ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL OF SCIENCE ENGINEERING AND TECHNOLOGY

QUANTUM DYNAMICS IN COUPLED CAVITY ARRAYS WITH VARIATIONAL MATRIX PRODUCT STATES

M.Sc. THESIS

Nur ASLAN

Department of Physics Engineering

Physics Engineering Programme

JANUARY 2015

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL OF SCIENCE ENGINEERING AND TECHNOLOGY

QUANTUM DYNAMICS IN COUPLED CAVITY ARRAYS WITH VARIATIONAL MATRIX PRODUCT STATES

M.Sc. THESIS

Nur ASLAN
(509121132)

Department of Physics Engineering

Physics Engineering Programme

Thesis Advisor: Assoc. Prof. Dr. A. Levent SUBAŞI

JANUARY 2015

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

VARYASYONEL MATRİS ÇARPIM DURUMLARI METODUNUN BAĞLI KOVUK ZİNCİRİNDE KUANTUM DİNAMİĞİNE UYGULANMASI

YÜKSEK LİSANS TEZİ

Nur ASLAN
(509121132)

Fizik Mühendisliği Anabilim Dalı

Fizik Mühendisliği Programı

Tez Danışmanı: Doç. Dr. A. Levent SUBAŞI

OCAK 2015

Nur ASLAN, a M.Sc. student of ITU Graduate School of Science Engineering and Technology, student ID 509121132, successfully defended the thesis entitled **"QUANTUM DYNAMICS IN COUPLED CAVITY ARRAYS WITH VARIATIONAL MATRIX PRODUCT STATES"**, which he/she prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor :	**Assoc. Prof. Dr. A. Levent Subaşı** Istanbul Technical University	.....
Co-advisor :	**Assist. Prof. Dr. Ş. Ekin Kocabaş** Koç University	.....
Jury Members :	**Assoc. Prof. Dr. Ali Yıldız** Istanbul Technical University	.....
	Prof. Dr. Özgür Esat Müstecaplıoğlu Koç University	.....
	Chief Researcher Dr. Serkan Ateş TÜBİTAK BİLGEM UEKAE	.....

Date of Submission : **13 December 2014**
Date of Defense : **22 January 2015**

FOREWORD

We would like to acknowledge the ITU-BAP program support for a portable computer and a monitor under project "Güçlü Etkileşen Foton Sistemlerinin Bir Boyutlu Örgü Üzerinde Zaman Evrimi". We would also like to acknowledge the use of the computer cluster in ITU Statistical Physics and Complex Systems Research Group's computational rheology laboratory.

January 2015

Nur ASLAN
Physics Engineer

TABLE OF CONTENTS

Page

ABBREVIATIONS

TN	**:** Tensor Networks
MPS	**:** Matrix Product State
MPO	**:** Matrix Product Operator
VMPS	**:** Variational Matrix Product States
RWA	**:** Rotating Wave Approximation
CRW	**:** Counter Rotating Wave
CCA	**:** Coupled Cavity Arrays
CQED	**:** Cavity Quantum Electrodynamics
TLS	**:** Two Level System

LIST OF TABLES

Page

LIST OF FIGURES

Page

QUANTUM DYNAMICS IN COUPLED CAVITY ARRAYS WITH VARIATIONAL MATRIX PRODUCT STATES

SUMMARY

Matrix product state (MPS) representation is an alternative way to describe a quantum many body system on a one dimensional lattice by means of an interconnected local tensor array. The exponential growth of the Hilbert space of a quantum many body system is a major limitation in numerical calculations. In variational matrix product state (VMPS) method, the Hilbert space of a system is represented by a reduced number of basis vectors, therefore allowing the simulation of larger systems. This truncation of the Hilbert space in terms of product of local states is related to the entanglement in the system.

We have applied VMPS method to a model of coupled quantum cavity arrays. In our model in one of the cavities, there is a two level system (TLS) interacting with the cavity mode. We have studied the system in both the so-called rotating wave approximation (RWA) and beyond including counter rotating wave (CRW) terms in Rabi model.

We have calculated the ground state particle distributions of the system with different coupling strengths. The dependence of ground state convergence on local index dimensions is studied for a small chain. The ground state calculation provides the initial state that we used in scattering calculations. Because of limited resources the ground state on the large chain is approximately obtained with small index dimensions.

Using the Suzuki-Trotter decomposition, we have investigated time dependent dynamics of emission from the TLS. In RWA, we have observed three regimes as a function of coupling strength. In weak coupling regime, there is a purely exponential decay to the ground state. As the interacting strength is increased, exponential decay with secondary Rabi oscillations emerges. In strongly interacting regime, fractional decay with significant oscillations occur which are related to photon-atom bound states. Beyond the RWA, the population trapping and field localization effects are much stronger.

Finally, scattering of a Gaussian shaped pulse from a TLS is simulated within RWA and including CRW terms. In the case of single photon scattering, photon-atom bound state does not come into play. However, two photon scattering leads the residual occupation of the excited level after the scattering (population trapping). This is the effect of photon-atom bound states. Additionally, we have examined scattering from a TLS initially starting from the ground state instead of the vacuum state for large couplings.

Using MPS it is possible to study more general systems. The computational tools prepared for this thesis can easily be extended to similar problems such as coupled quantum cavity arrays interacting with multiple TLSs or three level systems. We want to study these systems by variational matrix product state method in the future.

VARYASYONEL MATRİS ÇARPIM DURUMLARI METODUNUN BAĞLI KOVUK ZİNCİRİNDE KUANTUM DİNAMİĞİNE UYGULANMASI

ÖZET

Matris çarpım durumları (İngilizce baş harflerinden kısaltılarak MPS) temsili, kuantum mekaniksel bir sistemin durumunun birbirine bağlı yerel tensörler cinsiden bir ifadesidir. Her tensörün bir tane fiziksel ve iki tane kendisini komşularına bağlayan sanal bağ indeksi olmak üzere üç indeksi vardır. Matris çarpım durumları ile kuantum mekaniğindeki iki durum vektörünün iç çarpımı, bir işlemcinin beklenen değeri gibi hesaplar yapılabilir. Bu hesaplar, MPS bağlamında birer tensör çarpım işlemine dönüşür.

Herhangi bir çok parçacık kuantum durumu, matris çarpım durumları ile ifade edilebilir. Bu geçiş tekil değer ayrıştırma (singular value decomposition, SVD) ile yapılır. Herhangi bir A matrisi için

$$A_{nm} = U_{nn'}\Lambda_{n'm'}V^{\dagger}_{m'm}$$

olur. Burada Λ, A matrisinin tekil değerlerini büyükten küçüğe sıralı şekilde ($\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_r$) taşıyan köşegen bir matristir.

Güçlü etkileşimli çok parçacıklı kuantum sistemlerinin çözümü genellikle sayısal yöntemler gerektirir. Sayısal yöntemlerle bu sistemlerin büyüklükleri Hilbert uzayındaki üstel büyüme nedeniyle kullanılan bilgisayarın hafızası miktarınca sınırlıdır. MPS ile bir sistemin enerji özdurumlarını ve özvektörlerini varyasyonel bir hesapla bulmak mümkündür. Bu metod, varyasyonel matris çarpım durumları metodu olarak geçer. Varyasyonel matris çarpım durumları metodunda sistemin Hilbert uzayı daha az miktardaki sayıda durum vektörüyle, yani bir yaklaşıklıkla ifade edilir. Bu yaklaşıklığın bir ölçütü olarak Von Neumann dolanıklık entropisi şöyle verilir:

$$S = -Tr\left(\rho \log \rho\right)$$

Burada $\rho = |\psi\rangle\langle\psi|$ sistemin yoğunluk matrisidir. Tensörlerin bağ boyutu (D) ve yoğunluk matrisi arasındaki ilişki ise

$$\sum_{i=1}^{D} \rho_{ii} \approx 1$$

olur.

Böylelikle MPS'teki tensörlerin bağ boyutu D, yukarıdaki ilişkileri sağlayacak şekilde küçük seçilebilir. Buradaki D boyutu aynı zamanda tekil değerleri içeren Λ matrisinin sıfırdan farklı köşegen elemanlarından $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_D$ 'ye kadar olan tekil değerlerini tutup, kalan tekil değerleri ($\lambda_{D+1} \geq \cdots \geq \lambda_r$) ihmal etmeye karşılık gelir. MPS metoduyla ilgili tek kısıtlama fiziksel sistemin dolanıklık entropisi ile ilgilidir. Bunun dışında tensor ağları (tensor networks, TN) metodu farklı sınır koşulları, sonlu ya da sonsuz büyüklüğe sahip ya da bir boyut ve daha yüksek boyutlardaki sistemlere uygulanabilir.

Bu yöntemde tıpkı durum vektörlerinde olduğu gibi kuantum operatörlerinin de matris çarpımları şeklinde ifade edilmesi gerekir. Operatörler için tanımlanan bu yeni forma matris çarpım operatörleri (MPO) denir. Bir MPO, MPS üzerine uygulandığında sonuç başka bir MPS verir. Bu yeni MPS'nin bağ boyutu MPO'dan gelen indekslerin katkısıyla daha büyük olur.

Varyasyonel matris çarpım metodu bir sistemin enerji özdeğer ve özvektörlerini bulmak için bir Lagrange çarpanı λ ile aşağıdaki gibi kullanılır:

$$\min_{\langle\Psi|}[\langle\Psi|H|\Psi\rangle - \lambda\,\langle\Psi|\Psi\rangle]$$

Burada, MPS ifadesindeki her bir tensör elemanı için minimizasyon yapılarak sistem için optimal MPS ifadesi elde edilir. Bunun için önceden belirlenen bir ε yaklaşıklığı sağlanana kadar zincir üzerinde sağa ve sola doğru ilerlenir. Bu süpürme işlemi (sweeping) olarak bilinir. Süpürme işlemi hesaptaki yaklaşıklığı artıran bir işlemdir ve gerektiği kadar tekrarlanır.

Işık-madde etkileşimleri, yoğun madde fiziğinde oldukça çalışılan bir konudur. Bu etkileşimlerle ilgili bir takım modeller geliştirilmiştir. Bu modellerden en basiti tek modlu kuantize bir ışık ile iki seviyeli bir sistemin etkileşimini anlatan kuantum Rabi modelidir. Kuantize ışığın modu, kovuk modu olarak bilinir. Bu modele göre iki seviyeli sistem, kuantize ışık moduyla etkileşimi sonucunda bir miktar uyarılır ve sonra soğurduğu ışığı salar. Bu döngü Rabi frekansı denen, kovuk modu ile atomun etkileşim gücüne bağlı olan bir sıklıkta olur. Bu salınımlara Rabi salınımları denir. Sistemin uyarılma sayısı kovuk modunda bulunan foton sayısı ve iki seviyeli sistemin uyarılma sayısı toplamıdır. Kuantum Rabi modelinde aynı anda kavitede bir photon yaratan (yok eden) ve iki seviyeli sistemi uyaran (taban durumuna geçiren) süreçlerin varlığı nedeniyle toplam uyarılma sayısı korunmaz. Bu süreçleri ihmal edince toplam uyarılma sayısı korunan bir büyüklük haline gelir. Bu yaklaşıma ingilizce baş harflerinin kısaltmasıyla RWA denir. Bu model Jaynes Cummings modelidir ve analitik çözümü mevcuttur. Bu nedenle Jaynes Cummings modeli kuantum mekaniksel ışık-madde etkileşimi problemlerinde temel bir kilometre taşıdır.

Teknolojik yenilikler ve yeni deneysel tekniklerle birlikte ışık-madde etkileşimi alanındaki çalışmalarda daha karmaşık sistemlerin çözümüyle ilgili uğraşılar baş göstermiştir. Bu tür sistemlerdeki çalışmalarda etkileşimlerin kuvvetini ayarlayabilmek önemlidir, çünkü etkileşimlerin gücü sistemde farklı fiziksel olgulara yol açabilir. Bu tezde, iki seviyeli bir sistemin kuantum kovuklardan oluşan bir zincirdeki kovuklardan biriyle etkileştiği bir model üzerinde çalışmalar yapılmıştır. Bu çalışmalar, yukarıda bahsi geçen varyasyonel matris çarpım durumları metodu kullanılarak yapılmıştır. Böylelikle matris çarpım durumlarıyla hesap yapma ve bu hesaplardan yararlanarak bazı fiziksel sonuçlar elde etme tez çalışması boyunca bir amaç olarak görülmüştür.

İki seviyeli sistem-N kovuk etkileşimi modelinin varyasyonel matris çarpım durumları metoduna uyarlayarak sistemin enerji özdurum ve özdeğerleri hem RWA hem de RWA ötesi durumlarında elde edilmiştir. Sistemin enerji özdeğer ve özvektörleri düşükten yükseğe farklı etkileşim güçlerinde hesaplanarak belirli bir etkileşim gücü değerinde g kuantum faz geçisi olduğu tespit edilmiştir.

Matris çarpım metoduyla zamanda evrim hesapları da yapılabilmektedir. Bu tez çalışmasında, zamanda ilerleme operatörü için MPO formunda Suzuki-Trotter açılımı kullanarak zamana bağlı hesaplar yapılmıştır.

Bu hesaplardan biri, başlangıç anında uyarılmış durumda olan iki seviyeli sistemin yaptığı emisyonun zamana bağlı olarak hem RWA hem RWA ötesi durumlarda incelenmesidir. RWA durumunda, iki seviyeli sistemin emisyonu ile ilgili üç farklı rejim görülmüştür. İlki iki seviyeli sistemin kendiliğinden emisyonuna karşılık gelen üstel bir davranıştır. Bu durum zayıf etkileşim değerlerinde görülür. Etkileşim artırıldığında, iki seviyeli sistem üstel olarak taban durumuna geçmeye devam ederken aynı zamanda salınımlar yapar. Bu, iki seviyeli sistem etrafında yerelleşmeye başlayan bir foton dağılımının habercisidir. Yüksek etkileşimde ise iki seviyeli sistem emisyon yaparken aynı zamanda düzgün olmayan büyük genlikli salınımlar yapar. Buna foton-atom bağlı durumları yol açar.

İkiden fazla seviyeli bir sistemin emisyonunu incelemek de aynı metod ile mümkün olacaktır. Zincir üzerinde birden fazla sayıda iki ya da daha fazla seviyeli sistem olması durumunda emisyon dinamiği incelenebilir.

Zamana bağlı diğer bir hesap olarak ise bir foton barındıran Gaussyen bir dalga paketinin iki seviyeli sistemden saçılması simüle edilmiştir. Bu simülasyon yüksek etkileşimde yapılmıştır. Bir foton saçılımında ilginç bir fiziksel olgu ortaya çıkmazken aynı saçılma simülasyonu iki foton taşıyan bir Gaussyen dalga paketiyle yapıldığında, çarpışmadan sonra iki seviyeli sistemin tamamen taban durumuna geçmediği görülmüştür. İki foton saçılma simülasyonu farklı bir şekilde tekrarlanmıştır. Bir foton taşıyan iki Gaussyen dalga paketi zıt yönlerden gönderilmiş ve iki seviyeli sistemin uyarılma sayısının sıfır olmadığı burada da gözlenmiştir. Foton-atom bağlı durumlarının etkisinin iki ve daha fazla sayıda foton saçılmalarında ortaya çıktığı anlaşılmıştır. Burada foton-TLS bağlı durumunun etkisini incelemek için saçılma simülasyonları farklı etkileşim güçlerinde hem RWA'da hem RWA ötesinde yapılmıştır.

Saçılma simülasyonlarında foton-atom bağlı durumlarını incelemek için gelen dalga paketi farklı şekillerde hazırlanabilir. Birden fazla dalga paketi zıt yönlerden farklı momentumla ya da başlangıç pozisyonlarından başlatılarak gönderilebilir.

Emisyon ve saçılma simülasyonlarında kaviteler için kalite faktörü eklenebilir. Kalite faktörü büyüklüğünce kavitelerde kayıplar olacaktır. Bu şekilde fiziksel sistem daha gerçekçi bir hale getirilmiş olup, kalite faktörünün foton-atom bağlı durumlarına etkisi incelenebilir.

Varyasyonel matris çarpım metodu ile yapılan hesaplarda iki farklı hata kaynağı mevcuttur. Bunlardan biri MPS'in bağ boyutundan D gelen budama hatasıdır. Hesabın doğruluğu büyük ölçüde bu hatanın büyüklüğüne bağlıdır. Bu hatanın mertebesi bilgisayar temsili hatasına kadar düşürülebilir. Diğer bir hata kaynağı ise zamanda ilerleme operatörünün Suzuki-Trotter açılımından gelen hatadır. Bu hata zaman adımının büyüklüğü ve kullanılan açılımın mertebesi ile ilgilidir. Yüksek mertebedeki açılımları kullanarak ve zaman adımını yeterince küçük alarak bu hata asgari seviyeye indirilebilir. Bu hesaplarla ilgili olan en önemli nokta bağ boyutunun MPS'i yeterince iyi ifade edebilecek şekilde seçilmesidir. Eğer bağ boyutu MPS'i yeterince iyi ifade edemezse, hesaplardaki baskın hata kaynağı bu olacaktır.

Varyasyonel matris çarpım durumları metoduyla daha geliştirilmiş fiziksel problemleri çözmek gelecekteki amacımızdır. Bu sistemler arasında N kovuk modu ile birden fazla iki seviyeli sistem ya da üç seviyeli sistem etkileşimlerini içeren fiziksel modeller olacaktır. Bu modellerle ilgili faz geçişleri, atomun emisyon dinamiği ve saçılmalar ilgi odağımız olacaktır.

1. INTRODUCTION

Light-matter interactions in the quantum mechanical approach have an important place in quantum many body physics. When the interactions are weak, the problem can be tackled with perturbative methods. Whereas in the case of strong interactions new physics emerges.

Technological developments have lead to various experimental techniques in field of quantum optics. This has allowed to tune the interactions between light and matter. Therefore, it became possible to investigate the new physics in existence of the strong interactions. One of these control mechanisms is cavity quantum electrodynamics (CQED) system consisting of a two level system (TLS) coupled to a single mode of the electromagnetic field [1].

The simplest CQED model is Quantum Rabi model which describes the TLS interacting with a single mode electromagnetic resonator. Jaynes Cummings model which is RWA approximation of the Quantum Rabi model has an analytical solution. For this reason it is a milestone for theoretical and experimental QCED problems [2]. Coupled quantum cavity arrays are formed to create strongly correlated many body system in the quantum light-matter basis. There are some motivations to investigate these systems. These systems can be used as quantum simulators [3]. With optical lattices and Josephson junctions it is hard to examine individual site physics experimentally due to the small separation between nearest neighbor sites, whereas in quantum cavity arrays (QCA), this separation is on the order of micrometers. It is therefore possible to optically access these sites [4].

Besides the development of these new experimental techniques, theoretical studies on these systems have been done. Due to the strong interactions, quantum many body problems are tackled with numerical methods. The main problem of studying a many particle quantum system is the rapidly growing Hilbert space dimension. Some numerical methods including series expansion [5], quantum monte carlo [6] and the numerical renormalization group (NRG) [7] have been developed over the past few decades to cope with the large number of degrees of freedom of such systems. The density matrix

renormalization group (DMRG) method has been introduced by S. White based on NRG in order to tackle with strongly correlated 1-D lattice models where NRG fails to give accurate results [8]. However, DMRG method does not give accurate results in two dimensional systems and periodic boundary conditions [9].

Recently, tensor networks have been popularly used to investigate strongly correlated many body systems [10]. Quantum state of the system is represented by a network of interconnected local tensors in this context. Physical properties of the system are studied by variational calculations using tensor networks. These are tensor network methods. Unlike DMRG, tensor networks methods are applicable to two dimensional systems [11] and periodic boundary conditions [12]. Therefore it is highly promising to study new physics a variety range of strongly correlated systems with these methods.

One dimensional tensor networks are matrix product states. In this thesis, our aim is to give basic concepts about matrix product states and how to do calculations by the matrix product states. These discussions are given in Chapter 2. In Chapter 3, we give motivation of the system of coupled quantum cavity arrays interacting with a two level system. Ground state and time-dependent calculation results of the physical system are presented in Chapter 4. A general outlook for the calculations and results are given in Chapter 5.

2. MATRIX PRODUCT STATES (MPS)

2.1 Introduction to MPS

Strongly correlated quantum many body systems can be handled by some numerical methods such as density matrix renormalization group (DMRG) [13] and tensor networks (TN) [10]. The common feature of these methods is representing the Hilbert space of the system by a truncated basis by selecting the most relevant state vectors. Tensor networks have come into prominence in recent years. It is a representation of the quantum many body state by local interconnected tensors. One dimensional tensor network is a matrix product state (MPS).

There is an area law between the number of basis states (D) that will be kept to represent the state faithfully and the entanglement property in the ground state of the system [14]. According to this area law, the number of basis vectors are bounded by the entanglement. A measure of the entanglement of the state is the so-called Von Neumann entropy

$$S = -\mathrm{Tr}(\rho \log \rho) \tag{2.1}$$

The rank of the density matrix of a system gives the information about entanglement. In the case that the rank of the density matrix is $D = 1$, there is only one nonzero eigenvalue which is equal to 1. This makes Von Neumann entropy in Eq. (2.1) $S = 0$ meaning that the state which is represented by the density matrix ρ is a state with no entanglement. Some entangled states with $D = 2$ are AKLT state, W state, GHZ state etc [10]. An example for the $N = 3$, GHZ state is given in Appendix A.

Consider a general many body state with N particles

$$|\Psi\rangle = \sum_{i_1,i_2,\cdots,i_N} C_{i_1 i_2 \cdots i_N} |i_1\rangle \otimes |i_2\rangle \otimes \cdots \otimes |i_N\rangle \tag{2.2}$$

Let d denote the Hilbert space dimension of the single particle. $|i_1\rangle, |i_2\rangle, \cdots, |i_N\rangle$ are the orthonormal basis vectors for the Hilbert space $\mathscr{H}_N$ and $C_{i_1 i_2 \cdots i_N} \in \mathbb{C}$ whose size grows exponentially with the number of particles, $dim(\mathscr{H}_N) = d^N$. This causes a restriction on the size of the system that can be stored on a computer. It is possible to split $C_{i_1 i_2 \cdots i_N}$

into smaller tensors through singular value decomposition (SVD). Each tensor represents a local site state in this way.

Eq. (2.2) can be written alternatively as

$$|\Psi\rangle = \sum_{\{i\}} C_{i_1 i_2 \cdots i_N} |i_1 i_2 \cdots i_N\rangle \tag{2.3}$$

which is the convention that we use in this thesis.

2.2 Singular Value Decomposition (SVD) and Schmidt Basis

Any $n \times m$ matrix A can be written in terms of its singular values in the form $A = U\Lambda V^\dagger$ where U and V are $n \times n$ and $m \times m$ unitary matrices and Λ $n \times m$ is a diagonal matrix whose entries are the non-negative singular values $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_r$ of the matrix A.

Schmidt decomposition is an expression of a vector in terms of tensor product of two vectors. Eq. (2.3) can be written in the so-called Schmidt basis for two subparts. For this, we split the chain of sites into two parts in a way that there are n sites on the left side and $N-n$ sites on the right side. Let us rewrite the Eq. (2.3) for this decomposition:

$$|\Psi\rangle = \sum_{\{i\}} C_{i_1 \cdots i_n, i_{n+1} \cdots i_N} |i_1 \cdots i_n\rangle \otimes |i_{n+1} \cdots i_N\rangle \tag{2.4}$$

By the SVD decomposition, $C_{i_1 \cdots i_n, i_{n+1} \cdots i_N} = U_{i_1 \cdots i_n, \alpha_n} \Lambda_{\alpha_n, \alpha_n} V_{\alpha_n, i_{n+1} \cdots i_N}$ so that

$$|\Psi\rangle = \sum_{\alpha_n=1}^{D} \Lambda_{\alpha_n, \alpha_n} \left(\sum_{i_1 \cdots i_n} U_{i_n \alpha_n} |i_1 i_2 \cdots i_n\rangle \right) \otimes \left(\sum_{i_{n+1} \cdots i_N} V^\dagger_{\alpha_n, i_{n+1} \cdots i_N} |i_{n+1} \cdots i_N\rangle \right) \tag{2.5}$$

with the definitions of $|\alpha_n\rangle^L = U_{i_n, \alpha_n} |i_1 i_2 \cdots i_n\rangle$ and $|\alpha_n\rangle^R = V^\dagger_{\alpha_n, i_{n+1} \cdots i_N} |i_{n+1} \cdots i_N\rangle$ we obtain

$$|\Psi\rangle = \sum_{\alpha_n=1}^{D} \Lambda_{\alpha_n, \alpha_n} |\alpha_n\rangle^L \otimes |\alpha_n\rangle^R \tag{2.6}$$

in the Schmidt basis. It is possible to obtain the spectrum of the singular value matrix Λ by calculating the reduced density matrix of the left (or right) side:

$$\rho_L = \mathrm{Tr}_R(|\Psi\rangle \langle\Psi|) \tag{2.7}$$

$$= \sum_{\alpha_n} \Lambda^2_{\alpha_n, \alpha_n} |\alpha_n\rangle^{L\,L} \langle\alpha_n| \tag{2.8}$$

The trace of the reduced density matrix is $\mathrm{Tr}(\rho_L) = 1$ for the complete spectrum of Λ^2. It is possible to cut the spectrum of Λ^2 leading an approximate representation for the state vector Eq. (2.6). This truncation corresponds to set the bond dimension $dim(\alpha_n)$ to a minimal number so that the trace of the reduced density matrix $\mathrm{Tr}(\rho_L) \approx 1$ (or $\mathrm{Tr}(\rho_R) \approx 1$).

