) \sigma_-(t) \rangle = -i \lim_{t' \to t} G^<(t,t') \,. \tag{5.82}$$

Thus, the equation of motion for the number operator will be given by

$$\frac{\partial n(t)}{\partial t} = -i \lim_{t' \to t} \left(\frac{\partial}{\partial t} + \frac{\partial}{\partial t'} \right) G^{<}(t,t') \; . \tag{5.83}$$

The equations of motion for the lesser Green's function are[78]

$$\begin{aligned} \left[i\frac{\partial}{\partial t} + \omega_0 \right] G^{<}(t,t') &= \int_{t_0}^{\infty} dt_1 \Sigma^{r}(t,t_1) G^{<}(t_1,t') \\ &\quad + \int_{t_0}^{\infty} dt_1 \Sigma^{<}(t,t_1) G^{a}(t_1,t') \, , \end{aligned} \tag{5.84}$$

and

$$\begin{aligned} \left[-i\frac{\partial}{\partial t'} + \omega_0 \right] G^{<}(t,t') &= \int_{t_0}^{\infty} dt_1 G^{<}(t,t_1) \Sigma^{a}(t_1,t') \\ &\quad + \int_{t_0}^{\infty} dt_1 G^{r}(t,t_1) \Sigma^{<}(t_1,t') \, , \end{aligned} \tag{5.85}$$

where t_0 is the time when the reservoir is connected to the system (we have taken $t_0 = 0$), and we have used the retarded and advanced self-energies, $\Sigma^{r,a}(t,t')$ and Green's functions $G^{r,a}(t,t')$. The retarded self-energy is given by

$$\Sigma^{r}(t,t') = \sum_{k} |g_k|^2 \, d_k^{r}(t,t') \; , \tag{5.86}$$

where

$$d_k^{r}(t,t') = -\imath \Theta(t-t') \, e^{-\imath \omega_k (t-t')} \, , \tag{5.87}$$

is the bosonic (or can take as fermionic) retarded Green's function. For the particular case of the Lorentzian reservoir above, the retarded self-energy is

$$\Sigma^{r}(\omega) = \frac{\kappa\gamma}{2} \frac{1}{\omega - \omega_0 + \imath\gamma} \, , \tag{5.88}$$

which in real-time can be related to the Born memory kernel

$$\begin{aligned} \Sigma^{r}(t,t') &= -\imath\theta(t-t') \, e^{\imath\omega_0 (t-t')} k(t-t') \\ &= [\Sigma^{a}(t',t)]^{\star} \; . \end{aligned}$$

The lesser self-energy $\Sigma^{<}(t,t')$ is given by

$$\begin{aligned}\Sigma^{<}(t,t') &= \sum_k |g_k|^2 \, d_k^{<}(t,t') &(5.89)\\ &= i\sum_k |g_k|^2 \left\langle b_k^{\dagger}(t')b_k(t)\right\rangle_R \ , &(5.90)\end{aligned}$$

where $d_k^{<}(t,t')$ is the bosonic lesser Green's function. When the reservoir is at zero temperature, the bosonic correlation functions in 5.89 are zero. Thus, this finite temperature source for particles drops out at zero temperature. Taking this into account and subtracting equation 5.84 from equation 5.85 we get

$$\begin{aligned}-i\left(\frac{\partial}{\partial t}+\frac{\partial}{\partial t'}\right)G^{<}(t,t') &= \int_{t_0}^{\infty} dt_1 \left[-\Sigma^r(t,t_1)G^{<}(t_1,t')\right.\\ &\quad \left. G^{<}(t,t_1)\Sigma^a(t_1,t')\right] \ . &(5.91)\end{aligned}$$

Taking the limit $t' \to t$ now, while recognizing that $\Sigma^a(t,t') = [\Sigma^r(t',t)]^{\star}$and that $G^{<}$ is defined as the correlation function with an i in front (so the Im part of this equation is equivalent to the Re part of 5.47) gives us equation 5.47, which we derived above with an alternative method. However, with the Keldysh formalism we can go further by using the integral equation [78]

$$\begin{aligned}G^{<}(t,t') &= \int_{t_0}^{\infty} dt_2 \int_{t_0}^{\infty} dt_3 G^r(t,t_2)\Sigma^{<}(t_2,t_3)G^a(t_3,t')\\ &\quad +G^r(t,t_0)G^{<}(t_0,t_0)G^a(t_0,t') \, . &(5.92)\end{aligned}$$

At zero temperature there is no source so this equation reduces to

$$\begin{aligned}G^{<}(t,t') &= G^r(t,t_0)G^{<}(t_0,t_0)G^a(t_0,t')\\ &= \imath G^r(t,t_0)n(t_0)G^a(t_0,t') \ . &(5.93)\end{aligned}$$

We can use this zero-temperature relationship in 5.91. First, however, lets write the lesser Green's function in terms of some intermediate time particle number. At time t_1, we have

$$n\left(t_1\right) = G^r\left(t_1,t_0\right) n\left(t_0\right) G^a\left(t_0,t_1\right) \ . \tag{5.94}$$

Inverting this and putting into 5.93,

$$G^{<}\left(t,t'\right) = \imath G^r\left(t,t_0\right)\left[G^r\left(t_1,t_0\right)\right]^{-1} n\left(t_1\right)\left[G^a\left(t_0,t_1\right)\right]^{-1} G^a\left(t_0,t'\right) \ . \tag{5.95}$$

Taking the limit $t' \to t$ of 5.91,

$$\frac{\partial n(t)}{\partial t} = \int_{t_0}^{\infty} dt_1 \left[-\Sigma^r(t,t_1) G^<(t_1,t) + G^<(t,t_1) \Sigma^a(t_1,t) \right] .$$

Then we can use $G^<$ in terms of $n(t_1)$ and use that $G^a(t_1,t_1) = \imath = -G^r(t_1,t_1)$ to get

$$\begin{aligned} \frac{\partial n(t)}{\partial t} &= \int_{t_0}^{\infty} dt_1 \Big\{ -\imath \Sigma^r(t,t_1) n(t_1) \left[G^a(t_0,t_1)\right]^{-1} G^a(t_0,t) \\ &\quad + \imath G^r(t,t_0) \left[G^r(t_1,t_0)\right]^{-1} n(t_1) \Sigma^a(t_1,t) \Big\} \\ &= 2\Im \left\{ \int_{t_0}^{\infty} dt_1 \Sigma^r(t,t_1) n(t_1) \left[G^a(t_0,t_1)\right]^{-1} G^a(t_0,t) \right\} . \end{aligned} \tag{5.96}$$

But of course, by taking the limit $t' \to t$ in 5.93 we already know

$$\begin{aligned} n(t) &= G^r(t,t_0) n(t_0) G^a(t_0,t) \\ &= \left| G^r(t-t_0) \right|^2 n(t_0) , \end{aligned} \tag{5.97}$$

where we used that $G^a(t,t') = [G^r(t',t)]^\star$ and that the retarded Green's function is only dependent on the time difference from the initial time t_0. Due to the form of the number operator in equation 5.97, it is guaranteed to be positive. We also see where the absolute value squared comes from in the exact solution obtained above.

Let us consider equation 5.96. Because we are dealing with a number conserving system, the equation 5.96 is one term in the master equation for the full density matrix (the other diagonal term will be minus this term, and the off-diagonal elements are decoupled from the diagonal elements, where we can find the form of their derivative if we wanted it). The equation looks like a typical memory kernel master equation, if we identify the memory kernel with the exact kernel

$$k_{ex}(t,t_1) = \Im \left\{ 2\Sigma^r(t,t_1) \left[G^a(t_0,t_1)\right]^{-1} G^a(t_0,t) \right\} . \tag{5.98}$$

This shows, though, that the memory kernel that contains physics beyond the weak coupling regime involves the system propagator in the presence of the reservoir. Writing the advanced Green's function in terms of the conjugate of the retarded Green's function

$$G^r(t-t') = \left[G^a(t'-t)\right]^\star , \tag{5.99}$$

for which the Fourier transform is

$$G^r(\omega) = \frac{1}{\omega - \omega_0 - \Sigma^r(\omega)} . \tag{5.100}$$

Using the Lorentzian spectral function, the retarded Green's function comes out to be

$$G^r(t) = e^{-\gamma t/2}\left[\frac{1}{\sqrt{1-2R}}\sinh\left[\frac{\gamma t}{2}\sqrt{1-2R}\right] + \cosh\left[\frac{\gamma t}{2}\sqrt{1-2R}\right]\right] . \tag{5.101}$$

This gives solution above for the amplitude decay in the presence of a Lorentzian reservoir.

Now let us turn our attention to the exact memory kernel, equation 5.98. There are three observations that we can make concerning the form of the non-Markovian equation 5.96 with the kernel 5.98:

- We see that the memory kernel itself is a more complicated of an object than the solution 5.97 (or consider the case of the Lorentzian reservoir, with equation 5.101 in equation 5.98, which is brutal compared to 5.61). This has fairly profound implications for the usage of non-Markovian master equations for non-perturbative treatments. What it shows for this case is that to get a non-perturbative non-Markovian master equation, one has to do calculations that are just as complex as finding the solutions to begin with. This is going to be general, however. Only in situations like in 3.5.3 will master equations be useful. In that case, a transformation was applied to the Hamiltonian to non-perturbatively take into account the system-reservoir coupling, but a perturbative treatment was used for the system degrees of freedom. This is not to say that this case is not interesting, but nonetheless, perturbative.

- The memory kernel is a complex combination of system and reservoir interaction. Thus, it is not clear when one can reduce this to a less complicated or more generic form.

- It is interesting to examine in what case can we identify

$$\left[G^a\left(t_0,t_1\right)\right]^{-1} G^a\left(t_0,t\right) \to G^a\left(t_1,t\right) . \tag{5.102}$$

  This will be possible when we have a Markovian reservoir and the Green's functions $G^{r,a}$ are a simple exponential. However, in that case, the whole master equation will collapse into a Markovian one. It is also interesting that this case will sort of look like the SL equation, it is probably because the SL equation does not go beyond Markovian dynamics (also see below) by the insertion of the propagator $e^{\mathcal{L}t}$ into the memory function.

Thus, we see that, although we have only looked at a particularly simple case, we have learned some things about the structure of non-Markovian equations. Based on these observations, it is our believe that non-Markovian equations are not a reasonable approach to strongly dissipative systems.

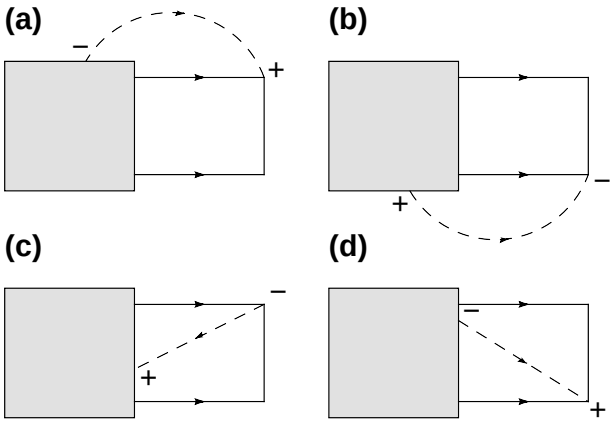


Figure 5.3: Kinetic equation diagrams for the Born approximation. The $-/+$ stand for a σ_-/σ_+ acting on the corresponding side of the density matrix. Each diagram contributes to a different matrix element in the self-energy superoperator. (a) takes $|1\rangle\langle 1| \Rightarrow |1\rangle\langle 1|$ and $|1\rangle\langle 0| \Rightarrow |1\rangle\langle 0|$, (b) takes $|1\rangle\langle 1| \Rightarrow |1\rangle\langle 1|$ and $|0\rangle\langle 1| \Rightarrow |0\rangle\langle 1|$, (c) takes $|1\rangle\langle 1| \Rightarrow |0\rangle\langle 0|$, and (d) takes $|1\rangle\langle 1| \Rightarrow |0\rangle\langle 0|$.

5.4.2 Full equation for the density matrix

The equation of motion for the full density matrix can also be written using the the Keldysh approach, we will not go into much detail. This full equation is called the kinetic equation and has the form

$$\frac{\partial \rho^I(t)}{\partial t} = \int_{t_0}^{\infty} dt' \Omega(t,t') \rho^I(t') \,, \tag{5.103}$$

where $\Omega(t,t')$ is a self-energy superoperator. Thus, if we write our density matrix as a superket, $|\rho^I(t')\rangle\rangle_\sharp$ (using the notation of Ref. 58) in the basis

$$\begin{pmatrix} |1\rangle\langle 1| \\ |0\rangle\langle 0| \\ |1\rangle\langle 0| \\ |0\rangle\langle 1| \end{pmatrix} , \tag{5.104}$$

we can represent the superoperator $\Omega(t,t')$ as a 4×4 matrix. The diagrams that contribute to the Born approximation of Ω are shown in figure 5.3.

The exact kinetic equation for the density matrix has a structure similar to the Born approximation. The diagrams in this case are shown in figure 5.4. The difference is the double line propagator, which is given by the Dyson equation in figure 5.5. Actually, this sweeps some stuff under the rug. Really one has a a partial evolution backward in time, and then a forward evolution with the exact propagator. This makes the diagrams more complex.

We do not learn anything further from the kinetic equation, as the observations one can make are essentially the same as the equation of motion for the number operator. Part of the reason for this is

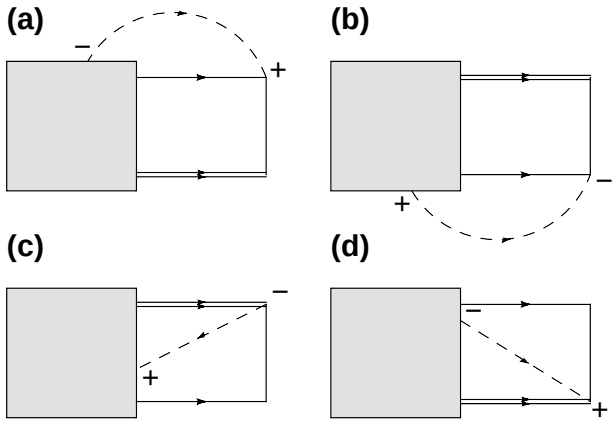


Figure 5.4: Kinetic equation diagrams for the exact kinetic equation. The $-/+$ stand for a σ_-/σ_+ acting on the corresponding side of the density matrix. Each diagram contributes to a different matrix element in the self-energy superoperator. (a) takes $|1\rangle\langle 1| \Rightarrow |1\rangle\langle 1|$ and $|1\rangle\langle 0| \Rightarrow |1\rangle\langle 0|$, (b) takes $|1\rangle\langle 1| \Rightarrow |1\rangle\langle 1|$ and $|0\rangle\langle 1| \Rightarrow |0\rangle\langle 1|$, (c) takes $|1\rangle\langle 1| \Rightarrow |0\rangle\langle 0|$, and (d) takes $|1\rangle\langle 1| \Rightarrow |0\rangle\langle 0|$.

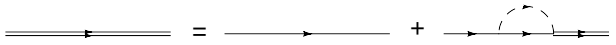


Figure 5.5: Propagator for the exact kinetic equation.

that the off-diagonal elements of the density matrix decay independently of the diagonal elements, which is because the Hamiltonian is particle conserving. However, figure 5.5 actually does give us a point of reference to compare to the SL equation.

5.4.3 Shabani-Lidar revisited

The SL equation can be represented diagrammatically the same as the exact equation in figure 5.4 but with the exact propagator of figure 5.5 replaced with the Markovian propagator in figure 5.6. It is interesting to note then that the physical scenario of the SL equation is to merge the Born and Markovian approximations into a non-Markovian master equation. However, by using the Markovian evolution operator, or equivalently the propagator in figure 5.6, the SL equation is not systematically expanding in the memory time of the reservoir, but rather is introducing a cutoff corresponding to the system-reservoir coupling strength. If this coupling gets to large compared to the memory time, the Markovian propagator will exponentially cutoff the memory. Thus, it is a very unphysical approach and it is unclear about whether the SL equation meets the goals given in Ref. 61.

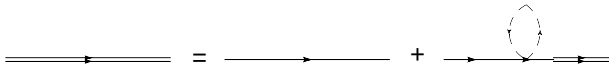


Figure 5.6: Markovian propagator for the SL equation. The propagator contains instantaneous interactions with the reservoir.

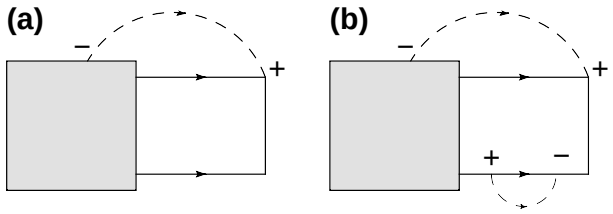


Figure 5.7: (a) A contribution to the Born approximation. (b) A contribution beyond the Born approximation.

5.4.4 Breakdown of positivity

Now we can see diagrammatically why the Born approximation is not consistent. Consider the two diagrams shown in figure 5.7. Diagram (a) is included in the Born approximation. If we suppose this second-order process is important, we are stating that the wavefunction of the system is evolving for a time τ such that

$$\tau \gtrsim \frac{1}{\kappa} \, . \tag{5.105}$$

That is, our system is evolving for long enough that it is likely an interaction will take place between the system and the reservoir. Given that, though, we also have to permit multiple interactions with the reservoir if its memory time is such that

$$\frac{1}{\gamma} \gtrsim \frac{1}{\kappa} \, . \tag{5.106}$$

For example, the process shown in figure 5.7(b) is likely to happen if the reservoir memory time is long compared to the interaction time of the system and reservoir. Exactly when the breakdown will occur will be dependent on the specific model, but consistency requires that the parameter γ/κ be small for a second-order approach to be valid. If γ/κ is not small, the additional processes like those in figure 5.7(b) are needed and these will tend to damp out the long time memory effects.

5.5 Conclusions

We have examined many issues in this chapter. First, we studied a simple non-Markovian process and found that if one indiscriminately uses a non-Markovian term in a master equation, that the resulting evolution is actually unstable with respect to positivity when one turns on an additional interaction in the Hamiltonian. Further, it is well known that approximate non-Markovian terms themselves can lose positivity when the system-reservoir coupling gets too strong compared to the reservoir memory time. Given this poor behavior, we proposed an approach to non-Markovian dynamics that does not rely on the construction of non-Markovian master equations. This approach is based on transforming the reservoir Hamiltonian into a form that allows for generic approximations

to be made. We saw for a specific case that after a single transformation of the Hamiltonian that one is left with a Markovian master equation that is, in some cases, exact, and in other cases, approximate, but at least completely positive and physically justified for long memory times. In addition, physical systems with long-time memories arising, for instance, from ohmic dissipation, can also be generically treated with this transformation approach as well. And indeed, this is the basis of the numerical renormalization group, and also our proposal later for branched lattice simulations. One final note on this method is that it is only valid for non-interacting reservoirs. However, if one takes the construction of a semi-infinite branch as the reservoir and uses MPS to do the simulation, one can use an interacting reservoir, but it will not necessarily be clear how this corresponds to an impurity-like reservoir.

To further explore the inadequacy of using non-Markovian master equations we have examined three treatments of open system dynamics for a particular Hamiltonian. Namely, we studied the exact solution and two approximate solutions: the Born approximation and a recently proposed "post-Markovian" master equation. The equation of motion of the particle number for the exact solution can be written as an integro-differential equation with a second order reservoir correlation function. However, the reservoir correlation function is integrated with a two-time correlation function of the full dynamics. The approximate solutions replace this two-time correlation function with one using some reduced description of the dynamics. The Born approximation uses only the bare Hamiltonian dynamics and the SL equation uses the two-time correlation function obtained from Markovian dynamics. Because of the lack of a systematic expansion in the reservoir timescale, though, the SL equation is not physically valid beyond the parameter regime of the Born approximation, in agreement with the findings of Ref. 72. However, the SL equation restores positivity to regions of parameter space where the Born approximation is not positive. Using a Keldysh approach, we saw that it restores positivity in a very similar way as the exact solution does: by modifying the memory kernel. But the SL equation does it without systematically taking into account the reservoir memory.

Further, by using Keldysh techniques to acquire a somewhat general non-Markovian master equation form, we have shown that obtaining a proper memory kernel that retains the physics (and extends into new parameter regimes the validity of the Born approximation) is tantamount to solving the complete problem. We believe this is quite general when the total system is separated into a system and reservoir: whenever you want a consistent memory kernel it is necessary to modify the kernel using the real-time retarded and advanced Green's function for the system with the same set of system-reservoir self-energy diagrams. Because of this, the proper memory kernel cannot be generic, e.g., cannot be just a property of the reservoir, and it will depend on properties of the system (e.g., the system Hamiltonian has to be first diagonalized and used with system-reservoir self-energy to modify the memory kernel). The final conclusion here is that to have simulations

of discrete systems strongly connected to a reservoir, one has to abandon the idea of dividing the Hamiltonian into system and reservoir components and integrating out the reservoir. This is not to say that this partitioning is not helpful in some cases, such as deriving the effective action to be used in Monte Carlo simulations, but in these lattice simulations, it does not seem to be the fruitful approach.

Chapter 6

Infinite system tensor network simulations

We give a method to simulate systems that are composed of a one-dimensional lattice connected to branches at each lattice site. The physical scenario represented by this structure is that of a one-dimensional systems where each site is connected to a local, independent reservoir. The intention of the method is to be able to simulate systems in the presence of strong dissipation, e.g., Josephson junction arrays in the presence of a resistive shunt or quasi-particle dissipation. It has a similarity with the numerical renormalization group. However, the method is formulated for lattice systems rather than single impurities. Also, the implementation is drastically different than NRG since here we use MPS as a variational class of states over which we find the ground state rather than performing a less efficient iterative diagonalization procedure. In this chapter we give a translationally invariant formulation, although it can be directly used for finite lattices as well.

6.1 Introduction

In the previous chapter, we gave a procedure for simulating strongly non-Markovian processes that involved a series of successive transformations on the reservoir/bath Hamiltonian. We saw that this series can be terminated in cases where we can represent the bath as a finite collection of modes damped by Markovian reservoirs. However, for strong dissipation, one has to continue the series of transformations and one can only terminate the process artificially. This latter case, though, is the one that the numerical renormalization group has been developed to handle. Thus, when complete, the method presented here will be like an infinite system numerical renormalization group. In this chapter, though, we only develop the technique to handle simulations of branched lattices, as shown in Figure 6.1, and the full method, including choosing the representation of the bath, will only be briefly discussed. Below, we first discuss the algorithm, then we give some examples of infinite system simulations with and without branches.