2.3 Matrix Product State Representation

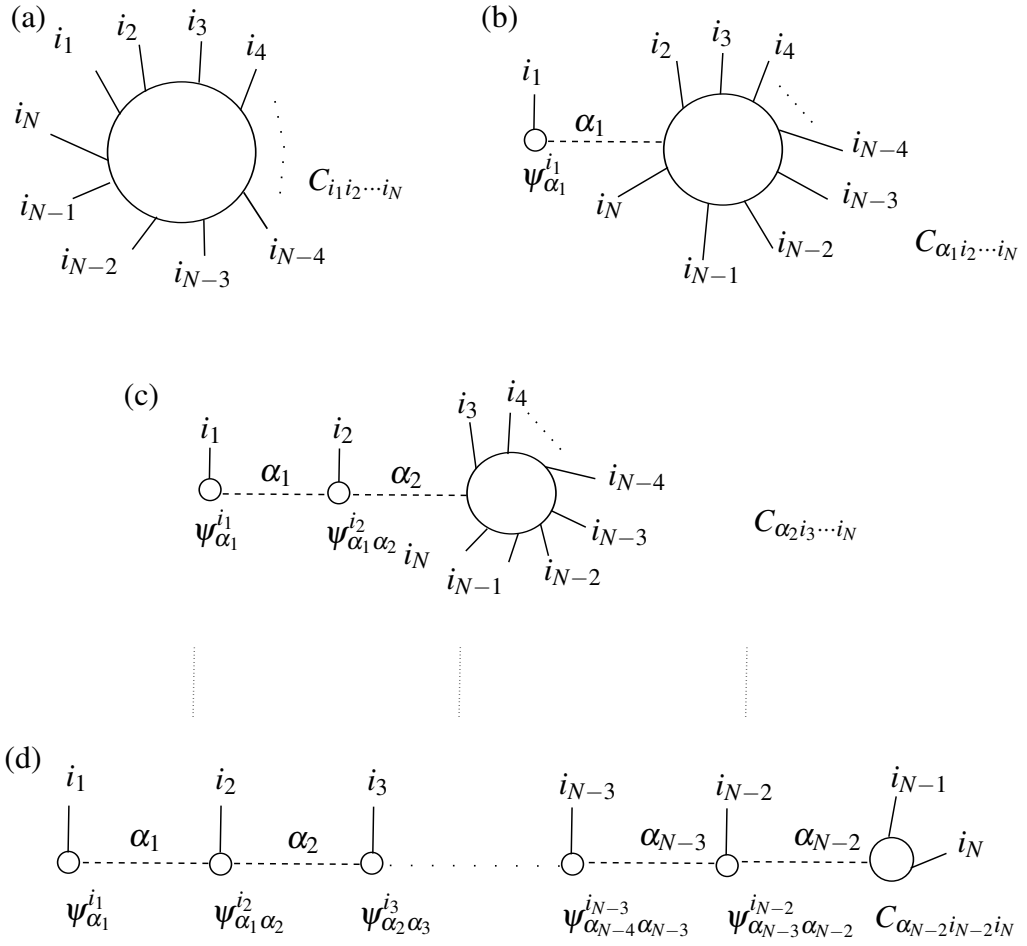

Figure 2.1: Graphical representation of breaking the tensor $C_{i_1 i_2 \cdots i_N}$ into smaller locally interconnected pieces of tensors. Sizes of tensors are decreased with the number of indices.

The graphical representation for the derivation of MPS is shown in Fig. 2.1. In Fig. 2.1.(a), $C_{i_1 i_2 \cdots i_N}$ tensor in Eq. (2.3) is shown. In order to obtain MPS, we should apply SVD for each separation ($n = 1, 2, \cdots, N-1$) of the chain. Let us first separate the site $n = 1$ from the remaining $N-1$ in Eq. (2.3) by introducing the matrix $C_{i_1,(i_2 i_3 \cdots i_N)} = U_{i_1,\alpha_1} \Lambda_{\alpha_1,\alpha_1} V^{\dagger}_{\alpha_1,(i_2 i_3 \cdots i_N)}$ where $(i_2 i_3 \cdots i_N)$ is a combined index.

$$|\Psi\rangle = \sum_{\alpha_1=1}^{D} \sum_{i_1,i_2,\cdots,i_N=1}^{d} U_{i_1,\alpha_1} \Lambda_{\alpha_1,\alpha_1} V^{\dagger}_{\alpha_1,(i_2 i_3 \cdots i_N)} |i_1 i_2 \cdots i_N\rangle \qquad \textbf{(2.9)}$$

or equivalently

$$|\Psi\rangle = \sum_{\alpha_1=1}^{D} \sum_{i_1,i_2,\cdots,i_N=1}^{d} \psi_{\alpha_1}^{i_1} \quad C_{\alpha_1,i_2i_3\cdots i_N} |i_1 i_2 i_3 \cdots i_N\rangle \tag{2.10}$$

where $C_{\alpha_1,i_2i_3\cdots i_N} = \Lambda_{\alpha_1} V^{\dagger}_{\alpha_1,(i_2i_3\cdots i_N)}$. Here we changed the notation $U_{i_1,\alpha_1} = \psi_{\alpha_1}^{i_1}$. This notation change emphasizes that $\psi_{\alpha_1}^{i_1}$ is a tensor whose elements depends on the local physical index i_1 and α_1 is called a bond index (virtual index). This is shown in Fig. 2.1.(b).

We continue to perform SVD decomposition between the site $i = 2$ and the remaining $N-2$ sites by reshaping the matrix $C_{\alpha_1,i_2i_3\cdots i_N}$ as $C_{\alpha_1 i_2,i_3\cdots i_N}$. Here, $(\alpha_1 i_2)$ and $(i_3 i_4 \cdots i_N)$ are combined indices. The SVD decomposition of this matrix is $C_{\alpha_1 i_2,i_3\cdots i_N} = U_{\alpha_1 i_2,\alpha_2} \Lambda_{\alpha_2,\alpha_2} V^{\dagger}_{\alpha_2,i_3 i_4\cdots i_N}$.

Hence Eq. (2.10) becomes

$$|\Psi\rangle = \sum_{\alpha_1,\alpha_2=1}^{D} \sum_{i_1,i_2,\cdots,i_N=1}^{d} \psi_{\alpha_1}^{i_1} \quad U_{\alpha_1 i_2,\alpha_2} \Lambda_{\alpha_2} V^{\dagger}_{\alpha_2,i_3 i_4\cdots i_N} |i_1 i_2 i_3 \cdots i_N\rangle \tag{2.11}$$

Re-write Eq. (2.11) by changing the notation for the matrix to the tensor notation $U_{\alpha_1 i_2,\alpha_2} = \psi^{i_2}_{\alpha_1\alpha_2}$

$$|\Psi\rangle = \sum_{\alpha_1,\alpha_2=1}^{D} \sum_{i_1,i_2,\cdots,i_N=1}^{d} \psi_{\alpha_1}^{i_1} \psi^{i_2}_{\alpha_1\alpha_2} C_{\alpha_2,i_3 i_4\cdots i_N} |i_1 i_2 i_3 \cdots i_N\rangle \tag{2.12}$$

where $C_{\alpha_2,i_3 i_4\cdots i_N} = \Lambda_{\alpha_2,\alpha_2} V^{\dagger}_{\alpha_2,i_3 i_4\cdots i_N}$. This is indicated in Fig. 2.1.(c).

It follows that when Schmidt decomposition is done through the lattice iteratively, the final state that will be obtained in the MPS form as

$$|\Psi\rangle = \sum_{\{\alpha\}} \sum_{\{i\}} \psi_{\alpha_1}^{i_1} \psi^{i_2}_{\alpha_1\alpha_2} \cdots \psi^{i_N}_{\alpha_{N-1}} |i_1 i_2 \cdots i_N\rangle \tag{2.13}$$

for the open boundaries ($i_1 \neq i_{N+1}$). By the use of Einstein summation convention for the virtual dimensions $\{\alpha\}$ Eq. (2.13) can be written as

$$|\Psi\rangle = \sum_{\{i\}} \psi_{\alpha_1}^{i_1} \psi^{i_2}_{\alpha_1\alpha_2} \cdots \psi^{i_N}_{\alpha_{N-1}} |i_1 i_2 \cdots i_N\rangle \tag{2.14}$$

Fig. 2.2 shows the graphical representation of the MPS with open boundaries. In Fig. 2.1 and Fig. 2.2, the circles on the chain represent the local tensors. The continuous lines and the dashed lines indicate the physical indices and the bond indices of tensors respectively.

Figure 2.2: Graphical representation of MPS where the circles indicate the tensors associated with each site. The number of the legs for each site is equal to the number of indices of the related tensor.

Notice from the fig that the $\psi^{i_1}_{\alpha_1}$ and $\psi^{i_N}_{\alpha_{N-1}}$ have only one subscript indicating that they are vectors (rank 1 tensor), while the ones with two indices are matrices (rank 2 tensor). This is valid in the case of open boundary conditions.

When the periodic boundary condition ($i_{N+1} = i_1$) is valid for the physical model, the first site and the last site tensors have an extra virtual bond dimension (α_0) connecting to these tensors. Therefore, the MPS in Eq. (2.13) is written as

$$|\psi\rangle = \sum_{\{i\}} \psi^{i_1}_{\alpha_0\alpha_1}\psi^{i_2}_{\alpha_1\alpha_2}\cdots\psi^{i_N}_{\alpha_{N-1}\alpha_0}|i_1 i_2\cdots i_N\rangle \tag{2.15}$$

$$= \mathrm{Tr}\left(\psi^{i_1}\psi^{i_2}\cdots\psi^{i_N}\right)|i_1 i_2\cdots i_N\rangle \tag{2.16}$$

where trace indicates the sum over virtual dimensions (bond dimensions). MPS with periodic boundaries is shown in Fig. 2.3.

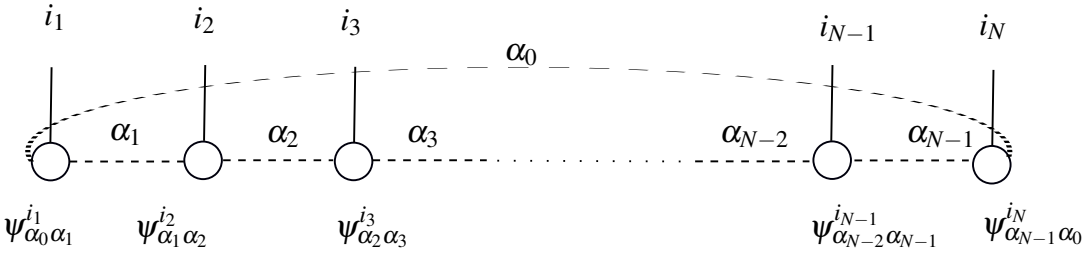

Figure 2.3: MPS representation with periodic boundary condition.

2.4 Matrix Product Operator Representation

We want to keep the MPS form of the state during numerical calculation. To do this, it is useful to work with operators which are in the product form of matrices, i.e. Matrix Product Operators (MPOs).

Our aim in this section is to obtain the MPO form of a quantum mechanical operator shown in Fig. 2.4 from the general definition by SVD decomposition.

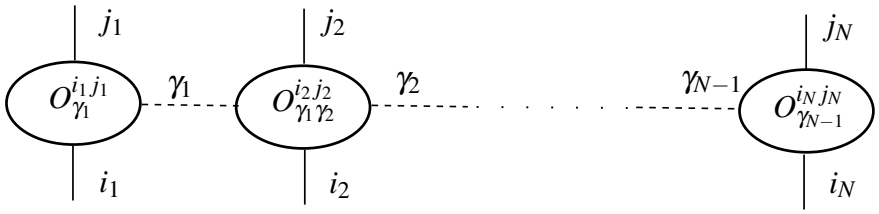

Figure 2.4: Graphical representation of MPO where the physical indices are shown by the continuous lines and the virtual indices shown by the dashed lines.

A quantum mechanical many body operator is defined as

$$\hat{\mathscr{O}} = \sum_{\{i\},\{j\}} O_{i_1 i_2 \cdots i_N, j_1 j_2 \cdots j_N} \left| i_1 i_2 \cdots i_N \right\rangle \left\langle j_1 j_2 \cdots j_N \right| \tag{2.17}$$

where $O_{i_1 i_2 \cdots i_N, j_1 j_2 \cdots j_N}$ denote the matrix elements. Consider splitting the chain into two part as one site ($n = 1$) is on the left and remaining $N-1$ sites are on the right. In order to do that one needs to reshape the coefficient matrix $O_{i_1 i_2 \cdots i_N, j_1 j_2 \cdots j_N} := O_{i_1 j_1, i_2 j_2 \cdots i_N j_N}$ and do the SVD decomposition as $O_{i_1 j_1, i_2 j_2 \cdots i_N j_N} = U_{i_1 j_1, \gamma_1} \Lambda_{\gamma_1, \gamma_1} V^{\dagger}_{\gamma_1, i_2 j_2 \cdots i_N j_N}$. It is important to emphasize that $(i_1 j_1)$ and $(i_2 j_2 \cdots i_N j_N)$ are combined indices here. Now, let us rewrite the Eq. (2.17) in terms of the SVD matrices:

$$\hat{\mathscr{O}} = \sum_{\{i\},\{j\}} U_{i_1 j_1, \gamma_1} \Lambda_{\gamma_1, \gamma_1} V^{\dagger}_{\gamma_1, i_2 j_2 \cdots i_N j_N} \left| i_1 i_2 \cdots i_N \right\rangle \left\langle j_1 j_2 \cdots j_N \right| \tag{2.18}$$

and change notation for the matrices (make a tensor) $U_{i_1 j_1, \gamma_1} := O^{i_1 j_1}_{\gamma_1}$ and $\Lambda_{\gamma_1, \gamma_1} V^{\dagger}_{\gamma_1, i_2 j_2 \cdots i_N j_N} := O_{\gamma_1, i_2 j_2 \cdots i_N j_N}$. Therefore Eq. (2.19) becomes

$$\hat{\mathscr{O}} = \sum_{\{i\},\{j\}} O^{i_1 j_1}_{\gamma_1} O_{\gamma_1, i_2 j_2 \cdots i_N j_N} \left| i_1 i_2 \cdots i_N \right\rangle \left\langle j_1 j_2 \cdots j_N \right| \tag{2.19}$$

We continue with the SVD decomposition of the configuration where the two ($n = 2$) sites are to the left and the remaining $N-2$ sites are to the right. In order to do that, we reshape the coefficient matrix as $O_{\gamma_1,i_2j_2\cdots i_Nj_N} := O_{\gamma_1 i_2 j_2, i_3 j_3 \cdots i_N j_N}$. Here, $(\gamma_1 i_2 j_2)$ and $(i_3 j_3 \cdots i_N j_N)$ are the combined indices. By the SVD decomposition we write, $O_{\gamma_1 i_2 j_2, i_3 j_3 \cdots i_N j_N} = U_{\gamma_1 i_2 j_2, \gamma_2} \Lambda_{\gamma_2,\gamma_2} V^{\dagger}_{\gamma_2, i_3 j_3 \cdots i_N j_N}$.

Rewriting Eq. (2.19) in terms of the SVD matrices with the definitions of $U_{\gamma_1 i_2 j_2, \gamma_2} := O^{i_2 j_2}_{\gamma_1 \gamma_2}$ and $\Lambda_{\gamma_2,\gamma_2} V^{\dagger}_{\gamma_2, i_3 j_3 \cdots i_N j_N} := O_{\gamma_2, i_3 j_3 \cdots i_N j_N}$, we obtain in the following expression:

$$\hat{\mathcal{O}} = \sum_{\{i\},\{j\}} O^{i_1 j_1}_{\gamma_1} O^{i_2 j_2}_{\gamma_1 \gamma_2} O_{\gamma_2, i_3 j_3 \cdots i_N j_N} |i_1 i_2 \cdots i_N\rangle \langle j_1 j_2 \cdots j_N| \tag{2.20}$$

It is obvious that when the SVD decomposition is continued recursively to the end of the chain, the final expression is

$$\hat{\mathcal{O}} = \sum_{\{i\},\{j\}} O^{i_1 j_1}_{\gamma_1} O^{i_2 j_2}_{\gamma_1 \gamma_2} \cdots O^{i_N j_N}_{\gamma_{N-1}} |i_1 i_2 \cdots i_N\rangle \langle j_1 j_2 \cdots j_N| \tag{2.21}$$

which is in the MPO form.

The graphical representation of the MPO is shown in Fig. 2.4. The virtual indices (γ_n) and the physical indices ($\{i\},\{j\}$) in Eq. (2.21) are shown by the dashed and the continuous lines respectively.

When we apply $\hat{\mathcal{O}}$ in Eq. (2.21) on the MPS form of $|\psi\rangle$ in Eq. (2.13) we get

$$\begin{aligned} \hat{\mathcal{O}}|\psi\rangle = \sum_{\{i\},\{j\},\{k\}} & O^{i_1 j_1}_{\gamma_1} \psi^{k_1}_{\alpha_1} O^{i_2 j_2}_{\gamma_1 \gamma_2} \psi^{k_2}_{\alpha_1 \alpha_2} \cdots O^{i_N j_N}_{\gamma_{N-1}} \psi^{k_N}_{\alpha_{N-1}} \\ & \times |i_1 i_2 \cdots i_N\rangle \langle j_1 j_2 \cdots j_N | k_1 k_2 \cdots k_N\rangle \end{aligned} \tag{2.22}$$

which is equal to

$$\hat{\mathcal{O}}|\psi\rangle = \sum_{\{i\},\{j\}} O^{i_1 j_1}_{\gamma_1} \psi^{j_1}_{\alpha_1} O^{i_2 j_2}_{\gamma_1 \gamma_2} \psi^{j_2}_{\alpha_1 \alpha_2} \cdots O^{i_N j_N}_{\gamma_{N-1}} \psi^{j_N}_{\alpha_{N-1}} |i_1 i_2 \cdots i_N\rangle \tag{2.23}$$

where the repeating physical indices j_n are summed over. By definition of $\phi^{i_n}_{(\gamma_{n-1}\alpha_{n-1})(\gamma_n \alpha_n)} = O^{i_n j_n}_{\gamma_{n-1}\gamma_n} \psi^{j_n}_{\alpha_{n-1}\alpha_n}$ where $(\gamma_{n-1}\alpha_{n-1})$ and $(\gamma_n \alpha_n)$ are combined indices we obtain,

$$\begin{aligned} |\Phi\rangle =& \hat{\mathcal{O}}|\psi\rangle \\ =& \sum_{\{i\}} \phi^{i_1}_{(\gamma_1 \alpha_1)} \phi^{i_2}_{(\gamma_1 \alpha_1)(\gamma_2 \alpha_2)} \cdots \phi^{i_N}_{(\gamma_{N-1}\alpha_{N-1})} |i_1 i_2 \cdots i_N\rangle \end{aligned} \tag{2.24}$$

which is again in the MPS form.

For an operator which has the form sum of product of operators, an efficient way of choosing the $O^{i_n j_n}_{\gamma_{n-1}\gamma_n}$ matrices is given for one body and two body operators in [15] is shown below.

- One body operators

  For a one body operator of the form

  $$\hat{\mathcal{O}}_1 = \sum_{n=1}^{N} \hat{X}_n \tag{2.25}$$

  the the local matrix representation of the operator $\hat{\mathcal{O}}_1$ is

  $$O^{i_n j_n} = \begin{pmatrix} \delta^{i_n j_n} & 0 \\ X_n^{i_n j_n} & \delta^{i_n j_n} \end{pmatrix} \tag{2.26}$$

  for $n = 2, 3, \cdots, N-1$ where $O^{i_n j_n} = \langle i_n | X_n | j_n \rangle$. The first and the last local matrices are $O^{i_1 j_1} = (X_1^{i_1 j_1}, \delta^{i_1 j_1})$ and $O^{i_N j_N} = (\delta^{i_N j_N}, X_N^{i_N j_N})^T$ respectively.

- Two body operators

  For a two body operator of the form

  $$\hat{\mathcal{O}}_2 = \sum_{n=1}^{N} X_n Y_{n+1} \tag{2.27}$$

  the matrix product operator representation can be

  $$O^{i_n j_n} = \begin{pmatrix} \delta^{i_n j_n} & 0 & 0 \\ Y_n^{i_n j_n} & 0 & 0 \\ 0 & X_n^{i_n j_n} & \delta^{i_n j_n} \end{pmatrix} \tag{2.28}$$

  whereas $O^{i_1 j_1} = (0, X_1^{i_1 j_1}, \delta^{i_1 j_1})$ and $O^{i_N j_N} = (\delta^{i_N j_N}, Y_N^{i_N j_N}, 0)^T$.

2.5 Calculations with MPS

In this section, scalar product of two MPSs and expectation value of a MPO are discussed.

- Scalar Product

  The graphical representation of scalar product in MPS formulation is given in Fig. 2.5.

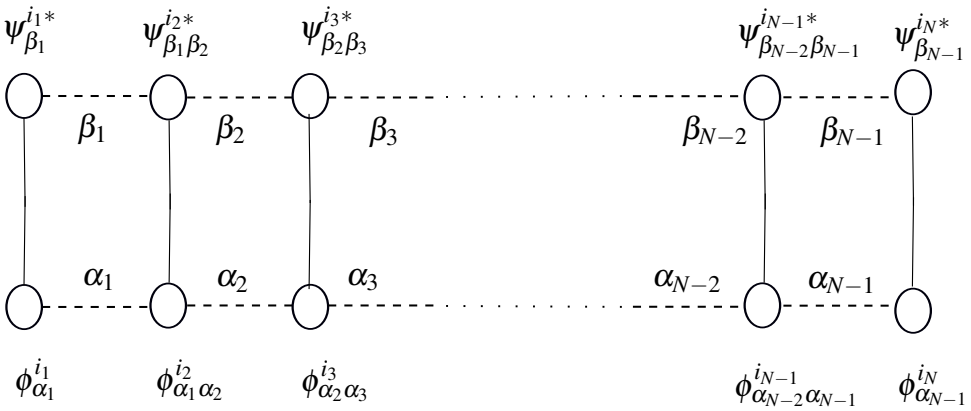

Figure 2.5: Graphical representation of the scalar product for MPSs.

The scalar product of a wave vector in MPS form as in Eq, (2.13) is calculated as in the following arguments:

$$\begin{aligned}\langle\Psi|\Phi\rangle &= \sum_{\{i\},\{j\}} \psi^{j_1*}_{\beta_1}\psi^{j_2*}_{\beta_1\beta_2}\cdots\psi^{j_N*}_{\beta_{N-1}}\phi^{i_1}_{\alpha_1}\phi^{i_2}_{\alpha_1\alpha_2}\cdots\phi^{i_N}_{\alpha_N}\langle j_1 j_2\cdots j_N|i_1 i_2\cdots i_N\rangle \\ &= \sum_{\{i\}} \psi^{i_1*}_{\beta_1}\phi^{i_1}_{\alpha_1}\psi^{i_2*}_{\beta_1\beta_2}\phi^{i_2}_{\alpha_1\alpha_2}\cdots\psi^{i_N*}_{\beta_{N-1}}\phi^{i_N}_{\alpha_{N-1}} \\ &= \xi_{(\beta_1\alpha_1)}\xi_{(\beta_1\alpha_1)(\beta_2\alpha_2)}\cdots\xi_{(\beta_{N-1}\alpha_{N-1})}\end{aligned} \tag{2.29}$$

where $\xi_{\beta_n\alpha_n} := \psi^{i_n*}_{\beta_n}\phi^{i_n}_{\alpha_n}$. All combined repeated indices $(\beta_n\alpha_n)$ are summed over in the final expression.

- Expectation value of an MPO The graphical representation of expectation value of an MPO is indicated in the Fig. 2.6.