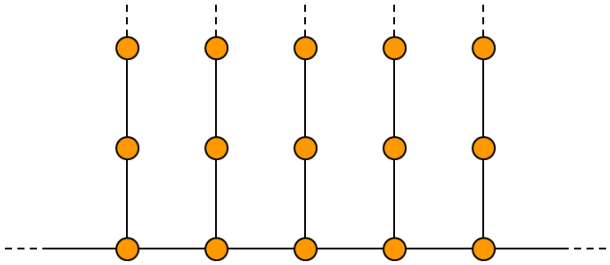


Figure 6.1: Branched lattice under consideration in this chapter.

6.2 Algorithm

In this section we show a matrix product state approach to a translationally invariant one-dimensional lattice and how to use this for translationally invariant lattices with branches. The translationally invariant algorithm is a specific application of the time-evolving block decimation (TEBD) algorithm and was given only recently by Vidal.[79]. The algorithm has been dubbed *iTEBD*.

Consider the case where one is given a translationally invariant state represented as a matrix product state, as shown in Figure 6.2. If we now want to evolve that state in time according to some Hamiltonian

$$H = \sum_l H_{l,l+1}, \tag{6.1}$$

which translationally invariant and a sum of local terms, we have to apply the unitary operator

$$U = e^{-\imath H t} = \left(e^{-\imath H \delta t}\right)^N . \tag{6.2}$$

We can Trotter decompose this evolution operator

$$U = \left(\prod_l e^{-\imath H_{l,l+1}\delta t}\right)^N + \mathcal{O}\left(\delta t\right) , \tag{6.3}$$

which gives it as a product of local gates. Each gate is identical, and further, the odd l gates act on disjoint sets of sites (and likewise for the even). The idea behind iTEBD then, is that we can parallelize the application of the odd and even gates.

To perform this parallelize application of the odd and even gates, one has to break the translational symmetry of the state in Figure 6.2(a). The new state has the structure as in Figure 6.2(b). With this new structure, all the odd (even) gates can be applied simultaneously to the whole, infinite lattice. By continuing to perform the operations and alternating between odd and even, the state is evolved in time. Similarly, this idea can be applied to imaginary time evolution. This allows one to compute ground states of the infinite lattice and also to get quantities like correlation functions.[†]

[†]However, in this case, unlike the unitary gates above, imaginary time evolution destroys the orthogonality of the

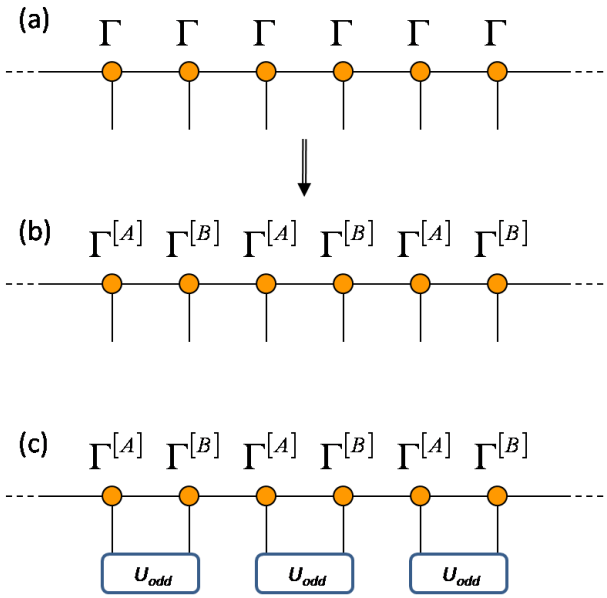


Figure 6.2: (a) Translationally invariant matrix product state that is (b) broken into pairs of sites. Each pair of sites is identical, such that, the application of the evolution operator can be done all at once for the infinite lattice, as shown in (c), where one alternates between applying the odd and even gates.

The computational cost of this method is just $Nd^3\chi^3$.

The infinite lattice system with branches is based on the iTEBD. Schematically, we have the MPS shown in Figure 6.3. On the left in this figure, a lattice site connected to a branch is shown. This lattice site is part of a translationally invariant lattice where all sites are identically configured. We call each lattice site plus its branch a *lattice group*. Following the iTEBD, we break the lattice groups into pairs of lattice groups so that we can perform operations on nearest neighbor sites.

In addition to the procedure for handling the infinite system lattice, we have to manipulate MPS with higher-dimensional tensors at the lattice site. In fact, "matrix product state" is not a great name anymore, as the state no longer has the "matrix product" form. This is why we call these states *tensor networks*. Manipulating these tensors can be done in the same fashion as the matrix product tensors[33], however, it is first required to collapse all the bonds that are not being operated on into one.

The efficiency of simulating the branched lattice structure in Figure 6.1 is $N\left(d^3\chi^3 + L_b d_b^3\right)\chi_b^3$, where N is the number of time steps, χ is the matrix product dimension on the lattice, d is the local Hilbert space dimension on the lattice, χ_b is the matrix product dimension on the branch, and d_b is the local Hilbert space dimension on the branch. Obviously, there is a $\sim \chi^6$ scaling here which is much worse than the usual χ^3. However, since we are performing translationally invariant

Schmidt vectors in the SD. Thus, the parallelized application of imaginary gates is not justified on a formal level. In addition to numerical evidence that the procedure works well, one can also argue that with very small time steps in the imaginary time evolution, the orthogonality is not destroyed too much and the simulation should work.

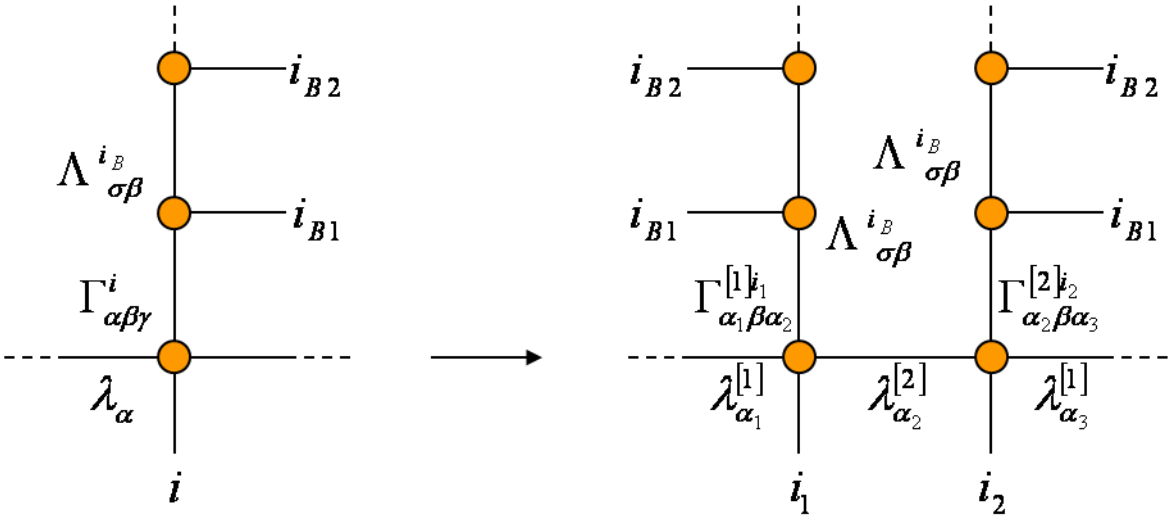


Figure 6.3: On the left, we show single lattice site connected to a one-dimensional reservoir. This lattice site is part of a translationally invariant lattice. On the right, to perform simulations, we break the translational symmetry, as in iTEBD.

simulations, we do not have a factor of L corresponding to the lattice size, which is definitely a significant improvement.

6.2.1 Examples

In the following examples, we consider the Ising model with a transverse field:

$$H = \sum_l \left(\sigma_x^l \sigma_x^{l+1} + g\sigma_z^l\right) . \tag{6.4}$$

Specifically, we look the correlations of the system for various values of the coupling g, with and without branches connected to the lattice.

6.2.1.1 One-dimensional Ising

In Figure 6.4, we plot the correlation function

$$\langle \sigma_z^0 \sigma_z^l \rangle - \langle \sigma_z^0 \rangle^2 , \tag{6.5}$$

for several values of g as computed with the iTEBD method. We find that the correlations decay exponentially, except at the critical point, $g = 1$, where they decay polynomially. These simulations take negligible time. In Figure 6.5, we plot the relative error of the correlation function versus the position. Despite having a MPS dimension of 16, the correlation function has an a relative error of only a few percent.

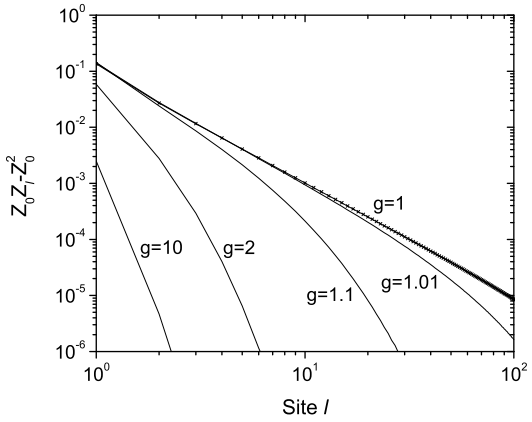


Figure 6.4: ZZ correlations of the Ising model on a translationally invariant lattice. We show the correlations for several values of g. At the critical point, $g = 1$, the correlations decay polynomially $(1/l^2)$. Overlaid on the $g = 1$ results are the exact correlations shown as crosses. For $g \neq 1$, the correlations decay exponentially.

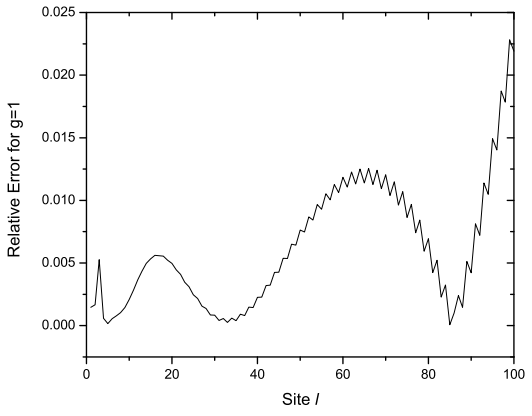


Figure 6.5: The relative error of the ZZ correlations for $g = 1$. The error is only a few percent, which is quite amazing because $\chi = 16$.

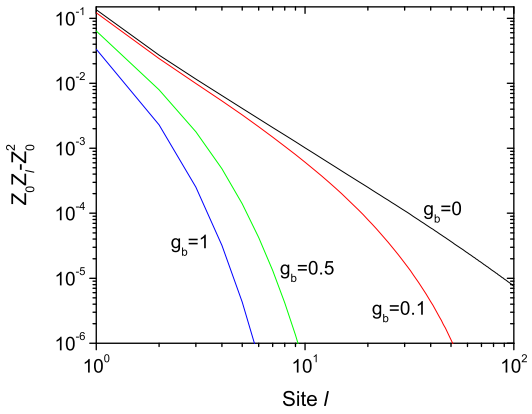


Figure 6.6: ZZ correlations at $g = 1$ on the main lattice. For no connection to the side chain, $g_b = 0$, the correlations decay polynomially. However, adding connections to the 16 site side chain makes the correlations decay exponentially.

6.2.1.2 Ising with branches

Now we consider the same Ising Hamiltonian 6.4, but now we connect up branch to each lattice site. The branches have the same Hamiltonian as the main lattice. However, each branch is connected to a site l with interaction

$$H_I^l = g_b \sigma_x^l \sigma_x^{B_1^l} , \tag{6.6}$$

where B_1^l labels the first site in the branch. Using the tensor network simulation on the branched lattice, we simulate the infinite system with $L_B = 16$ length branches and use $\chi = 16$ and $\chi_b = 4$. The correlation functions on the lattice are shown in Figure 6.6. We find that turning on the interaction to the branches destroys the polynomial decay of the correlations.

6.2.2 Logarithmic covering of reservoir modes

Above we gave example calculations with a Ising Hamiltonian on all of the structure. However, the case of particular interest is when the one-dimensional system is strongly connected to reservoirs that represent an environment. The environment could be the dissipative environment represented as a set of bosonic modes. In any case, one can not perfectly represent the infinite number of modes of the environment. Instead, one is restricted to a finite number of modes. The original formulation of the numerical renormalization group[27] gives an ansatz to represent the infinite number of modes. The idea behind the ansatz is that only the low-energy degrees of freedom are important to the long-time behavior of the system. Thus, one introduces a discretization of the reservoir modes based on a logarithmic covering of the energy scales. The discretization is shown schematically in Figure 6.7 for a band of reservoir modes. After choosing a finite collection of modes with this

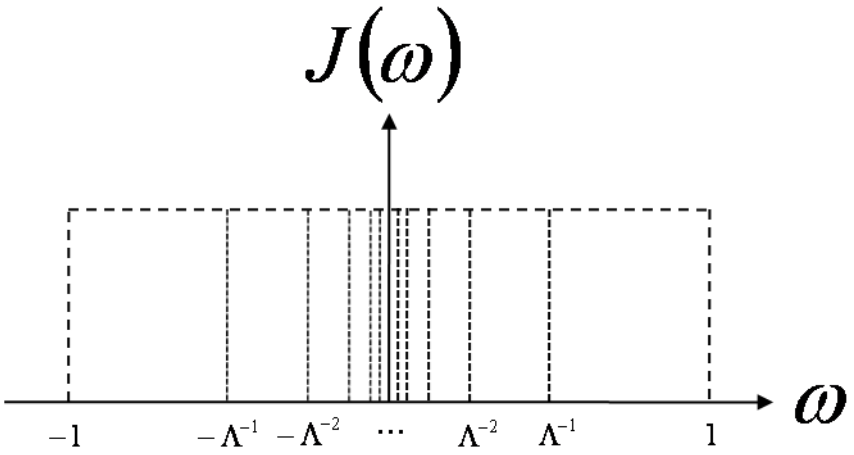


Figure 6.7: Logarithmic covering of a band of reservoir modes. The modes further from zero are grouped into larger blocks. Each block of modes is replaced by a single mode at the block's mean frequency.

covering, the transformation procedure of Chapter 5 is applied to take this collection of modes into a one-dimensional lattice, which can then be attached onto the main lattice.

Actually, this idea of a logarithmic covering was used in another form in Chapter 3. There, we used a logarithmic covering of the memory kernel in a non-Markovian master equation. However, the idea is exactly analogous. In the covering of the polynomially decaying kernel, the error was controlled by the fact that the decay gets smoother as one goes back further in time. Here, the error is controlled because as one goes to higher energy scales, the relative importance of the modes also gets "smoother." For instance, in terms of perturbation theory, a larger denominator appears at the higher energies. Thus, the high energy modes, as seen by an impurity, look increasingly the same for higher and higher energies, in addition to having a much lower weighting than the low energy modes. This is the same idea as the increasingly smooth memory kernel. In both cases, what is important is that any errors that are incurred are controllable by, in this case, taking $\Lambda \to 1$.

6.3 Conclusions

In this chapter we demonstrated a method to simulate branched lattices, where the branches, for instance, could represent an environment composed of bosons. The method was formulated for a translationally invariant system. This tremendously helps in the computational cost. However, it can be directly carried over to a finite lattice. This is especially relevant if the ideas of Chapter 7 are to be used. For instance, if one wants to look at the real-time behavior of the current in the presence of superconducting leads, then one could look at a finite lattice version of the above. To use this method to study dissipative systems, one has to incorporate the ideas and analysis of the numerical renormalization group. This is work in progress.

Chapter 7

Dynamical approach to transport

We propose a method of investigating transport properties of one-dimensional systems via the use of matrix product state simulations. The method is based on the introduction of Markovian leads which drive particles through the one-dimensional system. Using the method of Chapter 2, one can either examine the real-time dynamics of the system or find the steady states of the Lindbladian. The use of Markovian leads may be justifiable based on the presence of other dissipative/decohering processes that erase the electrons memory of the injection process. Correspondingly, errors resulting from the replacement of the true non-Markovian reservoirs by the Markovian reservoirs will not matter. Although results obtained by this method will not be directly applicable to experiments, we believe the method can be used to assess approximations made in other transport approaches. Also, we discuss specific examples of issues arising in transport (and transport approaches).

7.1 Introduction

Understanding the transport properties of physical systems is centerfold to the development of new technologies. In particular, there is a lot of interest in predicting transport properties of nanoscale structures. However, to do so is a complex problem and usually involves many uncontrolled approximations. A typical approach is to use density-functional theory (DFT) calculations to calculate the electronic structure of a nanoscale configuration of atoms connected to leads, then to use this electronic structure in a calculation of transport properties using the Landauer formalism. In addition to the use of approximate exchange-correlation functionals, it is not clear what effect the replacement of the dynamical many-body problem with a DFT-based scattering approach has.

We propose a method to calculate the transport properties of one-dimensional systems that uses matrix product state techniques. Specifically, we introduce Markovian reservoirs that drive particles through the system. The introduction is rather *ad hoc*, however, we believe that it is justifiable based on the grounds that in real physical system there will be other relaxation processes. These processes will erase the electrons memory as it flows through the leads toward the junction, thus erasing any

non-Markovian effects at the boundaries. Although due to the restriction of one-dimensionality and a set of its own approximations, the results obtained with this method will probably not be applicable directly to experiment. But one may be able to use the method to assess approximations of other methods.

We discuss examples of what other transport approaches may miss. Specifically, using time-dependent current-density functional theory, we derive analytically the dynamical exchange-correlation correction to the DC conductance of nanoscale junctions. The correction pertains to the conductance calculated in the zero-frequency limit of time-dependent density-functional theory within the adiabatic local-density approximation. In particular, we show that in linear response the correction depends non-linearly on the gradient of the electron density; thus, it is more pronounced for molecular junctions than for quantum point contacts. We provide specific numerical examples to illustrate these findings.

We also investigate steady state solutions to transport in quantum systems. Specifically, we investigate states which meet a criterion of instantaneous steady states. These are states which at some time minimize the change in density throughout all space and have a given current density flowing from one part of the system to another. In a classical system, this criterion gives a unique state for a given current density and particle number (energy is not a free variable). However, in a quantum mechanical system, there are many states with a given energy and particle number which minimize the change in density throughout space. Taking an example of spinless fermions on a one-dimensional lattice, we explicitly show the phase space of instantaneous steady states. There exists states with the same energy and particle number but with different values of the steady-state current.

7.2 Transport with MPS

If we had a very large computer, we may consider modeling a transport experiment exactly as it occurs, e.g., taking into account each process that happens in the materials involved in the experiment. Obviously, this is well beyond reach, but we could try to push toward this limit as much as possible. For instance, we could consider the model in Figure 7.1 for a transport experiment through a nanoscale junction. The model has several components. First, there are particle reservoirs that add and subtract particles into the system at the boundaries, which are far away from the junction region that contains the actual system of interest. Second, the complete system includes the metallic leads that are partitioned by the nanoscale junction. Within the leads, the relaxation processes of the electrons are modeled explicitly, and likewise, electron-electron interactions are also considered. Third, the nanoscale junction is allowed to be a fully interacting system.

The idea behind this setup is not specifically for use with MPS. However, taking now a more

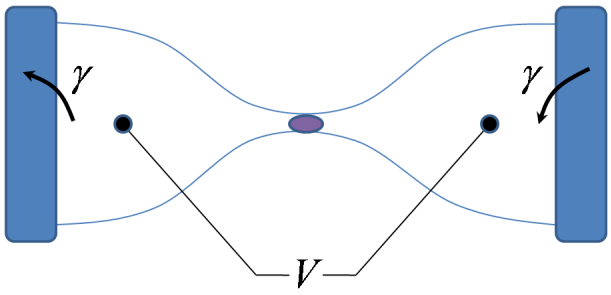


Figure 7.1: Computational setup of transport through a nanoscale junction. Two particle reservoirs, a source and sink shown as blue rectangles, inject or subtract particles from the system at a rate γ. The system includes two metallic leads, which are separated by a nanoscale restriction that contains some object of interest (the purple oval). The introduction of particles with the rate γ is phenomenological and is not necessarily related to an external bias. Thus, one uses a four-probe setup and measures the voltage drop V internally across the system.

practical viewpoint, we can put restrictions on the processes and our system. For instance, consider only one-dimensional systems with local interactions and relaxation processes. These are severe restrictions, but we can keep the main idea behind the model: to incorporate the processes, if only approximately, that occur in a physical system, while retaining a quantum treatment of the problem. We still, though, have to determine a way to model these processes.

First comes the injection/depletion process. One would normally think of this as a non-Markovian process. Taking, for instance, a reservoir of electrons with Hamiltonian

$$H_R = \sum_{k,\mu\in L,R} \epsilon_{\mu k}(t)\, c^\dagger_{\mu k} c_{\mu k}\,, \tag{7.1}$$

where the L, R stand for the left and right reservoirs and we have given the electrons a time-dependent on-site energy (which can be used to drive the electrons away from or into the reservoir). Integrating out such a reservoir would result in a non-Markovian process, except in very restricted settings. However, since a physical system has relaxation processes in the leads, the injection/depletion processes of the reservoirs need not be modeled perfectly. In fact, introducing Markovian reservoirs may be sufficient, because the relaxation processes will erase the memory of the electrons as the travel through the leads. Thus, errors due to using a Markovian rather than a non-Markovian reservoir may be washed out. We will therefore consider Markovian reservoirs, which comes in the form of a wide-band limit assumption (we make a further simplifying assumption below).

For the relaxation processes in the leads, one has to make more serious approximations. Of course, we would like to model relaxation due to interactions with phonons. However, it is not clear how much of this will be possible. Certainly, given the one-dimensional setup, one can take a bosonic mode that lives at each site and which is simulated explicitly (with some Markovian damping). This would roughly correspond to a situation of a one-dimensional chain of atoms where

the bosonic mode represents the longitudinal vibration of each atom. In addition, it would give a non-local relaxation mechanism, e.g., electrons could interact with this phonon chain, which would then propagate this interaction along the lattice. On the other hand, it might be sufficient just to consider local Markovian relaxation of the electrons (for instance, one could define a local Fröhlich-like Hamiltonian[80]). For the specific example below, this is the approach we take.