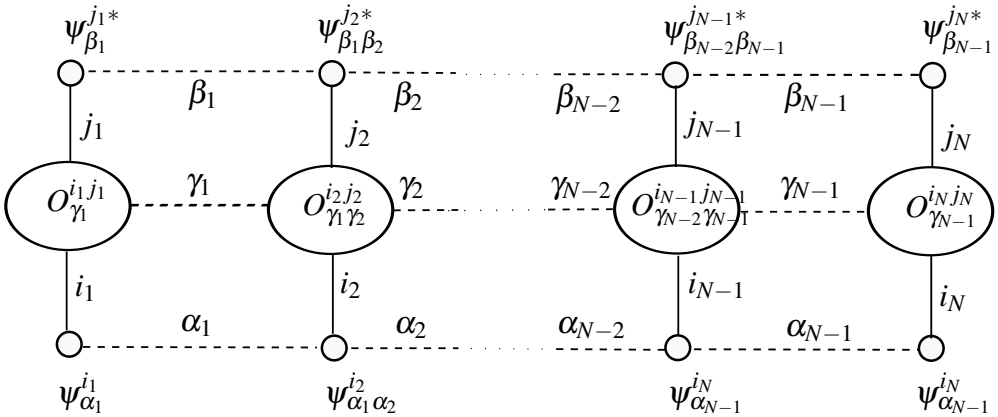

Figure 2.6: Graphical representation of the expectation value of an MPO.

Expectation value of an operator in the form as in Eq. (2.21) is calculated by sandwiching between two MPS as

$$\begin{aligned}\langle\psi|\hat{\mathscr{O}}|\psi\rangle &= \sum_{\{i\},\{j\},\{k\},\{l\}} \psi^{j_1*}_{\beta_1} O^{k_1 l_1}_{\gamma_1} \psi^{i_1}_{\alpha_1} \psi^{j_2*}_{\beta_1\beta_2} O^{k_2 l_2}_{\gamma_1\gamma_2} \psi^{i_2}_{\alpha_1\alpha_2} \cdots \psi^{j_N*}_{\beta_{N-1}} O^{k_N l_N}_{\gamma_{N-1}} \psi^{i_N}_{\alpha_{N-1}} \\ &\times \underbrace{\langle j_1 j_2 \cdots j_N | k_1 k_2 \cdots k_N\rangle}_{\delta_{j_1 k_1},\delta_{j_2 k_2},\cdots} \underbrace{\langle l_1 l_2 \cdots l_N | i_1 i_2 \cdots i_N\rangle}_{\delta_{l_1 i_1},\delta_{l_2 i_2},\cdots} \\ &= \psi^{j_1*}_{\beta_1} O^{j_1 i_1}_{\gamma_1} \psi^{i_1}_{\alpha_1} \psi^{j_2*}_{\beta_1\beta_2} O^{j_2 i_2}_{\gamma_1\gamma_2} \psi^{i_2}_{\alpha_1\alpha_2} \cdots \psi^{j_N*}_{\beta_{N-1}} O^{j_N i_N}_{\gamma_{N-1}} \psi^{i_N}_{\alpha_{N-1}} \\ &= \phi_{\beta_1\gamma_1\alpha_1} \phi_{(\beta_1\gamma_1\alpha_1)(\beta_2\gamma_2\alpha_2)} \cdots \phi_{\beta_{N-1}\gamma_{N-1}\alpha_{N-1}}\end{aligned} \tag{2.30}$$

where $\phi_{(\beta_{n-1}\gamma_{n-1}\alpha_{n-1})(\beta_n\gamma_n\alpha_n)} = \psi^{j_n*}_{\beta_{n-1}\beta_n} O^{j_n i_n}_{\gamma_{n-1}\gamma_n} \psi^{i_n}_{\alpha_{n-1}\alpha_n}$. Einstein summation convention is used for the repeating indices. According to the figure, firstly the physical indices with continues lines are contracted. Contractions over the virtual indices (bond dimensions) are done after the physical indices.

Expectation value of an MPO in Eq. 2.30 can be expressed by the scalar product of two MPSs as $\langle\psi|\phi\rangle$ where $|\phi\rangle = \mathscr{O}|\psi\rangle$. It is indicated in Fig. 2.5 and in Eq. 2.29.

2.6 Variational Matrix Product State Method

Suppose that we want to use the MPS formulation to approximate the ground state of a physical system. We need to find the best MPS representation for the ground state in Eq. (2.13). This is an optimization problem for the MPS where the each local tensor coefficient $\psi^{i_n*}_{\alpha_{n-1}\alpha_n}$ is a variational parameter. This turns out to be a minimization problem for an effective Hamiltonian for a specific separation for the chain. The minimization is done according to

$$\min_{\langle\Psi|}[\langle\Psi|\mathscr{H}|\Psi\rangle - \lambda\,\langle\Psi|\Psi\rangle] \tag{2.31}$$

The minimization problem Eq. (2.31) can be written by replacing Eq. (2.13) and Eq. (2.21) where $\mathscr{H}$ is the Hamiltonian operator and λ is the Lagrange multiplier for the normalization by

$$\begin{aligned}\min_{\psi^{i_1*}_{\beta_1},\psi^{i_2*}_{\beta_1\beta_2},\cdots} \Bigg(&\sum_{i,k,l,m} \psi^{i_1*}_{\beta_1} \psi^{i_2*}_{\beta_1\beta_2} \cdots \psi^{i_N*}_{\beta_{N-1}} O^{k_1 l_1}_{\gamma_1} O^{k_2 l_2}_{\gamma_1\gamma_2} \cdots O^{k_N l_N}_{\gamma_{N-1}} \\ &\times \psi^{m_1}_{\alpha_1} \psi^{m_2}_{\alpha_1\alpha_2} \cdots \psi^{m_N}_{\alpha_{N-1}} \langle i_1 i_2 \cdots i_N | k_1 k_2 \cdots k_N\rangle \langle l_1 l_2 \cdots l_N | m_1 m_2 \cdots m_N\rangle \\ &-\lambda \sum_{i,m} \psi^{i_1*}_{\beta_1} \psi^{i_2*}_{\beta_1\beta_2} \cdots \psi^{i_N*}_{\beta_{N-1}} \psi^{m_1}_{\alpha_1} \psi^{m_2}_{\alpha_1\alpha_2} \cdots \psi^{m_N}_{\alpha_{N-1}} \Bigg)\end{aligned} \tag{2.32}$$

where $\langle i_1 i_2 \cdots i_N | k_1 k_2 \cdots k_N \rangle = \delta_{i_1 k_1} \delta_{i_2 k_2} \cdots \delta_{i_N k_N}$ and $\langle l_1 l_2 \cdots l_N | m_1 m_2 \cdots m_N \rangle = \delta_{l_1 m_1} \delta_{l_2 m_2} \cdots \delta_{l_N m_N}$.

By defining the vectorization of the tensors as $x_{i_n} := \psi^{i_n}_{\alpha_{n-1}\alpha_n}$ whose dimensions are $d \times D \times D$, Eq. (2.32) becomes

$$\min_{x^*_{i_1}, x^*_{i_2}, \cdots} \left(\sum_{i,m} x^*_{i_1} x_{m_1} x^*_{i_2} x_{m_2} \cdots x^*_{i_N} x_{m_N} O^{i_1 m_1}_{\beta_1} O^{i_2 m_2}_{\beta_1 \beta_2} \cdots O^{i_N m_N}_{\beta_N} - \lambda \sum_i x^*_{i_1} x_{i_1} x^*_{i_2} x_{i_2} \cdots x^*_{i_N} x_{i_N} \right) \tag{2.33}$$

In Eq. (2.33), there are several minimization parameters, $x^*_{i_n}$. Minimization according to all $x^*_{i_n}$ simultaneously requires a huge computational effort. There is a variational method of doing the minimization by taking the $x^*_{i_n}$ as a variational parameter and treating the remaining $N-n$ constant ($n = 1, 2, \cdots$) [10]. This is the so called alternating least squares method. The same procedure is applied for each local vector coefficient ($x^*_{i_n}$) one by one along the chain.

Suppose that we have decomposed the chain as $n-1$ sites to the left and $N-n-1$ sites to the right as in Fig. 2.7. The minimization parameters are the coefficients $x^*_{i_n}$, in this configuration. Since $x^*_{i_n}$ and $O^{i_n m_n}_{\gamma_{n-1}\gamma_n}$ are just numbers, their places can be changed in the Eq. (2.33) so that it becomes:

$$\min_{x^*_{i_n}} \left(x^*_{i_n} H^L_{\beta_{n-1}\gamma_{n-1}\alpha_{n-1}} O^{i_n m_n}_{\gamma_{n-1}\gamma_n} H^R_{\beta_n \gamma_n \alpha_n} x_{i_n} - \lambda \sum_i x^*_{i_n} x^*_{i_1} x_{i_1} x^*_{i_2} x_{i_2} \cdots x^*_{i_N} x_{i_N} x_{i_n} \right) \tag{2.34}$$

with respect to $x^*_{i_n}$ as an optimization parameter. Here

$$H^L_{\beta_{n-1}\gamma_{n-1}\alpha_{n-1}} = x^*_{i_1} O^{i_1 m_1}_{\gamma_1} x_{m_1} x^*_{i_2} O^{i_2 m_2}_{\gamma_1 \gamma_2} x_{m_2} \cdots x^*_{i_{n-1}} O^{i_{n-1} m_{n-1}}_{\gamma_{n-2}\gamma_{n-1}} x_{m_{n-1}} \tag{2.35}$$

and

$$H^R_{\beta_n \gamma_n \alpha_n} = x^*_{i_{n+1}} O^{i_{n+1} m_{n+1}}_{\gamma_n \gamma_{n+1}} x_{m_{n+1}} x^*_{i_{n+2}} O^{i_{n+2} m_{n+2}}_{\gamma_{n+1}\gamma_{n+2}} x_{m_{n+2}} \cdots x^*_{i_N} O^{i_N m_N}_{\gamma_{N-1}} x_{m_N} \tag{2.36}$$

which are left and right block operators for this chain separation. Instead of doing physical index contractions for all sites through the chain for each minimization step, we contract the local tensors once and keep in memory. We use memory efficiently in this way.

Graphical representation for the minimization step (n^{th} site) is shown in Fig. 2.7.

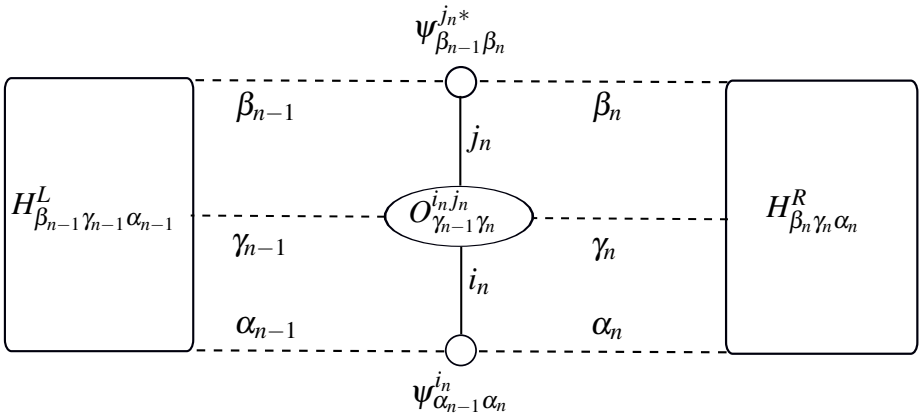

Figure 2.7: Graphical representation of the effective Hamiltonian (H_{eff}) for the n^{th} site.

Defining H_{eff} and N_{eff} which are effective Hamiltonian and effective normalization operator respectively

$$H_{eff} = \sum_{i_n, m_n} \left(H^L_{\beta_{n-1}\gamma_{n-1}\alpha_{n-1}} O^{i_n m_n}_{\beta_{n-1}\beta_n} H^R_{\beta_n\gamma_n\alpha_n} \right)$$

$$N_{eff} = \sum_{\{i\}\backslash i_n} \left(x^*_{i_1} x_{i_1} x^*_{i_2} x_{i_2} x^*_{i_3} x_{i_3} \cdots x^*_{i_N} x_{i_N} \right)$$

and re-writing Eq. (2.33) in terms of the H_{eff} and N_{eff}, we obtain the following expression:

$$\min_{x^*_{i_n}} \left(x^*_{i_n} H_{eff} x_{i_n} - \lambda x^*_{i_n} N_{eff} x_{i_n} \right) \tag{2.37}$$

The minimization of Eq. (2.37) with respect to $x^*_{i_n}$ gives a generalized eigenvalue problem.

$$H_{eff} x_{i_n} = \lambda N_{eff} x_{i_n} \tag{2.38}$$

In the case of the open boundary conditions, it is possible to make $N_{eff} = \mathbb{I}$, so the Eq. 2.38 reduces to the ordinary eigenvalue problem. Details are given in reference [16].

Minimization procedure is done with respect to $x^*_{i_1}, x^*_{i_2}, x^*_{i_3}, \cdots$ one by one back and forth until a good convergence for the MPS representation is satisfied. This is the sweeping

process which improves the convergence of the target state. For each decomposition H_{eff} is calculated and updated through the sweeping.

2.6.1 Excited state calculation

It is possible to calculate the first excited state in a similar fashion. By Lagrange multiplier to ensure orthogonality to the ground state

$$\min_{\langle\psi_1|}[\langle\psi_1|\mathscr{H}|\psi_1\rangle - \lambda\langle\psi_1|\psi_1\rangle - \mu\langle\psi_0|\psi_1\rangle] \tag{2.39}$$

where $|\psi_0\rangle$ denotes the ground state in Eq. (2.13). Let us introduce the excited state with the following notation

$$|\psi_1\rangle = \sum_{\{j\}} \chi^{j_1}_{\beta_1}\chi^{j_2}_{\beta_1\beta_2}\cdots\chi^{j_N}_{\beta_{N-1}}|j_1 j_2\cdots j_N\rangle \tag{2.40}$$

to avoid confusion and the Hamiltonian is in Eq. (2.21).

The expression in the Eq. (2.39) with the vectorization definitions of the first excited state $x_{i_n} := \chi^{i_n}_{\beta_{n-1}\beta_n}$ and the ground state $y_{i_n} := \psi^{i_n}_{\alpha_{n-1}\alpha_n}$ becomes after some manipulations as

$$\min_{x^*_{i_n}}\left[x^*_{i_n}H_{eff}x_{i_n} - \lambda x^*_{i_n}N_{eff}x_{i_n} - \mu y^*_{i_n}V_{eff}x_{i_n}\right] \tag{2.41}$$

The solution of the Eq. (2.41) is given by the generalized eigenvalue equation

$$PH_{eff}Px_{i_n} = \lambda PN_{eff}Px_{i_n} \tag{2.42}$$

where $PH_{eff}P$ and $PN_{eff}P$ are the projected effective Hamiltonian and the projected effective normalization operator of that separation, respectively.

Since we are seeking x_{i_n}, the projection operator, P, is chosen as the matrix combined from the orthonormal vectors which are all orthogonal to $y^*_{i_n}V_{eff}$.

Eq. (2.42) is solved for the x_{i_n} for the each separation ($n = 1,2,\cdots,N$) of the chain.

2.6.2 Variational matrix product states (VMPS) algorithm

We start with the MPS form which has random coefficients at first and then do SVD to orthonormalize the basis. Here the algorithm of the variational matrix product state method is summarized in a few steps:

(i) Create a MPS for N sites with the bond dimension D of the form in Eq. (2.13), whose tensor coefficients ($\psi^{i_n}_{\alpha_{n-1}\alpha_n}$) are random.

(ii) Form MPO form of the Hamiltonian as in the Eq. (2.21).

In order to minimize the expectation value of the Hamiltonian (Eq. (2.21)):

(iii) Start from the first decomposition $(n = 1)$ for the chain:

(a) Reshape the MPS matrices ($\psi^{i_n}_{\alpha_{n-1}\alpha_n}$) in a convenient way to the decomposition. Do SVD decomposition to obtain the MPS in terms of the $\psi^{i_n}_{\alpha_{n-1}\alpha_n} = U_{\alpha_{n-1} i_n \alpha_n}$, matrices with local physical indices.

(b) Form H_{eff} for the decomposition (see Sec 2.6) by summing over the all degrees of freedom of the chain except for the n^{th} site as in Fig. 2.7.

(c) Diagonalize H_{eff}, take the minimum eigenvalue as the ground state energy and the corresponding eigenvector. Take this eigenvector as the updated vectorization of the tensor $x_{i_n} = \psi^{i_n}_{\alpha_{n-1}\alpha_n}$.

(iv) Continue to the decomposition with the next site. It is significant to do SVD for the each chain decomposition in turn ($n = 1, 2 \cdots$). Use the updated MPS representation from the previous step for calculation of H_{eff}. This is the key step of the method. Do the steps (a), (b), (c) for each decomposition up to the end of the chain.

(v) Repeat the (iii)-(iv) for the reverse direction until some convergence criteria (i.e. $O(10^{-8})$) for the energy is obtained. This criteria would be change in the order of the energies between the current and the previous steps.

2.7 Time Evolution of MPS

Time evolution of the MPS in Eq. (2.13) is done by applying the time evolution operator formed by the Hamiltonian $\mathscr{H}$. In order to save MPS form of $|\psi\rangle$ during the time evolution, the time evolution operator, $\mathscr{U}$, is also required in the MPO form as in Eq. (2.21) [16]. This can be handled by the variational minimization where the minimized function is the error in the time evolution in each time step, δt. The minimization is done to the norm of the difference of state $|\psi\rangle$ at t and $t + \delta t$:

$$\| \mathscr{U} |\psi(t)\rangle - |\psi(t+\delta t)\rangle \|^2 \qquad \textbf{(2.43)}$$

A critical point here is that when an MPO is applied to an MPS the bond dimension D of the MPS gets bigger (see Eq. (2.24) and Appendix B). Therefore, the minimization in Eq. (2.43) is done by searching the best MPS with the given bond dimension D representing the state of the system.

Eq. (2.43) can be written explicitly in terms of Eq. (2.13) and Eq. (2.21) as

$$\|\mathscr{U}\,|\psi(t)\rangle - |\psi(t+\delta t)\rangle\,\|^2 = \|\mathscr{U}\,|\psi(t)\rangle\,\|^2 + \|\psi(t+\delta t)\|^2 - 2\,\langle\psi(t)|\mathscr{U}^\dagger|\psi(t+\delta t)\rangle \tag{2.44}$$

where

$$\|\mathscr{U}\,|\psi(t)\rangle\,\|^2 = \psi^{i_1*}_{\alpha_1} O^{j_1 i_1 *}_{\gamma_1} O^{l_1 j_1}_{\mu} \psi^{l_1}_{\beta_1} \psi^{i_2 *}_{\alpha_1\alpha_2} O^{j_2 i_2 *}_{\gamma_1\gamma_2} O^{l_2 j_2}_{\mu_1\mu_2} \psi^{l_2}_{\beta_1\beta_2} \cdots \psi^{i_N *}_{\alpha_{N-1}} O^{j_N i_N *}_{\gamma_{N-1}} O^{l_N j_N}_{\mu_{N-1}} \psi^{l_N}_{\beta_{N-1}} \tag{2.45}$$

and

$$\|\psi(t+\delta t)\|^2 = \chi^{i_1*}_{\alpha_1}\chi^{i_1}_{\beta_1}\chi^{i_2*}_{\alpha_1\alpha_2}\chi^{i_2}_{\beta_1\beta_2}\cdots\chi^{i_N*}_{\alpha_{N-1}}\chi^{i_N}_{\beta_{N-1}} \tag{2.46}$$

and

$$\langle\psi(t)|\mathscr{U}^\dagger|\psi(t+\delta t)\rangle = \psi^{i_1*}_{\alpha_1} O^{i_1 j_1 *}_{\gamma_1} \chi^{j_1}_{\beta_1} \psi^{i_2 *}_{\alpha_1\alpha_2} O^{i_2 j_2 *}_{\gamma_1\gamma_2} \chi^{j_2}_{\beta_1\beta_2} \cdots \psi^{i_N *}_{\alpha_{N-1}} O^{i_N j_N *}_{\gamma_{N-1}} \chi^{j_N}_{\beta_{N-1}} \tag{2.47}$$

by the use of orthonormalization property of the basis vectors as done before. Defining the vectorization of $\psi^{i_n}_{\alpha_{n-1}\alpha_n}(t) := y_{\alpha_{n-1}i_n\alpha_n}$ and $\psi^{i_n}_{\alpha_{n-1}\alpha_n}(t+\delta t) := x_{\alpha_{n-1}i_n\alpha_n}$ so that Eq. (2.44), we write for the n^{th} decomposition as

$$y^\dagger_{\alpha_{n-1}i_n\alpha_n}\tilde{O}_{eff}y_{\beta_{n-1}l_n\beta_n} + x^\dagger_{\alpha_{n-1}i_n\alpha_n}N_{eff}x_{\beta_{n-1}i_n\beta_n} - 2y_{\alpha_{n-1}i_n\alpha_n}O_{eff}x_{\beta_{n-1}j_n\beta_n} \tag{2.48}$$

where $\tilde{O}_{eff}$, N_{eff} and O_{eff} are the contracted tensors in Eq. (2.44) out of the tensors belonging to the site n.

Minimization with respect to $y^\dagger_{\alpha_{\gamma_{n-1}i_n\gamma_n}}$ gives

$$\tilde{O}_{eff}y_{\beta_{n-1}l_n\beta_n} = x_{\beta_{n-1}l_n\beta_n} \tag{2.49}$$

2.7.1 Suzuki-Trotter expansion of the time evolution operator

Time evolution of a state $|\psi\rangle$ is done by successive iterations of time evolution operator by time-steps δt as

$$|\psi(t)\rangle = (e^{-i\mathscr{H}\delta t}\cdots e^{-i\mathscr{H}\delta t})\,|\psi(0)\rangle \tag{2.50}$$

where $\mathscr{H}$ could be written as a sum of local bond Hamiltonian operators

$$\mathscr{H} = \mathscr{H}_1 + \mathscr{H}_2 + \cdots + \mathscr{H}_{N-1} \tag{2.51}$$

where N is the number of sites on the chain and $\mathscr{H}_i$ denote the bond operators.

The local bond operators $\mathscr{H}_1, \mathscr{H}_2, \cdots, \mathscr{H}_{N-1}$ in general do not commute each other, i.e. $[\mathscr{H}_i, \mathscr{H}_{i+1}] \neq 0$. Hence it is not correct to split the time evolution operator as

$$e^{-i\mathscr{H}\delta t} \neq e^{-i\mathscr{H}_1\delta t} e^{-i\mathscr{H}_2\delta t} \cdots e^{-i\mathscr{H}_{N-1}\delta t} \tag{2.52}$$

In order to break up the expression in an appropriate way, we group the local bond operators as not to have any common sites so that each group consist of commuting local bond operators. Let us group even local bond operators; $\mathscr{H}_A : \mathscr{H}_2, \mathscr{H}_4, \mathscr{H}_6, \cdots$ and the other odd local bond operators; $\mathscr{H}_B : \mathscr{H}_1, \mathscr{H}_3, \mathscr{H}_5, \cdots$ and $\mathscr{H} = \mathscr{H}_A + \mathscr{H}_B$. Hence we can split the time evolution operator in terms of the commuting local bond operators as

$$e^{-i\mathscr{H}_A\delta t} = e^{-i\mathscr{H}_2\delta t} e^{-i\mathscr{H}_4\delta t} \cdots \tag{2.53a}$$

$$e^{-i\mathscr{H}_B\delta t} = e^{-i\mathscr{H}_1\delta t} e^{-i\mathscr{H}_3\delta t} \cdots \tag{2.53b}$$

where $\mathscr{H}_A$, $\mathscr{H}_B$ represent even bond operator group and odd bond operator group respectively. Graphical representation of this is shown in Fig. 2.8.

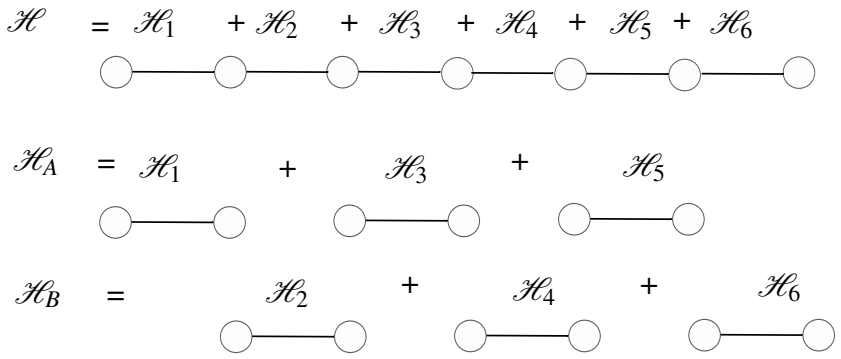

Figure 2.8: Graphical representation for even-odd bond separation of time evolution operator

The time evolution operator is given by the second order Suzuki-Trotter expansion which approximates to $e^{-i\mathscr{H}\delta t} = 1 - i\mathscr{H}\delta t - \frac{\mathscr{H}^2}{2!}\delta t^2 + \cdots$ expansion up to the third order terms in δt.

$$e^{-i(\mathscr{H}_A+\mathscr{H}_B)\delta t} = e^{-i\mathscr{H}_A\delta t/2} e^{-i\mathscr{H}_B\delta t} e^{-i\mathscr{H}_A\delta t/2} + O(\delta t^3) \tag{2.54}$$

The error in Eq. (**2.54**) is calculated explicitly as

$$
\begin{aligned}
e^{-i\mathscr{H}_A\delta t/2}e^{-i\mathscr{H}_B\delta t}e^{-i\mathscr{H}_A\delta t/2} &= (1-i\mathscr{H}_A\delta t-\frac{\mathscr{H}_A^2}{2!}\delta t^2-i\frac{\mathscr{H}_A^3}{3!}\delta t^3+\cdots)\\
\times(1-i\mathscr{H}_B\delta t-i\frac{\mathscr{H}_B^2}{2!}\delta t^2-i\frac{\mathscr{H}_B^3}{3!}\delta t^3+\cdots)&\times(1-i\mathscr{H}_A\delta t-\frac{\mathscr{H}_A^2}{2!}\delta t^2-i\frac{\mathscr{H}_A^3}{3!}\delta t^3+\cdots)\\
&= 1-i(\mathscr{H}_A+\mathscr{H}_B)\delta t-(\mathscr{H}_A+\mathscr{H}_B)\frac{\delta t^2}{2}\\
&+\left(-i\frac{\mathscr{H}_A^3}{3!}\frac{1}{2^3}-i\frac{\mathscr{H}_B^3}{3!}\frac{1}{2^3}-i\frac{\mathscr{H}_B^3}{3!}+i\frac{\mathscr{H}_A\mathscr{H}_B^2}{2!}+i\frac{\mathscr{H}_A^2\mathscr{H}_B}{2}+i\frac{\mathscr{H}_A^3}{4}\right)\delta t^3
\end{aligned}
\tag{2.55}
$$

The terms which are proportional to the third power of δt (δt^3) are not equal to the third power terms in the $e^{-i\mathscr{H}\delta t}$ expansion. Hence we see that the error of the expansion is the order of δt^3 with small time-steps.

2.8 Reduction of the Bond Dimension for an MPS

When an MPO is applied to an MPS, the resulting MPS has a bigger bond dimension. This is shown in Eq. (2.24). Bond dimension of the new MPS can be reducible. In other words, state can be represented by a bond dimension less than the resulting bond dimension. In this case, we do reduction for the bond dimension of MPS.

During the time evolution, we apply unitary time evolution operator (MPO) to the state (MPS). We reduce the bond dimension while solving the Eq. (2.43). To do the reduction, we search an MPS with a minimal bond dimension. This minimal bond dimension is equal to the bond dimension of the original MPS, $\dim(\alpha_n)$ in Eq. 2.14.

3. COUPLED QUANTUM CAVITY ARRAYS

3.1 Quantum Rabi Model

Quantum Rabi model is a model for the interaction between a single mode of electromagnetic field and a two level atom which is given by the Hamiltonian

$$\mathscr{H}_R = \mathscr{H}_A + \mathscr{H}_F + \mathscr{H}_I \tag{3.1}$$

Here, $\mathscr{H}_A$ is the atom Hamiltonian given by

$$\mathscr{H}_A = \sum_i E_i \left|i\right\rangle \left\langle i\right| \tag{3.2}$$

where $\left|i\right\rangle$ denotes the states of the atom. The single mode field Hamiltonian is

$$\mathscr{H}_F = \hbar\omega \left(a^\dagger a + \frac{1}{2}\right) \tag{3.3}$$

where $a^\dagger$ (a) is creation (annihilation) operator for the photon mode and the interaction term is

$$\mathscr{H}_I = -e\boldsymbol{r}\cdot\boldsymbol{E} \tag{3.4}$$

where $\boldsymbol{r}$ is the position operator of the electron and $\boldsymbol{E}$ is the electric field operator in the dipole approximation.

Re-write the $e\boldsymbol{r}$ by inserting the completeness relation for the atom basis as

$$e\boldsymbol{r} = \sum_{i,j} \left|i\right\rangle \left\langle i\right| \boldsymbol{r} \left|j\right\rangle \left\langle j\right| \tag{3.5}$$

$$= \sum_{i,j} \mu_{ij}\sigma_{ij} \tag{3.6}$$

where $\mu_{ij} = \langle i|\boldsymbol{r}|j\rangle$ is the electric dipole transition element and $\sigma_{ij} = |i\rangle\langle j|$. The quantized electric field is given as in [17]

$$\boldsymbol{E} = \hat{e}\varepsilon(a + a^{\dagger}) \tag{3.7}$$

Here, $\varepsilon = \sqrt{\hbar\omega/2\varepsilon_0 V}$ where V is the volume of the quantization box and ε_0 is the vacuum permittivity, $\hat{e}$ is the polarization unit vector. By these changes the interaction term becomes

$$\mathscr{H}_I = -\sum_{i,j} g_{ij}\sigma_{ij}(a^{\dagger} + a) \tag{3.8}$$

where $g_{ij} = -\frac{\mu_{ij}\hat{e}\varepsilon}{\hbar}$. The two level system has two possible states $|i\rangle = \{|0\rangle, |1\rangle\}$. Electric dipole transition is the same for transition from the ground state to the excited state with the transition from the excited state to the ground state. Therefore, $g_{01} = g_{10} = g$ and $g_{11} = g_{00} = 0$. We use the notation for the electric dipole transition moment from ground state to the excited state operator as $\sigma_+ = \sigma_{10}$ and for the reverse process $\sigma_- = \sigma_{01}$. Hence, Quantum Rabi Hamiltonian becomes

$$\mathscr{H}_R = \omega a^{\dagger}a + \frac{\Omega}{2}\sigma_z + g(\sigma_- + \sigma_+)(a^{\dagger} + a) \tag{3.9}$$

where raising and lowering operators are respectively $\sigma_+ = |1\rangle\langle 0|$ and Ω is the transition frequency for the two level atom (TLS), $\sigma_- = |0\rangle\langle 1|$. σ_z is the Pauli spin matrix.

Quantum Rabi model in Eq. 3.9 does not have an exact solution due to the counter rotating terms ($\sigma_+ a^{\dagger}$, $\sigma_- a$), but it is solvable with some approximations. Ignoring the counter rotating terms corresponds to rotating wave approximation (RWA). RWA is not sufficiently good at strong coupling limit in comparison with the numerical solution of the Rabi Hamiltonian in Eq. (3.9). There are some other approximations for the Rabi model to capture the physics in the strong coupling limit. One of them is offered by Irish in [18] which is so-called generalized RWA (GRWA). This approximation is held by a basis transformation for the Hamiltonian in Eq. 3.9.

Fig. 3.1 shows ground state and the first excited state energies with different approximations.

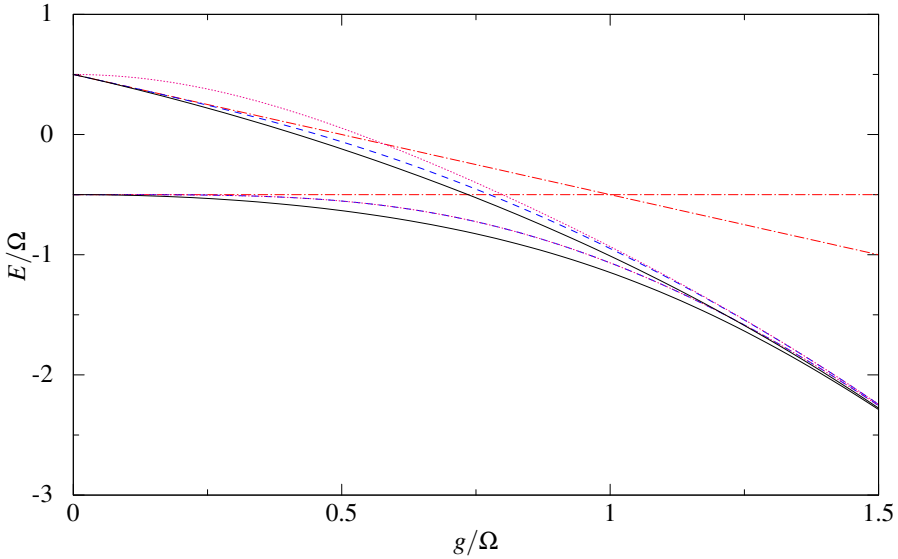

Figure 3.1: Numerical, RWA, generalized RWA and adiabatic approximation energies versus coupling strength. Red dotted dashed lines indicate the RWA, black solid lines numerical solution, blue dashed lines generalized RWA (GRWA) and pink dotted lines adiabatic approximation energies.

We see from the figure that adiabatic approximation works well in strong coupling limit, whereas RWA breaks down around strong coupling limit $g = 0.5$. GRWA merges the behavior of adiabatic approximation in strong coupling limit and RWA in weak coupling limit. Therefore, it is convenient with the analytical solution of the Rabi model in Eq. 3.9.

3.1.1 Jaynes-Cummings model

Quantum Rabi model is depicted in Fig. 3.2.

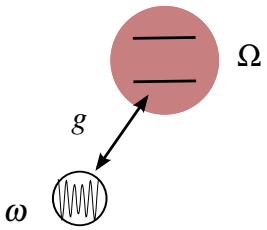

Figure 3.2: Graphical representation of single quantum cavity and TLS

Jaynes Cummings Model is obtained by the RWA of the Rabi model in Eq. (3.9) introduced in [19].

$$\mathscr{H}_{JC} = \omega a^{\dagger} a + \frac{\Omega}{2}\sigma_z + g(\sigma_+ a + \sigma_- a^{\dagger}) \tag{3.10}$$

The Hilbert space of the system is spanned by the set of states $\{|n\rangle|g\rangle, |n\rangle|e\rangle\}$ where n is the number of photons and e (g) indicates the atom is in the excited state (ground state).

Due to the RWA Hamiltonian commutes with the total excitation number operator which is given by

$$\mathscr{N}_e = a^\dagger a + \sigma_+\sigma_- \tag{3.11}$$

hence, $[\mathscr{H}, \mathscr{N}_e] = 0$. Thus, the total excitation number is a conserved quantity of the system. The excitation number operator acts on the bare states as

$$\mathscr{N}_e|n,g\rangle = \quad n|n,g\rangle \tag{3.12}$$

$$\mathscr{N}_e|n,e\rangle = \quad n|n,e\rangle + 1|n,e\rangle \tag{3.13}$$

giving the n and $n+1$ excitations respectively. It is only possible to couple the states with the same excitation number i.e. $\{|n,e\rangle, |n+1,g\rangle\}$. Therefore Hamiltonian has a block diagonal structure with 2×2 matrices

$$\mathscr{H}_{JC} = \mathscr{H}_0 \oplus \mathscr{H}_1 \oplus \cdots \oplus \mathscr{H}_N \tag{3.14}$$

each subscript indicating the excitation number for each block.

Hamiltonian of $n+1$ excitation is

$$\mathscr{H}_n = \begin{pmatrix} \frac{\Omega}{2} + \omega n & g\sqrt{n+1} \\ g\sqrt{n+1} & -\frac{\Omega}{2} + \omega(n+1) \end{pmatrix} \tag{3.15}$$

Energy eigenvalues of Eq. (3.15) are

$$\varepsilon_n = (n+\frac{1}{2})\omega \pm \frac{1}{2}\Omega_R \tag{3.16}$$

where $\Omega_R = \sqrt{\delta^2 + 4g^2(n+1)}$ is the Rabi frequency of the atom and $\delta = \Omega - \omega$ detuning frequency of the field [20].

The eigenstates are the dressed states of atom-photon field system

$$|n,+\rangle = \quad \cos\theta_n|n,e\rangle + \sin\theta_n|n+1,g\rangle \tag{3.17}$$

$$|n,-\rangle = \quad -\sin\theta_n|n,e\rangle + \cos\theta_n|n+1,g\rangle \tag{3.18}$$

with the definition of θ_n as

$$\theta_n = \arctan\left(\frac{2g\sqrt{n+1}}{\delta}\right) \tag{3.19}$$

In Fig. 3.3, Rabi oscillations of the TLS in the resonance condition with respect to different coupling strength is shown.

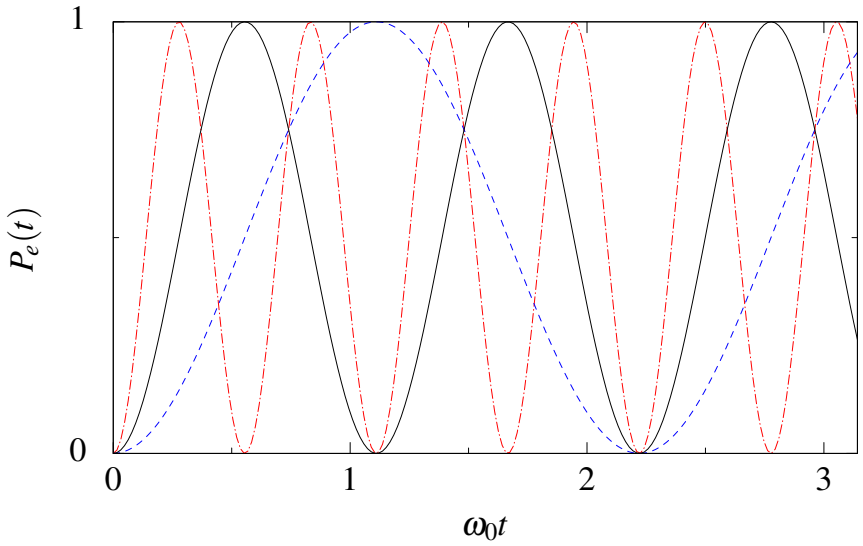

Figure 3.3: Rabi oscillations for different coupling constants on resonance condition ($\delta = \Omega - \omega = 0$). Red dashed dotted line corresponds to $g = 2$, black solid line $g = 1$ and the blue dashed line $g = 0.5$

Rabi frequency is $\Omega_R = 2g\sqrt{n+1}$ with the resonance condition ($\delta = 0$). Therefore, as the coupling increases the frequency of the Rabi oscillations increases.

In Fig. 3.4, change on Rabi oscillations with fixed coupling respect to the detuning δ is shown.

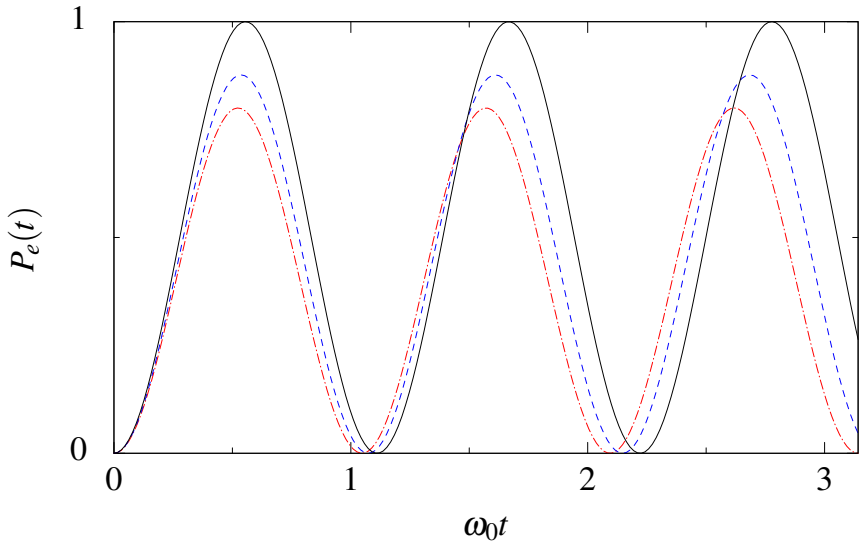

Figure 3.4: Rabi oscillations versus the detuning $\delta = \Omega - \omega$. Red dashed dotted line corresponds to $\delta = 1$, blue dashed line $\delta = 0.75$ and the black solid line $\delta = 0$ (resonance).

As the detuning increases, Rabi frequency increases. In the resonance condition $\delta = 0$, excitation probability of the TLS is at most $P_e = 1$. With the detuning, this probability is lower.

3.2 Coupled Cavity Array (CCA)

Interaction between the TLS and photon field is tunable when they are confined in a cavity. In order to examine many body properties of such systems, coupled cavity arrays is used in experiments. The one we are interested in is the coupled cavity array, one of the cavities is interacting with a TLS. Photon hopping occurs between the nearest neighboring cavities due to the overlap of the cavity modes [4]. The model is depicted in Fig. 3.5.

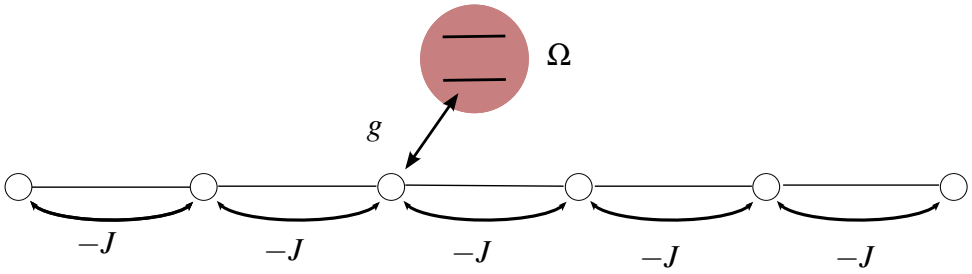

Figure 3.5: Graphical representation of coupled cavity array (CCA) interacting with a TLS.

The model is a real space version of Dicke model in which the TLS interacts with multi-mode cavity radiation field [21]. To capture this point of view, we generalize

the Quantum Rabi model $\mathscr{H}_R$ in Eq. (3.9) including coupled quantum cavity array interactions.

$$\mathscr{H} = \mathscr{H}_F + \mathscr{H}_A + \mathscr{H}_I \tag{3.20}$$

$$= \sum_k \omega_k a_k^\dagger a_k + \frac{\Omega}{2}\sigma_z + \sum_k g_k(\sigma_+ a_k + \sigma_- a_k^\dagger + \sigma_+ a_k^\dagger + \sigma_- a_k) \tag{3.21}$$

where k is the number of modes of the wave inside the cavity. This is known as Dicke Hamiltonian. We do Fourier transform for the real space as

$$a_k^\dagger = \frac{1}{\sqrt{N}} \sum_{i=1}^{N} e^{ikx} a_i^\dagger \tag{3.22}$$

where a is the lattice parameter and $x = ia$. Applying Eq. (3.22) to the Dicke Hamiltonian in Eq. (3.21) we get for the $\mathscr{H}_F$ with the nearest neighbor ($n.n$) interaction

$$\mathscr{H}_F = \frac{1}{N} \sum_k \omega_k \sum_{i,i'=1}^{N} e^{ik(x-x')} a_i^\dagger a_{i'} \tag{3.23}$$

$$= \omega \sum_{i=1}^{N} a_i^\dagger a_i + \sum_{i=1}^{N-1} J_{i,i+1} a_i^\dagger a_{i+1} + J_{i-1,i} a_{i+1}^\dagger a_i \tag{3.24}$$

where $J_{i,i+1} = J_{i-1,i} = \frac{1}{N}\sum_k \omega_k e^{-ika}$ hopping coefficient to the nearest neighbor is the same for both directions and ω_c is the cavity mode. We take $J_{i,i+1} = J_{i-1,i} = -J$ [21].

The interaction term H_I becomes

$$\mathscr{H}_I = \frac{1}{\sqrt{N}} \sum_k \sum_{i=1}^{N} \left(e^{ikx} \sigma_+ a_i^\dagger + e^{-ikx} \sigma_- a_i \right) \tag{3.25}$$

assuming atom (TLS) is in the cavity $i = i_0$ and only interacts with the field mode in this cavity $\mathscr{H}_I$ becomes

$$\mathscr{H}_I = g \left(\sigma_+ a_{i_0} + \sigma_- a_{i_0}^\dagger + \sigma_+ a_{i_0}^\dagger + \sigma_- a_{i_0} \right) \tag{3.26}$$

where $g = \frac{1}{\sqrt{N}} \sum_k g_k e^{\pm ikx_0}$ and $x_0 = i_0 a$.

Therefore, Rabi model with coupled quantum cavities is given by the Hamiltonian

$$\mathscr{H} = \omega \sum_{i=1}^{N} a_i^\dagger a_i - J \sum_{i=1}^{N-1} (a_i a_{i+1}^\dagger + a_i^\dagger a_{i+1}) + \frac{\Omega}{2}\sigma_z + g(\sigma_+ a_{i_0} + \sigma_- a_{i_0}^\dagger + \sigma_- a_{i_0} + \sigma_+ a_{i_0}^\dagger) \tag{3.27}$$

Neglecting the counter-rotating terms (RWA) gives the multi-mode Jaynes Cummings Hamiltonian as

$$\mathscr{H} = \omega \sum_{i=1}^{N} a_i^\dagger a_i - J \sum_{i=1}^{N} (a_{i+1}^\dagger a_i + h.c.) + \frac{\Omega}{2}\sigma_z + g(\sigma_+ a_{i_0} + h.c.) \qquad \textbf{(3.28)}$$

The excitation number operator $\mathscr{N}_e$ on the lattice becomes

$$\mathscr{N}_e = \sum_{i=1}^{N} a_i^\dagger a_i + \sigma_+ \sigma_- \qquad \textbf{(3.29)}$$

satisfies the commutation relation with the Hamiltonian in Eq. (3.28) $[\mathscr{H}, \mathscr{N}_e] = 0$.

The eigenstate of the $\mathscr{N}_e$ operator with zero eigenvalue defines the so-called vacuum state which will serve as a reference state in our numerical calculations.

4. RESULTS

The model Hamiltonian studied for CCA has four parameters. We choose the hopping parameter J as the energy unit and fix the cavity mode energy $\omega = 2J$ and the TLS level spacing $\Omega = 2J$ corresponding to near resonance condition for photons modes in the linear dispersion regime. Scattering simulations have Gaussian pulses centered with momentum in this regime. Accordingly the natural time unit is $\hbar/J$. The TLS is always located in the middle of the chain.

4.1 Ground State of CCA

We have calculated ground state particle distributions for different coupling strengths g with RWA and including CRW terms. The simulation parameters are set as: $N = 61$, $i_0 = 31$, $J = 0.05$, $\Omega = 2$ and $\omega = 2$. This implies near resonance condition. Energy unit in the ground state calculations with $J = 0.05$ is not the hopping parameter.

In Fig. 4.1, particle distributions in the ground state are shown. The physical index dimension is chosen as $d = 5$ in the RWA and $d = 10$ without RWA which are shown in the left and the right panels respectively. Increasing the physical index dimension beyond these values does not change energy and convergence.

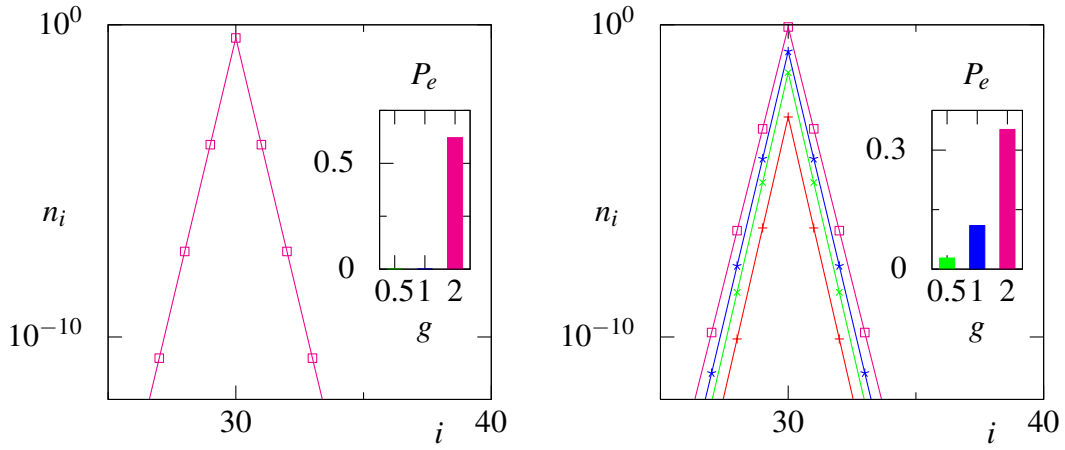

Figure 4.1: Ground state particle distributions for 61 sites with $J = 0.05$. $d = 5$ for various coupling strengths g with RWA in the left panel and $d = 10$ without RWA on the right panel. The insets show the TLS excitation probability with these coupling strengths.

In RWA, TLS-photon make a bound state when $g = 2$ which results in non-zero field localization and excited state occupation. The ground state is the vacuum state for other coupling strengths.

Without the RWA the ground state is a TLS-photon bound state for all coupling strengths. The expected value of number of excitations in the ground state increases with the coupling strength.

We have calculated particle distributions with different coupling strengths for a longer chain with $N = 201$, $i_0 = 101$, $J = 1$, $\Omega = 2$ and $\omega = 2$. These are shown in Fig. 4.2. We have set the physical index dimension $d = 2$ in RWA and $d = 4$ for beyond RWA. These are not the true ground state particle distributions due to the restricted physical index dimension d but are both shown for illustration purposes and used as initial state distribution in scattering simulations to be presented below.