Last, but certainly not least, is the electronic interactions within the leads and the junction. Since we use MPS, it is wise to restrict the interactions to just nearest neighbor. This can be done in the usual way. That is, we can use a Hubbard type Hamiltonian. Actually, the approximations that one has to make here may not be important to answering the question of how the interactions change the dynamical picture and resulting physical predictions.

For a specific example, lets consider a Markovian Lindbladian

$$\mathcal{L} = \mathcal{L}_H + \mathcal{L}_R + \mathcal{L}_P \,. \tag{7.2}$$

The first term, $\mathcal{L}_H \circ \equiv -\imath \left[H_S, \circ\right]$, is due to the system (junction and leads) Hamiltonian, which we take to be that of noninteracting spinless fermions

$$H_S = \sum_l J_l \left(c_l^\dagger c_{l+1} + c_{l+1}^\dagger c_l \right) + \sum_l \mu_l c_l^\dagger c_l \,, \tag{7.3}$$

where we have variable nearest-neighbor hopping strength and on-site potential which we can use to create impurities. Particle injection/depletion will be taken care of by a perfect source/sink (e.g., this is stronger than the wide-band limit mentioned above):

$$\mathcal{L}_P \circ = \gamma \left(c_1 \circ c_1^\dagger + c_L^\dagger \circ c_L - \frac{1}{2} \circ c_1^\dagger c_1 - \frac{1}{2} c_1^\dagger c_1 \circ - - \frac{1}{2} \circ c_L c_L^\dagger - \frac{1}{2} c_L c_L^\dagger \circ \right) , \tag{7.4}$$

where we are injecting particles on site L and taking them away on site one, e.g., at the ends of the lattice. We will take $\mathcal{L}_R$ to be local dephasing, which loosely corresponds to a relaxation process,

$$\mathcal{L}_R \circ = \kappa \sum_l \left(n_l \circ n_l - \frac{1}{2} \circ n_l^2 - \frac{1}{2} n_l^2 \circ \right) . \tag{7.5}$$

In Figures 7.2 and 7.3 we show an example simulation for weak local dephasing. We find what is expected, the system is initially set out of equilibrium and slowly it goes into a steady state, which has a constant current. The magnitude of this current is dependent on the strength of the injection. However, we can get a current voltage relation, for instance, by examining the distribution of fermions in the system, see Figure 7.3. This looks like typical biased system, where we have a set of excess fermions on the right side, coupled by a strong drop at the impurity site, and a depletion of fermions on the left side. The weaker the local dephasing, the flatter the distributions will become

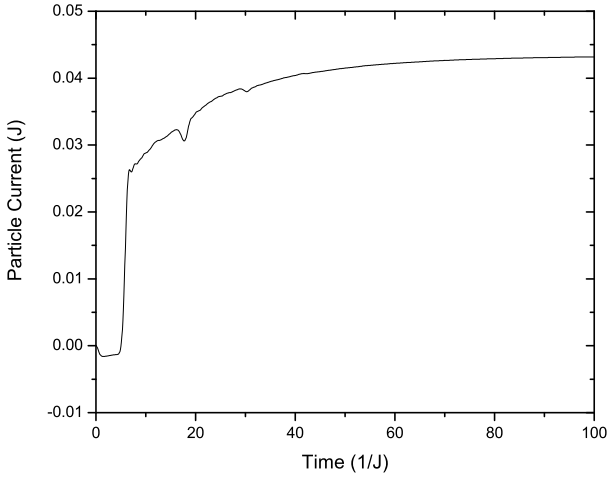


Figure 7.2: Current through the center site versus time. The system is a 41 site one-dimensional lattice of noninteracting, spinless fermions. The injection rate is $\gamma = 0.2$ and the local dephasing rate is $\kappa = 0.1$. The nearest-neighbor hopping is $J = -1$, except to an impurity in the center of the lattice where the coupling is $J_{imp} = -0.5$.

on each side.[†]

We have seen in this section an idea for a model system to study transport with MPS. The idea behind the model is to set up a situation that the takes into account the many different processes that happen in a physical system, if only in an approximate way. With this method implemented using the algorithm of Chapter 2, one should be able to study and assess approximations within other transport approaches, specifically with regard to how interactions change the dynamical picture.

7.3 Dynamical corrections to the DFT-LDA electron conductance

Recent attempts to apply static density-functional theory[81,82] (DFT) to electronic transport phenomena in nanoscale conductors have met with some success. Typical examples are atomic-scale point contacts, where the conductance, calculated with DFT within the local-density approximation (LDA), is found to be in excellent agreement with the experimental values[83]. When applied

[†]If the system has really strong local dephasing, the steady state should be a solution to the classical diffusion equation in steady state, e.g., it will satisfy Fick's law:

$$j(x) \equiv j_0 = -D(x) \frac{\partial n(x)}{\partial x}, \tag{7.6}$$

with diffusion coefficient

$$D(x) = J(x), \tag{7.7}$$

and boundary condition

$$n(0) = \frac{J(0)}{\gamma} j_0 . \tag{7.8}$$

For a system with no impurity, this will just give a density profile that is a straight line. For lattices with impurities, there will be jumps at the locations of the impurities.

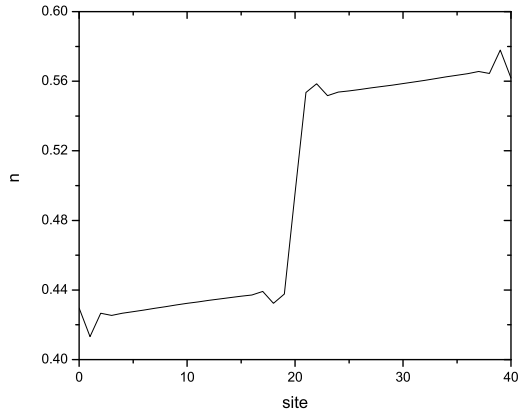


Figure 7.3: Particle density profile of the system near to its steady state. We see that there exists particle enhancement on one side of the junction and depletion on the other. The slight slope of the particle density is due to the local dephasing, which creates resistance in the "leads."

to molecular junctions, however, similar calculations have not been so successful, yielding theoretical conductances typically larger than their experimental counterparts[84]. These discrepancies have spurred research that has led to several suggested explanations. One should keep in mind, however, that these quantitative comparisons often neglect some important aspects of the problem. For instance, the experimentally reported values of single-molecule conductance seem to be influenced by the choice of the specific experimental setup[85,86,87,88,89]: different fabrication methods lead to different conductances even for the same type of molecule, with recent data appearing to be significantly closer to the theoretical predictions[88,89] than data reported in earlier measurements[90]. In addition, several current-induced effects such as forces on ions[91] and local heating[92,93] are generally neglected in theoretical calculations. These effects may actually generate substantial structural instabilities leading to atomic geometries different than those assumed theoretically. However, irrespective of these issues, one is naturally led to ask the more general question of whether static DFT, within the known approximations for the exchange-correlation (xc) functional, neglects *fundamental* physical information that pertains to a truly non-equilibrium problem. In other words, how accurate is a static DFT calculation of the conductance within the commonly used approximation for the xc functional, LDA, compared to a time-dependent many-body calculation in the zero-frequency limit?

In this section we provide an answer to this question. Specifically, we seek to analytically determine the correction to the conductance calculated within the static DFT-LDA approach and illustrate the results with specific examples. A few recent attempts were made in this direction. For instance, Delaney *et al.*[94] used a configuration-interaction based approach to calculate currents from many-body wavefunctions. While this scheme seems to yield a conductance for a specific molecule of the same order of magnitude as found in earlier experiments on the same system, it relies on strong

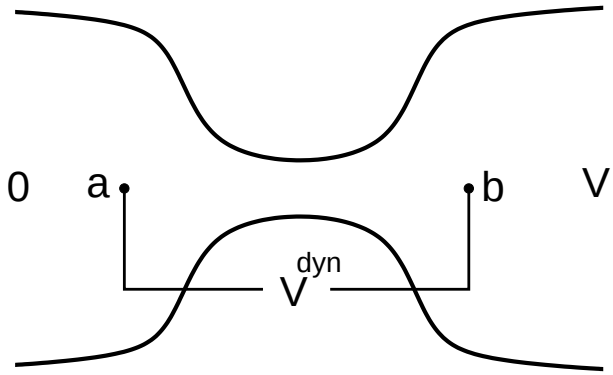


Figure 7.4: Schematic of a nanoscale junction between electrodes with applied bias V. Due to dynamical exchange-correlation (xc) effects there is an xc field which gives rise to an additional voltage drop V^{dyn} compared to the electrostatic potential difference.

assumptions about the electronic distribution of the particle reservoirs[94]. Following Gonze *et al.*[95], Evers *et al.*[96] suggested that approximating the xc potential of the true non-equilibrium problem with its static expression is the main source of discrepancy between the experimental results and theoretical values. However, these authors do not provide analytical expressions to quantify their conclusion.

Our system is the nanojunction illustrated in Figure 7.4, which contains two bulk electrodes connected by a constriction. In order to understand the dynamical current response, one must formulate the transport problem beyond the static approach using time-dependent density functional theory[97,98] (TDDFT).[†] In the low frequency limit, the so-called adiabatic local-density approximation (ALDA) has often been used to treat time-dependent phenomena in inhomogeneous electronic systems. However, it is essential to appreciate that the dynamical theory includes an additional xc field beyond ALDA[99] – *a field that does not vanish in the DC limit.*[‡] This extra field, when acting on the electrons, induces an additional resistance $R^{\rm dyn}$, which is otherwise absent in the static DFT calculations or TDDFT calculations within the ALDA.[§] Our goal is to find an analytic expression for this resistance and then estimate its value in realistic systems. We will show that the dynamical xc field opposes the purely electrostatic field: one needs a larger electric field to achieve the same current, implying that the resistance is, as expected, increased by dynamical xc effects.

We proceed to calculate the xc electric field. This quantity was determined by Vignale, Ull-

[†]For a strictly finite and isolated system the true many-body total current is given *exactly* by the one-electron total current, obtained from TDDFT provided one knows the *exact* xc functional [M. Di Ventra and T. N. Todorov, J. Phys. Cond. Matt. **16**, 8025 (2004); G. Stefanucci and C.-O. Almbladh, Europhys. Lett. 67, 14 (2004)]. Here we ask the question of what percentage of the total conductance originates from dynamical effects beyond ALDA.

[‡]The additional field is a strongly non-local functional of the density, but becomes a local functional when expressed in terms of the current density.

[§]We implicitly assume that we can identify a physical initial state such that, in the zero-frequency limit, the conductance obtained from a TDDFT calculation within the ALDA for the xc kernel equals the conductance obtained using static DFT within LDA. The assumption requires non-resonant scattering and is thus satisfied by quantum point contacts and molecular junctions in linear response when the Fermi level falls between the highest occupied and the lowest unoccupied molecular orbitals[84].

rich and Conti using time-dependent current-density functional theory (TDCDFT)[100,101]. In their notation, this field is

$$E_\alpha(\mathbf{r},\omega) = E_\alpha^{ALDA}(\mathbf{r},\omega) - \frac{1}{en}\partial_\beta \sigma_{\alpha\beta}(\mathbf{r},\omega), \tag{7.9}$$

where $E_\alpha^{ALDA}(\mathbf{r},\omega)$ is the ALDA part of the xc contribution, $\sigma_{\alpha\beta}(\mathbf{r},\omega)$ is the xc stress tensor, and e is the electron charge. Here $n = n_0(\mathbf{r})$ is the ground-state number density of an inhomogeneous electron liquid. We are interested in the dynamical effects that are related to the second term on the RHS of equation (7.9), i.e.,

$$E_\alpha^{\rm dyn}(\mathbf{r},\omega) \equiv -\frac{1}{en}\partial_\beta \sigma_{\alpha\beta}(\mathbf{r},\omega). \tag{7.10}$$

In the static limit[100], we can transform $E_\alpha^{\rm dyn}$ into an associated potential by integrating between points a and b inside the electrodes lying on the z-axis. We take the z-direction to be parallel to the current flow. The end points a and b are in regions where the charge density does not vary appreciably, i.e., $\partial_z(1/n)|_a^b \sim 0$ (see Figure 7.4). This yields

$$\begin{aligned} V^{\rm dyn} &= -\int_a^b \lim_{\omega\to 0} {\rm Re}\,\mathbf{E}^{\rm dyn}\cdot d\mathbf{l} \\ &= \int_a^b \frac{1}{en}\lim_{\omega\to 0} {\rm Re}\,\partial_\beta \sigma_{z\beta}(\mathbf{r},\omega) dz. \end{aligned} \tag{7.11}$$

Importantly, we include here only the part of the electric field that varies in phase with the current (i.e., we take the real part of the stress tensor). In general—e.g., at finite frequency—the electric field has both an in-phase (dissipative) and an out-of-phase (capacitive) component, where the latter is related to the shear modulus of the inhomogeneous electron liquid. Such capacitive components play a crucial role in the theory of the dielectric response of insulators. We ignore them here on the basis that they do not contribute to the resistance.¶

The general xc stress tensor in TDCDFT[101] is given by

$$\begin{aligned} \sigma_{\alpha\beta}(\mathbf{r},\omega) &= \tilde{\eta}(n,\omega)\left(\partial_\beta u_\alpha + \partial_\alpha u_\beta - \frac{2}{3}\nabla\cdot\mathbf{u}\,\delta_{\alpha\beta}\right) \\ &\quad + \tilde{\zeta}\,(n,\omega)\nabla\cdot\mathbf{u}\,\delta_{\alpha\beta}, \end{aligned} \tag{7.12}$$

where $\tilde{\eta}(n,\omega)$ and $\tilde{\zeta}(n,\omega)$ are the frequency-dependent viscoelastic coefficients of the electron liquid, while $\mathbf{u} = \mathbf{j}/n$ and $\mathbf{j}$ are the velocity field and the particle current density, respectively, induced by a small, time-dependent potential.

¶In the current-density functional theory, the capacitive effect is described by a static shear modulus. Within the local density approximation for a metallic system, this quantity is expected to vanish in the frequency regime relevant to DC conduction.

The viscoelastic coefficients are given by

$$\tilde{\eta}(n,\omega) = -\frac{n^2}{i\omega} f^h_{xc,T}(\omega)\,, \tag{7.13}$$

and

$$\tilde{\zeta}(n,\omega) = -\frac{n^2}{i\omega}\left\{ f^h_{xc,L}(\omega) - \frac{3}{4} f^h_{xc,T}(\omega) - \epsilon''_{xc} \right\}, \tag{7.14}$$

where $f^h_{xc,L}(\omega)$ and $f^h_{xc,T}(\omega)$ are, respectively, the longitudinal and transverse xc kernel of the homogeneous electron gas evaluated at the local electron density $n = n_0(\mathbf{r})$, while ϵ''_{xc} is simply

$$\epsilon''_{xc} = \left.\frac{d^2\epsilon_{xc}(n)}{dn^2}\right|_{n_0(\mathbf{r})}. \tag{7.15}$$

In the representative systems that we examine below, the derivatives in the transverse directions x and y account for only a small fraction of the total dynamical xc field and can hence be ignored. We thus obtain

$$E_z = -\frac{1}{en}\partial_z \sigma_{zz}. \tag{7.16}$$

We then see that

$$\sigma_{zz} = \frac{4\eta}{3}\partial_z u_z\,, \tag{7.17}$$

where the viscosity $\eta = \lim_{\omega\to 0} \mathrm{Re}\tilde{\eta}(\omega)$ is a function of n, and therefore of z. The real part of $\tilde{\zeta}(\omega)$ vanishes in the limit of $\omega \to 0$. Under the same assumptions of negligible transverse variation in current density,[‖] we can write

$$u_z = \frac{I}{enA_c}\,, \tag{7.18}$$

where $I > 0$ is the total current (independent of z), and A_c is the cross-sectional area.[**] Substituting this into the equation for the voltage drop and integrating by parts we arrive at

$$V^{\mathrm{dyn}} = -\frac{4I}{3e^2A_c}\int_a^b \eta\frac{(\partial_z n)^2}{n^4}dz\,. \tag{7.19}$$

Because η is positive – a fact that follows immediately from the positivity of energy dissipation in the liquid – we see that the right hand side of this equation is negative-definite: the electrostatic voltage is *always opposed* by the dynamical xc effect. We identify the quantity on the right hand side of equation (7.19) with $-R^{\mathrm{dyn}}I$, where R^{dyn} is the additional resistance due to dynamical effects. According to TDCDFT, the current that flows in the structure in response to an electric potential

[‖] The current density varies much slower than the particle density in the contact region. [See, e.g., N. D. Lang, Phys. Rev. B **36**, R8173 (1987).] If we relaxed this approximation, we would obtain a larger value for the dynamical contribution to the resistance.

[**] Note that we have assumed the right electrode is positively biased (see Figure 7.4) so that electrons flow from left to right. The current is thus defined as flowing in the opposite direction.

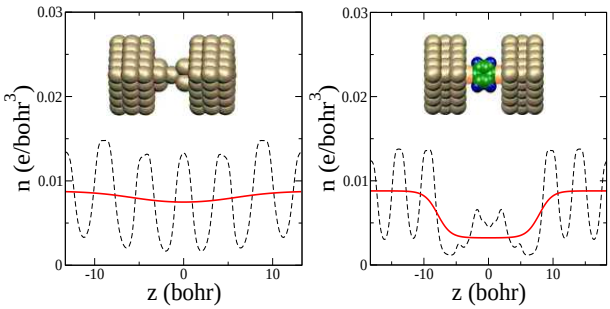


Figure 7.5: Planar (dashed line) and macroscopic (solid line) averages of the charge density for a gold point contact (left) and a molecular junction (right).

difference V is given by

$$I = G_s(V + V^{\rm dyn}) = G_s(V - R^{\rm dyn} I)\,, \tag{7.20}$$

where G_s is the conductance calculated in the absence of dynamical xc effects (e.g., by means of the techniques of Ref. 102). Solving equation (7.20) leads immediately to the following expression for the total resistance $R = V/I$:

$$R = \frac{1}{G_s} + R^{\rm dyn}\ , \tag{7.21}$$

where

$$R^{\rm dyn} = \frac{4}{3e^2 A_c}\int_a^b \eta \frac{(\partial_z n)^2}{n^4} dz. \tag{7.22}$$

The dynamical xc term thus *increases* the resistance.

This is the central result of this section. It shows that the dynamical effects (beyond ALDA) on the resistance depend nonlinearly on the variation of the charge density when the latter changes continuously from one electrode to the other across the junction. In a nanojunction, this correction is non-zero only at the junction-electrode interface where the electron density varies most. Knowing the charge density one can then estimate this resistance.

Let us thus consider two specific examples that have attracted much attention, namely the gold point contact and the molecular junction formed by a benzene-dithiolate (BDT) molecule between two bulk electrodes (see insets in Figure 7.5) and estimate the error made by the (A)LDA calculation in determining the resistance. In order to make a connection between the microscopic features of these junctions and the density in equation (7.22), we model the charge density $n = n_0(\mathbf{r})$ as a product of smoothed step functions in every direction, i.e.,

$$\begin{aligned} n_0(\mathbf{r}) \quad &= \quad n_e \Xi(z, d, \gamma_e) \\ &\quad + n_c \Omega(x, h, \gamma_c)\Omega(y, w, \gamma_c)\Omega(z, d, \gamma_e). \end{aligned} \tag{7.23}$$

The smoothed step function is given by

$$\Theta(z,l,\gamma) = \frac{1}{e^{(z+l/2)/\gamma}+1}, \tag{7.24}$$

where l is the full-width at half-maximum and γ is the decay length. Here, $\Omega(z,l,\gamma) = \Theta(z,-l,\gamma) - \Theta(z,l,\gamma)$ and $\Xi(z,l,\gamma) = 1 - \Omega(z,l,\gamma)$; $\Omega(z,l,\gamma)$ represents the density distribution of the junction n_c, which smoothly connects to the constant bulk density n_e of the two electrodes separated by a distance d. Finally, h and w represent the lateral dimensions of the junction.

The electron densities are obtained from self-consistent static DFT calculations with the xc functional treated within the LDA.[††] The (111) gold surface orientation is chosen for both the point contact and the molecular junction (see schematics in Figure 7.5).

In Figure 7.5 we plot the planar and macroscopic averages of the self-consistent electron densities for both systems as a function of distance from the center of the junction along the z-direction. The macroscopic average is then fitted to the simple charge model in equation (7.23). The fitted density is then substituted in equation (7.22) to find the correction to the resistance. The estimated value of $R^{\rm dyn}$[‡‡] for the point contact is $\sim$0.2 KΩ (the static DFT resistance is about 12 KΩ[103]), while for the BDT molecule is $\sim$40KΩ (to be compared with the linear response static DFT resistance of about 360 KΩ[102]). As expected, $R^{\rm dyn}$ for BDT is larger than that for the point contact due to the larger variation of the average density between the bulk electrodes and the molecule. In Figure 7.6 we plot the resistance in equation (7.22) as a function of the ratio n_e/n_c and the decay constant γ, where we fix n_e to the value of bulk gold ($r_s \approx 3$). The resistances of the two specific examples are indicated by dots in the figure. It is clear that the dynamical contributions to the resistance can become substantial when the gradient of the density at the electrode-junction interface becomes large. These corrections are thus more pronounced in organic/metal interfaces than in nanojunctions formed by purely metallic systems.

In summary, we have shown that dynamical effects in the xc potential contribute an additional resistance on top of the static DFT-LDA one. The magnitude of the additional resistance, within linear response and the zero-frequency limit, depends on the gradient of the charge density across the junction. This additional resistance is thus larger in molecular junctions than in quantum point contacts.

[††]The calculations have been performed with the `Abinit` package (X. Gonze *et al*, http://www.abinit.org); the present calculations are performed using norm-conserving pseudopotentials. (N. Troullier, J. L. Martins, Phys. Rev. B **43**, 1993 (1991).)

[‡‡]We have used the viscosity $\eta = \frac{\hbar}{120}\left[\frac{k_F}{\pi a_0}\right]^{3/2}$ where k_F and a_0 are the Fermi wavevector of the bulk electrodes and Bohr radius, respectively.

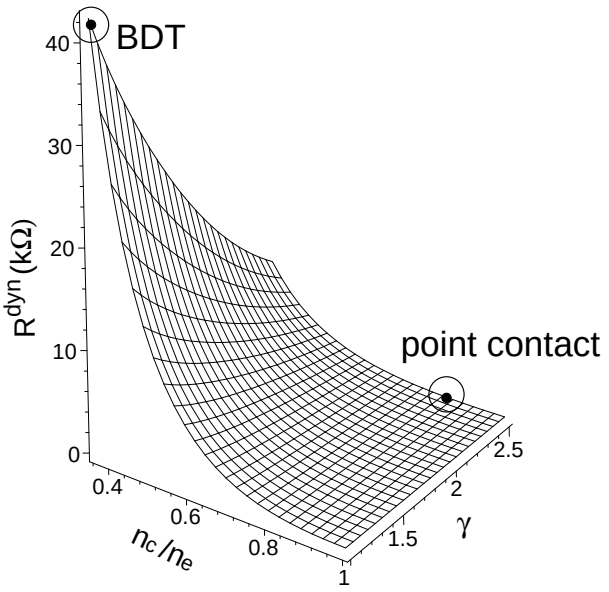


Figure 7.6: The resistance due to dynamical effects as calculated from equation 7.22 with the charge density determined from DFT-LDA calculations as a function of the main parameters discussed in the text. The resistance of a gold point contact and a BDT molecular junction are indicated by dots.

7.4 Steady-state solutions in quantum transport

Understanding the physics of systems set strongly out-of-equilibrium is at the core of applied and fundamental research. This often takes the form of understanding electron or phonon currents being driven through some smaller system, a junction or central region, by a larger system, e.g., a reservoir. For instance, in nanoscale electronics, one wants to examine electron transport across metallic point contacts or molecular junctions. In addition to electron currents, phonon currents in these systems are important to the understanding of local-heating, which can be detrimental to device operation. Common approaches to investigating these issues involve the use of the non-equilibrium Green's function formalism or the Lippmann-Schwinger equation. Within these approaches, one always has to make assumptions concerning the description of the central region to obtain a tractable problem. For example, it is often assumed that a static density-functional theory description of the system Hamiltonian is adequate to understand coherent electron transport. In some cases this can yield accurate results, but in others not. [104]

Here we are interested in examining the assumptions, not about the central region, but about the combined reservoir-junction system. Typically it is assumed the system is connected to a non-interacting particle reservoir infinitely far back in the past and rapidly reaches a steady state where dependence on the initial conditions has been wiped out. It is not necessarily true though that the dynamical situation will lead to the steady state calculated as above. That is, the steady state can be a function of initial conditions.

Recently, a new conceptual framework for examining transport has been put forward which can help understand some of the issues above.[91] This framework is based on setting finite systems strongly out-of-equilibrium by choosing appropriate initial conditions. In the context of electronic transport, the basic idea is to have two very large metallic electrodes connected by a nanoscale junction, which contains a molecule or some other arrangement of atoms. The electrodes can be charged, then at $t = 0$ a junction can connect the two electrodes allowing them to discharge. Or at time $t = 0$, an applied bias or excess charge can be added to the electrodes. This will cause a flow of charge from one electrode to the other. Depending on particular aspects of the system like the size of the metal electrodes, the system can reach a quasi-steady state.[105]

In this section, we examine a variational procedure proposed in Ref. 91. We solve for the instantaneous steady states in a one-dimensional spinless fermion system. We have found that these instantaneous steady states have an associated phase space. That is, for any given value of the current across the junction, there are many instantaneous steady states with the same energy and particle number. Further, there is overlap in the phase space for different values of the steady state current. This gives an indication that the steady state of the full system will be dependent on initial conditions.

We organize this section as follows. In subsection 7.4.1, we outline the variational procedure of Ref. 91 for finding microscopic steady states of a physical system and introduce the concept of an instantaneous steady state. In section 7.4.2, we look at particles on a lattice. In subsection 7.4.3, we first discuss how the number of free parameters scales with the lattice size. We then perform the variational procedure for non-interacting particles to find the instantaneous steady states, showing that there are many microscopic states that give the same energy, particle number, and steady-state current. In subsection 7.4.4, we show an explicit example of a single-particle on a lattice. Finally, we conclude in subsection 7.4.5.

7.4.1 Variational method

Here we wish to outline the approach we take to understand the nature of steady-state solutions in quantum transport. We want to go beyond the typical approach of finding a stationary state of a central region by incorporating the physics of the whole system. In this context, finding the microscopic steady states has been cast in the form of minimizing the functional[91]

$$A\left[\rho\right] = \int_{t_1}^{t_2} dt \int_{allspace} d\mathbf{r} \left(\frac{\partial \rho\left(\mathbf{r}, t\right)}{\partial t} \right)^2 , \tag{7.25}$$

where $\rho\left(\mathbf{r}, t\right)$ is the density of the particle(s), which can be either quantum or classical. Minimizing the functional 7.25 is equivalent to finding microscopic states with very slowly changing density throughout the whole system for a finite time interval. To get a feel for what the functional means,

we can perform an unconstrained minimization of $A[\rho]$ with respect to $\rho(\mathbf{r},t)$. Ignoring that the system is finite and then using the continuity equation gives[91]

$$\frac{\mathrm{d}I(t)}{\mathrm{d}t} = 0\ ,\ \forall t \in (t_1,t_2) \tag{7.26}$$

for the current $I(t)$ from one part of the system into the other. That is, states which minimize 7.25 give a steady state for a finite time interval. We call these states *quasi-steady states*.

Instead of the harder problem of finding the global quasi-steady states for a time interval (t_1,t_2), we find the *instantaneous steady states*[91] by minimizing the functional

$$\begin{aligned} B &= B[|\psi(t)\rangle] \\ &= \int_{allspace} d\mathbf{r}\left(\frac{\partial\rho(\mathbf{r},t)}{\partial t}\right)^2 \\ &= \int_{allspace} d\mathbf{r}\left(\nabla\cdot\mathbf{j}(\mathbf{r},t)\right)^2\ , \end{aligned} \tag{7.27}$$

where in the third line we used the continuity equation

$$\frac{\partial\rho(\mathbf{r},t)}{\partial t} + \nabla\cdot\mathbf{j}(\mathbf{r},t) = 0\,. \tag{7.28}$$

The functional 7.27 is the functional 7.25 but for a specific instance of time.

First lets consider a quantum mechanical system in general terms. The state of the system can be expanded in terms of the energy eigenbasis

$$|\psi(t)\rangle = \sum_i c_i e^{-iE_i t}|\psi_i\rangle \tag{7.29}$$

where, since we consider only finite closed systems, the states $\psi_i(\mathbf{r})$ are all bound states with energies E_i respectively. Thus, we get for equation 7.27

$$\begin{aligned} B[\{c_i\}] = &\sum_{i,i',i'',i'''} c_i^* c_{i'} c_{i''}^* c_{i'''} e^{-it_0(E_{i'}+E_{i'''}-E_i-E_{i''})} \\ &\times \int_{allspace} d\mathbf{r}\left(\nabla\cdot\mathbf{j}_{ii'}(\mathbf{r})\right)\left(\nabla\cdot\mathbf{j}_{i''i'''}(\mathbf{r})\right)\ , \end{aligned} \tag{7.30}$$

where $\mathbf{j}_{ii'}(\mathbf{r}) = \langle\psi_i|j(\mathbf{r})|\psi_{i'}\rangle$. The following constraints imposing the total energy E_0, normalization,

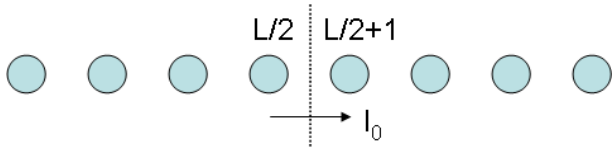


Figure 7.7: Schematic of a one-dimensional fermionic lattice. We consider a lattice with a nearest neighbor hopping Hamiltonian with equal couplings between all nearest neighbor lattice sites. The current between the two halves of the system is enforced to be I_0.

and the central region steady state current I_0 have to be satisfied:

$$I_0 = \sum_{i,i'} c_i^* c_{i'} e^{-it_0(E_{i'}-E_i)} \int_S \mathbf{j}_{ii'}(\mathbf{r}) \cdot d\mathbf{S}\,, \tag{7.31}$$

$$E_0 = \sum_i c_i^* c_i E_i\,, \tag{7.32}$$

$$1 = \sum_i c_i^* c_i\,. \tag{7.33}$$

Before applying this formalism, we first discuss states for a particle on a lattice.

7.4.2 Particles on a lattice

In order to give a tangible example and make a connection between simulations and many-body theory, we examine spinless fermions on a one-dimensional lattice. We consider the system in Figure 7.7, where we have a linear lattice of spinless fermions. We use fermions because we want to make an immediate connection with electronic transport in nanoscale systems, but the conceptual results of this section are quite generic and will apply in all dimensions for an arbitrary Hamiltonian and for bosons and fermions alike.

For simplicity, we choose a nearest-neighbor hopping Hamiltonian

$$H = -J \sum_{l=1}^{L-1} \left(c_l^\dagger c_{l+1} + c_{l+1}^\dagger c_l \right)\,, \tag{7.34}$$

where we have taken an L site lattice and we will set the hopping parameter, J, to be 1. The Hamiltonian is only important to find the actual form of the instantaneous steady states, but not in the number of free parameters.

7.4.2.1 Single fermion on the lattice

The simplest case is to place one fermion on the lattice. We discuss this case first and then go onto the N particle states. A general one fermion state on a one-dimensional lattice is

$$|\psi_1\rangle = \sum_{l=1}^{L} a_l e^{-\imath\theta_l} |1\rangle_l \bigotimes_{i\neq l} |0\rangle_i \,, \tag{7.35}$$

where each coefficient was decomposed into an amplitude and a phase ($a_l, \theta_l \in \mathrm{R}$ and $a_l > 0$). The amplitude and phase are functions of time, but since we are finding the instantaneous steady state, we can ignore the time dependence. Normalization gives the condition

$$\sum_{l=1}^{L} a_l^2 = \sum_{l=1}^{L} n_l \equiv N = 1 \,, \tag{7.36}$$

and there is a meaningless overall phase. Thus, we have $2L$ free parameters a_l, θ_l minus one overall phase, minus 1 condition for normalization, or $2\,(L-1)$ free parameters to describe a one-particle state.

Below we will use the correlation matrix to perform the variational procedure. The one-particle state 7.35 has correlation functions

$$\langle c_l^\dagger c_l \rangle = a_l^2 \equiv n_l \,, \tag{7.37}$$

$$\langle c_l^\dagger c_k \rangle = a_l a_k e^{-\imath(\theta_k - \theta_{il})} \,. \tag{7.38}$$

Since the a_l are strictly positive by construction,

$$\langle c_i^\dagger c_j \rangle = n_i^{1/2} n_j^{1/2} e^{-\imath(\theta_j - \theta_i)} \,. \tag{7.39}$$

In addition, we define variables for the real and imaginary parts of the correlation matrix

$$\epsilon_{ij} = 2\Re\langle c_i^\dagger c_j \rangle = 2a_i a_j \cos\left(\theta_i - \theta_j\right) \,, \tag{7.40}$$

and

$$I_{ij} = 2\Im\langle c_i^\dagger c_j \rangle = 2a_i a_j \sin\left(\theta_i - \theta_j\right) \,. \tag{7.41}$$

An important relationship between elements of the correlation matrix for single particle states is thus

$$\epsilon_{ij}^2 + I_{ij}^2 = 4 n_i n_j \,. \tag{7.42}$$

The energy is expressed solely in terms of $\{\epsilon_{ij}\}$

$$E = \sum_{i=1}^{L-1} \epsilon_{ii+1} \,, \tag{7.43}$$

and the functional B can be expressed solely in terms of I_{ii+1} as we will see below.

7.4.2.2 Many noninteracting particles

Many non-interacting particles on a lattice can be succinctly described by the correlation matrix

$$M = \begin{bmatrix} \langle c_1^\dagger c_1 \rangle & \langle c_1^\dagger c_2 \rangle & \cdots & \langle c_1^\dagger c_{L-1} \rangle & \langle c_1^\dagger c_L \rangle \\ \langle c_2^\dagger c_1 \rangle & \langle c_2^\dagger c_2 \rangle & \cdots & \langle c_2^\dagger c_{L-1} \rangle & \langle c_2^\dagger c_L \rangle \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \langle c_{L-1}^\dagger c_1 \rangle & \langle c_{L-1}^\dagger c_2 \rangle & \cdots & \langle c_{L-1}^\dagger c_{L-1} \rangle & \langle c_{L-1}^\dagger c_L \rangle \\ \langle c_L^\dagger c_1 \rangle & \langle c_L^\dagger c_2 \rangle & \cdots & \langle c_L^\dagger c_{L-1} \rangle & \langle c_L^\dagger c_L \rangle \end{bmatrix} . \tag{7.44}$$

However, as is discussed in section 7.4.3.1 and 7.4.6, this is not the must compact description. We will use it anyway in finding the current distribution which minimizes 7.27, because this fact does not matter there.

We can treat the real and imaginary parts as independent variables. Thus, lets set

$$M = \frac{\epsilon + iI}{2} \,. \tag{7.45}$$

In section 7.4.6, we give the conditions the correlation matrix must fulfill to be a proper (pure) quantum state. Like one-particle states, we can derive a relationship between the real and imaginary elements of the correlation matrix. However, it is an inequality instead of an equality:

$$\epsilon_{ij}^2 + I_{ij} \leq 4 n_i n_j \,, \tag{7.46}$$

To minimize B, we need an expression for the time derivative of the density in terms of the correlation matrix. The change in density with time is

$$\frac{\partial n_l(t)}{\partial t} = i \langle \psi(t) | \left[H, c_l^\dagger c_l \right] | \psi(t) \rangle \tag{7.47}$$

$$= I_{l-1l} - I_{ll+1} \,, \tag{7.48}$$

where we can express the change in density in terms of the current into the site I_{l-1l} and the current out of the site I_{ll+1}.

7.4.2.3 Interacting particle states

Interacting particle states explore the whole of Fock space, thus they live on an exponentially growing space and there is no compact form as there is with noninteracting particles. This exponential growth of the state space shows up in the number of free parameters below. We will not work explicitly with interacting states.

7.4.3 Variational Procedure

In this section we are going to perform the variational procedure for many noninteracting particles on a lattice by using the correlation matrix as the (complete) description of the state of the system. We want to minimize the functional 7.27, which on a lattice is given by the expression

$$B\left[\{n_l\}\right] = \sum_{l=1}^{L} \left(\frac{\partial n_l\left(t\right)}{\partial t} \right)^2 , \tag{7.49}$$

where

$$n_l\left(t\right) = \langle c_l^\dagger c_l \rangle_t \, . \tag{7.50}$$

The standard approach to minimize 7.49, with constraints 7.31, 7.32, and 7.33, would be to use Lagrange multipliers. However, since we have more free parameters (see below) than will be set by the minimization, we can perform an unconstrained minimization except keeping the current constraint. Further, the time derivatives of the density at each lattice site are independent except two of them are set by overall number conservation, $\dot{N}_L + \dot{N}_R = 0$, where $N_{L(R)}$ is the total particle number on the left (right) half, and $\dot{N}_L = -I_0$ due to the current constraint. Thus, we can minimize each term separately giving $L-2$ constraints. These $L-2$ constraints plus the three constraints 7.31 gives $L+1$ total constraints. This is independent of the model (Hamiltonian). Since the number of parameters in quantum models of interest will always grow linearly with lattice size or faster, we are guaranteed to have free parameters. We can now discuss the free parameter growth for the specific case of a one-dimensional spinless fermion lattice.

7.4.3.1 Free parameters

Single fermion on a lattice. The simplest case is a single fermion on a lattice of L sites. A state of this system has $2L$ parameters, as shown in equation 7.35, minus one parameter which is a meaningless phase. Thus, the number of free parameters, F_{NI}, for an instantaneous steady state on a non-interacting, one-particle, L-site lattice is

$$F_{NI}\left(1, L\right) = L - 2 \, . \tag{7.51}$$

That is, the number of free parameters grows linearly with the lattice size.

Many non-interacting fermions on a lattice. Now consider N non-interacting fermions on a lattice of L sites. To count the number of parameters, we can just fill up the lattice by constructing orthogonal single particle states. For each fermion i, one can choose a single particle wavefunction $|\psi_i\rangle$, given by equation 7.35, which has $2L$ parameters. This gives $2NL$ parameters total. But each state must be orthonormal, $\langle\psi_i|\psi_j\rangle \equiv \delta_{ij}$, thus giving N^2 conditions (e.g., N conditions for the diagonal elements and $2\frac{N(N-1)}{2}$ for the off-diagonal elements). Once you have chosen the N states, in that basis the correlation matrix will look like

$$M = \begin{bmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & & 0 & & \\ & & & & \ddots & \\ & & & & & 0 \end{bmatrix}. \tag{7.52}$$

This matrix has a $U(N)$ symmetry, e.g., we can rotate it with $\mathrm{V} \oplus \mathrm{I}$, where V is an $N \times N$ unitary matrix and I is the $(L-N)\times(L-N)$ identity matrix. Thus, we have $N^2 = 2N^2 - N^2$ parameters of the unitary matrix V which do not change the state. Since normalization (and particle number) is already fixed by this procedure, we have only L constraints from the minimization, current, and energy constraints. Overall then, we have free parameters

$$\begin{aligned} F_{NI}(N,L) &= 2NL - 2N^2 - (L) && (7.53)\\ &= 2N(L-N) - L\,, && (7.54) \end{aligned}$$

The maximum number of free parameters as a function of N comes at half filling

$$N = \frac{L}{2}\,. \tag{7.55}$$

In addition, if we consider larger and larger lattices, but keep the average density constant

$$N = \alpha L, \tag{7.56}$$

then

$$F_{NI}(N,L) = 2\alpha(1-\alpha)L^2 - L\,. \tag{7.57}$$

That is, the number of free parameters grows quadratically in L, opposed to linearly in L for a single particle on the lattice.

Many interacting fermions on a lattice. If we consider an interacting fermion system with fixed particle number N on a lattice of size L, then the number of basis states we have is

$$\frac{L!}{(L-N)!N!}\,. \tag{7.58}$$

Thus the number of parameters to describe a state is $2\frac{L!}{(L-N)!N!}$ minus an overall phase and normalization. Again we fix the energy, middle current, and the minimization values. Thus, the total number of parameters is

$$F_I\left(N,L\right) = 2\frac{L!}{(L-N)!N!} - L - 2\,. \tag{7.59}$$

Using Stirling's formula

$$\ln n! \approx n\ln n - n\,, \tag{7.60}$$

or

$$n! \approx \left(\frac{n}{e}\right)^n\,, \tag{7.61}$$

then

$$F_I\left(N,L\right) \approx 2\left(\frac{L}{e}\right)^L\left(\frac{L-N}{e}\right)^{N-L}\left(\frac{N}{e}\right)^{-N}\,. \tag{7.62}$$

Again, taking the average density to stay constant with lattice size, $N=\alpha L$, we get

$$\begin{aligned} F_I\left(N,L\right) &\approx 2\alpha^{-\alpha L}\left(1-\alpha\right)^{(\alpha-1)L} &(7.63)\\ &\approx 2\left(\frac{1}{\alpha}\right)^{\alpha L}\left(\frac{1}{1-\alpha}\right)^{(1-\alpha)L}\,. &(7.64)\end{aligned}$$

Since $0\leq\alpha\leq 1$, the number of free parameters grows exponentially in L.

7.4.3.2 Minimization for noninteracting fermions

We deal with noninteracting fermions so all we need are the single particle correlation functions, regardless of whether we consider 1 or N particles on the lattice. Thus, we can just minimize all quantities and satisfy all constraints by using the correlation functions. The change in density in time is given by equation 7.48 and the current from left to right halves is

$$I_{L\to R} = I_{L/2L/2+1} \equiv I_0\,. \tag{7.65}$$

In these terms, $B[n_l]$ is given by

$$\begin{aligned} B[n_l] &= \sum_{l=1}^{L} \left(\frac{\partial n_l}{\partial t} \right)^2 \\ &= I_{12}^2 + I_{L-1,L}^2 \\ &\quad + \sum_{l=2}^{L-1} \left(I_{l-1,l}^2 - 2 I_{l-1,l} I_{l,l+1} + I_{l,l+1}^2 \right) , \end{aligned} \tag{7.66}$$

which is what we want to minimize. We can minimize this function term by term by taking the derivatives $\frac{\partial B}{\partial I_{ll+1}}$ and setting them to zero to get the minimization conditions.

Starting on the left side of the lattice, the derivative with respect to I_{12} gives

$$4I_{12} - 2I_{23} = 0 , \tag{7.67}$$

or

$$I_{12} = \frac{1}{2} I_{23} . \tag{7.68}$$

The derivative with respect to I_{23} gives

$$4I_{23} - 2I_{12} - 2I_{34} = 0 , \tag{7.69}$$

or putting in 7.68,

$$4I_{23} - I_{23} - 2I_{34} = 0 , \tag{7.70}$$

giving

$$I_{23} = \frac{2}{3} I_{34} . \tag{7.71}$$

Continuing through the lattice gives, for $l < L/2$,

$$I_{ll+1} = \frac{l}{l+1} I_{ll+1} . \tag{7.72}$$

For $L/2$, we have to fix the current by constraint 7.65,

$$I_{L/2} \equiv I_0 \tag{7.73}$$

so

$$I_{ll+1} = \frac{2l}{L} I_0 , \forall l \leq L/2 . \tag{7.74}$$

A more eloquent way to find this solution is the following: First, recognize that except for the

first, middle and last site, we have the equation

$$4I_{ll+1} - 2I_{l-ll} - 2I_{l+1l+2} = 0\,, \tag{7.75}$$

which is equivalent to

$$\begin{aligned} \frac{I_{l+1l+2} - I_{ll+1}}{\Delta x} - \frac{I_{ll+1} - I_{l-1l}}{\Delta x} &= 0\,, & (7.76)\\ \frac{1}{\Delta x}\left(\frac{\Delta I_{i+1}}{\Delta x} - \frac{\Delta I_i}{\Delta x}\right) &= 0\,, & (7.77) \end{aligned}$$

where Δx is the lattice spacing. This is the finite difference of the second derivative

$$\frac{\Delta^2 I_{l+1/2}}{\Delta x^2} = 0\,. \tag{7.78}$$

The general solution is then

$$I_{ll+1} = \begin{cases} u_{Left}l + v_{Left} & l \leq L/2 \\ u_{Right}l + v_{Right} & l \geq L/2 \end{cases}\,, \tag{7.79}$$

where we divided the lattice into two halves, since in the middle there is a fixed current I_0. Then we can just fix the slopes and intersections using $I_{L/2L/2+1} = I_0$ and the minimization equations for the two end sites.