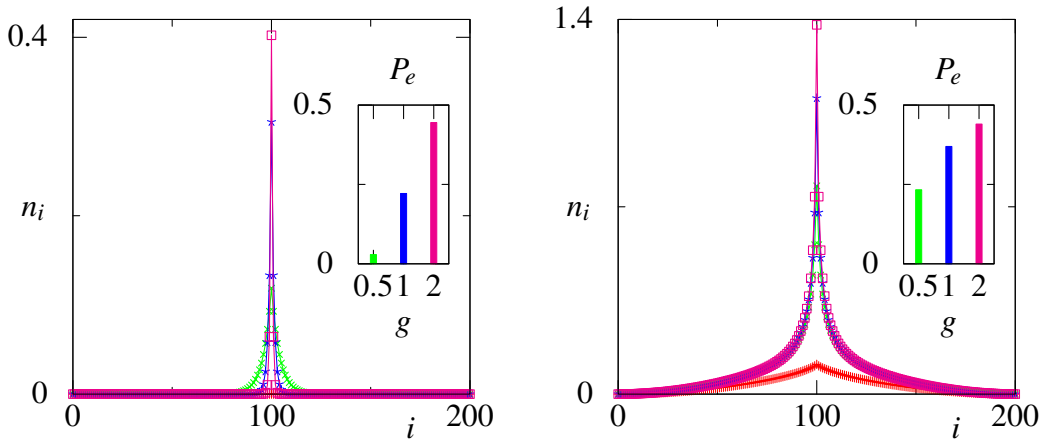

Figure 4.2: Photon distribution in CCA with 201 sites for coupling strengths $g = 0.1, 0.5, 1, 2$ in RWA (left panel) and with CRW terms (right panel). Inset shows the excitation probability of the TLS for corresponding coupling strength.

In the left panel of Fig. 4.2, photon distribution in the minimum energy state found with $d = 2$ with different coupling strengths ($g = 0.1, 0.5, 1, 2$) within RWA are shown. In weak coupling regime ($g = 0.1$), minimum energy state is the vacuum state. Therefore there is no excitation of the TLS and on the chain. At a critical coupling strength ($g \approx 0.5$ for this chain), minimum energy state photon distribution of the system differs from that of the vacuum state. This is a critical point where a quantum phase transition occurs and this critical point is dependent on the size of the system. The TLS excitation is shown in the inset. As coupling gets stronger, excitations on the chain localize around the TLS.

In the right panel of Fig. 4.2, photon distribution beyond RWA with $d = 4$ are shown. The minimum energy state is not vacuum for any of the coupling strengths. As the coupling strength is increased, chain excitations localize around the TLS as in the case of RWA.

4.2 Emission from a Two Level System

In this section we show the results on emission from the TLS which is initially in the excited state at $t = 0$.

Following the work of Lombardo et al. [22], we investigated emission from the TLS as a function of coupling in RWA. We have reproduced the same results with the MPS method. Additionally, we have repeated the same calculations without RWA. Fig. 4.3 shows scape-time graphics for the photon distributions on the chain.

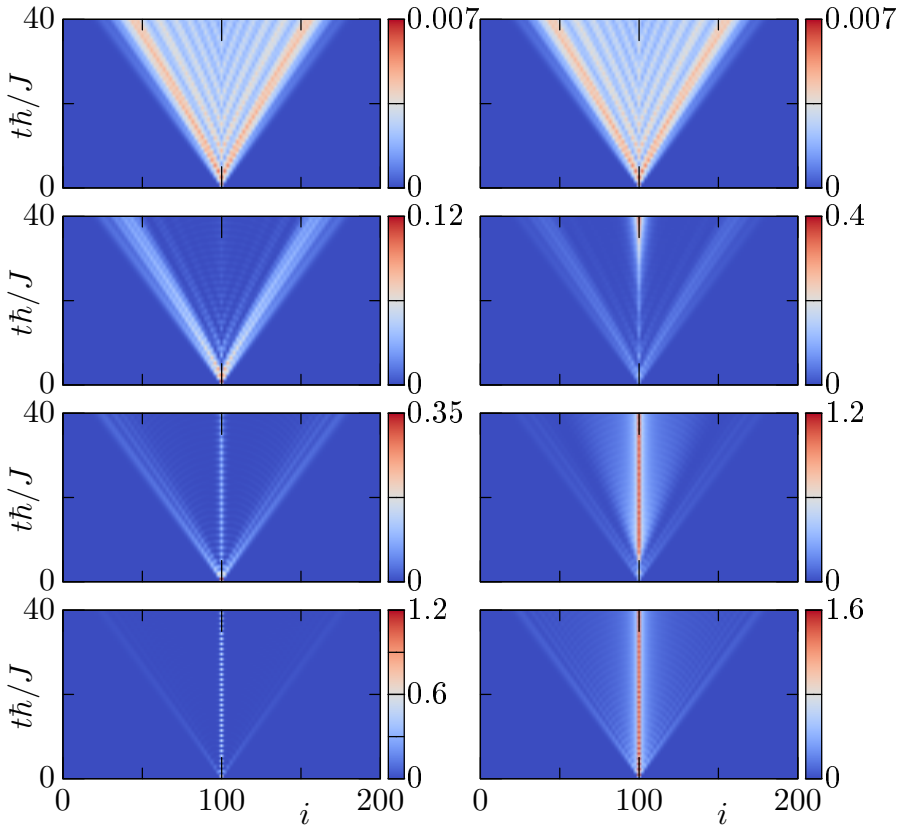

Figure 4.3: Emission from TLS shown as function of space and time within RWA (left panel) and including CRW terms (right panel) for different coupling strengths ($g = 0.1, 0.5, 1, 2$ from top to bottom).

Fig. 4.4 shows TLS probability distributions and photon number expectation values in the cavity in which TLS is located (n_{i_0}).

In Fig. 4.4 and Fig. 4.3, there are three regimes in the emission from the TLS with respect to coupling strength g. For weak coupling $g = 0.1$, TLS does spontaneous emission and

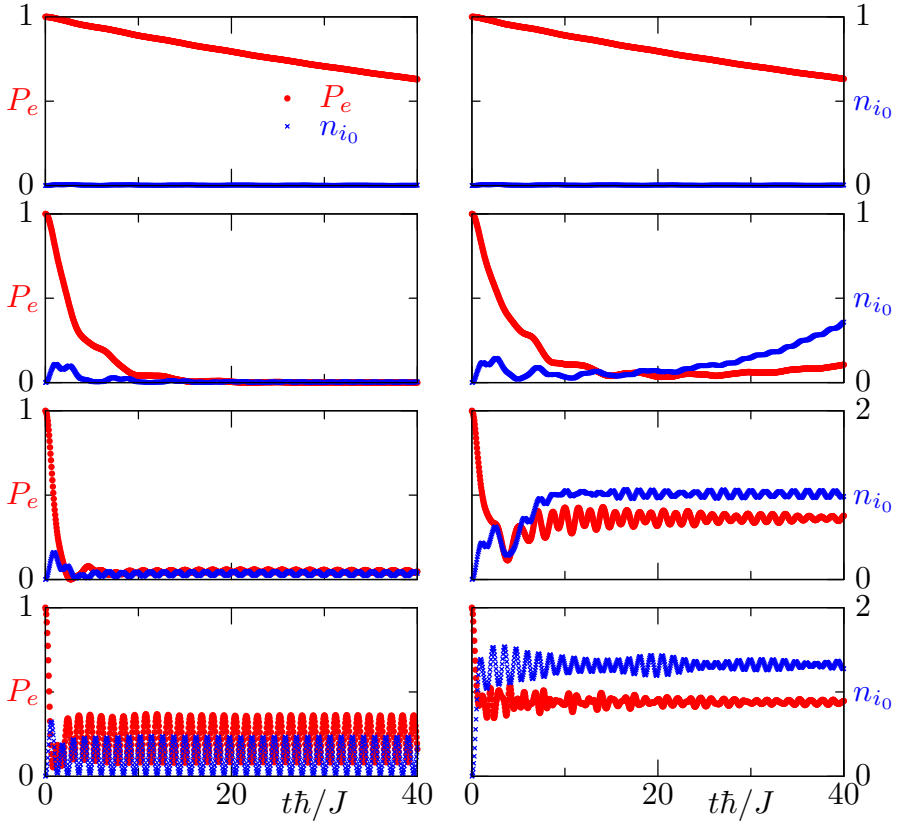

Figure 4.4: Emission dynamics from TLS as a function of time within RWA (left panel) and with CRW terms (right panel) for different coupling strengths ($g = 0.1, 0.5, 1, 2$ from top to bottom). P_e and n_{i_0} denote the excitation probability of the TLS and occupation of the cavity mode on site i_0 respectively.

it emits all of its excitation to the field in long time scale in RWA case as shown in upper left side. This is the exponential decay regime. Emitted photons go away from the TLS. In order to see complete emission from TLS, a longer chain is required. Results are shown up to $t = 40\hbar/J$ to avoid boundary effects. For the coupling strength $g = 0.1$, there is not any significant difference between RWA and non-RWA. For $g = 0.5$, TLS has an oscillating decay in a short time in RWA. Without RWA, some residual excitation remains in the TLS and some field localization starts, but the general emission behavior is similar. There is another decay regime for the coupling $g \approx 1$ where the Rabi oscillations of the TLS begin. This is the effect of a bound state of TLS and photons. Bound state causes the emitted photons to be localized around the TLS. Therefore, TLS can re-absorb and re-emit these photons. For the couplings $g = 1$ and $g = 2$, TLS oscillations never go to zero due to the bound state. When the coupling strength is $g = 2$, photon population is mostly located around TLS. Therefore, this is the third regime in [22], where fractional oscillations occur.

In Fig. 4.3, in weak coupling $g = 0.1$, the emitted photons tend to go away from TLS. We see the Rabi oscillations of the TLS with a big period in $g = 0.1$. As the coupling strength is increased period of the Rabi oscillations gets smaller. Due to the population trapping, TLS does not emit its all excitation to the field. Excitation probability of the TLS is greater beyond RWA for strong coupling strengths. In these cases, the Rabi oscillations have a vanishing amplitude in time.

4.3 Scattering from a Two Level System

We have studied the scattering of a Gaussian wave packet with one and two photons from the TLS in the vacuum state. We have used the same parameters as Longo et al. [21].

One photon Gaussian wave packet prepared with momentum $k = \pi/2$, width $\omega_0 = 9$, centered around $i_c = 35$ is shown in Fig. 4.5. This initial state is created by applying appropriate photon creation operators to the vacuum MPS. The TLS transition frequency is $\Omega = 0.2$ and the coupling constant is $g = 1$. The approximate detuning frequency for the incoming pulse with $k = \pi/2$ is $\Delta = 0.2$. Time evolution is done with 2^{nd} order Suzuki-Trotter expansion with the time step $dt = 0.1$.

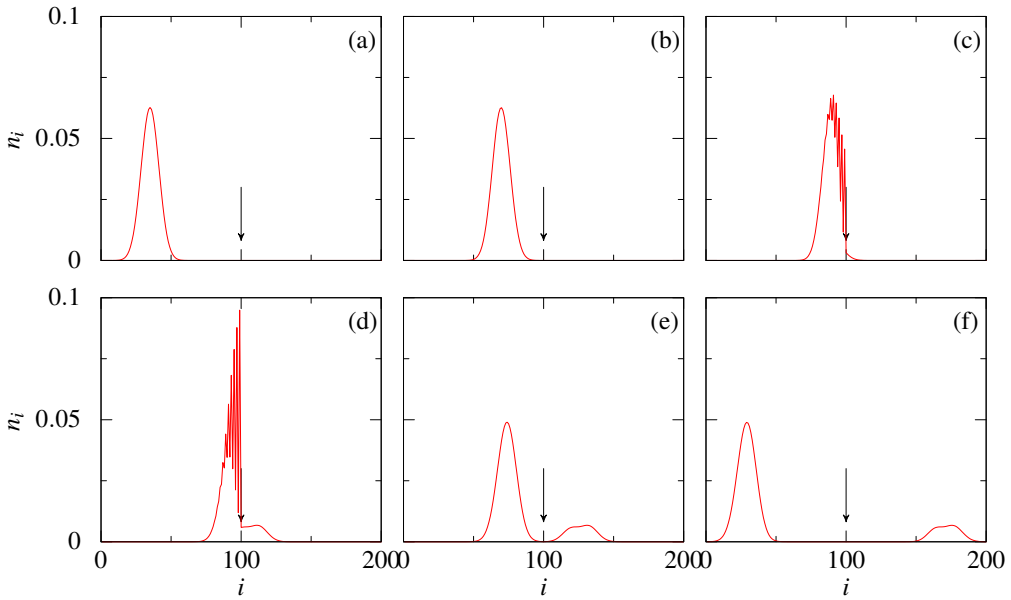

Figure 4.5: Snapshots in time of a Gaussian wave packet with one photon from the TLS. Little black arrow shows the TLS location on the chain. Snapshots are at times $t[\hbar/J]$ = (a) 0, (b) 17.5, (c) 27.5, (d) 37.5, (e) 47.5, (f) 70.

In Fig. 4.6, the incoming, reflected, transmitted pulses are shown as a surface plot in the left panel. In the right panel, TLS excitation probability and expectation value of photon number n_{i_0} are shown. After the scattering, the pulse splits into two parts one of which

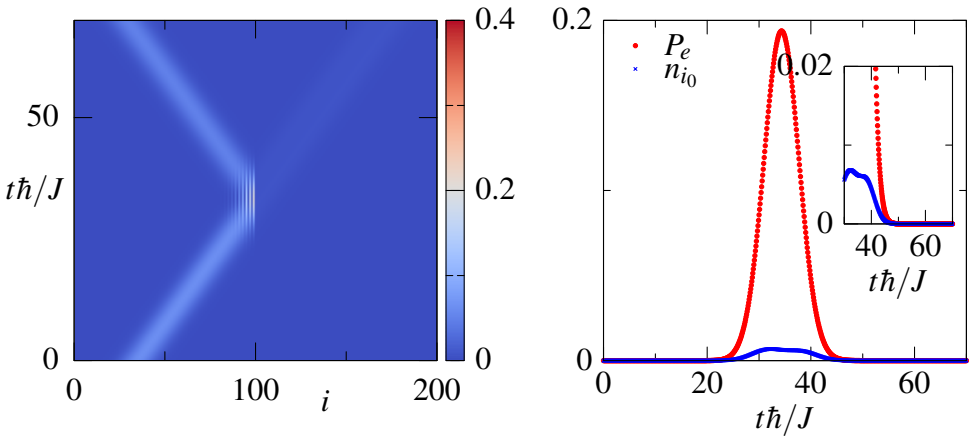

Figure 4.6: Left panel shows photon distribution during the scattering of one photon from the TLS. Right panel shows TLS and TLS-cavity excitation as a function of time.

is the reflected pulse and the other one is the transmitted pulse as seen in Fig. 4.5(d). During the scattering process, TLS absorbs and re-emits photons, so the deformation in the shape of the transmitted pulse in Fig. 4.5.(d)-(e)-(f) is the results of interference effect of directly transmitted pulse and the re-emitted ones. The TLS completely goes to the ground state after scattering. The major part of the pulse is reflected by TLS after the scattering while the transmitted pulse has a small amplitude.

As shown in the left panel of Fig. 4.6, during interaction of the incoming pulse and the TLS, photon population increases around the TLS. After the scattering, field localization and TLS excitation vanish as shown in the right panel of Fig. 4.6.

Scattering of a Gaussian wave packet prepared with two photons is shown in Fig. 4.7. Left panel of Fig. 4.8 shows the transmitted and the reflected pulses where the right panel shows the TLS excitation probability and expectation value of photon number n_{i_0}. In the case of two photon scattering from the TLS, there is a physical phenomenon which was studied by Longo et al. in [23]. After scattering, the photon-atom bound state occurs, therefore a residual occupation remains in the excited state of the TLS as shown in the right panel of Fig. 4.8 and in Fig. 4.7(d)-(f). (The reason of the complete emission from the TLS for one photon scattering is that photon-atom bound states are energetically far away from the dispersion region of the system [23].) The TLS undergoes Rabi oscillations after scattering because of the bound state. Transmitted pulse amplitude is bigger than case of one photon scattering.

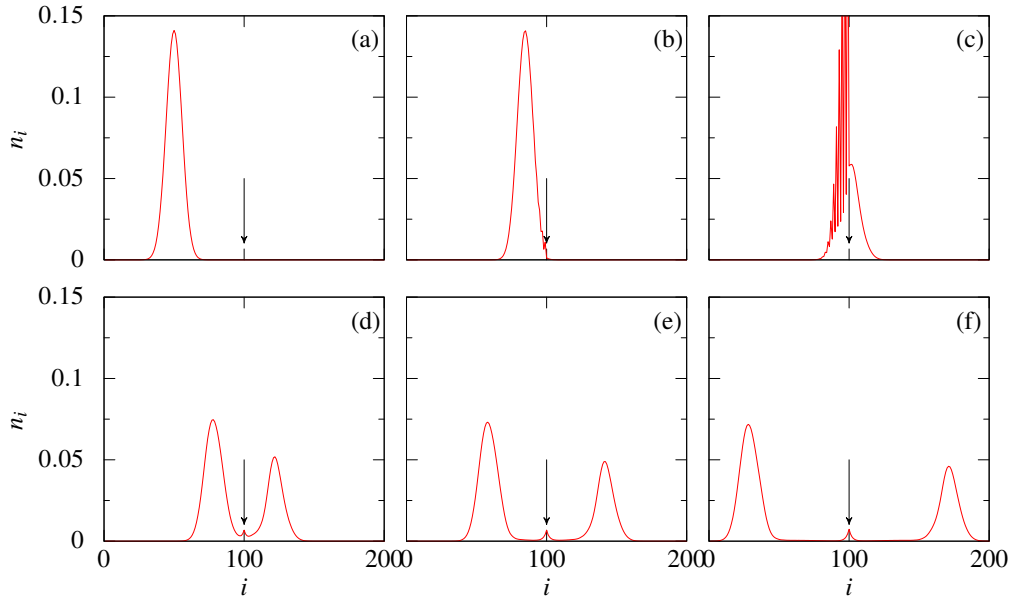

Figure 4.7: Scattering snapshots of the Gaussian with two photons from the TLS. Snapshots are at times $t[\hbar/J] =$ (a) 0, (b) 17.5, (c) 27.5, (d) 37.5, (e) 47.5, (f) 62.5.

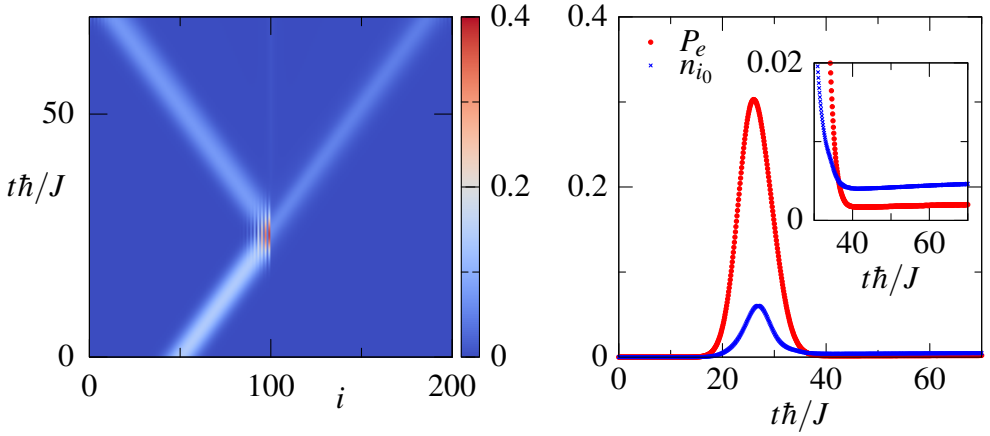

Figure 4.8: Left panel shows photon distribution during the scattering of a two photon pulse from the TLS. Right panel shows TLS and TLS-cavity excitation as a function of time.

In order to examine the effect of the direction of the incoming pulses on the photon-TLS bound state, we studied two incoming pulses from opposite directions to the TLS. The two Gaussian wave packets are prepared with momentum $k^{(1)} = -k^{(2)} = 3\pi/4$, width $\omega_0^{(1)} = \omega_0^{(2)} = 7$, centered around $i_c^{(1)} = 50$ and $i_c^{(2)} = 150$ and each having one photon. The detuning frequency is near zero (resonance) for $|k| = 3\pi/4$. Transition frequency of TLS is $\Omega = \sqrt{2}$ and the coupling constant is $g = 1$. Time evolution is done with 2^{nd} order Suzuki-Trotter expansion with the time step $dt = 0.1$. These parameters are taken from the reference [23]. Fig. 4.9 indicates the initial, reflected and the transmitted wave

packets before and after the scattering. In the left panel of Fig. 4.10 the space-time plot of this event is shown.

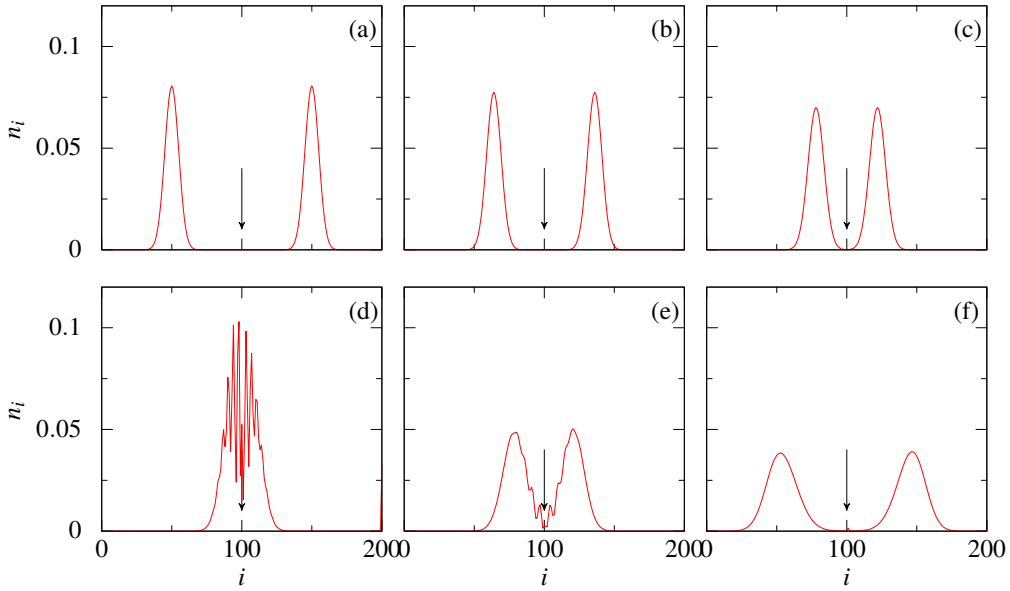

Figure 4.9: Scattering of two Gaussian wave packets from the TLS incoming from different sides. Snapshots are at times $t[\hbar/J] =$ (a) 0, (b) 10 (c) 20, (d) 30, (e) 50, (f) 68.

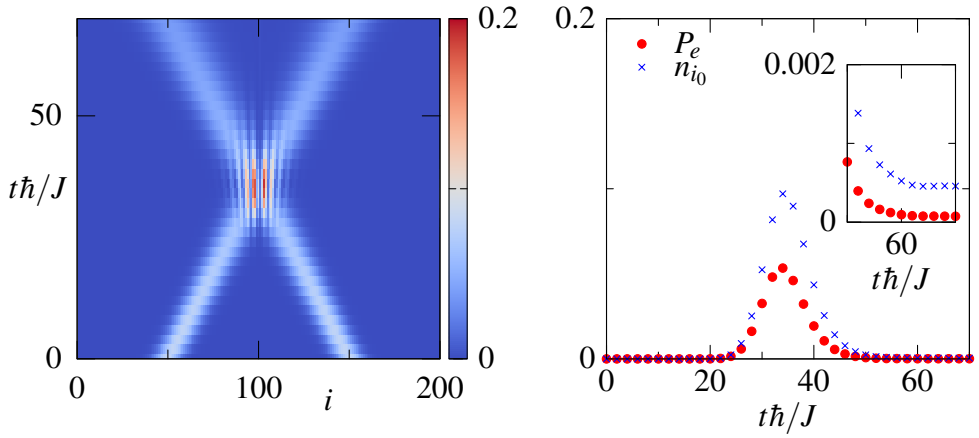

Figure 4.10: Left panel shows photon distribution during the scattering of a two photon pulses incoming from both sides of the TLS. Right panel shows TLS and TLS-cavity excitation as a function of time.

Notice that occupation is not zero around the TLS in Fig. 4.9(e)-(f), because of the photon-atom bound states as before.

Photon occupation on site i_0 and excitation probability of the TLS goes to nearly zero after the scattering as shown in the right panel of Fig. 4.10.

We have investigated the role of the coupling strength in the scattering from the TLS in RWA and beyond. For these simulations, we have taken the minimum energy states as

initial states at $t = 0$. Photon distributions to the minimum energy states was shown in Fig. 4.2.

Fig 4.11 shows surface plots of scattering from the TLS in RWA (left) and beyond (right) for different coupling strengths $g = 0.1, 0.5, 1, 2$. Incoming pulse is a Gaussian with the parameters $k = \pi/2$, $\omega_0 = 9$ and $i_c = 35$. Fig. 4.12 shows the excitation probability of

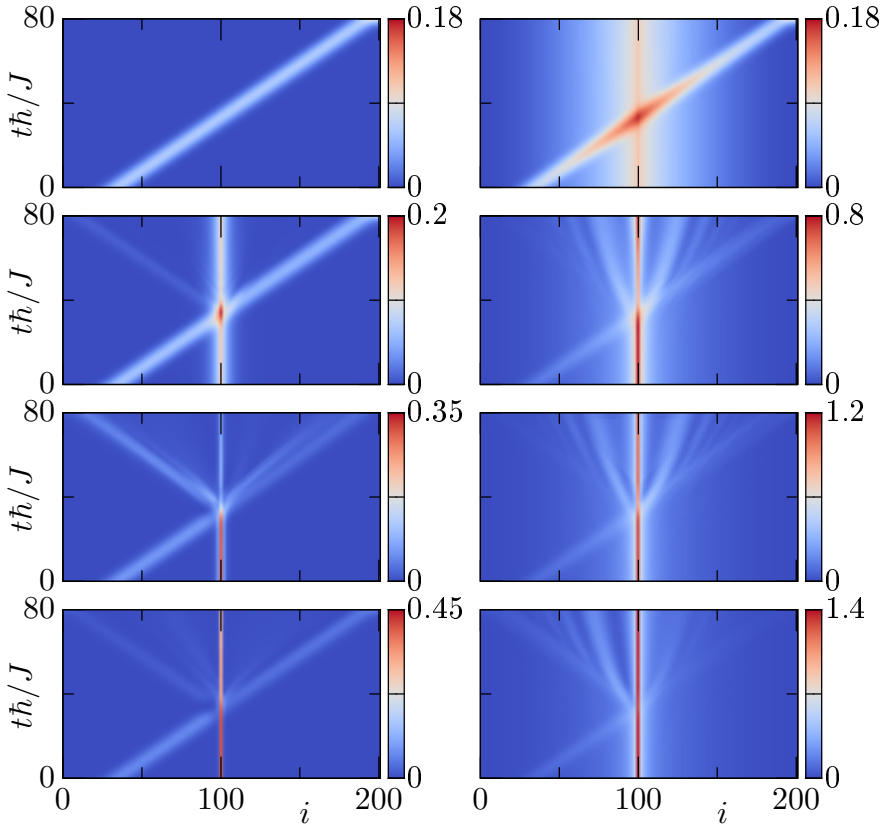

Figure 4.11: One photon scattering from the TLS in RWA (left) and beyond (right) with coupling $g = 0.1, 0.5, 1, 2$ (from top to bottom).

the TLS and photon occupation at site i_0 (n_{i_0}) in time.

In weak coupling ($g = 0.1$), incoming photon and the TLS do not interact noticeably. For $g = 0.5$, they interact weakly. Major part of the incoming pulse is transmitted after the scattering in RWA. Without RWA, this interaction changes the photon distribution on the chain. For $g = 1$, the reflected part of the pulse increases while the TLS excitation clearly decreases in RWA. Beyond RWA, photon distribution on the chain changes but the TLS excitation does not change significantly. For $g = 2$, interaction does not change the field localization around the TLS and TLS excitation significantly.

In Fig. 4.12, for $g = 0.1$, photon population increases during the interaction, but this does not change the excitation probability of the TLS. For $g = 0.5, 1, 2$, the TLS does not

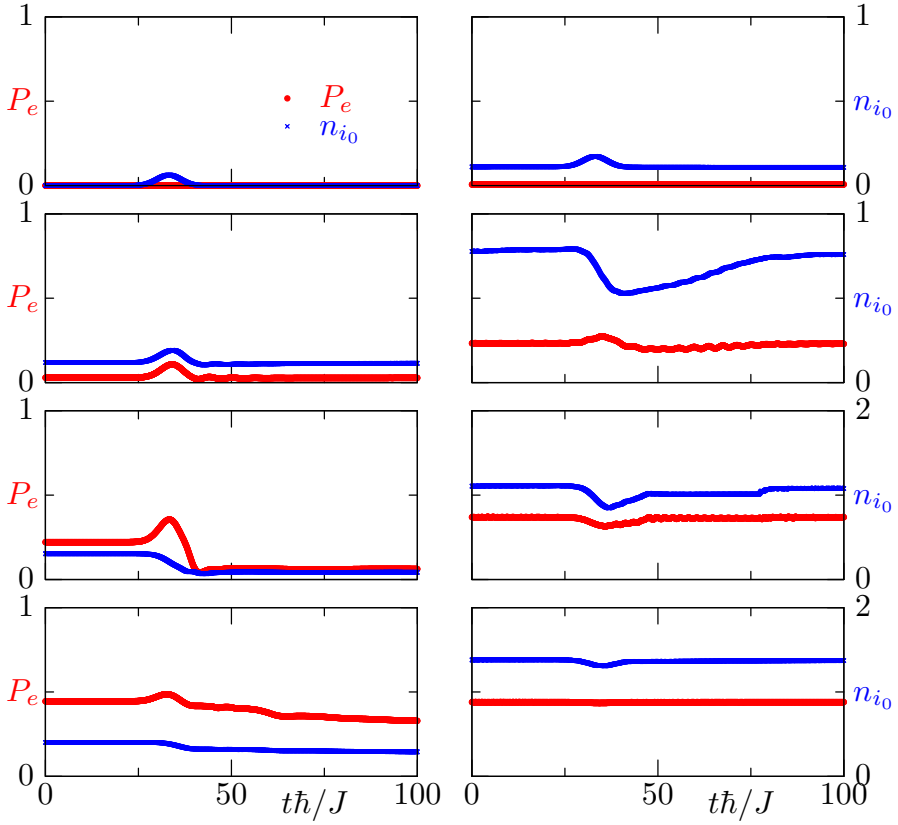

Figure 4.12: TLS excitation probability and photon occupancy on TLS site i_0 as a function of time in RWA (left) and beyond (right) for coupling $g = 0.1, 0.5, 1, 2$ (from top to the bottom).

completely go to the ground state. For $g = 0.5$, the TLS excitation increases during the interaction and it returns to values close to its initial values after the scattering. For $g = 1$ and $g = 2$, the TLS excitation is lower after the scattering in RWA, while beyond RWA it remains around the same excitation probability. For $g = 0.5, 1, 2$, excitation probability increases during interaction and becomes smaller than the initial values in RWA. Beyond RWA, for $g = 0.5$ it increases during the interaction whereas for $g = 1$ it decreases during the interaction. For $g = 2$ it is not affected by the interaction too much.

Fig. 4.13 shows the surface plot of scattering two photon pulse from the TLS in RWA (left panel) and beyond (right panel) for increasing coupling strengths from top to the bottom $g = 0.1, 0.5, 1, 2$. Results are qualitatively similar to one photon scattering.

Fig. 4.14 shows the excitation probability of the TLS and photon occupation at site i_0 (n_{i_0}) in time.

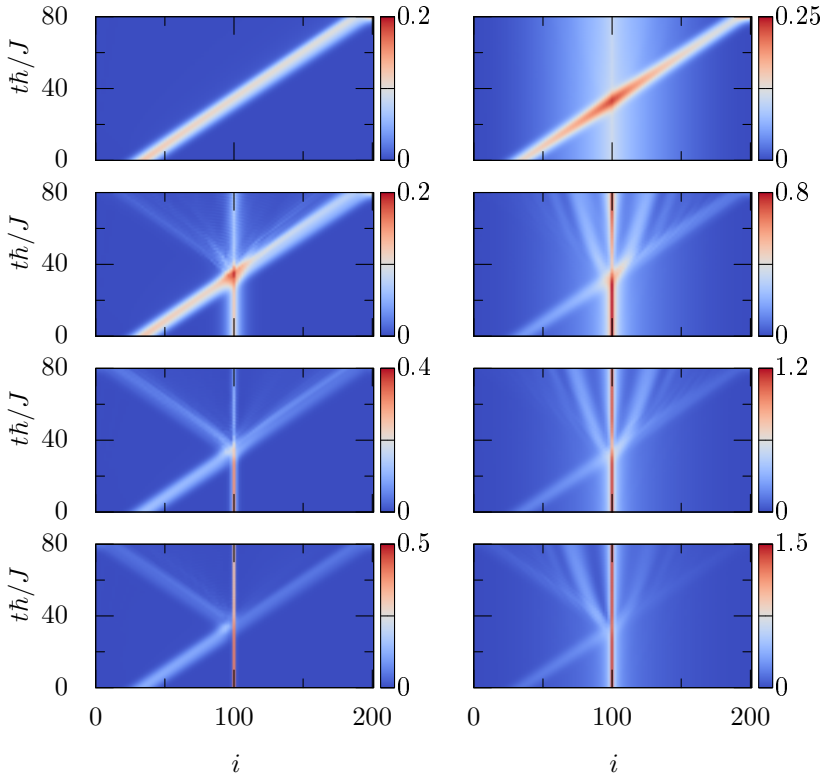

Figure 4.13: Two photon scattering from the TLS in RWA (left) and beyond (right) with coupling $g = 0.1, 0.5, 1, 2$ (from top to bottom).

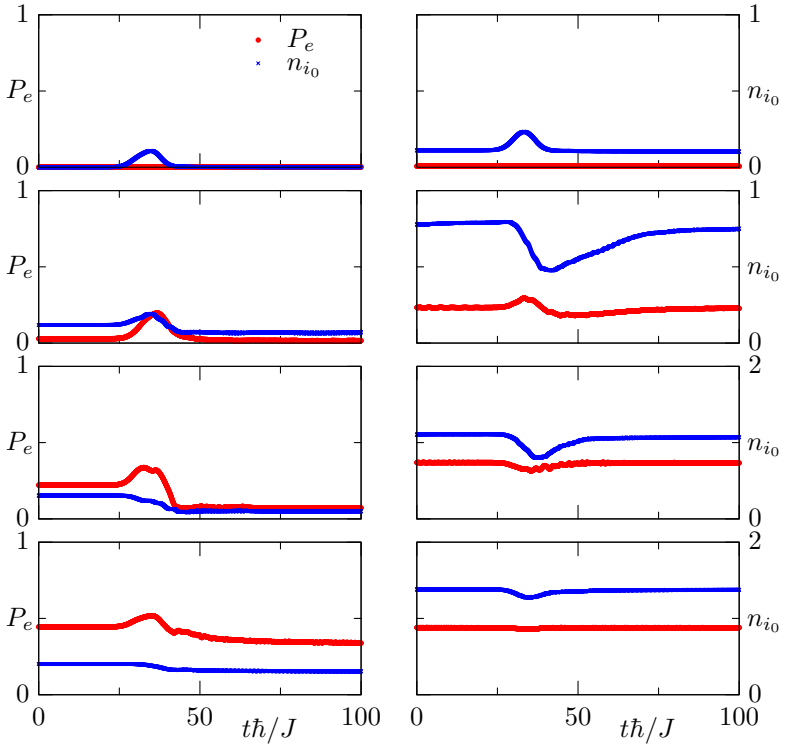

Figure 4.14: TLS excitation probability and photon occupancy on TLS site i_0 as a function of time for two photon pulse in RWA (left) and beyond (right) for coupling $g = 0.1, 0.5, 1, 2$ (from top to the bottom).

5. CONCLUSIONS AND OUTLOOK

In this thesis we have performed numerical calculations on coupled quantum cavity arrays interacting with a two level system using MPSs. We studied time dependent dynamics including emission and scattering from the TLS within RWA and beyond.

Following the work of Lombardo et al. [22] we studied the emission dynamics from a TLS. We confirmed that emission from the TLS has three regimes depending on the coupling strength. The first regime is in the weak coupling limit. In this regime, the TLS decays exponentially. This is the spontaneous emission from the TLS. In the second regime with higher coupling, the TLS undergoes Rabi oscillations while it decays to the ground state. The third regime is in the strong coupling limit where the Rabi oscillations become dominant. These last two regimes result from the photon-atom bound states.

Scattering from the TLS calculations are done by following the references [21] and [23]. The first one is the scattering of a Gaussian wave packet with single photon from a TLS. The second one is the scattering of a Gaussian wave packet with two photon from a TLS. In the second case, we have observed a residual occupation of the excited state in the TLS. This is the effect of photon-atom bound states again. As a third calculation, we have investigated the scattering of two pulses approaching the TLS from different sides resulting in population trapping as before.

We repeated the scattering simulations for different coupling strengths in RWA and including CRW terms starting from the minimum energy state obtained from the ground state calculations.

We are planning to study on coupled quantum cavity arrays interacting with multiple TLSs or a three level system with MPS method in the future.

REFERENCES

[1] **Schiró, M.**, **Bordyuh, M.**, **Öztop, B. and Türeci, H.E.** (2012). Phase Transition of Light in Cavity QED Lattices, *Physical Review Letters*, ***109***(5), 053601.

[2] **Solano, E.** (2011). The dialogue between quantum light and matter, *Physics*, ***4***, 68.

[3] **Rundquist, A.**, **Majumdar, A.**, **Bajcsy, M.**, **Dasika, V.D.**, **Bank, S. and Vuckovic, J.** (2013). Photonic Crystal Coupled Cavity Arrays for Quantum Simulation, *CLEO: 2013*, Optical Society of America, p.CM4F.7.

[4] **Hartmann, M.**, **Brandão, F. and Plenio, M.** (2008). Quantum many-body phenomena in coupled cavity arrays, *Laser & Photonics Reviews*, ***2***(6), 527–556.

[5] **Gelfand, M.P.** (1996). Series expansions for excited states of quantum lattice models, *Solid State Communications*, ***98***(1), 11 – 14.

[6] **Binder, K.** (1995). *The Monte Carlo Method in Condensed Matter Physics*, Springer.

[7] **Wilson, K.G.** (1975). The renormalization group: Critical phenomena and the Kondo problem, *Rev. Mod. Phys.*, ***47***, 773–840.

[8] **Noack, R. and White, S.R.** (1999). The Density Matrix Renormalization Group, *Lecture Notes In Physics*, ***528***, 27–66.

[9] **White, S.R.** (1993). The Density Matrix Algorithms for Quantum Renormalization Groups, *Physical Review B*, ***48***(14), 10345–10356.

[10] **Orus, R.** (2013). A Practical Introduction to Tensor Networks: Matrix Product States and Projected Entangled Pair States, `http://arxiv.org/abs/1306.2164`.

[11] **Jordan, J.**, **Orús, R.**, **Vidal, G.**, **Verstraete, F. and Cirac, J.** (2008). Classical Simulation of Infinite-Size Quantum Lattice Systems in Two Spatial Dimensions, *Phys. Rev. Lett.*, ***101***, 250602.

[12] **Pippan, P.**, **White, S. and Evertz, H.** (2010). Efficient matrix-product state method for periodic boundary conditions, *Phys. Rev. B*, ***81***, 081103.

[13] **Feiguin, A.E.** (2011). The Density Matrix Renormalization Group and its Time Dependent Variants, *AIP Conference Proceedings*, ***1419***, 5–92.

[14] **Eisert, J.**, **Cramer, M. and Plenio, M.B.** (2010). Colloquium: Area laws for the entanglement entropy, *Reviews of Modern Physics*, ***82***, 277–306.

[15] **Sánchez-Burillo, E.**, **Zueco, D.**, **García-Ripoll, J. and Martín-Moreno, L.** (2014). Scattering in the ultrastrong regime: nonlinear optics with one photon, *ArXiv e-prints*.

[16] **Verstraete, F.**, **Murg, V. and Cirac, J.** (2008). Matrix product states, projected entangled pair states, and variational renormalization group methods for quantum spin systems, *Advances in Physics*, ***57***(2), 143–224.

[17] **Scully, M.O. and Zubairy, M.** (1997). *Quantum Optics*, Cambridge University Press, 1 edition.

[18] **Irish, E.** (2007). Generalized Rotating-Wave Approximation for Arbitrarily Large Coupling, *Phys. Rev. Lett.*, ***99***, 173601.

[19] **Jaynes, E. and Cummings, F.** (1963). Comparison of quantum and semiclassical radiation theories with application to the beam maser, *Proceedings of the IEEE*, ***51***(1), 89–109.

[20] **Fox, M.** (2006). *Quantum Optics: An Introduction*, Oxford Master Series in Physics.

[21] **Longo, P.**, **Schmitteckert, P. and Busch, K.** (2009). Dynamics of photon transport through quantum impurities in dispersion-engineered one-dimensional systems, *Journal of Optics A: Pure and Applied Optics*, ***11***(11), 114009.

[22] **Lombardo, F.**, **Ciccarello, F. and Palma, G.M.** (2014). Photon localization versus population trapping in a coupled-cavity array, *Phys. Rev. A*, ***89***, 053826.

[23] **Longo, P.**, **Schmitteckert, P. and Busch, K.** (2011). Few-photon transport in low-dimensional systems, *Physical Review A*, ***83***(6), 063828.

APPENDICES

APPENDIX A: MPS Examples for two and three sites
APPENDIX B: Tensor Contraction Implementation in Matlab/Octave
APPENDIX C: Variational Matrix Product State Method Matlab Code

APPENDIX A: MPS Examples for two and three sites

Two Sites MPS General Formulation

Consider a two site state vector

$$|\psi\rangle = \sum_{i_1,i_2} C_{i_1 i_2} |i_1 i_2\rangle \tag{A.1}$$

where $C_{i_1 i_2}$ is the coefficient matrix. It is a tensor with two physical indices as shown in Fig. A.1.(a). To decompose the $C_{i_1 i_2}$ tensor, we form a matrix C_{i_1,i_2} as in Fig. A.1.(b). We apply SVD for this decomposition as

$$C_{i_1,i_2} = U_{i_1,\alpha} \Lambda_{\alpha,\alpha} V^{\dagger}_{\alpha,i_2} \tag{A.2}$$

and we change the notation to $\psi^{i_1}_{\alpha} := U_{i_1,\alpha}$ and $\psi^{i_2}_{\alpha} := \Lambda_{\alpha,\alpha} V^{\dagger}_{\alpha,i_2}$. We rewrite Eq. (**A.1**) as

$$|\psi\rangle = \sum_{i_1,i_2} \psi^{i_1}_{\alpha} \psi^{i_2}_{\alpha} |i_1 i_2\rangle \tag{A.3}$$

which is shown as graphically in Fig. A.1.(c) where the virtual index (bond index) is shown by the dashed line.

An example for two site wave vector is given in the proceeding discussions.

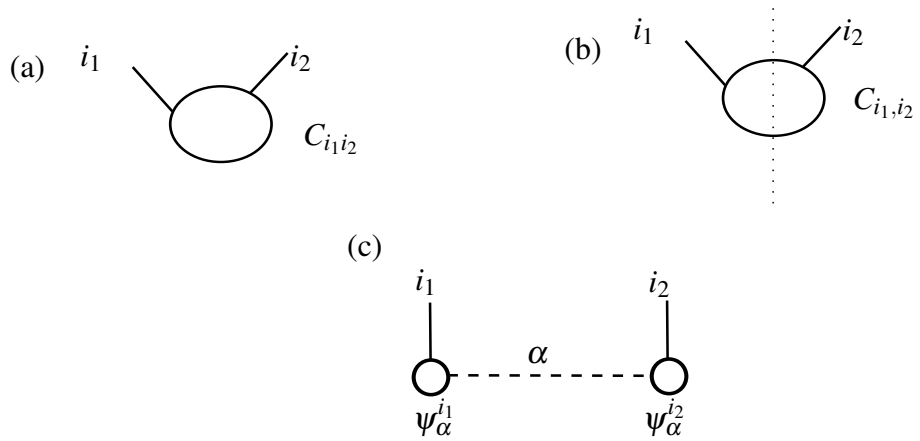

Figure A.1: Graphical description for the derivation of the two sites MPS

Three Sites MPS General Formulation

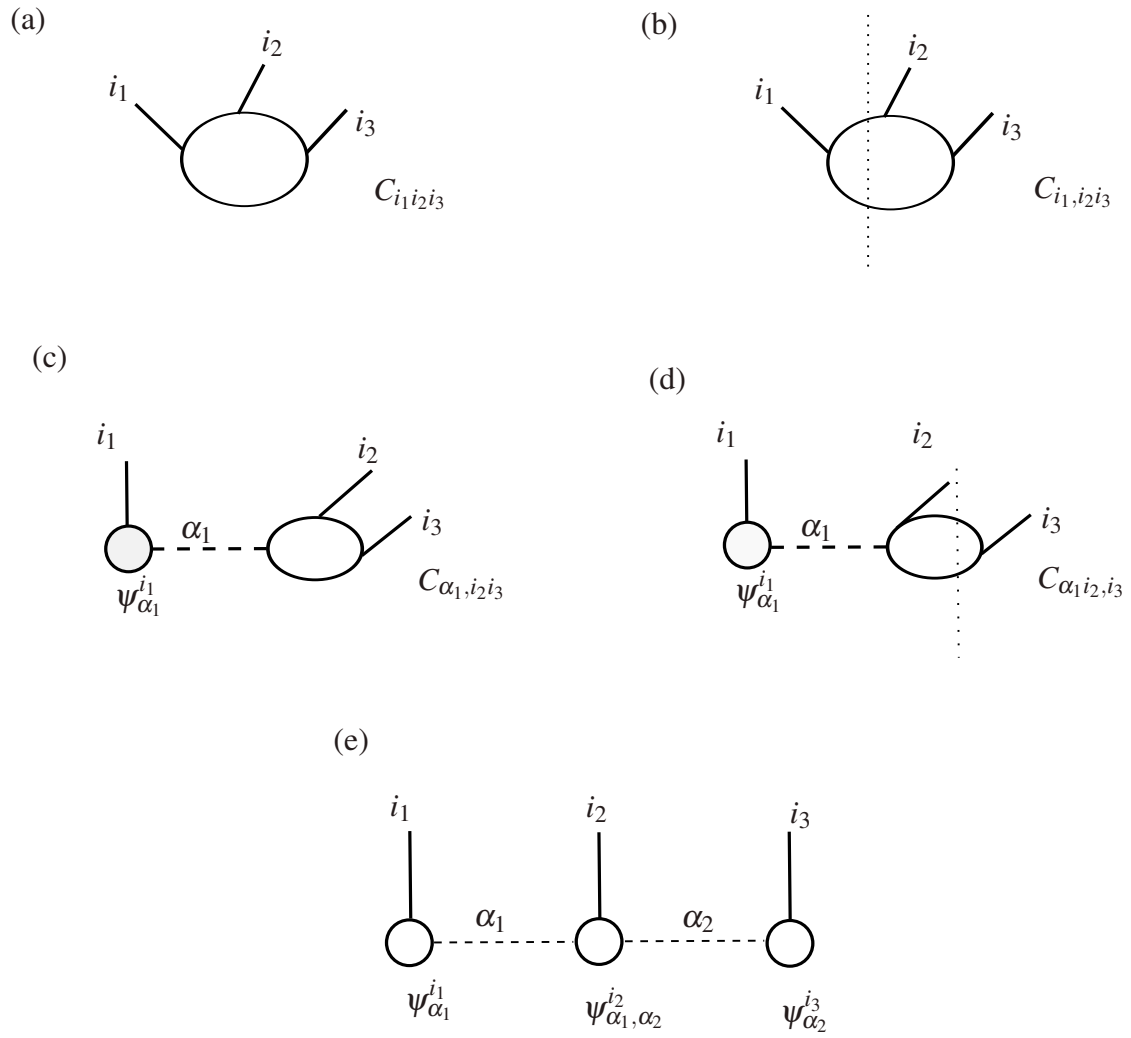

Figure A.2: Graphical description for derivation of three sites MPS

A three site wave vector is given as

$$|\psi\rangle = \sum_{i_1,i_2,i_3} C_{i_1 i_2 i_3} |i_1 i_2 i_3\rangle \tag{A.4}$$

where $C_{i_1 i_2 i_3}$ tensor is depicted in Fig. A.2.(a). We form the matrix $C_{i_1,i_2 i_3}$ where $(i_2 i_3)$ is a combined index, as it is shown in Fig. A.2.(b). We apply SVD to the matrix $C_{i_1,i_2 i_3}$ as

$$C_{i_1,i_2 i_3} = U_{i_1,\alpha_1} \Lambda_{\alpha_1,\alpha_1} V^{\dagger}_{\alpha_1,i_2 i_3} \tag{A.5}$$

and we change the notation to $\psi^{i_1}_{\alpha_1} := U_{\alpha_1,i_1}$ and $C_{\alpha_1,i_2 i_3} := \Lambda_{\alpha_1,\alpha_1} V^{\dagger}_{\alpha_1,i_2 i_3}$. Eq. (**A.4**) is rewritten in terms of the decomposed matrix as

$$|\psi\rangle = \sum_{i_1 i_2 i_3} \psi^{i_1}_{\alpha_1} C_{\alpha_1,i_2 i_3} |i_1 i_2 i_3\rangle \tag{A.6}$$

which is shown in Fig. A.2.(c). The virtual index (bond index, α_1) is shown by the dashed line.