Going through the second half of the lattice in the same way will give the overall current distribution

$$I_{ll+1} = \begin{cases} \frac{2l}{L} I_0 & l \leq L/2 \\ -\frac{2l}{L} I_0 + 2I_0 & l \geq L/2 \end{cases}\,. \tag{7.80}$$

Thus, there is a unique current profile which minimizes the functional 7.27, which we show in Figure 7.8. This profile, though, only sets $L-1$ parameters. We now discuss an explicit example of states with this current profile as well as a given energy and particle number.

7.4.4 Example

We consider an example of one particle on a four site lattice. We start by explicitly writing the wavefunction 7.35 for a single particle on a four site lattice

$$\begin{aligned} |\psi\rangle &= a_1 e^{i\theta_1}|1000\rangle + a_2 e^{i\theta_2}|0100\rangle \\ &\quad + a_3 e^{i\theta_3}|0010\rangle + a_4 e^{i\theta_4}|0001\rangle\,. \end{aligned} \tag{7.81}$$

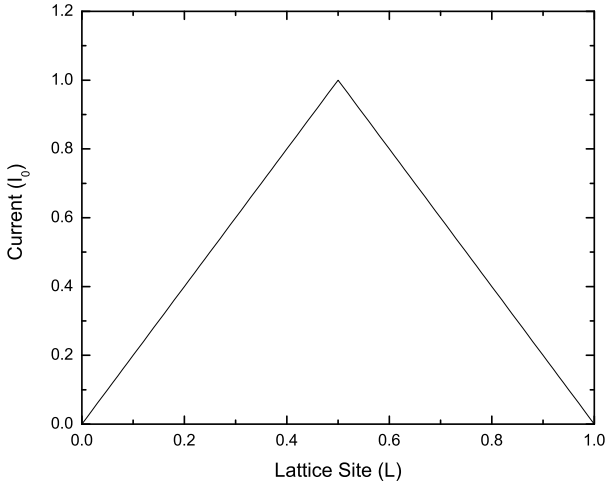


Figure 7.8: Current distribution given by equation 7.80 which minimizes the function $B[n_l]$ in equation 7.27. The current is enforced to be I_0 in the middle of the lattice. Minimizing the change in density throughout the rest of the lattice gives linear slope with zero current into/out of the lattice at the end sites. Since there are additional free parameters after we minimize B, we can construct normalized instantaneous steady states with the desired energy and particle number.

The rest of the conditions are explicitly ,with the Hamiltonian as in equation 7.34,

$$1) \qquad a_1^2 + a_2^2 + a_3^2 + a_4^2 = 1\,, \tag{7.82}$$

$$2) \qquad E = \sum_{i=1}^{3} 2a_i a_{i+1} \cos\left(\theta_i - \theta_{i+1}\right)\,, \tag{7.83}$$

$$3) \qquad I_0 = 2a_2 a_3 \sin\left(\theta_2 - \theta_3\right) = I_{23}\,, \tag{7.84}$$

$$4) \qquad \frac{I_0}{2} = 2a_1 a_2 \sin\left(\theta_1 - \theta_2\right) = I_{12}\,, \tag{7.85}$$

$$5) \qquad \frac{I_0}{2} = 2a_3 a_4 \sin(\theta_3 - \theta_4) = I_{34}\,, \tag{7.86}$$

$$6) \qquad \text{Overall meaningless phase}\,, \tag{7.87}$$

where, again, $J = 1$ in the Hamiltonian. There are then two free parameters. To get an idea about the states, we fix

$$a_1 = a_4 \equiv a\,, \tag{7.88}$$

e.g., we fix the particle number on sites one and four to be the same.[†] In addition, we can get rid of the overall phase by just considering the difference in phases

$$\Delta_{ii+1} \quad = \quad \theta_i - \theta_{i+1}\,. \tag{7.89}$$

[†]There are two other possibilities to fix the extra free parameter, which are probably better arbitrary conditions: *(i)* Set the energy of the left half the system, E_L, to be equal to $E_R + \Delta V$. ΔV can loosely play the role of a bias, which we set. *(ii)* Set a particle-hole like symmetry around the mean particle density 1/4, e.g., $a_1 = 1/4 - b$ and $a_4 = 1/4 + b$ where b is the free parameter.

Again, to make things simpler, we restrict the phase differences to be in the first quadrant

$$0 < \Delta_{ii+1} < \frac{\pi}{2} \,. \tag{7.90}$$

Now we can incorporate the the conditions for the current into the expression for the energy, writing everything in terms of particle number $n_i = a_i^2$,

$$E = \left(4n_1n_2 - \frac{I_0^2}{4}\right)^{1/2} + \left(4n_2n_3 - I_0^2\right)^{1/2} + \left(4n_3n_4 - \frac{I_0^2}{4}\right)^{1/2} . \tag{7.91}$$

Putting in the arbitrary condition 7.88 and the normalization through $n^2 = \frac{1}{2}\left(1 - n_2 - n_3\right)$ then

$$\begin{aligned} E &= \left(4n_2n_3 - I_0^2\right)^{1/2} + \left(2\left(1 - n_2 - n_3\right)n_2 - \frac{I_0^2}{4}\right)^{1/2} \\ &\quad + \left(2\left(1 - n_2 - n_3\right)n_3 - \frac{I_0^2}{4}\right)^{1/2} . \end{aligned} \tag{7.92}$$

Since we must have

$$0 \leq n_2 + n_3 \leq 1 \,, \tag{7.93}$$

it is best to set

$$n_2 = r^2 \cos\left(\phi\right) \,, \tag{7.94}$$

$$n_3 = r^2 \sin\left(\phi\right) \,, \tag{7.95}$$

to ensure that the computations stay within a physical range.

Fixing the current, then looking at the allowed densities and energies consistent with the instantaneous steady state, we find the contours in Figure 7.9. What we see is that, for a given energy and total number of particles, there is a range of different densities that give an instantaneous steady state.

7.4.5 Conclusions

In this section we have taken a closed system approach to quantum transport. We have found that using a variational procedure to find the microscopic states corresponding to some steady state current gives a large phase space of states. That is, for a given current, many states can realize the steady state. But we have to speculate on what this means in practice. If the basic intuition carries over to more complex systems (including quasi-steady states), then one would conclude that the phase space of states for given macroscopic conditions might lead to noise in the system. This noise would be due to the state of the system switching between the different steady states, either

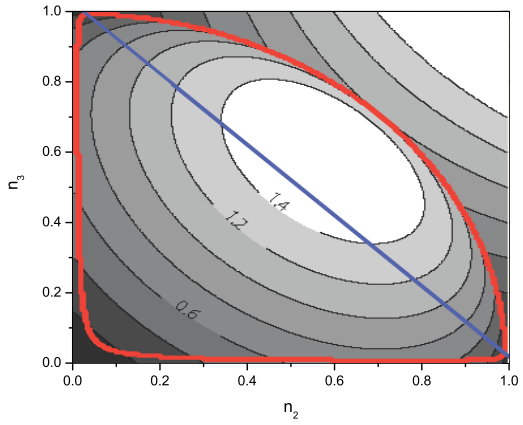


Figure 7.9: Steady state contours of constant energy. The parameters are one particle, $I_0 = 0.006\,J$, $J = 1$ a.u., and $n_1 = n_4$ (assumed for simplicity). The red line represents the boundary inside of which $\Im E = 0$. The blue line gives the region where $n_2 + n_3 \leq 1$.

under perturbations or by its own dynamics. However, it remains to be seen whether this basic idea carries over.

7.4.6 Subsidiary calculations

In this subsidiary section, we discuss some details of noninteracting states on a fermionic lattice. We work with fermionic operators, $c_l, c_l^\dagger$, for each site in the lattice, which satisfy the usual anticommutation relations

$$\begin{aligned} \left[c_l, c_k^\dagger\right]_+ &= \delta_{lk}\,, \\ [c_l, c_k] &= 0\,, \\ \left[c_l^\dagger, c_k^\dagger\right] &= 0\,. \end{aligned} \tag{7.96}$$

For noninteracting states on a lattice of L sites with a definite number of particles, the correlation matrix

$$M = \begin{bmatrix} \langle c_1^\dagger c_1\rangle & \langle c_1^\dagger c_2\rangle & \cdots & \langle c_1^\dagger c_{L-1}\rangle & \langle c_1^\dagger c_L\rangle \\ \langle c_2^\dagger c_1\rangle & \langle c_2^\dagger c_2\rangle & \cdots & \langle c_2^\dagger c_{L-1}\rangle & \langle c_2^\dagger c_L\rangle \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \langle c_{L-1}^\dagger c_1\rangle & \langle c_{L-1}^\dagger c_2\rangle & \cdots & \langle c_{L-1}^\dagger c_{L-1}\rangle & \langle c_{L-1}^\dagger c_L\rangle \\ \langle c_L^\dagger c_1\rangle & \langle c_L^\dagger c_2\rangle & \cdots & \langle c_L^\dagger c_{L-1}\rangle & \langle c_L^\dagger c_L\rangle \end{bmatrix}, \tag{7.97}$$

completely specifies the state.[‡]

There are L^2 complex parameters in this matrix. However, there are restrictions M must satisfy in order to describe a proper quantum state, and in particular, a pure quantum state since we are dealing with closed systems. A pure noninteracting state of N particles can always be written as

$$M = U \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \ddots & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} U^\dagger \tag{7.99}$$

where U is a unitary matrix and there are N 1's on the diagonal and the rest of the elements are zeros. That is, M must define a projection operator in order for it to describe a pure quantum state. Diagonalizing the noninteracting Hamiltonian by a unitary transformation on the creation/annihilation operators, which gives

$$b_k = \sum_j U_k^j c_j \,, \tag{7.100}$$

$$b_k^\dagger = \sum_j U_k^{\dagger j} c_j^\dagger \,, \tag{7.101}$$

so

$$\langle b_l^\dagger b_k \rangle = \sum_{i,j} U_l^{\dagger i} U_k^j \langle c_i^\dagger c_j \rangle \,. \tag{7.102}$$

This gives

$$M^2 = UPU^\dagger UPU^\dagger = UPU^\dagger = M \,, \tag{7.103}$$

that is, M is a projector. Since

$$M_{ij} = \langle c_i^\dagger c_j \rangle \,, \tag{7.104}$$

then

$$\left(M^2\right)_{ij} = \sum_k \langle c_i^\dagger c_k \rangle \langle c_k^\dagger c_j \rangle \equiv \langle c_i^\dagger c_j \rangle \,. \tag{7.105}$$

We can treat the real and imaginary parts as independent variables. Thus, lets set

$$M = A + iB \,. \tag{7.106}$$

[‡]We can extend this to noninteracting states without definite particle number by just including correlation functions with 2 creation or 2 annihilation operators, but for our interests we have taken

$$\langle c_l^\dagger c_k^\dagger \rangle = \langle c_l c_k \rangle = 0 \,. \tag{7.98}$$

This is different notation than above. Since M is Hermitian,

$$\begin{aligned} A_{ij} &= A_{ji}\,, \\ B_{ij} &= -B_{ji}\,, \\ B_{ii} &= 0\,. \end{aligned} \tag{7.107}$$

The conditions to be a pure, proper quantum state are then

$$\sum_k \left(A_{ik}A_{kj} - B_{ik}B_{kj}\right) = A_{ij}\,, \tag{7.108}$$

$$\sum_k \left(B_{ik}A_{kj} + A_{ik}B_{kj}\right) = B_{ij}\,. \tag{7.109}$$

The conditions just given are way too many, which is why it is hard to work with the correlation matrix to construct the instantaneous steady states. This is why, in the section 7.4.3.1, we work directly with the unitary matrix elements to give the number of free parameters.[§]

7.5 Conclusions

In this Chapter, we first introduced a methodology that we believe will help examine issues associated with computing transport properties of nanoscale objects. The idea behind the methodology is to model, in some way or another, all processes that physically occur: particle injection/depletion, relaxation, and interactions. Then, make a four-probe measurement of the current-voltage characteristics. By taking this approach, we argued physically that one can make at least one drastic approximation, e.g., model the injection/depletion processes as Markovian. The relaxation processes can also be approximated. Although approximations in this regard will prohibit direct quantitative comparison with experiment, which is not such a loss as this is not the intention of the method. With these simplifications, one can use MPS simulations (e.g., the method in Chapter 2) to do transport calculations.

There are a number of questions one can examine with this method. Two in particular we introduced within the chapter: one is the examination of interaction (even weak interaction) effects on nonequilibrium transport, e.g., how does the replacement of the interacting dynamical problem by a (static or dynamic) density-functional problem affect the calculated physical properties? Two is an examination of the allowed steady states of the model, e.g., under what conditions can what get multiple steady states (i.e., the zero right eigenvalues of the Lindbladian), how do these differ in their

[§]In terms of the correlation matrix, we want to minimize $B = 4B_{12}^2 + 4\sum_{l=2}^{L-1}\left(B_{i-1,i} - B_{i,i+1}\right)^2 + 4B_{L-1,L}$, subject to the constraints $2B_{L/2,L/2+1} = I_0$, $\sum_{l=1}^{L-1}\left(A_{i,i+1} + A_{i+1,i}\right) = E_0$, $\mathrm{tr}A = N_0\left(= \frac{L}{2}\right)$, $A^2 - B^2 = A$, $[B,A]_+ = B$, $B^T = -B$, and $A^T = A$. The constraints are too many in number here, hence why it is best to work with the unitary transformations.

physical properties, and what perturbations can create transitions between them? There are other questions we did not discuss: the method can also be used to look at conditions for local equilibrium in the leads, e.g., how far does one have to go away from the junction in order to get to a point where the electrons look like a set of noninteracting particles? Given the wealth of potential questions that can be examined and their tremendous importance, we believe that the method proposed, albeit still in the early development phase, is worth further investigation.

Chapter 8

Entropy of "multi-channel" models

We investigate how the entanglement entropy behaves on star and tree lattices of free fermions, which we use as test systems for multi-channel models. In particular, we examine the XX model in one dimension and its fermionic extension to branched lattices. We find that the von Neumann entropy on the star lattice, for a branch plus the center site, goes through a maximum and then decays to 1 as a function of the number of branches. However, its decay to 1 goes as $(\log_2 N)/N$, where N is the number branches. Correspondingly, the $\alpha-$Renyi entropy decays to 1 as $1/N^\alpha$. This gives an indication that it is not efficient to simulate branched star lattices using tensor networks that have the identical geometry as the lattice. We also consider a tree geometry which is related to the physical situation of a one-dimensional lattice connected to independent, non-interacting reservoirs/baths. For this latter case, we find that entanglement increases with the logarithm of the number of sites contained in the block, not just the linear length of the block.

8.1 Introduction

In this Chapter, we examine the behavior of the entanglement entropy of free fermions on different lattice geometries besides just the one-dimensional case. The two lattice geometries we consider are shown in Figure 8.1. These types of geometries show up in strongly dissipative systems (see Chapters 5 and 6) and in multi-channel models (e.g., multi-channel Kondo models[106,107]). One can simulate lattices of this type using tensor networks which preserve the geometry of the system. That is, at each lattice site, one replaces the site with a tensor that has, in addition to the local dimension, indices corresponding directly to the bonds on the lattice. However, it is not clear that this is the most efficient way, or even that it is an efficient way, to do the simulation. This is what we seek to determine in this chapter. Some of the basics of the calculations are shown in Appendices A and B.

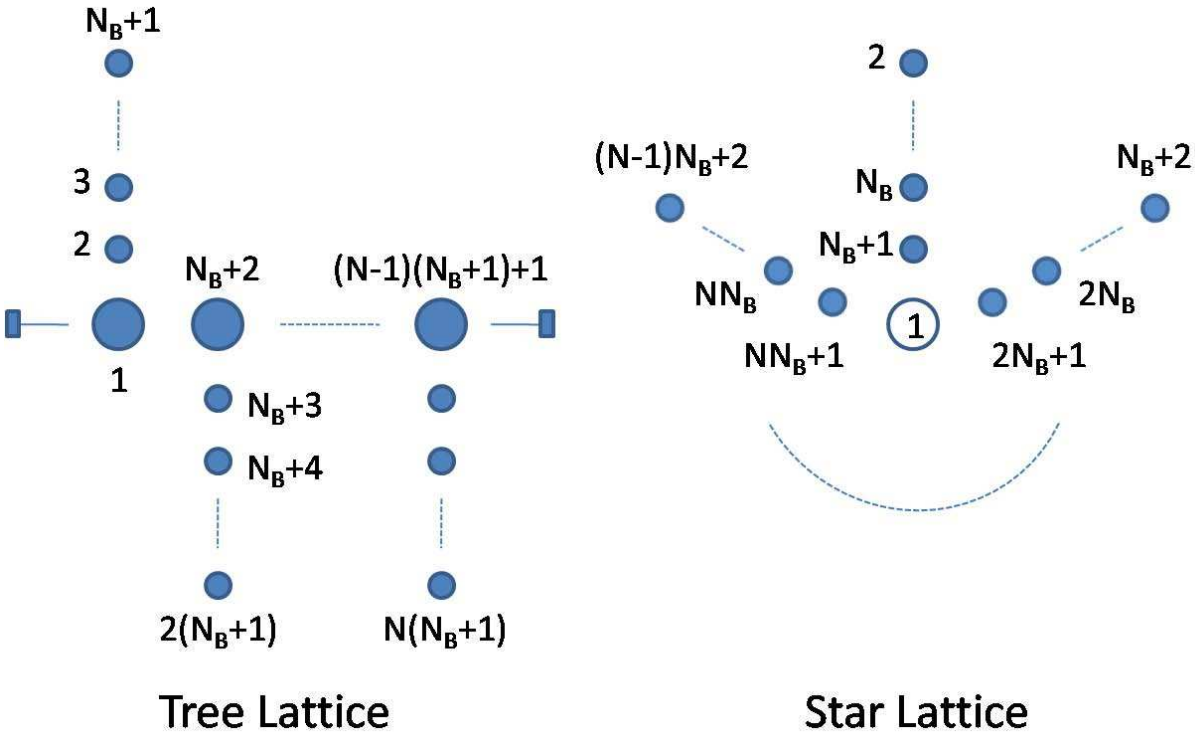


Figure 8.1: The tree and star lattice geometries together with a numbering system if one wanted to transform their respective Hamiltonians back into a spin model. The squares on the ends of the tree lattice represent taking a periodic system. With the numbering system shown, flip operators on the different branches will appear (together with a flip around the ring for the tree lattice) when writing the respective spin Hamiltonian.

8.2 Star lattices

The Hamiltonian of the star lattice is

$$H = 2gc_1^\dagger c_1 + \sum_{l=1}^{N}\left[-c_1^\dagger c_{l_{N_B}} - c_{l_{N_B}}^\dagger c_1 - \sum_{k=1}^{N_B-1}\left(c_{l_k}^\dagger c_{l_k+1} + c_{l_k+1}^\dagger c_{l_k}\right) + 2g\sum_{k=1}^{N_B} c_{l_k}^\dagger c_{l_k}\right], \tag{8.1}$$

where here $l_k \equiv N_B\,(l-1)+k+1$, which divides the lattice up into N branches of N_B sites each, and there is one center site. The star geometry is nice because the branches are permutationally invariant. Thus, we expect that the entanglement across the bonds to the branches should decrease for large N in correspondence with mean field theory and the quantum de Finetti theorem. However, we find that there is a locally bound fermion that always keeps the entropy finite when the block of sites contains the middle site. In the following subsections, we describe our computational approach and results, and some exact results for the form of the entropy for large N.

8.2.1 Computational approach

The Hamiltonian on the star lattice can be written as, ignoring a uniform on-site potential of $2g$,

$$H = \sum_{n=1}^{N}\left(H_{0n} + H_n\right), \tag{8.2}$$

where

$$H_{0n} = -\left(c_{nN_B}^\dagger c_0 + c_0^\dagger c_{nN_B}\right) \tag{8.3}$$

is the interaction Hamiltonian of the center site (mode c_0) with branch n**,** and

$$H_n = -\sum_{l=1}^{N_B-1}\left(c_{nl}^\dagger c_{nl+1} + c_{nl+1}^\dagger c_{nl}\right) \tag{8.4}$$

is the Hamiltonian of the branch. We have relabeled the operators according to their branch n and site l on the branch. We can diagonalize all the branches by a sine transform,

$$U_{ST} = \mathrm{I}_1 \oplus \left(\mathrm{I}_{N_B} \otimes V_{ST}\right), \tag{8.5}$$

where

$$(V_{ST})_{lk} = \left(\frac{2}{N_B+1}\right)^{1/2} \sin\frac{k\pi l}{N_B+1}, \tag{8.6}$$

e.g.,

$$b_{nk}^\dagger = \left(\frac{2}{N_B+1}\right)^{1/2}\sum_{l=1}^{N_B} c_{nl}^\dagger \sin\frac{k\pi l}{N_B+1}, \tag{8.7}$$

and $k = 1, \ldots, N_B$. This gives the branch Hamiltonian as

$$H_n = \sum_{k=1}^{N_B} \epsilon_k b_{nk}^\dagger b_{nk} \,, \tag{8.8}$$

with $\epsilon_k = -2\cos\left(k\pi/N_B + 1\right)$, and again, the on-site potential is being ignored. The interaction of the branch n with the center site will take the form

$$H_{0n} = -\left(\frac{2}{N_B+1}\right)^{1/2} \sum_{k=1}^{N_B} \sin\frac{k\pi N_B}{N_B+1} \left(b_{nk}^\dagger b_0 + b_0^\dagger b_{nk}\right) \,, \tag{8.9}$$

where $b_0 \equiv c_0$.

Then we can Fourier transform all the branches with the real orthogonal transformation

$$(V_{FT})_{nm} = \frac{1}{\sqrt{2}}\frac{1}{\sqrt{N}} \left\{ e^{\imath\frac{2\pi}{N}mn} + e^{\imath\frac{2\pi}{N}(N-m)n} \right\} \,, \; m = 1 \ldots \lceil N/2 \rceil - 1 \,, \tag{8.10}$$

and

$$(V_{FT})_{n(N-m)} = \frac{\imath}{\sqrt{2}}\frac{1}{\sqrt{N}} \left\{ e^{\imath\frac{2\pi}{N}mn} - e^{\imath\frac{2\pi}{N}(N-m)n} \right\} \,, \; m = 1 \ldots \lceil N/2 \rceil - 1 \,, \tag{8.11}$$

which takes

$$\left(d_{\{m\}k}^\dagger\right) = \left(b_{\{n\}k}^\dagger\right) V_{FT} \,. \tag{8.12}$$

This gives the total Hamiltonian as

$$H = \sum_{m=1}^{N-1}\sum_{k=1}^{N_B} \epsilon_k d_{mk}^\dagger d_{mk} + \sum_{k=1}^{N_B} \left[\epsilon_k d_{Nk}^\dagger d_{Nk} - \sqrt{N} \left\{ \left(V_{ST}^\dagger\right)_{kN_B} d_{Nk}^\dagger d_0 + d_0^\dagger d_{Nk} \left(V_{ST}\right)_{kN_B} \right\} \right] \,, \tag{8.13}$$

where we have used that the center site (now, $d_0 \equiv b_0 \equiv c_0$) only couples to the symmetric mode of the branches, which comes from the relation

$$\sum_{n=1}^{N} e^{-\imath\frac{2\pi}{N}mn} = N\delta_{m,N} \,. \tag{8.14}$$

If we want, we can sine transform back the operators d_{Nk},

$$f_l^\dagger \equiv \sum_{k=1}^{N_B} \left(V_{ST}^\dagger\right)_{kN_B} d_{Nk}^\dagger \,, \tag{8.15}$$

and set , yet again, $f_0 \equiv c_0$, to get

$$H = \sum_{m=1}^{N-1}\sum_{k=1}^{N_B} \epsilon_k d_{mk}^\dagger d_{mk} - \sqrt{N}\left(f_{N_B}^\dagger f_0 + f_0 f_{N_B}\right) - \sum_{l=1}^{N_B}\left(f_l^\dagger f_{l+1} + f_{l+1}^\dagger f_l\right) \,. \tag{8.16}$$

Thus, we have a set of modes independent of the center site, and then a nearest-neighbor hopping lattice for the center site with the symmetric modes of the branches. Actually, the fact that only the symmetric channel couples to the center site allows us to write down exactly how the entropy will behave for a large number of channels.

8.2.2 Exact entropy scaling

Note that the coupling strength in equation 8.16 of the center mode with the "lattice" increases with the square root of the number of branches N. This, in the limit of large N, creates a bound fermion on a state that is an equal superposition of the center site and the symmetric combination of end sites on the branches, with only small corrections. The rest of the energy eigenstates will be symmetric combinations of the sites at the same position on the branches and there will be $\sim N_B/2$ fermions to put into these states (e.g., there will be half filling of the completely symmetric "branch").

For large N and $N \gg N_B$, then, one expects that tracing out all but one branch and the center site, that the filled states of the first term in equation 8.16 will not create entanglement and that only the mixing of the symmetric modes will create partially filled levels to order $1/N$ (because the correlation functions come from the expectation values of creation/annihilation operators). Define the block B as one of the branches (and the block $B+0$ as one of the branches plus the center site) and let us calculate the entropy of that block. The local filling factors on the branch will be of the form

$$\lambda_l = 0 + \frac{C_l}{N}\,,\, 1 - \frac{C_l}{N}\,, \tag{8.17}$$

except for one mode, which is the bound mode to the center of the star, which has filling

$$\frac{1}{2} + \frac{C_0}{N}\,. \tag{8.18}$$

From Appendix B, we know that the entanglement entropy comes from the filling of local fermionic modes, therefore, we can write out an expression of the entanglement entropy in the large N (and $N \gg N_B$) limit. For the center mode, the contribution to the von Neumann entropy will be

$$-\left(\frac{1}{2}+\frac{C_0}{N}\right)\log_2\left(\frac{1}{2}+\frac{C_0}{N}\right) - \left(\frac{1}{2}-\frac{C_0}{N}\right)\log_2\left(\frac{1}{2}-\frac{C_0}{N}\right) = 1 + \mathcal{O}\left(\frac{1}{N^2}\right)\,, \tag{8.19}$$

and each other mode will have a contribution

$$-\left(\frac{C_l}{N}\right)\log_2\left(\frac{C_l}{N}\right) - \left(1-\frac{C_l}{N}\right)\log_2\left(1-\frac{C_l}{N}\right) = \frac{C_l}{N}\left(\log_2 N - \log_2 C_l + \frac{1}{\ln 2}\right)\,. \tag{8.20}$$

We can sum up the contributions and write the functional form for S,

$$S_{B+0} = \mathcal{A}\left(\frac{\log_2 N + \mathcal{C}}{N}\right) + 1\,. \tag{8.21}$$

That is, we expect that there is a polynomial decay to entropy 1, with a logarithmic correction. Similarly, for just a branch, the entropy will behave the same way but without the center site contribution, thus

$$S_B = \mathcal{A}\left(\frac{\log_2 N + \mathcal{C}}{N}\right)\,. \tag{8.22}$$

The asymptotic value of the entropy in 8.21 is due to having a half fermion on the center site, and indeed, this local contribution will occur in any other geometry where a single site is connected to N branches, such as the pom-pom geometry (which has a stem with an end site connected to the branches, the star geometry is a particular case of the pom-pom). In addition, even if one rescales the connections to the center site in equation 8.16 by a factor proportional to $1/\sqrt{N}$, there will still be a nonzero asymptotic value, which will depend, though, on the length of the branch. For large branches, there will be half-filling on the "lattice" containing the center site, thus creating the same asymptotic value of one.

We can also examine how we expect the Renyi entropy to behave. The contribution of the center mode to the Renyi entropy 9.30 is

$$\frac{1}{1-\alpha}\log_2\left(\left(1-\frac{1}{2}-\frac{C_0}{N}\right)^{\alpha}+\left(\frac{1}{2}+\frac{C_0}{N}\right)^{\alpha}\right) = 1+\mathcal{O}\left(\frac{1}{N^2}\right)\,. \tag{8.23}$$

The contribution of each other mode is

$$\frac{1}{1-\alpha}\log_2\left(\left(1-\frac{C_l}{N}\right)^{\alpha}+{\frac{C_l}{N}}^{\alpha}\right) = \mathcal{A}\left(\frac{N^{1-\alpha}-\mathcal{C}\alpha}{N}\right)+\mathcal{O}\left(\frac{1}{N^2}\right)\,, \tag{8.24}$$

where at large N it will behave as $1/N^{\alpha}$ (where we only consider $0<\alpha<1$). Thus, the functional form of the Renyi entropy will be

$$S^{\alpha}_{B+0} = \mathcal{A}\left(\frac{N^{1-\alpha}-\mathcal{C}\alpha}{N}\right)+1\,. \tag{8.25}$$

8.2.3 Computational results

The main results for the von Neumann entropy on the star geometry are shown in Figures 8.2 and 8.3. These two figures show the behavior of the entropies S_{B+0} and S_B as a function of the number of branches. The branch length in these results is $N_B = 4$, and the on-site potential is $g = 0^+$ (see below for a definition of 0^+). At small number of branches, the entropy first increases, peaks, then decays. For large N ($N \gg N_B$), the entropy decays as in equation 8.21. Further, the form of

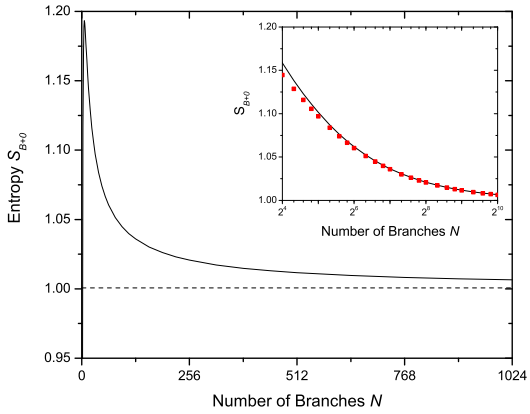


Figure 8.2: Entanglement entropy of a branch plus the center site as a function of the number of branches. The decay is as discussed above. The inset shows the best fit to the form 8.21, with the black line the fit and the red squares the data, in the range shown, which gives $\mathcal{A} = 0.7143$ and $\mathcal{C} = -0.4502$. Even at smaller values of $N < 16$, the basic form 8.21 has the same shape as the curve, including the peak (see below).

equation 8.21 seems to be essentially valid all the way to the smallest $N = 2$, but one has to take $N \to N + N_0$, where N_0 is some constant.

We can also look at the opposite limit by examining $N_B \gg N$, i.e., look at the growth of the entropy as a function of the branch length. Figure 8.4 shows the results for $N = 4$ and $g = 0^+$. As expected, the entropy grows with the logarithm of the branch length. The coefficient of this logarithm will drop off as $1/N$, as the finite number of fermions $\sim N_B/2$ will only show up in the coefficient $\mathcal{C}$ in equation 8.21 and 8.22. If one holds N fixed, the entanglement will always diverge logarithmically in N_B, regardless of the value of N - only the coefficient of the divergence will change.

Also, there is a peak as a function of the number of branches, see Figure 8.5. The peak shifts to lower values of N as we increase N_B. One further situation we can look at is when $N = N_B$. Does this entanglement diverge or decay? We find that the entanglement decays, but much slower than the functional forms above, see Figure 8.6.

We will discuss in the conclusions what these results mean for simulations using tensor networks.

8.3 Tree lattices

The Hamiltonian of the tree lattice is

$$H = \sum_{l=1}^{N} \left[-c^\dagger_{l_0} c_{(l+1)_0} - c^\dagger_{(l+1)_0} c_{l_0} - \sum_{k=0}^{N_B-1} \left(c^\dagger_{l_k} c_{l_k+1} + c^\dagger_{l_k+1} c_{l_k} \right) + 2g \sum_{k=0}^{N_B} c^\dagger_{l_k} c_{l_k} \right] , \tag{8.26}$$

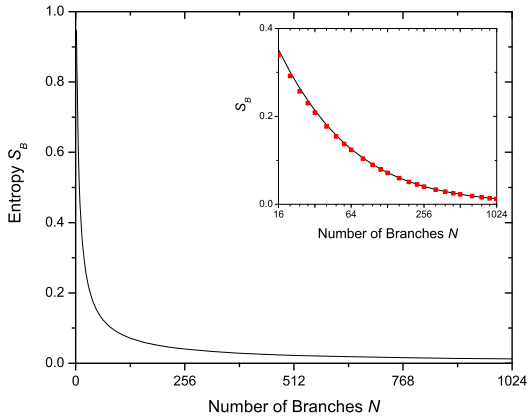


Figure 8.3: Entanglement entropy of a branch as a function of the number of branches. The decay is as discussed above. The inset shows the best fit to the form 8.22, with the black line the fit and the red squares the data, in the range shown, which gives $\mathcal{A} = 1.1989$ and $\mathcal{C} = +0.7021$. Even at smaller values of $N < 16$, the basic form 8.22 produces the curve well.

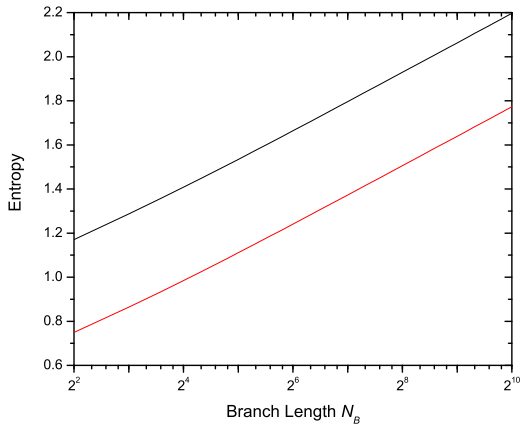


Figure 8.4: Entanglement entropy of the branch and branch+0 as a function of branch length. At large N_B, these curves behave as $\mathcal{A}\log_2 N_B + \mathcal{C}$, with slope 0.1321. For small N_B, the curves must taper off due to the bound state.

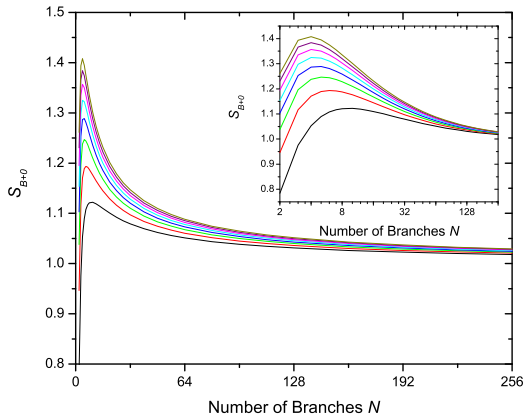


Figure 8.5: Entanglement entropy as a function of N, for several values of $N_B = 2, 4, 6, 8, 10, 12, 14, 16$. We see that the peak at small N shifts to lower values of N as we increase N_B. For large N_B, this peak shifts to $N = 3$ where it stays.

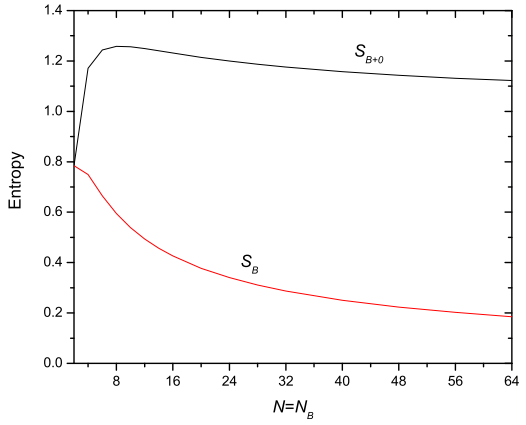


Figure 8.6: Entanglement entropy as a function of N with $N = N_B$. We see that the entanglement does decay, but is not clear that S_{B+0} is decaying to 1 and S_B to 0, rather than some other finite constant.

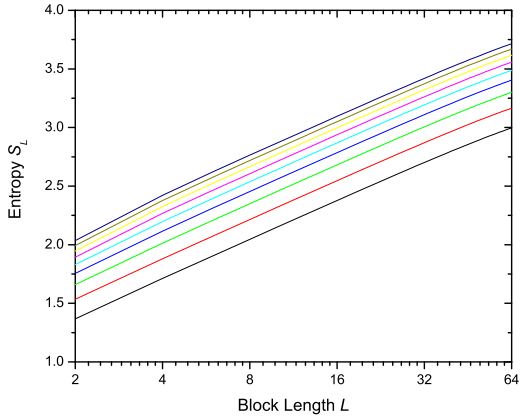


Figure 8.7: Entanglement Entropy as a function of block length (measured in terms of the number of lattice groups) for successive values of $N_B = 0, 2, 4, 6, 8, 12, 14, 16$. The entropy increases as a function of N_B. Fitting each of the curves gives a coefficient of the logarithm between 0.324 and 0.33, with values increasing with N_B. There are still significant finite size effects, as is clear for $N_B = 0$ where the slope should be $1/3$ [108], so the increasing coefficient trend is not significant. For instance, taking $N = 1024$ and $N_B = 0, 2, 4$, one finds coefficients 0.3339, 0.3340, 0.3349 when fitting from size 2 to size 64 blocks (even in this case, there are effects from not taking blocks much bigger than N_B).

where we have defined $l_k \equiv N_B\,(l-1) + l + k$, which divides up the lattice into N *lattice groups* of $N_B + 1$ sites. This just corresponds to a uniform hopping model on the tree lattice geometry in Figure 8.1. For part of this work, we will consider a periodic lattice, identifying the $(N+1)^{th}$ lattice group with the 1^{st} lattice group. Otherwise, we will drop the hopping term at the end of the lattice.

The entanglement entropy we are interested in is S for blocks of L contiguous lattice groups as a function of both L and N_B. Also of interest is the correlation function $\langle \sigma^x_l \sigma^x_{l+n} \rangle$ where n is restricted to run over the main lattice sites, e.g., $\langle \sigma^x_{l_0} \sigma^x_{l_0+n_0} \rangle$. Notice that the determinant 9.20 will have elements running over the branches.

8.3.1 Entanglement and correlations for $N \gg N_B, L$

Some results for a periodic lattice of length $N = 256$ and at $g = 0^+$ (for the meaning of $0^\pm$ see below) are shown in Figures 8.7, 8.8, 8.9, and 8.10. These results are for $N \gg N_B, L$. That is, we have a large lattice and we examine the effect of short side chains on the properties of the lattice at short, e.g., $\sim N_B$, to medium distances, e.g., $\sim L$. The entropies all grow logarithmically with the block length, L, where the blocks include the side chains and the length is measured in terms of the number of lattice groups. The correlations also decay with a power law.

There are not strong definitive conclusions that we can draw from the above results. From the growth of the entanglement entropy, we expect that the MPS dimension grows at approximately

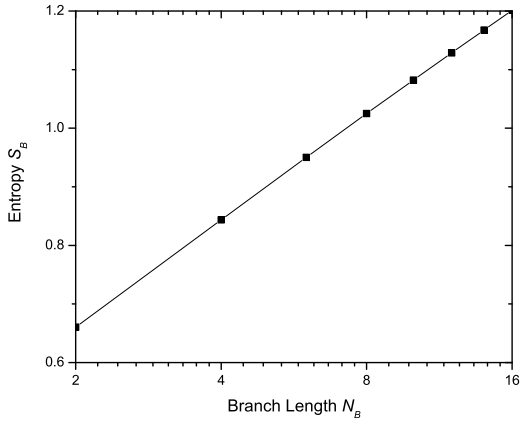


Figure 8.8: Entanglement entropy of a single branch as a function of branch length. The slope of this curve is 0.18, but is expected to go to 1/6 as $N_B \to \infty$ (with $N_B \ll N$).

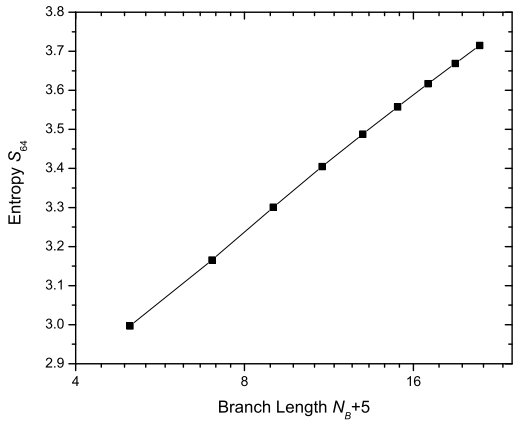


Figure 8.9: Entanglement entropy for block length $L = 64$ as a function of N_B. If one plots S_{64} directly versus N_B, one finds that for these short branch lengths that the growth seems faster than $\log_2 N_B$. However, it is clear that the growth should not be $\log_2 N_B$ as S_{64} has to go to a finite value at $N_B = 0$. Thus, we plot the entropy versus $\log_2 (N_B + N_0)$ with $N_0 = 5$.

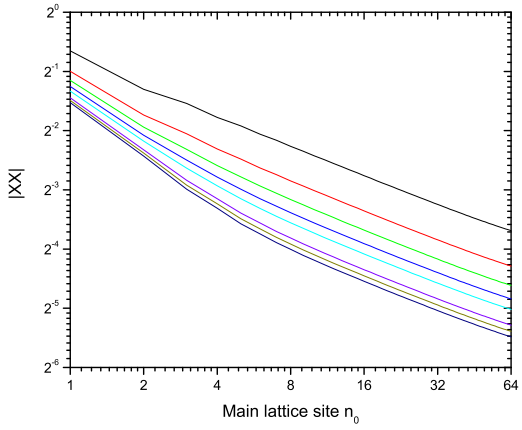


Figure 8.10: The correlation function $|\mathrm{XX}| \equiv \left|\langle \sigma^x_{l_0} \sigma^x_{l_0+n_0} \rangle\right|$ where $n_0 \equiv l$ is restricted to live on the main lattice. All curves display power law behavior at long distances, with a decay of $1/n_0^{1/2}$ (e.g., fitting the curves above from $n_0 = 16 \to 64$ give the power as 0.46, which converges rapidly to 0.5 as we increase N, but still fit from $n_0 = 16 \to 64$ to reduce the correlations from feeling the periodic effect. For instance, for $N = 1024$ and $N_B = 0, 2, 4$, we get 0.498,0.499,0.4999, respectively, for the power).

the same rate for the lattice with and without short side chains. However, the absolute value of the entropy is larger and we expect that we do need a larger dimension. On the other hand, the magnitude of the correlations are reduced. The increase in entropy thus seems to be determined only by a shift in the constant in the expression $S = a \log_2 L + b$. This shift is most likely due to simply the short distance correlations of the side chains across the boundary. To determine whether this is the case, we could examine the conditional entropies and mutual information between the blocks, with and without some of the side chains (near the edges of the block), and the rest of the lattice. This may be a worthwhile calculation, although it seems clear the branches near the boundary are increasing the constant in the entropy.

8.3.2 Generic computational errors

It turns out that within these calculations there is a finite size error that breaks the translational symmetry of the lattice. This error actually is detrimental to obtaining accurate correlation functions for finite systems, and is in fact easy to fix. It also has relevance to simulations with iTEBD as that method breaks translationally symmetry and thus can incur the same type of error, although it is not so easy to fix in that case.

Consider a periodic lattice with no branches, for example. There is a degeneracy at the Fermi level for left and right movers. The computer does not know how to fill in these levels, and they will end up being randomly filled depending on slight numerical errors (of machine precision). That is,

the left and right movers at the Fermi level have zero energy for $g = 0$, but numerically each one will acquire a slightly positive or negative energy. Regardless of the filling, there will be an error proportional to $1/N$ at each site (there may or may not be the correct filling of the levels at the Fermi level, giving ± 1 extra particle, or breaking the symmetry and having one extra left or right mover). Therefore, we can only accurately calculate correlators out to a distance $L \ll N$. A suitable choice for L would be $\sqrt{N}$, so that errors will go to zero as one increases N. We have gone beyond that in the calculations, but this is acceptable so long as one recognizes that there are random errors present.