Indices of C_{α_1,i_2i_3} are separated by the new decomposition $C_{\alpha_1 i_2,i_3}$ where $(\alpha_1 i_2)$ is a combined index. This is shown in Fig. A.2.(d). We apply SVD for the new matrix as

$$C_{\alpha_1 i_2,i_3} = U_{\alpha_1 i_2,\alpha_2}\Lambda_{\alpha_2,\alpha_2}V^{\dagger}_{\alpha_2,i_3} \tag{A.7}$$

Changing the notation to $\psi^{i_2}_{\alpha_1\alpha_2} := U^{i_2}_{\alpha_1,\alpha_2}$ and $\psi^{i_3}_{\alpha_2} := \Lambda_{\alpha_2,\alpha_2}V^{\dagger}_{\alpha_2,i_3}$, we rewrite Eq. (**A.6**) as

$$|\psi\rangle = \sum_{i_1,i_2,i_3} \psi^{i_1}_{\alpha_1}\psi^{i_2}_{\alpha_1\alpha_2}\psi^{i_3}_{\alpha_2}|i_1 i_2 i_3\rangle \tag{A.8}$$

which is shown graphically in Fig. A.2.(e). Note here that $\psi^{i_1}_{\alpha_1}$ and $\psi^{i_3}_{\alpha_2}$ which are the first and the last site tensors have only one virtual index (bond index). This is the case of open boundary conditions as mentioned before in Chapter 2.

A three site MPS example is given in the proceeding examples.

Simple MPS Derivation Examples

Here, we give some two and three sites examples of MPS with $D = 1$ and $D = 2$ bond dimensions.

Pure State

Pure state is the most simple exercise with $D = 1$ meaning that there is no entanglement.

Consider the product state of $i_n = \{\uparrow\downarrow\}$, $N = 3$ particle state

$$\begin{aligned} |\Psi\rangle &= |\downarrow\downarrow\uparrow\rangle \\ &= \sum_{i_1,i_2,i_3} \psi^{i_1}\psi^{i_2}\psi^{i_3}|i_1 i_2 i_3\rangle \\ &= \psi^{\downarrow}\psi^{\downarrow}\psi^{\uparrow}|\downarrow\downarrow\uparrow\rangle \end{aligned}$$

in terms of product of matrices. Since there is not any contraction index of the tensors, they are just numbers and all $\psi^{i_n} = \{1,0\}$.

Entangled States

Spin singlet state and GHZ are the entangled states with the bond dimension $D = 2$.

(i) Singlet state

$$|\Psi\rangle = \frac{1}{\sqrt{2}}(|\uparrow\downarrow\rangle - |\downarrow\uparrow\rangle) \tag{A.9}$$

which can be written in the MPS form as

$$|\Psi\rangle = \sum_{i_1,i_2=\{\uparrow,\downarrow\}} \psi^{i_1}_{\alpha}\psi^{i_2}_{\alpha}|i_1 i_2\rangle$$

in which α is the contraction index, hence $\psi_\alpha^{i_n}$ are rank 1 tensors (vectors). Apparently we can choose the vectors as

site index	1	2
$\psi_\alpha^{\uparrow}$	$(1 \quad 0)$	$-(0 \quad 1)^\intercal/\sqrt{2}$
$\psi_\alpha^{\downarrow}$	$(0 \quad 1)$	$(1 \quad 0)^\intercal/\sqrt{2}$

Table A.1: Singlet state vectors

Now, let us verify the Table A.1 by the SVD method. Firstly, Eq. (**A.9**) is written in the general form by means of the local physical variables (i_1, i_2).

$$|\Psi\rangle = \sum_{i_1,i_2} C_{i_1 i_2} |i_1, i_2\rangle \tag{A.10}$$

where the coefficient matrix is

$$C_{i_1,i_2} = \frac{1}{\sqrt{2}} \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \tag{A.11}$$

with the consistency in Eq. (**A.9**). SVD decomposition is done as $C_{i_1,i_2} = U_{i_1,\alpha_1} \Lambda_{\alpha_1,\alpha_1} V^\dagger_{\alpha_1,i_2}$

where Λ matrix is formed by the square roots of the eigenvalues of the matrix $A = \psi^\dagger_{i_1,i_2} \psi_{i_1,i_2}$, which is

$$\Lambda = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{A.12}$$

and V matrix is formed by the eigenvectors of the A as in the following

$$V = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \tag{A.13}$$

and

$$U = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \tag{A.14}$$

where $U_{i_1,\alpha} = C_{i_1,i_2} V_{i_2,\alpha'} \Lambda^\dagger_{\alpha',\alpha}$.

Now, let us re-write Eq. (**A.10**) in terms of U, Λ, V matrices

$$|\Psi\rangle = \sum_{i_1,i_2} U_{i_1,\alpha_1} \Lambda_{\alpha_1,\alpha_1} V^\dagger_{\alpha_1,i_2} |i_1, i_2\rangle \tag{A.15}$$

$$= \sum_{i_1,i_2} \psi^{i_1}_{\alpha_1} \psi^{i_2}_{\alpha_1} |i_1, i_2\rangle \tag{A.16}$$

where $\psi^{i_2}_{\alpha_1} := \Lambda_{\alpha_1,\alpha_1} V^\dagger_{\alpha_1,i_2}$. It is easy to check if the matrix elements give the coefficients in Eq. (**A.9**).

(ii) GHZ state with $N = 3$

$$\begin{aligned} |GHZ\rangle &= \frac{1}{\sqrt{2}}(|\uparrow\uparrow\uparrow\rangle + |\downarrow\downarrow\downarrow\rangle) \\ &= \sum_{\{i\}} C_{i_1 i_2 i_3} |i_1 i_2 i_3\rangle \\ &= \sum_{\{i\}} \psi^{i_1}_{\alpha_1} \psi^{i_2}_{\alpha_1 \alpha_2} \psi^{i_3}_{\alpha_2} |i_1 i_2 i_3\rangle \end{aligned}$$

Note here that the first (ψ_{α_1}) and the last tensors (ψ_{α_2}) have one index while the tensor in the middle has two indices. Now, let us find the $\psi^{i_n}_{\alpha_{n-1}\alpha_n}$ tensors in MPS by the SVD decomposition.

The first decomposition is done to the matrix $C_{i_1,i_2 i_3}$ in which one of the sites is on left side the remaining two are on the right side.

$$C_{i_1,i_2 i_3} = \begin{pmatrix} 1/\sqrt{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & -1/\sqrt{2} \end{pmatrix} \tag{A.17}$$

where non-zero elements are the coefficients of the states $\uparrow, \uparrow\uparrow$ and $\downarrow, \downarrow\downarrow$. By the SVD decomposition $C_{i_1,i_2 i_3} = U_{i_1,\alpha_1} \Lambda_{\alpha_1 \alpha_1} V^{\dagger}_{\alpha_1,i_1 i_2}$. It is simple to find the relevant matrices as

$$U_{i_1,\alpha_1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{A.18}$$

which is equal to the local tensor of the first site $\psi^{i_1}_{\alpha_1}$, and

$$\Lambda_{\alpha_1,\alpha_1} = \begin{pmatrix} 1/\sqrt{2} & 0 & 0 & 0 \\ 0 & 1/\sqrt{2} & 0 & 0 \end{pmatrix} \tag{A.19}$$

and finally

$$V_{\alpha_1,i_2 i_3} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \end{pmatrix} \tag{A.20}$$

In Eq. (**A.20**) the first two rows corresponds to the non-zero singular values which are $(\alpha_1)_{1,2} = 1/\sqrt{2}$ and the last two rows correspond to the zero singular values of the $\Lambda_{\alpha_1,\alpha_1}$ matrix.

After the first SVD decomposition we obtain the matrix $\psi_{\alpha_1,i_2 i_3} = \Lambda_{\alpha_1,\alpha_1} V^{\dagger}_{\alpha_1,i_2 i_3}$ for the right side of the chain, that is

$$\psi_{\alpha_1,i_2 i_3} = \begin{pmatrix} 1/\sqrt{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & -1/\sqrt{2} \end{pmatrix} \tag{A.21}$$

For the new decomposition, one need to obtain the matrix

$$\psi_{\alpha_1 i_2, i_3} = \begin{pmatrix} 1/\sqrt{2} & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & -1/\sqrt{2} \end{pmatrix} \tag{A.22}$$

where the non-zero elements correspond to $\{\lambda_{\alpha_1} i_2, i_3\} = 1/\sqrt{2}\uparrow,\uparrow$ and $\{\lambda_{\alpha_1} i_2, i_3\} = 1/\sqrt{2}\downarrow,\downarrow$ respectively. One do the SVD decomposition as $\psi_{\alpha_1 i_2, i_3} = U_{\alpha_1 i_2, \alpha_2} \Lambda_{\alpha_2, \alpha_2} V^{\dagger}_{\alpha_2, i_3}$ where the relevant matrices are

$$U_{\alpha_1 i_2, \alpha_2} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \tag{A.23}$$

which is equal to the local tensor $\psi^{i_2}_{\alpha_1 \alpha_2}$,

$$\Lambda_{\alpha_2, \alpha_2} = \begin{pmatrix} 1/\sqrt{2} & 0 \\ 0 & 1/\sqrt{2} \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \tag{A.24}$$

and

$$V_{\alpha_2, i_3} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \tag{A.25}$$

so that the local site tensor for the last site is $\psi_{\alpha_2, i_3} = \Lambda_{\alpha_2, \alpha_2} V^{\dagger}_{\alpha_2, i_3}$.

Notice from the Equations (**A.19**-**A.25**) that one could obtain the same results just keeping the only non-zero singular values and the corresponding columns of the V matrices.

APPENDIX B: Tensor Contraction Implementation in Matlab/Octave

Consider the tensors $X_{\alpha_1\alpha_2\cdots\alpha_N}$ with N indices and $Y_{\beta_1\beta_2\cdots\beta_M}$ with M indices. We would like to contract the set of mixed indices $\alpha_i\cdots\alpha_j$ of X and $\beta_k\cdots\beta_l$ of Y. Here it is explained how to contract these specific indices of X and Y in Eq. (**A.1**) in MATLAB in a few steps:

$$X_{\alpha_1\alpha_2\cdots\alpha_N}Y_{\beta_1\beta_2\cdots\beta_N} \tag{A.1}$$

where $\alpha_i\cdots\alpha_j \in \{\alpha_1\alpha_2\cdots\alpha_N\}$ and $\beta_k\cdots\beta_l \in \{\beta_1\beta_2\cdots\beta_M\}$.

1. Indices of X and Y are permuted by the command *permute*[1] in such a way that contraction indices of X, $\alpha_i\cdots\alpha_j$, are the rightmost indices of X and the contraction indices of Y, $\beta_k\cdots\beta_l$, are the leftmost indices of Y.

$$X_{\alpha_1\cdots\alpha_N\alpha_i\cdots\alpha_j}Y_{\beta_k\cdots\beta_l\beta_1\cdots\beta_M}$$

2. We make *combined index* from the contraction indices and non-contracted ones of X and Y as

$$X_{\alpha_1\cdots\alpha_N,\alpha_i\cdots\alpha_j}$$
$$Y_{\beta_k\cdots\beta_l,\beta_1\cdots\beta_M}$$

   by the *reshape*[2] command. Here, $\alpha_1\cdots\alpha_N$ and $\alpha_i\cdots\alpha_j$ are combined indices of X and $\beta_k\cdots\beta_l$ and $\beta_1\cdots\beta_M$ are combined indices of Y.

3. Do the matrix multiplication as in the following:

$$Z_{\alpha_1\cdots\alpha_N,\beta_1\cdots\beta_M} = X_{\alpha_1\cdots\alpha_N,\alpha_i\cdots\alpha_j}\delta_{\alpha_i,\beta_k}\cdots\delta_{\alpha_j,\beta_l}Y_{\beta_k\cdots\beta_l,\beta_1\cdots\beta_M} \tag{A.2}$$

   In Eq. (**A.2**) kronecker delta $\delta_{\alpha_i,\beta_k}$ is inserted to indicate that the column and row indices of the X and Y tensors have to be matched. Note that $\alpha_1\cdots\alpha_N$ and $\beta_1\cdots\beta_M$ do not contain $\alpha_i\cdots\alpha_j$ and $\beta_k\cdots\beta_l$ respectively anymore.

4. A new tensor can be set from the matrix $Z_{\alpha_1\cdots\alpha_N\beta_1\cdots\beta_M} := Z_{\alpha_1\cdots\alpha_N,\beta_1\cdots\beta_M}$ by the command *reshape*.

[1] `http://www.mathworks.com/help/matlab/ref/permute.html`
[2] `http://www.mathworks.com/help/matlab/ref/reshape.html`

Tensor Contraction Examples

In this section, the application an operator in the MPO form to a quantum state in the MPS form as in Eq. (2.24) is indicated. For this aim, consider we have chosen an operator as

$$\mathscr{O} = \mathscr{O}_1 \otimes \mathscr{O}_2 \tag{A.3}$$

where the $\mathscr{O}_1$ and $\mathscr{O}_2$ are applied to the first and the second sites respectively of a two sites chain. Hence, the general form of this quantum mechanical operator is of the form in Eq. (2.17).

We want to apply MPO form of Eq. (**A.3**) to the MPS form of a two site state as in Eq. (**A.3**). To do this, we need to find MPO representation for the Eq. (**A.3**) by applying SVD decomposition.

Coefficient matrix of Eq. (**A.3**) is $O_{i_1 i_2, j_1 j_2} = \langle i_1 i_2 | \mathscr{O} | j_1 j_2 \rangle$. As mentioned in Chapter 2, in order to do the SVD decomposition to the $O_{i_1 i_2, j_1 j_2}$ matrix in Eq. (**A.3**), it must be reshaped as $\tilde{O}_{i_1 j_1, i_2 j_2}$. Therefore, the indices i_2, j_1 change the places. To be more explicit, we write the matrix elements of the $\tilde{O}_{i_1 j_1, i_2 j_2}$ in terms of the $O_{i_1 i_2, j_1 j_2}$:

$$\tilde{O}_{i_1 j_1, i_2 j_2} = \begin{pmatrix} O_{11,11} & O_{11,12} & O_{21,11} & O_{12,12} \\ O_{11,21} & O_{11,22} & O_{12,21} & O_{12,22} \\ O_{21,11} & O_{21,12} & O_{22,11} & O_{22,12} \\ O_{21,21} & O_{21,22} & O_{22,21} & O_{22,22} \end{pmatrix} \tag{A.4}$$

Here, the practical way to obtain the MPO form for the Eq. (**A.3**) in MATLAB after obtaining the Eq. (**A.4**) is given in a few steps:

- $\tilde{O}_{i_1 i_2, j_1 j_2}$ matrix in Eq. (**A.4**) is converted to a tensor by the command $reshape(O_{i_1 i_2, j_1 j_2}, [d \quad d \quad d \quad d])$, whose each index has the local site Hilbert space dimension, d. This is the tensor $O_{i_1 i_2 j_1 j_2}$.
- The permutation procedure for the indices is done by the command $permute(O_{i_1 i_2 j_1 j_2}, [1 \quad 3 \quad 2 \quad 4])$ so that the indices $i_2 \leftrightarrow j_1$ change places. This gives the tensor $O_{i_1 j_1 i_2 j_2}$.
- We need to make it a matrix as in the Eq. (**A.4**). The command $reshape(O_{i_1 j_1 i_2 j_2}[d * d, d * d])$ gives the desired matrix $\tilde{O}_{i_1 j_1, i_2 j_2}$ in Eq. (**A.4**).
- Eq. (**A.4**) is of the form which we want to apply SVD decomposition to find the local MPO tensors.

The SVD decomposition of the $\tilde{O}_{i_1 j_1, i_2 j_2}$ is

$$\mathscr{O} = \sum_{\{i\},\{j\}} \tilde{O}_{i_1 j_1, i_2 j_2} |i_1 i_2\rangle \langle j_1 j_2| \tag{A.5}$$

$$= \sum_{\{i\},\{j\}} U_{i_1 j_1, \gamma} \Lambda_{\gamma,\gamma} V^{\dagger}_{\gamma, i_2 j_2} |i_1 i_2\rangle \langle j_1 j_2| \tag{A.6}$$

$$= \sum_{\{i\},\{j\}} O^{i_1 j_1}_{\gamma} O^{i_1 j_2}_{\gamma} |i_1 i_2\rangle \langle j_1 j_2| \tag{A.7}$$

where $O^{i_n j_n} = \langle i_n|\mathscr{O}|j_n\rangle$ is the coefficient matrix and $O_\gamma^{i_1 j_1} := U_{i_1 j_1,\gamma}$ and $O_\gamma^{i_2 j_2} := \Lambda_{\gamma,\gamma} V^\dagger_{\gamma,i_2 j_2}$.

A two site MPS is of the form

$$|\Psi\rangle = \sum_{i_1,i_2} \psi_{\alpha_1}^{i_1} \psi_{\alpha_1}^{i_2} |i_1 i_2\rangle \tag{A.8}$$

We apply the MPO to the MPS as in the following:

$$\begin{aligned} |\Phi\rangle &= \hat{\hat{O}}|\Psi\rangle \\ &= \sum_{\{i\}} O_\gamma^{i_1 j_1} \psi_\alpha^{j_1} O_\gamma^{i_2 j_2} \psi_\alpha^{j_2} |j_1 j_2\rangle \end{aligned} \tag{A.9}$$

Now, we turn back to the tensor notation for MPOs $O_{i_1 j_1 \gamma} := O_\gamma^{i_1 j_1}$ and $O_{\gamma i_2 j_2} := O_\gamma^{i_2 j_2}$ and for MPSs $\psi_{j_1 \alpha} := \psi_\alpha^{j_1}$ and $\psi_{\alpha, j_2} := \psi_\alpha^{j_2}$. Contraction is done by following the steps written above.

Eq. (**A.9**) becomes

$$|\Phi\rangle = \sum_{i_1,i_2} O_{i_1\gamma,j_1} \psi_{j_1,\alpha} O_{i_2\gamma,j_2} \psi_{j_2,\alpha} |j_1 j_2\rangle \tag{A.10}$$

$$= \sum_{i_1,i_2} \phi_{i_1\gamma,\alpha} \phi_{i_2\gamma,\alpha} |i_1 i_2\rangle \tag{A.11}$$

where $\phi_{i_1\gamma,\alpha} := O_{i_1\gamma,j_1}\psi_{j_1,\alpha}$ and $\phi_{i_2\gamma,\alpha} := O_{i_2\gamma,j_2}\psi_{j_2,\alpha}$. We rewrite the above expression by writing the physical indices as superscript as

$$|\Phi\rangle = \sum_{i_1,i_2} \phi_\eta^{i_1} \phi_\eta^{i_2} |i_1 i_2\rangle \tag{A.12}$$

where $\eta = \gamma\alpha$ combined index. Notice that the bond dimension gets bigger when we apply an MPO to the MPS.

We give some examples for Eq. (**A.12**) in this formulation with two site operators and states as in the following:

1. one body operator

$$\mathscr{O} = \sigma_x \otimes \mathbb{I}$$

This operator has a bond dimension one ($rank(\Lambda) = 1$). Therefore, it is already in the matrix product form. Here, we apply this operator to a product state in (a) and the singlet state in (b).

(a) $|\psi\rangle = |\uparrow\rangle \otimes |\downarrow\rangle$

Bond dimension of a product state is one ($rank(\Lambda) = 1$). When we apply $\mathscr{O}$ to the $|\psi\rangle$ Eq. (**A.12**) is

$$\phi = |\downarrow\rangle \otimes |\downarrow\rangle$$

where the bond dimension η is equal to one.

(b) $|\psi\rangle = \frac{1}{\sqrt{2}}(|\uparrow\rangle \otimes |\downarrow\rangle - |\downarrow\rangle \otimes |\uparrow\rangle)$

Bond dimension of the singlet state is two, since there is two non-zero singular value of Λ matrix. Since $\mathscr{O}$ matrix has a bond dimension one, $\mathscr{O}|\psi\rangle$ operation does not raise the bond dimension. Hence,

$$|\phi\rangle = \frac{1}{\sqrt{2}}(|\downarrow\rangle \otimes |\downarrow\rangle - |\uparrow\rangle \otimes |\uparrow\rangle)$$

2. two body operator

$$\mathscr{O} = \sigma_x \otimes \mathbb{I} + \mathbb{I} \otimes \sigma_x$$

As $\mathscr{O}$ operator has two non-zero singular values, it has bond dimension two. Therefore, $\mathscr{O}|\psi\rangle$ operation alters the bond dimension whatever MPS it is.

(a) $|\psi\rangle = |\uparrow\rangle \otimes |\downarrow\rangle$
Now, we apply an MPO with bond dimension two to a product state. This operation gives an MPS which is not a product state.

$$|\phi\rangle = \frac{1}{\sqrt{2}}(|\downarrow\rangle \otimes |\downarrow\rangle + |\uparrow\rangle \otimes |\uparrow\rangle)$$

It can be easily checked that $|\phi\rangle$ has only two non-zero singular values.

Bond dimension η in Eq. (**A.12**) can be bigger than the value $\eta = \gamma\alpha$ after the contraction. We know that MPS in Eq. (**A.12**) is represented faithfully by the bond dimension $\eta = \gamma\alpha$. Therefore, the bond dimension is reducible in these cases. However, with any bond dimension $\eta \geq \gamma\alpha$, quantum state vector is recovered from the MPS formulation by contracting the virtual indices in Eq. (**A.12**).

APPENDIX C: Variational Matrix Product State Method Matlab Code

Variational matrix product states code is mainly taken from the reference [16] and modified for the coupled cavity array with a TLS Hamiltonian in Eq. (3.27).

In main program, there is a logical parameter (flag) which controls the input for the initial MPS. For energy minimization, number of time step parameter is set to zero ($Nt = 0$). If $flag = 0$, ground state calculation is done and the MPS of the ground state is written in a *.mat* file. If $flag = 1$, it is possible to read the ground state MPS from **.mat** file and find the excited state calculations using it. When $Nt > 0$, time dependent algorithm works. In this case, if $flag = 1$ TLS is prepared in the excited state. This case is for the emission from TLS. If $flag = 2$, a wave packet is prepared for the scattering from TLS. We do preparation of the wave packet in two ways: Applying a Gaussian particle distribution to the vacuum state in MPS form and applying the Gaussian particle distribution to the ground state of the system in MPS form.