One can not get rid of these $1/N$ errors. However, one can take the randomness out of the results by just forcing there always to be one extra particle (or one less) which is put into (or taken out of) the zero energy state. This gives much more consistent results, but still has errors in solving the periodic Hamiltonian that are $1/N$ for local properties. Here is where we define $0^{\pm}$. The calculations are not done with $g = 0$ for the periodic system, but rather with $g = 0^{\pm}$ where $0 < 0^{+} < \min |\epsilon_k - \epsilon_{k'}|$ (for $k \neq k'$ and ignoring degeneracy). Likewise for 0^{-}. That is, $0^{\pm}$ represent some small constants with magnitude less than the level spacing so that the system either subtracts or adds one extra particle. Thus, we will not have exactly half-filling, but will have it to $\mathcal{O}(1/N)$. For a lattice with no branches, the level spacing as function of k is given by

$$|\epsilon_{k+1} - \epsilon_k| \cong \left| \frac{4\pi}{N} \sin \frac{2\pi k}{N} \right| \leq \frac{4\pi}{N} \,. \tag{8.27}$$

Thus, in particular, we choose $0^{+} \sim \varepsilon/N$, where ε is some small number and the factor of $1/N$ will ensure that 0^{+} decreases as we increase the lattice length (we could also put in a factor of $1/N_B$ to ensure that as we increase the side chain length there are not levels in between 0 and 0^{+}).

In addition, the above fix preserves a very important property: translational invariance (further, it gives a pure state with no currents in the system). What happens if one fills up a random superposition of left and right movers at the Fermi level is that the translational invariance will be broken to $\mathcal{O}(1/N)$, which has a tremendous effect on correlation functions: they cease to have power law behavior at even moderate distances. Using eigensystem routines will, in fact, produce a superposition of left and right movers, as the routines will find a real unitary matrix, which has to contain such a superposition (for left and right movers, not just the ones at the Fermi level). Translational symmetry will be broken unless either both left and right movers are filled/unfilled, or that only one of them is filled (but which then gives the state a current). Occupying any superposition of left and right movers will induce oscillations in properties (e.g., either the on-site correlation matrix elements or the nearest neighbor elements) on the lattice unless both conjugate superposition states are occupied.

This latter fact has significance for translationally invariant computations with MPS. Because,

for instance, the iTEBD algorithm breaks translationally symmetry from $\prod \Gamma$ into a symmetry $\prod \Gamma^A \Gamma^B$, any broken translationally invariance will translate into a poor representation of correlation functions, even at moderate distances. One suspects, as well, that without proper care, this will dominate the usability of MPS (rather than the effective χ). This is something to look into further.

8.3.3 Computational approach

One can take a brute force computational approach to the above problem, e.g., just diagonalize the $N_T \times N_T$ matrix H directly, fill up the correlation matrix in the eigenbasis, transform back to the lattice basis, and compute quantities such as the correlation functions and the entropies. However, a tremendous computational speedup can be achieved by doing part of the diagonalization of the Hamiltonian analytically. The approach is to rewrite the tree Hamiltonian in terms of levels and take the Fourier transform of each level, which gives us a block diagonal Hamiltonian, with each block being a Fourier component.

Take the Hamiltonian 8.26 and rewrite it as

$$H = \sum_{n=1}^{N_B+1} H_n + \sum_{n=1}^{N_B} H_{n,n+1} \,, \tag{8.28}$$

where n is labeling the level. Lets simply take $g = 0$ and a periodic system. Then, the $n = 1$ level has the Hamiltonian

$$H_1 = -\sum_{l=1}^{N} \left(c_{1l}^\dagger c_{1l+1} + c_{1l+1}^\dagger c_{1l} \right) \,, \tag{8.29}$$

and all other levels, $n > 1$ have $H_n = 0$. The fermion operators have been relabeled with the first index indicating their level and the second index their distance along the level. The level hopping Hamiltonian has the form

$$H_{n,n+1} = -\sum_{l=1}^{N} \left(c_{nl}^\dagger c_{n+1l} + c_{n+1l}^\dagger c_{nl} \right) \,, \tag{8.30}$$

for all n.

Fourier transforming the fermion operators within each level,

$$c_{nl} = \frac{1}{\sqrt{N}} \sum_{k=1}^{N} b_{nk} e^{\imath \frac{2\pi}{N} kl} \,, \tag{8.31}$$

we get the Hamiltonian

$$\begin{aligned} H &= \sum_{k=1}^{N} \left[\epsilon_k b_{1k}^\dagger b_{1k} - \sum_{n=1}^{N_B} \left(b_{nk}^\dagger b_{n+1k} + b_{n+1k}^\dagger b_{nk} \right) \right] \\ &\equiv \sum_{k=1}^{N} H_k \,, \end{aligned}$$

with $\epsilon_k = -2\cos(2\pi k/N)$. Now we can numerically diagonalize all the H_k, which has a cost of $N(N_B+1)^3$ (in practice it will scale better as each H_k is a tridiagonal matrix), to find the unitary transformations U_k and eigenvalues E_k.

Thus, the total unitary transformation is first a Fourier transform on each level defined by the elements

$$(V_{FT})_{lk} = \frac{1}{\sqrt{N}} e^{\imath \frac{2\pi}{N} kl} \,,\; k = 1 \ldots N \,, \tag{8.32}$$

(this matrix can be constructed, for instance, with the intrinsic Matlab routine ifft, but with appropriate normalization and a cyclic rotation of the columns and rows by -1 steps) where this matrix has to be blown to U_{FT} up using the indexing in equation 8.26:

$$U_{FT} = \mathrm{I}_n \otimes V_{FT} \,, \tag{8.33}$$

where I_n is the $n \times n$ identity matrix. Then, the total unitary transformation is

$$U = U_{FT} \bigoplus_k U_k \,. \tag{8.34}$$

The eigenvalues are

$$E = \bigoplus_k E_k \,. \tag{8.35}$$

By using this transformation, one can easily look at the Fermi level states, which has its surface at $k = N/4 \,,\, 3N/4$. Therefore, one can, say, have half-filling but break the left-right moving symmetry or, by mixing left and right moving states, break translational symmetry.

However, it turns out that the speed of the full program is much slower than the brute method. The diagonalization step is tremendously faster (reducing $N^3(N_B+1)^3$ down to $N(N_B+1)^3$, although the actual scaling will not be quite that reduction), but, one winds up with, since U is complex, a correlation matrix M that is complex. Thus, in particular, taking the determinant of part of M is much more difficult, and likewise so are the matrix multiplications of $N_T \times N_T$ matrices. There are a few things happening: there are many more nonzero terms in the unitary transformations due to errors in computing exponents (whereas the Lapack diagonalization routines are very good at eliminating numerical errors in the transformation matrices). To fix this, one can use the

intrinsic FFT routines (fft, dst, etc. for Matlab), which will insure unitarity to numerical precision. In addition, the complex matrix U_{FT} has many more non-zero elements than is necessary. If one instead uses a real version of U_{FT}, which is possible in this case because of the degeneracy of left and right movers, this reduces the number of matrix elements (essentially by half) because of the switch from complex to real. This has a profound effect both on the matrix multiplication using U_{FT} and the corresponding operations on the correlation matrix (which will now also be real).

We do this latter procedure. We find a real orthogonal transformation by computing

$$(V_{FT})_{lk} = \frac{1}{\sqrt{2}}\frac{1}{\sqrt{N}}\left\{e^{\imath\frac{2\pi}{N}kl} + e^{\imath\frac{2\pi}{N}(N-k)l}\right\}\,,\; k = 1\ldots\lceil N/2\rceil - 1\,, \tag{8.36}$$

and

$$(V_{FT})_{l(N-k)} = \frac{\imath}{\sqrt{2}}\frac{1}{\sqrt{N}}\left\{e^{\imath\frac{2\pi}{N}kl} - e^{\imath\frac{2\pi}{N}(N-k)l}\right\}\,,\; k = 1\ldots\lceil N/2\rceil - 1\,. \tag{8.37}$$

This will give a real U that diagonalizes the Hamiltonian, and also a real correlation matrix. Further computations with the correlation matrix will thus be much easier to perform.

There are many other tricks one can use. For instance, at zero temperature, one need not compute the correlation matrix by full matrix multiplications, but rather can take an isometry $\mathcal{U}$ formed from the columns of U which correspond to negative eigenvalues of H. Then M is just $\mathcal{U}\mathcal{U}^\dagger$, and this can be reduced further when computing the reduced correlation matrix, e.g., just take a matrix $\mathcal{U}$ formed from the columns of U with negative eigenvalue and the rows of U that correspond to the sites in the block. Indeed, all the properties of the reduced correlation matrix can be found from this reduced form of U. For instances, the eigenvalues of the reduced correlation matrix (all you need to look at entropies) can be found directly from the reduced U by performing a singular value decomposition (then squaring the singular values). However, we just work with the reduced correlation matrix.

8.3.4 Entanglement for $N \sim N_B$

The real physical limit that we are interested in is when the bath is as least as large as the correlations on the lattice. Thus, the above calculations look at a different limit then we are interested in. The calculation keeping $N \sim N_B$ might be more telling. However, in this case we have to look at finite lattices and examine how the entropy grows, for example, for a division of the lattice in half. The computational approach is similar to the above, but we have to use a sine transform instead of the Fourier transform. The main results are in Figure 8.11. What is found is that the entropy grows quite a bit faster than just the lattice by itself. Instead of logarithmic in the length of the block of sites, it is logarithmic in the number of sites in the block, which includes the branches. Although formally speaking, we can not draw a definitive conclusion regarding simulations with MPS, this

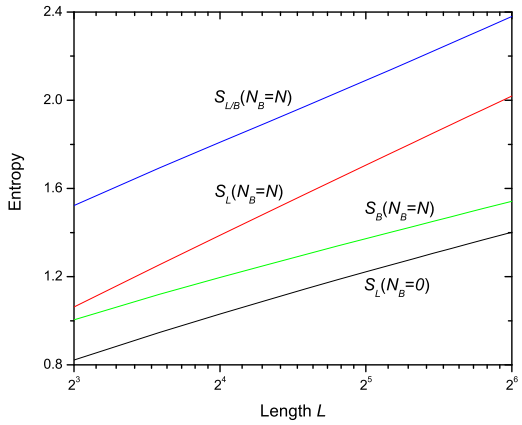


Figure 8.11: Entanglement entropies in the linear lattice which grows holding $N_B = N$. The different entropies stand for the following: S_L is the entropy of a block of L lattice groups, S_B is the entropy of a just a branch, and $S_{L/B}$ is the entropy of the block of L lattice groups without one branch near the boundary of the group.

finding is not comforting.

8.4 Conclusions

There is one important conclusion about simulating multi-channel models that is hinted at by the above results: for a large number of branches using tensor networks that preserve the geometry of the underlying lattice is a poor way to simulate the system. This is because it requires that one performs computational operations that scale as

$$\text{Cost} \propto \chi^{N+1}, \tag{8.38}$$

where χ is just the matrix product dimension and N is the number of channels. In addition, from a rigorous bound on one-dimensional lattices[38], we know that the size of χ is such that

$$\chi \leq A 2^{S^\alpha} \tag{8.39}$$

to get a good approximation using a MPS, where we will not concern ourselves with the prefactor A. But we found that the Renyi entropy scales with $1/N^\alpha$, which, of course, implies that the bound on χ will have a decreasing component due to the decrease in entropy. However, the decrease in the entropy is not fast enough to cancel the exponent in the cost, so that

$$\text{Cost} \propto \chi^{N^{1-\alpha}}, \tag{8.40}$$

e.g., the cost will be exponentially dependent on some power of N. To make these conclusions definite requires more care, as we have only shown that the upper bound will diverge and we have not examined the full behavior as a function of α nor the behavior of the prefactor. We can, though, ask the question, can we perform tensor network simulations in another way for which we can put rigorous bound on the computational cost that is polynomial in the number of branches? For instance, could we forget about using a geometry that matches the underlying lattice and use an alternate structure of the tensor network to do the simulations that is guaranteed to be efficient?

Chapter 9

Conclusions

In this thesis, we have developed several complimentary computational approaches for simulating open quantum systems. These approaches are based on numerical renormalization techniques. We first discussed a superoperator renormalization algorithm for one-dimensional systems that extended the ideas of matrix product states to a matrix product decomposition of the density matrix. This algorithm can be used to construct thermal states and to simulate real-time evolution as given by a generic Markovian master equation. Thus, it can be applied to the study of transport properties, finite temperature effects and dissipation in quantum systems far from equilibrium. The efficiency of the simulation depends on the amount of correlations between subsystems.

Further, we developed methods that can be used for strongly dissipative systems. The first such method was an algorithm to simulate integro-differential equations with long-time memory kernels. These types of integro-differential equations appear in systems such as the spin-boson model with strong dissipation. The second method used tensor network techniques to simulate branched lattices. Each branch can represent a local bath or reservoir. This method can be used to study the interplay between strong local dissipation and the many-body physics of the one-dimensional lattice.

We also set a conceptual foundation for simulating strongly non-Markovian systems. This foundation is a series of transformations that bring the environment's Hamiltonian into a form amenable to generic treatment (e.g., where general and controllable approximations can be made). In addition, we discussed potential applications of the methods to dissipative and driven quantum systems, and we also asked some interesting open questions about efficient ways to simulate branched lattices. We hope to apply the methods developed within this thesis to dissipatively driven phase transitions and non-equilibrium quantum systems relevant to technological applications.

Appendix A: Diagonalizing the XX model

In this Appendix, we show how the XX model is diagonalized with fermionic operators. Starting with the XX Hamiltonian,

$$H_{XX} = -\frac{1}{2}\sum_l \left(\sigma_l^x \sigma_{l+1}^x + \sigma_l^y \sigma_{l+1}^y\right) - g\sum_l \sigma_l^z \,, \tag{9.1}$$

we can perform the Jordan-Wigner transformation:

$$c_l = \left(\prod_{m=1}^{l-1} \sigma_m^z\right) \sigma_l^+ \,, \tag{9.2}$$

and

$$c_l^\dagger = \left(\prod_{m=1}^{l-1} \sigma_m^z\right) \sigma_l^- \,, \tag{9.3}$$

where $\sigma_l^+ = \left(\sigma_l^x + \imath\sigma_l^y\right)/2 = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$ and $\sigma_l^- = \left(\sigma_l^x - \imath\sigma_l^y\right)/2 = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}$. The new operators obey fermionic (anti)commutation relations $\left\{c_l, c_k^\dagger\right\} = \delta_{lk}$ and $\left\{c_l^\dagger, c_k^\dagger\right\} = \{c_l, c_k\} = 0$. The product of two operators gives

$$c_l^\dagger c_l = \sigma_l^- \sigma_l^+ = \left(I - \sigma_l^z\right)/2 \,, \tag{9.4}$$

and

$$c_l^\dagger c_{l+1} + c_{l+1}^\dagger c_l = \frac{1}{2}\left(\sigma_l^x \sigma_{l+1}^x + \sigma_l^y \sigma_{l+1}^y\right) \,. \tag{9.5}$$

This gives the XX Hamiltonian as

$$\begin{aligned} H_{XX} &= -\sum_l \left(c_l^\dagger c_{l+1} + c_{l+1}^\dagger c_l\right) + 2g\sum_i c_l^\dagger c_l \\ &= \sum_{i,j} H_{ij} c_i^\dagger c_j \,. \end{aligned}$$

We can diagonalize the Hermitian matrix H with a unitary matrix U,

$$H_{XX} = \sum_k \left(U^\dagger H U\right) b_k^\dagger b_k = \sum_k \epsilon_k b_k^\dagger b_k \,, \tag{9.6}$$

where U is the modal matrix (it has the eigenvectors as columns), which will give the new set of fermionic operators

$$\begin{pmatrix} b_1 \\ \vdots \\ b_N \end{pmatrix} = U^\dagger \begin{pmatrix} c_1 \\ \vdots \\ c_N \end{pmatrix}, \begin{pmatrix} b_1^\dagger & \cdots & b_N^\dagger \end{pmatrix} = \begin{pmatrix} c_1^\dagger & \cdots & c_N^\dagger \end{pmatrix} U \,. \tag{9.7}$$

Since the Hamiltonian is quadratic, all the information we need is contained in the correlation matrix

$$\tilde{M}_{kl} = \langle b_l^\dagger b_k \rangle \tag{9.8}$$

where, for instance, $\tilde{M}_{kl} = \delta_{lk} f_D\left(\epsilon_k\right)$ for a state in thermal equilibrium, with $f_D\left(\epsilon_k\right) = 1/\left(e^{\beta\epsilon_k}+1\right)$ as the Fermi-Dirac function and β is the inverse temperature. The correlation matrix can be transformed back into the original basis by

$$M_{ji} = \langle c_i^\dagger c_j \rangle = \left(U \tilde{M} U^\dagger\right)_{ji} \,. \tag{9.9}$$

Thus, this gives us a way to calculate the correlation functions of the fermion model, and also other properties such as the energy of a state, $E_M = \mathrm{tr}\,(HM)$.

Getting the correlation functions requires a little more work. The spin operators in terms of the fermion operators are

$$\sigma_l^+ = \left(\prod_{m=1}^{l-1} \left(1 - 2c_m^\dagger c_m\right)\right) c_l \,, \tag{9.10}$$

and

$$\sigma_l^- = \left(\prod_{m=1}^{l-1} \left(1 - 2c_m^\dagger c_m\right)\right) c_l^\dagger \,. \tag{9.11}$$

Or,

$$\sigma_l^x = \left(\prod_{m=1}^{l-1} \left(1 - 2c_m^\dagger c_m\right)\right) \left(c_l^\dagger + c_l\right) \,, \tag{9.12}$$

and

$$\sigma_l^z = 1 - 2c_l^\dagger c_l \,. \tag{9.13}$$

We will ignore σ^y, but we want $\langle \sigma_l^x \sigma_{l+n}^x \rangle$ and $\langle \sigma_l^z \sigma_{l+n}^z \rangle$. Following Ref. 109, we use the identity

$1 - 2c_l^\dagger c_l = \left(c_l^\dagger + c_l\right)\left(c_l^\dagger - c_l\right)$ to get

$$\begin{aligned}\langle\sigma_l^x\sigma_{l+n}^x\rangle &= \langle\left(c_l^\dagger + c_l\right)\left[\prod_{m=l}^{l+n-1}\left(c_m^\dagger + c_m\right)\left(c_m^\dagger - c_m\right)\right]\left(c_{l+n}^\dagger + c_{l+n}\right)\rangle \\ &= \langle\left(c_l^\dagger - c_l\right)\left[\prod_{m=l+1}^{l+n-1}\left(c_m^\dagger + c_m\right)\left(c_m^\dagger - c_m\right)\right]\left(c_{l+n}^\dagger + c_{l+n}\right)\rangle\,.\end{aligned}$$

Defining

$$A_l = c_l^\dagger + c_l\,,\; B_l = c_l^\dagger - c_l\,, \tag{9.14}$$

the correlator can be written as

$$\langle\sigma_l^x\sigma_{l+n}^x\rangle = \langle B_l A_{l+1} B_{l+1}\cdots A_{l+n-1}B_{l+n-1}A_{l+n}\rangle\,, \tag{9.15}$$

which can be evaluated using Wick's theorem. The contractions are

$$\langle A_l A_k\rangle = \delta_{lk} + 2\Im\left(\langle c_l^\dagger c_k^\dagger\rangle + \langle c_l^\dagger c_k\rangle\right)\,, \tag{9.16}$$

$$\langle B_l B_k\rangle = -\delta_{lk} + 2\Im\left(\langle c_l^\dagger c_k^\dagger\rangle - \langle c_l^\dagger c_k\rangle\right)\,, \tag{9.17}$$

and

$$\langle B_l A_k\rangle = -\langle A_k B_l\rangle = -\delta_{lk} + 2\Re\left(\langle c_l^\dagger c_k^\dagger\rangle + \langle c_l^\dagger c_k\rangle\right)\,. \tag{9.18}$$

There are a number of simplifications. In the ground state, the imaginary part of the correlation functions will be zero. Also, in the XX model, the correlation function $\langle c_l^\dagger c_k^\dagger\rangle$ is zero. Thus, only the contraction

$$\langle B_l A_k\rangle = -\delta_{lk} + 2\Re\langle c_l^\dagger c_k\rangle \equiv D_{l-k+1} \tag{9.19}$$

contributes to the correlation function 9.15. For a translationally invariant state, Wick's theorem gives the correlation function in terms of the determinant

$$\langle\sigma_l^x\sigma_{l+n}^x\rangle = \begin{vmatrix} D_0 & D_{-1} & \cdots & D_{-n+1} \\ D_1 & \ddots & & \\ \vdots & & D_0 & D_{-1} \\ D_{n-1} & & D_1 & D_0 \end{vmatrix}\,. \tag{9.20}$$

For a state that is not translationally invariant, which is the case we are interested in due to the branches, one has to modify the above expression by taking different D's as one goes down the

matrix:

$$\langle\sigma_l^x\sigma_{l+n}^x\rangle = \begin{vmatrix} 2\Re\langle c_l^\dagger c_{l+1}\rangle & 2\Re\langle c_l^\dagger c_{l+2}\rangle & \cdots & 2\Re\langle c_l^\dagger c_{l+n}\rangle \\ -1+2\Re\langle c_{l+1}^\dagger c_{l+1}\rangle & \ddots & & \\ \vdots & & 2\Re\langle c_{l+n-2}^\dagger c_{l+n-1}\rangle & 2\Re\langle c_{l+n-2}^\dagger c_{l+n}\rangle \\ 2\Re\langle c_{l+n-1}^\dagger c_{l+1}\rangle & & -1+2\Re\langle c_{l+n-1}^\dagger c_{l+n-1}\rangle & 2\Re\langle c_{l+n-1}^\dagger c_{l+n}\rangle \end{vmatrix} . \tag{9.21}$$

In other words, we can just take the determinant of the matrix

$$D \equiv \mathrm{ldiag}\left\{-1\cdots-1\right\} + 2\Re\mathcal{M}^T\,, \tag{9.22}$$

where ldiag stands for a lower diagonal matrix and $\mathcal{M}^T$ is the transpose of a skewed partial correlation matrix. We can also get the single point correlation function

$$\begin{aligned}\langle\sigma_l^x\rangle &= \langle\left[\prod_{m=l}^{l+n-1}\left(c_m^\dagger + c_m\right)\left(c_m^\dagger - c_m\right)\right]\left(c_{l+n}^\dagger + c_{l+n}\right)\rangle \\ &= 0\,,\end{aligned}$$

where the last line comes from the fact that there is always an unpaired fermionic operator.

The $\langle\sigma_l^z\sigma_{l+n}^z\rangle$ are given by

$$\begin{aligned}\langle\sigma_l^z\sigma_{l+n}^z\rangle &= 1-2\left(\langle c_l^\dagger c_l\rangle + \langle c_{l+n}^\dagger c_{l+n}\rangle\right) \\ &\quad +4\left(\langle c_l^\dagger c_l\rangle\langle c_{l+n}^\dagger c_{l+n}\rangle - \langle c_l^\dagger c_{l+n}^\dagger\rangle\langle c_l c_{l+n}\rangle - \langle c_l^\dagger c_{l+n}\rangle\langle c_{l+n}^\dagger c_l\rangle\right) \\ &= 1-2\left(\langle c_l^\dagger c_l\rangle + \langle c_{l+n}^\dagger c_{l+n}\rangle\right) + 4\left(\langle c_l^\dagger c_l\rangle\langle c_{l+n}^\dagger c_{l+n}\rangle - \langle c_l^\dagger c_{l+n}\rangle\langle c_{l+n}^\dagger c_l\rangle\right)\,,\end{aligned}$$

where in the last line we have dropped the term which is zero for the XX model. The single point correlation is

$$\langle\sigma_l^z\rangle = 1-2\langle c_l^\dagger c_l\rangle\,, \tag{9.23}$$

which is used to set the baseline for the two-point correlation function.

Appendix B: Entanglement entropy of noninteracting fermions

In this Appendix, we show how to obtain the entanglement entropy of noninteracting fermionic models directly from the correlation matrix. Given a Gaussian state with correlation matrix M, which is in the lattice basis, one first considers the reduced correlation matrix, M^R, as the matrix that removes the rows and columns of M of the sites not in a block of interest. The reduced correlation matrix contains all the information of the block of sites, since by Wick's theorem all correlation functions involving those sites can be written in terms of the single particle correlation functions in M^R, e.g., any subset of sites in a noninteracting state is also a noninteracting state. Then, one diagonalizes M^R to obtain its spectrum of eigenvalues $\lambda_k = \langle b_k^\dagger b_k \rangle$, which are the occupation factors of local fermionic levels.

We know that the form of a noninteracting fermion density matrix is

$$\rho = \prod_k \langle b_k^\dagger b_k \rangle b_k^\dagger b_k + \langle b_k b_k^\dagger \rangle b_k b_k^\dagger \,, \tag{9.24}$$

where we can identify the eigenstates k with the diagonal basis of the block of lattice sites. Taking this, choosing an ordering for the states, and performing a Jordon-Wigner transformation, we can write out the density matrix in all its glory, but that turns out to be simple:

$$\rho = \bigotimes_k \rho_k \,, \tag{9.25}$$

with

$$\rho_k = \begin{pmatrix} 1-\lambda_k & 0 \\ 0 & \lambda_k \end{pmatrix} . \tag{9.26}$$

The von Neumann entropy,

$$S \equiv -\mathrm{tr}\left(\rho \log_2 \rho\right) \,, \tag{9.27}$$

is then given by

$$S = -\sum_k \left[\lambda_k \log_2 \lambda_k + (1-\lambda_k) \log_2 (1-\lambda_k)\right] \,, \tag{9.28}$$

because the entropy function is additive for uncorrelated density matrices:

$$\begin{aligned} -\mathrm{tr}\left(\bigotimes_k \rho_k \log_2 \bigotimes_k \rho_k\right) &= -\mathrm{tr}\left(\rho_1 \log_2 \rho_1\right) \mathrm{tr}\left(\bigotimes_{k\neq 1} \rho_k\right) - \mathrm{tr}\left(\bigotimes_k \rho_k \log_2 \bigotimes_{k\neq 1} \rho_k\right) \\ &= -\mathrm{tr}\left(\rho_1 \log_2 \rho_1\right) - \mathrm{tr}\left(\bigotimes_k \rho_k \log_2 \bigotimes_{k\neq 1} \rho_k\right) \\ &= -\sum_k \mathrm{tr}\left(\rho_k \log_2 \rho_k\right) , \end{aligned}$$

where to go to the last line we went through each k and extracted it out of the second term. The Renyi entropy,

$$S^\alpha \equiv \frac{1}{1-\alpha} \log_2 \left(\mathrm{tr}\rho^\alpha\right) , \tag{9.29}$$

also acquires a nice form

$$S^\alpha = \frac{1}{1-\alpha} \sum_k \log_2 \left(\left(1-\lambda_k\right)^\alpha + \lambda_k^\alpha\right) , \tag{9.30}$$

because of

$$\begin{aligned} S^\alpha &= \frac{1}{1-\alpha} \log_2 \left(\mathrm{tr} \bigotimes_k \rho_k^\alpha\right) \\ &= \frac{1}{1-\alpha} \log_2 \left(\prod_k \mathrm{tr}\rho_k^\alpha\right) \\ &= \frac{1}{1-\alpha} \sum_k \log_2 \left(\mathrm{tr}\rho_k^\alpha\right) . \end{aligned}$$

Also, note that

$$S = \lim_{\alpha\to 1} S^\alpha , \tag{9.31}$$

thus the Renyi entropy is more comprehensive, in that as a function of α it gives more information about the character of the state than does the von Neumann entropy.

Bibliography

[1] Caldeira, A. O. & Leggett, A. J. Quantum tunnelling in a dissipative system. *Ann. Phys. - New York* **149**, 374 (1983).

[2] Leggett, A. J. *et al.* Dynamics of the dissipative 2-state system. *Rev. Mod. Phys.* **59**, 1 (1987).

[3] Caldeira, A. O. & Leggett, A. J. Path integral approach to quantum brownian motion. *Physica A: Statistical and Theoretical Physics* **121**, 587 (1983).

[4] Caldeira, A. O. & Leggett, A. J. Influence of damping on quantum interference: An exactly soluble model. *Phys. Rev. A* **31**, 1059 (1985).

[5] Tinkham, M. *Introduction to Superconductivity* (Dover Publications, Inc., Mineola, 2004).

[6] Schön, G. & Zaikin, A. D. Quantum description of dissipation in normal metals and short constrictions. *Phys. Rev. B* **40**, 5231 (1989).

[7] Ambegaokar, V., Eckern, U. & Schön, G. Quantum dynamics of tunneling between superconductors. *Phys. Rev. Lett.* **48**, 1745 (1982).

[8] Eckern, U., Schön, G. & Ambegaokar, V. Quantum dynamics of a superconducting tunnel junction. *Phys. Rev. B* **30**, 6419 (1984).

[9] Feynman, R. P. *Statistical Mechanics* (Westview Press, 1998).

[10] Weiss, U. *Quantum Dissipative Systems* (World Scientific Publishing, 1993).

[11] Wood, D. M. & Stroud, D. Charging effects and the phase-ordering transition in granular superconductors. *Phys. Rev. B* **25**, 1600 (1982).

[12] Simanek, E. Instability of granular superconductivity. *Phys. Rev. B* **22**, 459 (1980).

[13] Schön, G. & Zaikin, A. D. Quantum coherent effects, phase-transitions, and the dissipative dynamics of ultra small tunnel-junctions. *Phys. Rep.* **198**, 237 (1990).

[14] Chakravarty, S., Ingold, G.-L., Kivelson, S. & Zimanyi, G. Quantum statistical mechanics of an array of resistively shunted josephson junctions. *Phys. Rev. B* **37**, 3283 (1988).

[15] Chakravarty, S., Ingold, G.-L., Kivelson, S. & Luther, A. Onset of global phase coherence in josephson-junction arrays: A dissipative phase transition. *Phys. Rev. Lett.* **56**, 2303 (1986).

[16] Fisher, M. P. A. & Zwerger, W. Quantum brownian motion in a periodic potential. *Phys. Rev. B* **32**, 6190 (1985).

[17] Kampf, A. & Schön, G. Quantum effects and the dissipation by quasiparticle tunneling in arrays of josephson junctions. *Phys. Rev. B* **36**, 3651 (1987).

[18] Penttilä, J. S., Parts, U., Hakonen, P. J., Paalanen, M. A. & Sonin, E. B. "superconductor-insulator transition" in a single josephson junction. *Phys. Rev. Lett.* **82**, 1004 (1999).

[19] Miyazaki, H., Takahide, Y., Kanda, A. & Ootuka, Y. Quantum phase transition in one-dimensional arrays of resistively shunted small josephson junctions. *Phys. Rev. Lett.* **89**, 197001 (2002).

[20] Haviland, D. B. *et al.* Quantum phase transition in one-dimensional josephson junction arrays. *Physica C: Superconductivity* **352**, 55 (2001).

[21] Chow, E., Delsing, P. & Haviland, D. B. Length-scale dependence of the superconductor-to-insulator quantum phase transition in one dimension. *Phys. Rev. Lett.* **81**, 204 (1998).

[22] Kuo, W. & Chen, C. D. Scaling analysis of magnetic-field-tuned phase transitions in one-dimensional josephson junction arrays. *Phys. Rev. Lett.* **87**, 186804 (2001).

[23] Suzuki, M. *Quantum Monte Carlo Methods in Condensed Matter Physics* (World Scientific, 1993).

[24] Werner, P. & Troyer, M. Efficient simulation of resistively shunted josephson junctions. *Phys. Rev. Lett.* **95**, 060201 (2005).

[25] White, S. R. Density matrix formulation for quantum renormalization groups. *Phys. Rev. Lett.* **69**, 2863 (1992).

[26] White, S. R. Density-matrix algorithms for quantum renormalization groups. *Phys. Rev. B* **48**, 10345 (1993).

[27] Wilson, K. G. The renormalization group: Critical phenomena and the kondo problem. *Rev. Mod. Phys.* **47**, 773 (1975).

[28] Bulla, R., Costi, T. & Pruschke, T. The numerical renormalization group method for quantum impurity systems. *cond-mat/0701105* (2007).

[29] Pérez-Garzia, D., Verstraete, F., Wolf, M. M. & Cirac, J. I. Matrix product state representations. *quant-ph/0608197* (2006).

[30] Östlund, S. & Rommer, S. Thermodynamic limit of density matrix renormalization. *Phys. Rev. Lett.* **75**, 3537 (1995).

[31] Verstraete, F., Weichselbaum, A., Schollwöck, U., Cirac, J. I. & von Delft, J. Variational matrix product state approach to quantum impurity models. *cond-mat/0504305* (2005).

[32] Schollwöck, U. The density-matrix renormalization group. *Rev. Mod. Phys.* **77**, 259 (2005).

[33] Vidal, G. Efficient classical simulation of slightly entangled quantum computations. *Phys. Rev. Lett.* **91**, 147902 (2003).

[34] Vidal, G. Efficient simulation of one-dimensional quantum many-body systems. *Phys. Rev. Lett.* **93**, 040502 (2004).

[35] Nielsen, M. A. & Chuang, I. L. *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, 2000).

[36] Ekert, A. & Knight, P. L. Entangled quantum-systems and the schmidt decomposition. *American Journal of Physics* **63**, 415 (1995).

[37] Peres, A. *Quantum Theory: Concepts and Methods* (Kluwer Academic Publishers, Dordrecht, 1995).

[38] Verstraete, F. & Cirac, J. I. Matrix product states represent ground states faithfully. *Phys. Rev. B* **73**, 094423 (2006).

[39] Preskill, J. Lecture notes on quantum computation.

[40] Rommer, S. & Östlund, S. *Density matrix renormalization* (Springer, Berlin, 1999).

[41] Jeckelmann, E. Dynamical density-matrix renormalization-group method. *Phys. Rev. B* **66**, 045114 (2002).

[42] Cazalilla, M. A. & Marston, J. B. Time-dependent density-matrix renormalization group: A systematic method for the study of quantum many-body out-of-equilibrium systems. *Phys. Rev. Lett.* **88**, 256403 (2002).

[43] Cazalilla, M. A. & Marston, J. B. Cazalilla and marston reply. *Phys. Rev. Lett.* **91**, 049702 (2003).

[44] Luo, H. G., Xiang, T. & Wang, X. Q. Comment on "time-dependent density-matrix renormalization group: A systematic method for the study of quantum many-body out-of-equilibrium systems". *Phys. Rev. Lett.* **91**, 049701 (2003).

[45] White, S. R. & Feiguin, A. E. Real-time evolution using the density matrix renormalization group. *Phys. Rev. Lett.* **93**, 076401 (2004).

[46] Daley, A. J., Kollath, C., Schollw, ck, U. & Vidal, G. Time-dependent density-matrix renormalization-group using adaptive effective hilbert spaces. *J. Stat. Mech. - Theory E.* **2004**, P04005 (2004).

[47] Clark, S. R. & Jaksch, D. Dynamics of the superfluid to mott-insulator transition in one dimension. *Phys. Rev. A* **70**, 043612 (2004).

[48] Micheli, A., Daley, A. J., Jaksch, D. & Zoller, P. Single atom transistor in a 1d optical lattice. *Phys. Rev. Lett.* **93**, 140408 (2004).

[49] Fannes, M., Nachtergaele, B. & Werner, R. F. Finitely correlated states on quantum spin chains. *Comm. Math. Phys.* **144**, 443 (1992).

[50] Gardiner, C. W., Parkins, A. S. & Zoller, P. Wave-function quantum stochastic differential equations and quantum-jump simulation methods. *Phys. Rev. A* **46**, 4363 (1992).

[51] Gershenfeld, N. A. & Chuang, I. L. Bulk spin-resonance quantum computation. *Science* **275**, 350 (1997).

[52] Cory, D. G., Fahmy, A. F. & Havel, T. F. Ensemble quantum computing by nmrspectroscopy. *PNAS* **94**, 1634 (1997).

[53] Feynman, R. P. & Vernon, F. L. The theory of a general quantum system interacting with a linear dissipative system. *Ann. Phys. - New York* **24**, 118 (1963).

[54] Keldysh, L. V. Diagram technique for nonequilibrium processes. *Sov. Phys. JETP* **20**, 1018 (1965).

[55] Kadanoff, L. P. & Baym, G. *Quantum Statistical Mechanics* (W. A. Benjamin, Inc., New York, 1962).

[56] Breuer, H.-P. & Petruccione, F. *The Theory of Open Quantum Systems* (Oxford University Press, Oxford, 2002).

[57] Cushing, J. M. *Integrodifferential equations and delay models in population* (Springer-Verlag, Berlin, 1977).

[58] Zwolak, M. & Vidal, G. Mixed-state dynamics in one-dimensional quantum lattice systems: A time-dependent superoperator renormalization algorithm. *Phys. Rev. Lett.* **93**, 207205 (2004).

[59] Verstraete, F., Garcia-Ripoll, J. J. & Cirac, J. I. Matrix product density operators: Simulation of finite-temperature and dissipative systems. *Phys. Rev. Lett.* **93**, 207204 (2004).

[60] Zwolak, M. & Refael, G. Generic approach for simulating real-time, non-markovian dynamics (2007).

[61] Shabani, A. & Lidar, D. A. Completely positive post-markovian master equation via a measurement approach. *Phys. Rev. A* **71**, 020101 (2005).

[62] Barnett, S. M. & Stenholm, S. Hazards of reservoir memory. *Phys. Rev. A* **64**, 033808 (2001).

[63] Daffer, S., Wodkiewicz, K., Cresser, J. D. & McIver, J. K. Depolarizing channel as a completely positive map with memory. *Phys. Rev. A* **70**, 010304–4 (2004).

[64] Leathers, A. S. & Micha, D. A. Density matrix for non-markovian dissipative dynamics: A numerical method. *Chem. Phys. Lett.* **415**, 46 (2005).

[65] Brunner, H. & van der Houwen, P. J. *The Numerical Solution of Volterra Equations* (North-Holland, New York, 1986).

[66] Linz, P. *Analytical and Numerical Methods for Volterra Equations* (SIAM, Philadelphia, 1985).

[67] Lubich, C. Runge-kutta theory for volterra integrodifferential equations. *Numerische Mathematik* **40**, 119 (1982).

[68] Aslangul, C., Pottier, N. & Saintjames, D. Spin-boson systems - equivalence between the dilute-blip and the born approximations. *J. Physique* **47**, 1657 (1986).

[69] Mahan, G. *Many-Particle Physics* (Kluwer Academic/Plenum Publishers, New York, 2000), 3rd edn.

[70] Carmichael, H. J. *An Open Systems Approach to Quantum Optics* (Springer-Verlag, Berlin, 1993).

[71] Jauho, A. P., Wingreen, N. S. & Meir, Y. Time-dependent transport in interacting and noninteracting resonant-tunneling systems. *Phys. Rev. B* **50**, 5528 (1994).

[72] Maniscalco, S. & Petruccione, F. Non-markovian dynamics of a qubit. *Phys. Rev. A* **73**, 012111 (2006).

[73] Zwolak, M. Numerical ansatz for solving integro-differential equations with increasingly smooth memory kernels: spin-boson model and beyond. *cond-mat/0611412* (2006).

[74] Press, W. H., Teukolsky, S. A., Vetterling, W. T. & Flannery, B. P. *Numerical Recipes in C: The Art of Scientific Computing* (Cambridge University Press, Cambridge, 1992), 2nd edn.

[75] Garraway, B. M. Nonperturbative decay of an atomic system in a cavity. *Phys. Rev. A* **55**, 2290 (1997).

[76] Garraway, B. M. Decay of an atom coupled strongly to a reservoir. *Phys. Rev. A* **55**, 4636 (1997).

[77] Briegel, H. J. & Englert, B. G. Quantum optical master-equations - the use of damping bases. *Phys. Rev. A* **47**, 3311 (1993).

[78] Danielewicz, P. Quantum-theory of nonequilibrium processes I. *Ann. Phys. - New York* **152**, 239 (1984).

[79] Vidal, G. Classical simulation of infinite-size quantum lattice systems in one spatial dimension. *Phys. Rev. Lett.* **98**, 070201 (2007).

[80] Taylor, P. L. & Heinonen, O. *A quantum approach to condensed matter physics* (Cambridge University Press, Cambridge, 2002).

[81] Hohenberg, P. & Kohn, W. Inhomogeneous electron gas. *Physical Review* **136**, B864 (1964).

[82] Kohn, W. & Sham, L. J. Self-consistent equations including exchange and correlation effects. *Physical Review* **140**, A1133 (1965).

[83] Agrait, N., Yeyati, A. L. & van Ruitenbeek, J. M. Quantum properties of atomic-sized conductors. *Phys. Rep.* **377**, 81 (2003).

[84] Di Ventra, M., Pantelides, S. T. & Lang, N. D. First-principles calculation of transport properties of a molecular device. *Phys. Rev. Lett.* **84**, 979 (2000).

[85] Reed, M. A., Zhou, C., Muller, C. J., Burgin, T. P. & Tour, J. M. Conductance of a molecular junction. *Science* **278**, 252 (1997).

[86] Cui, X. D. *et al.* Reproducible measurement of single-molecule conductivity. *Science* **294**, 571 (2001).

[87] Reichert, J. *et al.* Driving current through single organic molecules. *Phys. Rev. Lett.* **88**, 176804 (2002).

[88] Xu, B. & Tao, N. J. Measurement of single-molecule resistance by repeated formation of molecular junctions. *Science* **301**, 1221 (2003).

[89] Xiao, X. Y., Xu, B. Q. & Tao, N. J. Measurement of single molecule conductance: Benzenedithiol and benzenedimethanethiol. *Nano Lett.* **4**, 267 (2004).

[90] Tomfohr, J., Ramachandran, G., Sankey, O. F. & Lindsay, S. M. Making contacts to single molecules: Are we there yet? In Cuniberti, G., Fagas, G. & Richter, K. (eds.) *Introducing Molecular Electronics* (Springer, New York, 2004).

[91] Di Ventra, M., Chen, Y. C. & Todorov, T. N. Are current-induced forces conservative? *Phys. Rev. Lett.* **92**, 176803 (2004).

[92] Montgomery, M. J., Todorov, T. N. & Sutton, A. P. Power dissipation in nanoscale conductors. *J. Phys.: Condens. Matter* **14**, 5377 (2002).

[93] Chen, Y. C., Zwolak, M. & DiVentra, M. Local heating in nanoscale conductors. *Nano Lett.* **3**, 1691 (2003).

[94] Delaney, P. & Greer, J. C. Correlated electron transport in molecular electronics. *Phys. Rev. Lett.* **93**, 036805 (2004).

[95] Gonze, X. & Scheffler, M. Exchange and correlation kernels at the resonance frequency: Implications for excitation energies in density-functional theory. *Phys. Rev. Lett.* **82**, 4416 (1999).

[96] Evers, F., Weigend, F. & Koentopp, M. Conductance of molecular wires and transport calculations based on density-functional theory. *Physical Review B (Condensed Matter and Materials Physics)* **69**, 235411 (2004).

[97] Runge, E. & Gross, E. K. U. Density-functional theory for time-dependent systems. *Phys. Rev. Lett.* **52**, 997 (1984).

[98] Burke, K., Car, R. & Gebauer, R. Density functional theory of the electrical conductivity of molecular devices. *Phys. Rev. Lett.* **94**, 146803 (2005).

[99] Vignale, G. & Kohn, W. Current-dependent exchange-correlation potential for dynamical linear response theory. *Phys. Rev. Lett.* **77**, 2037 (1996).

[100] Vignale, G., Ullrich, C. A. & Conti, S. Time-dependent density functional theory beyond the adiabatic local density approximation. *Phys. Rev. Lett.* **79**, 4878 (1997).

[101] Ullrich, C. A. & Vignale, G. Time-dependent current-density-functional theory for the linear response of weakly disordered systems. *Phys. Rev. B* **65**, 245102 (2002).

[102] Di Ventra, M. & Lang, N. D. Transport in nanoscale conductors from first principles. *Phys. Rev. B* **65**, 045402 (2001).

[103] Lagerqvist, J., Chen, Y.-C. & Ventra, M. D. Shot noise in parallel wires. *Nanotechnology* **15**, S459 (2004).

[104] Sai, N., Zwolak, M., Vignale, G. & Di Ventra, M. Dynamical corrections to the dft-lda electron conductance in nanoscale systems. *Phys. Rev. Lett.* **94**, 186810 (2005).

[105] Bushong, N., Sai, N. & DiVentra, M. Approach to steady-state transport in nanoscale conductors. *Nano Lett.* **5**, 2569 (2005).

[106] Gan, J. On the multichannel kondo model. *J. Phys.: Condens. Matter* **6**, 4547 (1994).

[107] Shaw, M., Zhang, X.-w. & Jin, D. Scaling solution of the overscreened multichannel kondo model. *Phys. Rev. B* **57**, 8381 (1998).

[108] Vidal, G., Latorre, J. I., Rico, E. & Kitaev, A. Entanglement in quantum critical phenomena. *Phys. Rev. Lett.* **90**, 227902 (2003).

[109] Sachdev, S. *Quantum Phase Transitions* (Cambridge University Press, New York, NY, 1999).