Listing C.1: main program

```
1
function main_mps
3
maxNumCompThreads(4);
5
global N d D j0 s0 k0 J U w0 V0 Omega Epsilon g0 g1 i0 dt Nt flag ↩
    precision CN Nex mps
7
t0=tic;
9 read_input('mps.in');
fid = fopen('mps.out','w');
11
[hset,n,mpo_odd,mpo_even]=prepare_hset(J,U,w0,V0,Omega,g0,g1,i0,N,d,↩
    CN,Nex);
13
if Nt==0
15     switch flag
           case 0  %ground state
17         [E0,mps0]=minimizeE(hset,D,i0,precision,[]);

19         case 1  % first excited state calculation
           load(strcat('mps0_psi.mat'), 'mps0');
21         [E0,mps1]=minimizeE(hset,D,i0,precision, mps0);
           mps0=mps1;
23     end

25       fprintf('E0 = %g\n',E0+CN*Nex^2);
         nexp=measurement(N,d,i0,n,mps0);
27       disp('initial expectation value')
         save(strcat('mps0_psi.mat'), 'mps0');
29       fprintf( fid, '%3d %9.3e \n', [(1:N+1);  nexp ] );
         fprintf( fid, '# E %f %f %f \n', [   E0+CN*Nex^2; g0; g1] );
31
elseif Nt>0
33
```

```
    %Evolution of a gaussian wave packet
    %randn('state',0)
    if flag==1
        mps=createinitialmps(N,D,d,i0,k0,j0,s0,precision,-1);

    elseif flag==2
        if exist('mps0_psi.mat', 'file')==2
            mps=createinitialmps(N,D,d,i0,k0,j0,s0,precision,0);
        else
            mps=createinitialmps(N,D,d,i0,k0,j0,s0,precision,-1);
        end
    end

        mps=createinitialmps(N,D,d,i0,k0,j0,s0,precision,flag);
        disp('initial state set')
        nexp=measurement(N,d,i0,n,mps);
        sum(nexp)

    fprintf( fid, '%3d %9.3e %9.3e \n', [(1:N+1); 0*dt*ones(1,size(nexp,2)); nexp] );
    fprintf( fid, '\n' );
    for it=1:Nt
        %fprintf('it %d: \n',it);

        [mps,K]=reduceD(mps,mpo_even,D,i0,precision);
        [mps,K]=reduceD(mps,mpo_odd ,D,i0,precision);
        [mps,K]=reduceD(mps,mpo_even,D,i0,precision);

         %if mod(it,20)==0
            fprintf('it :%d / %d \n',it,Nt);
            nexp=measurement(N,d,i0,n,mps);
            sum(nexp)
            fprintf( fid, '%3d %9.3e %9.3e \n', [(1:N+1);  it*dt*ones(1,size(nexp,2)); nexp ] );
            fprintf( fid, '\n' );
         %end
    end
         mps0=mps;
         save(strcat('mps_t_',num2str(Nt*dt,'%11.3e'),'_psi.mat'), 'mps0');
end
fprintf( fid, '# time= %12.4g \n', toc(t0) );
fclose(fid);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function read_input(inputfile)

global N d D j0 s0 k0 J U w0 V0 Omega Epsilon g0 g1 i0 dt Nt flag precision CN Nex

fid= fopen(inputfile, 'r');

  N          = fscanf(fid, 'N=%d\n', 1);
  d          = fscanf(fid, 'd=%d\n', 1);
  D          = fscanf(fid, 'D=%d\n', 1);
  j0         = fscanf(fid, 'j0=%f%f\n', 2);
  s0         = fscanf(fid, 's0=%f%f\n', 2);
  k0         = fscanf(fid, 'k0=%f%f\n', 2);
  J          = fscanf(fid, 'J=%f\n', 1);
```

```
    U          = fscanf(fid, 'U=%f\n', 1);
91  w0         = fscanf(fid, 'w0=%f\n', 1);
    V0         = fscanf(fid, 'V0=%f\n', 1);
93  Omega      = fscanf(fid, 'Omega=%f\n',1);
    Epsilon    = fscanf(fid, 'Epsilon=%f\n',1);
95  g0         = fscanf(fid, 'g0=%f\n',1);
    g1         = fscanf(fid, 'g1=%f\n',1);
97  i0         = fscanf(fid, 'i0=%f\n',1);
    dt         = fscanf(fid, 'dt=%f\n', 1);
99  Nt         = fscanf(fid, 'Nt=%d\n', 1);
    flag       = fscanf(fid, 'flag=%d\n', 1);
101 precision  = fscanf(fid, 'eps=%f\n', 1);
    CN         = fscanf(fid, 'CN=%f\n', 1);
103 Nex        = fscanf(fid, 'Nex=%d\n', 1);

105 fclose(fid);

107 end
```

Input file for the parameters.

Listing C.2: mps.in

```
1  N=          201
   d=          2
3  D=          2
   j0=    20  20
5  s0=       4  4
   k0=   1.57    1.57
7  J=          1
   U=          0
9  w0=        -2
   V0=         0
11 Omega=          1
   Epsilon=        0.0
13 g0=       0.5
   g1=         0
15 i0=        101
   dt=     0.01
17 Nt=         0
   flag=       0
19 eps=   1e-08
    CN=        1000
21 Nex=        1
```

This function minimizes energy of the system given by the Hamiltonian with N sites and M operators. This operators are kept in a cell function as *h=hset{i,j}*. Therefore, h is the i^{th} Hamiltonian operator with the bond dimension $\gamma = 1$ for the site j. This function calls functions calculating the effective Hamiltonian, doing minimization and updating MPS tensors. There are two outputs for this function one of which is minimized energy and the other one is the optimized MPS. Each local sites on the chain is updated by sweeping back and forth until *precision* condition is satisfied.

Listing C.3: minimize energy

```
1  function [E,mps]=minimizeE(hset,D,i0,precision,mpsB)
```

```
[M,N]=size(hset);
d=size(hset{1,1},1);
mps=createrandommps(N,D,d,i0);
mps=prepare(mps);

% storage-initialization
Hstorage=initHstorage(mps,hset,d,i0);

if ~isempty(mpsB), Cstorage=initCstorage(mps,[],mpsB,N); end
P=[];


% optimization sweeps
 while 1
     tic
  Evalues=[];

% ***************** cycle 1: j -> j+1 (from 1 to N-1) ***************
  for j=1:(N-1)
% projector-calculation
    if ~isempty(mpsB)
      B=mpsB{j};
      Cleft=Cstorage{j};
      Cright=Cstorage{j+1};
      P=calcprojector_onesite(B,Cleft,Cright);
    end

% optimization
Hleft=Hstorage(:,j);
Hright=Hstorage(:,j+1);
hsetj=hset(:,j);
[A,E]=minimizeE_onesite(hsetj,Hleft,Hright,P);
[A,U]=prepare_onesite(A,'lr');
mps{j}=A;
Evalues=[Evalues,E];

% storage-update
  for m=1:M
      if j==i0
       h=reshape(hset{m,j},[1,1,2*d,2*d]);
      else
       h=reshape(hset{m,j},[1,1,d,d]);
      end
      %h=reshape(hset{m,j},[1,1,d,d]);
    Hstorage{m,j+1}=updateCleft(Hleft{m},A,h,A);
  end

  if ~isempty(mpsB)
    Cstorage{j+1}=updateCleft(Cleft,A,[],B);
  end
 end

% ***************** cycle 2: j -> j-1 (from N to 2) *****************
  for j=N:(-1):2
% projector-calculation
    if ~isempty(mpsB)
      B=mpsB{j};
      Cleft=Cstorage{j};
      Cright=Cstorage{j+1};
      P=calcprojector_onesite(B,Cleft,Cright);
```

```
63    end

65  % minimization
      Hleft=Hstorage(:,j);
67    Hright=Hstorage(:,j+1);
      hsetj=hset(:,j);
69    [A,E]=minimizeE_onesite(hsetj,Hleft,Hright,P);
      [A,U]=prepare_onesite(A,'rl');
71    mps{j}=A;
      Evalues=[Evalues,E];
73
    % storage-update
75    for m=1:M
        if j==i0
77       h=reshape(hset{m,j},[1,1,2*d,2*d]);
        else
79       h=reshape(hset{m,j},[1,1,d,d]);
        end
81      %h=reshape(hset{m,j},[1,1,d,d]);
      Hstorage{m,j}=updateCright(Hright{m},A,h,A);
83    end

85    if ~isempty(mpsB)
        Cstorage{j}=updateCright(Cright,A,[],B);
87    end

89   end
      if (std(Evalues)/abs(mean(Evalues))< precision)
91      mps{1}=contracttensors(mps{1},3,2,U,2,1);
        mps{1}=permute(mps{1},[1,3,2]);
93      break;
      end
95    toc
    end
```

During time evolution we apply unitary time evolution operator in the MPO form to the MPS with this function. It does this by reducing the bond dimension of MPS to *DB*, since the bond dimension is increased after MPO is applied to the MPS.

Listing C.4: apply time evolution and reduce bond dimension for MPS

```
    function [mpsB, Kvalues]=reduceD(mpsA,mpoX,DB,i0,precision)
2
    N=length(mpsA);
4   d=size(mpsA{1},3);
    mpsB=createrandommps(N,DB,d,i0);
6   mpsB=prepare(mpsB);
    %mpsB=mpsA;
8   % initialization of the storage
    Cstorage=initCstorage(mpsB,mpoX,mpsA,N);
10
    % optimization sweeps
12  while 1
    Kvalues=[];
14  % ***************** cycle 1: j -> j+1 (from 1 to N-1) ***************
    for j=1:(N-1)
16
    % optimization
18  Cleft=Cstorage{j};
```

```
Cright=Cstorage{j+1};
A=mpsA{j}; X=mpoX{j};
[B,K]=reduceD2_onesite(A,X,Cleft,Cright);
[B,U]=prepare_onesite(B,'lr');
mpsB{j}=B;
Kvalues=[Kvalues,K];
% storage-update
Cstorage{j+1}=updateCleft(Cleft,B,X,A);
end
% ***************** cycle 2: j -> j-1 (from N to 2) *****************
for j=N:(-1):2
% optimization
Cleft=Cstorage{j};
Cright=Cstorage{j+1};
A=mpsA{j}; X=mpoX{j};
[B,K]=reduceD2_onesite(A,X,Cleft,Cright);
[B,U]=prepare_onesite(B,'rl');
mpsB{j}=B;
Kvalues=[Kvalues,K];
% storage-update
Cstorage{j}=updateCright(Cright,B,X,A);
end

if std(Kvalues)/abs(mean(Kvalues))<precision
mpsB{1}=contracttensors(mpsB{1},3,2,U,2,1);
mpsB{1}=permute(mpsB{1},[1,3,2]);
break;
end
end

% ********************** one-site optimization ↩
    ************************
function [B,K]=reduceD2_onesite(A,X,Cleft,Cright)
Cleft=contracttensors(Cleft,3,3,A,3,1);
Cleft=contracttensors(Cleft,4,[2,4],X,4,[1,4]);
B=contracttensors(Cleft,4,[3,2],Cright,3,[2,3]);
B=permute(B,[1,3,2]);
b=reshape(B,[prod(size(B)),1]);
K=-b'*b;
```

Auxiliary Functions

This function creates an MPS with virtual bond dimensions D and physical bond dimensions d.

Listing C.5: form random MPS

```
function [mps]=createrandommps(N,D,d,i0)

 mps=cell(1,N);
 mps{1}=randn(1,D,d)/sqrt(D);
 mps{N}=randn(D,1,d)/sqrt(D);

  for i=2:(N-1)
      if i==i0
        mps{i}=randn(D,D,d*2)/sqrt(D);
      else
        mps{i}=randn(D,D,d)/sqrt(D);
```

```
12        end
    end
```

In order to orthogonalize the basis for MPS, we use the function *prepare.m*

Listing C.6: make MPS entries orthogonal

```
function [mps]=prepare(mps)

% takes mps with random entries and calls the functions doing SVD ↩
    decompositon
 N=length(mps); % number of sites (cells)
  for i=N:-1:2
    [mps{i},U]=prepare_onesite(mps{i},'rl');
    mps{i-1}=contracttensors(mps{i-1},3,2,U,2,1);
    mps{i-1}=permute(mps{i-1},[1,3,2]); % make a cell again
  end
```

This is a part for the Hamiltonian that restricts the Hilbert space of the system to a particular particle number sector. This term is $C(N - N_{exc})^2$ (N is the total particle operator) which adds a cost for the energy quadratic in particle number if $N > N_{exc}$.

Listing C.7: restriction for the Hilbert space

```
function hset=HCN(hset,ad, sigmap, n, id, idimp, N, M, M0, CN, Nex,i0)

a=ad';
sigmam=sigmap';
M=M+(N+1)*(N+2);

for i=1:(N+1)*(N+2)
    for j=1:N
        if j==i0
          hset{M0+2+i,j}=kron(id,idimp);
        else
          hset{M0+2+i,j}=id;
        end
    end
end

for i=1:N
  for j=1:N
    if i==i0 && j~=i0
         hset{M0+2+(i-1)*(N+1)+j,i}=CN*kron(ad*a,idimp);
         hset{M0+2+(i-1)*(N+1)+j,j}=ad*a;
         hset{M0+2+N*(N+1)+j,i}=CN*kron(id,sigmap*sigmam);
         hset{M0+2+N*(N+1)+j,j}=ad*a;
    elseif j==i0 && i~=i0
         hset{M0+2+(i-1)*(N+1)+j,i}=CN*ad*a;
         hset{M0+2+(i-1)*(N+1)+j,j}=kron(ad*a,idimp);
         hset{M0+2+(i-1)*(N+1)+N+1,i}=CN*ad*a;
         hset{M0+2+(i-1)*(N+1)+N+1,j}=kron(id,sigmap*sigmam);
     elseif i==i0 && j==i0
         hset{M0+2+(i-1)*(N+1)+j,i}=CN*kron(ad*a,idimp)^2;
         hset{M0+2+N*(N+1)+j,i}=CN*kron(ad*a,sigmap*sigmam);
         hset{M0+2+(i-1)*(N+1)+N+1,i}=CN*kron(ad*a,sigmap*sigmam);
         hset{M0+2+N*(N+1)+N+1,i}=CN*kron(id,sigmap*sigmam)^2;
    else
```

```
        if i==j
36          hset{M0+2+(i-1)*(N+1)+j,i}=CN*ad*a*ad*a;
        else
38          hset{M0+2+(i-1)*(N+1)+j,i}=CN*ad*a;
            hset{M0+2+(i-1)*(N+1)+j,j}=ad*a;
40      end
    end
42  end
end
44
for j=1:N
46      if j==i0
            hset{M-(N+1)+j,j}=-2*CN*Nex*n{j,j};
48          hset{M,j}=-2*CN*Nex*n{N+1,i0};
        else
50          hset{M-(N+1)+j,j}=-2*CN*Nex*n{j,j};
        end
52  end
```

This is for preparing wave packet in the MPS form for two time dependent scenario: emission and scattering

Listing C.8: Initial MPS

```
function mps=createinitialmps(N,D,d,i0,k0,j0,s0,precision,flag)
2
global  ad id idimp mps
4
switch flag
6   case -1
        %vacuum state
8        mps{1}=zeros(D,d);
         mps{1}(1,1)=1;
10       mps{1}=permute(mps{1}, [3 1 2]);

12       for j=2:N-1
           if j==i0
14           mps{j}=zeros(D,D,d*2);
             mps{j}(1,1,2)=1;
16         else
             mps{j}=zeros(D,D,d);
18           mps{j}(1,1,1)=1;
           end
20       end

22       mps{N}=zeros(D,d);
         mps{N}(1,1)=1;
24       mps{N}=permute(mps{N}, [1 3 2]);
    disp('initial state set to vacuum state')
26
    case 0 % read initial mps from file
28       load('mps0_psi.mat', 'mps0');
         mps=mps0;
30  disp('initial state set from file')

32   case 1 % emission from excited state

34       mps{i0}(1,1,2)=0;
         mps{i0}(1,1,1)=1;
```

```
36
    case 2 % Gaussian wave packets
38      Opsi=cell(1,N);
        Gaussian=zeros(1,N);
40      Nph=length(k0);
        for k=1:Nph
42          for j=1:N
              Gaussian(j)=exp(1i*k0(k)*j-(j-j0(k))^2/(2*s0(k)^2));
44          end
            Opsi{1}=zeros(1,2,d,d); % identity and one-body operator ↩
                for each site
46
            for l=1:2
48            Opsi{1}(1,l,:,:)=id;
            end
50          Opsi{1}(1,1,:,:)=Gaussian(1)*ad;

52          for j=2:N-1
               if j==i0
54               Opsi{j}=zeros(2,2,d*2,d*2);
                 Opsi{j}(1,1,:,:)=eye(2*d,2*d);
56               Opsi{j}(2,2,:,:)=eye(2*d,2*d);
                 Opsi{j}(1,2,:,:)=zeros(2*d,2*d);
58               Opsi{j}(2,1,:,:)=kron(Gaussian(j)*ad,idimp);
                else
60               Opsi{j}=zeros(2,2,d,d);
                 Opsi{j}(1,1,:,:)=eye(d,d);
62               Opsi{j}(2,2,:,:)=eye(d,d);
                 Opsi{j}(1,2,:,:)=zeros(d,d);
64               Opsi{j}(2,1,:,:)=Gaussian(j)*ad;
               end
66          end
            Opsi{N}=zeros(2,1,d,d);
68
            for l=1:2
70            Opsi{N}(l,1,:,:)=id;
            end
72          Opsi{N}(2,1,:,:)=Gaussian(N)*ad;

74          [mps,K]=reduceD(mps,Opsi,D,i0,precision);
        end
76  end
    mps=prepare(mps);
```

Hamiltonian in the MPO form with the bond dimensions $\gamma = 1$ and unitary time evolution operators with Suzuki-Trotter expansion.

Listing C.9: MPO for Hamiltonian and Time Evolution Operator

```
    function [hset,n,mpo_odd,mpo_even]=prepare_hset(J,U,w0,V0,Omega,g0,g1↩
        ,i0,N,d,CN,Nex)
2   global dt Epsilon
    global a ad id idimp sigma_z sigmap sigmam
4
    M0=2*(N-1)+N; % number of terms in the Hamiltonian with on-site ↩
        interaction and external potential
6   M=M0+2;          % + Himp + Hint

8   hset=cell(M,N);
```

```
a=zeros(d, d);
for i=1:d-1
 a(i,i+1)=sqrt(i);
end
ad=a';
id=eye(d,d);
idimp=eye(2,2);       % TLS
sigma_z=[1  0; 0  -1];
sigmap=[0  1; 0  0];
sigmam=sigmap';

for m=1:M
    for j=1:N
        if j==i0
          hset{m,j}=kron(id,idimp);
        else
          hset{m,j}=id;
        end
    end
end

%TB Hamiltonian
for j=1:(N-1)
    if j==i0
     hset{2*(j-1)+1,j}=-J*kron(ad,idimp); hset{2*(j-1)+1,j+1}=a;
     hset{2*(j-1)+2,j}=-J*kron(a,idimp);  hset{2*(j-1)+2,j+1}=ad;
    elseif j==i0-1
     hset{2*(j-1)+1,j}=-J*ad; hset{2*(j-1)+1,j+1}=kron(a,idimp);
     hset{2*(j-1)+2,j}=-J*a;  hset{2*(j-1)+2,j+1}=kron(ad,idimp);
    else
     hset{2*(j-1)+1,j}=-J*ad; hset{2*(j-1)+1,j+1}=a;
     hset{2*(j-1)+2,j}=-J*a;  hset{2*(j-1)+2,j+1}=ad;
    end
end

% on-site interaction
for j=1:N
    if j==i0
     hset{2*(N-1)+j,j}=(U/2)*kron(ad*a*(ad*a-id),idimp)+w0*(kron(ad*a↩
         ,idimp)+0*kron(id,sigmap*sigmam))+V0*potential(j,N)*kron(ad*↩
         a,idimp);
    else
     hset{2*(N-1)+j,j}=(U/2)*ad*a*(ad*a-id)+w0*ad*a+V0*potential(j,N)↩
         *ad*a;
    end
end


% number operator
n=cell(N+1,N);
for i=1:N+1
    for j=1:N
        if j==i0
          n{i,j}=kron(id,idimp);
        else
          n{i,j}=id;
        end
    end
end
```

```
for j=1:N
    if j==i0
      n{j,j}=kron(ad*a,idimp);
      n{N+1,j}=kron(id,sigmap*sigmam);
    else
      n{j,j}=ad*a;
    end
end

for j=1:N
    hset{M0+1,j}=id;
end
hset{M0+1,i0}=Omega/2*kron(id,sigma_z)+Epsilon*kron(id,sigmap+sigmam);

for j=1:N
    hset{M0+2,j}=id;
end
hset{M0+2,i0}=g0*(kron(a,sigmap)+kron(ad,sigmam))+g1*(kron(a,sigmam)+kron(ad,sigmap));


%%%%%%%%%%%%%%%%%%%%%%% restriction on a particular number of particle sector %%%%%%%%%%%%%%%%%%%%%%%

%hset=HCN(hset, ad, sigmap, n, id, idimp, N, M, M0, CN, Nex, i0);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% time evolution operator
hbond=-J*(kron(a,ad)+kron(ad,a))...
      +(U/4)*( kron(ad*a*(ad*a-id), id)+kron(id, ad*a*(ad*a-id)) )...
      +w0/2*(kron(ad*a,id)+kron(id,ad*a));   %tight binding bond hamiltonian
Id=reshape(id,[1,1,d,d]);

mpo_even=cell(1,N);
mpo_odd=cell(1,N);
for j=1:N, mpo_even{j}=Id; mpo_odd{j}=Id; end;

w=expm(-1i*dt*hbond);
w=reshape(w,[d,d,d,d]); w=permute(w,[1,3,2,4]); w=reshape(w,[d^2,d^2]);
[Ub,S,Vb]=svd(w,'econ'); Vb=Vb'; eta=size(S,1);
Ub=Ub*sqrt(S); Vb=sqrt(S)*Vb;
Ub=reshape(Ub,[d,d,eta]); Ub=permute(Ub,[4,3,2,1]);
Vb=reshape(Vb,[eta,d,d]); Vb=permute(Vb,[1,4,3,2]);

for j=3:2:(N-1), mpo_odd{j}=Ub; mpo_odd{j+1}=Vb; end

w=expm(-1i*dt*( hbond+(U/4)*kron(id, ad*a*(ad*a-id))+w0/2*kron(id,ad*a) ));
w=reshape(w,[d,d,d,d]); w=permute(w,[1,3,2,4]); w=reshape(w,[d^2,d^2]);
[Ub,S,Vb]=svd(w,'econ'); Vb=Vb'; eta=size(S,1);
Ub=Ub*sqrt(S); Vb=sqrt(S)*Vb;
Ub=reshape(Ub,[d,d,eta]); Ub=permute(Ub,[4,3,2,1]);
Vb=reshape(Vb,[eta,d,d]); Vb=permute(Vb,[1,4,3,2]);

mpo_odd{1}=Ub; mpo_odd{2}=Vb;

w=expm(-1i*dt/2*hbond);
```

```
w=reshape(w,[d,d,d,d]); w=permute(w,[1,3,2,4]); w=reshape(w,[d^2,d^2]);
[Ub,S,Vb]=svd(w,'econ'); Vb=Vb'; eta=size(S,1);
Ub=Ub*sqrt(S); Vb=sqrt(S)*Vb;
Ub=reshape(Ub,[d,d,eta]); Ub=permute(Ub,[4,3,2,1]);
Vb=reshape(Vb,[eta,d,d]); Vb=permute(Vb,[1,4,3,2]);

for j=2:2:(N-3), mpo_even{j}=Ub; mpo_even{j+1}=Vb; end

w=expm(-1i*dt/2*( hbond+(U/4)*kron(ad*a*(ad*a-id),id)+w0/2*kron(ad*a,↩
    id) ));
w=reshape(w,[d,d,d,d]); w=permute(w,[1,3,2,4]); w=reshape(w,[d^2,d^2]);
[Ub,S,Vb]=svd(w,'econ'); Vb=Vb'; eta=size(S,1);
Ub=Ub*sqrt(S); Vb=sqrt(S)*Vb;
Ub=reshape(Ub,[d,d,eta]); Ub=permute(Ub,[4,3,2,1]);
Vb=reshape(Vb,[eta,d,d]); Vb=permute(Vb,[1,4,3,2]);

mpo_even{N-1}=Ub; mpo_even{N}=Vb;

% hbond for the impurity

id_i0=kron(id,idimp);
id_i0=reshape(id_i0,[1 1 2*d 2*d]);
mpo_even{i0}=id_i0;
mpo_odd{i0}=id_i0;

hbond_i0=-J*(kron(a,kron(ad,idimp))+kron(ad,kron(a,idimp)))...
          +Omega/4*kron(id,kron(id,sigma_z))...
          +g0/2*(kron(id,kron(a,sigmap))+kron(id,kron(ad,sigmam)))...
          +g1/2*(kron(id,kron(a,sigmam))+kron(id,kron(ad,sigmap)))...
          +w0/2*(kron(id,kron(ad*a,idimp))+kron(ad*a,kron(id,idimp)));

w=expm(-1i*dt*hbond_i0);
w=reshape(w,[2*d,d,2*d,d]); w=permute(w,[1,3,2,4]); w=reshape(w,[(2*d↩
    )^2, d^2]);
[Ub,S,Vb]=svd(w,'econ'); Vb=Vb'; eta=size(S,1);
Ub=Ub*sqrt(S); Vb=sqrt(S)*Vb;
Ub=reshape(Ub,[2*d,2*d,eta]); Ub=permute(Ub,[4,3,2,1]);
Vb=reshape(Vb,[eta,d,d]); Vb=permute(Vb,[1,4,3,2]);
mpo_odd{i0}=Ub; mpo_odd{i0+1}=Vb;

hbond_i0=-J*(kron(kron(ad,idimp),a)+kron(kron(a,idimp),ad))...
          +Omega/4*kron(kron(id,sigma_z),id)...
          +g0/2*(kron(kron(a,sigmap),id)+kron(kron(ad,sigmam),id))...
          +g1/2*(kron(kron(a,sigmam),id)+kron(kron(ad,sigmap),id))...
          +w0/2*(kron(kron(ad*a,idimp),id)+kron(kron(id,idimp),ad*a));

w=expm(-1i*dt/2*hbond_i0);
w=reshape(w,[d,2*d,d,2*d]); w=permute(w,[1,3,2,4]); w=reshape(w,[d↩
    ^2,(2*d)^2]);
[Ub,S,Vb]=svd(w,'econ'); Vb=Vb'; eta=size(S,1);
Ub=Ub*sqrt(S); Vb=sqrt(S)*Vb;
Ub=reshape(Ub,[d,d,eta]); Ub=permute(Ub,[4,3,2,1]);
Vb=reshape(Vb,[eta,2*d,2*d]); Vb=permute(Vb,[1,4,3,2]);
mpo_even{i0-1}=Ub; mpo_even{i0}=Vb;

%%%%%%%%%%

function vi=potential(i,N)
vi=(i-N/2.-.5)^2;
```

Listing C.10: effective Hamiltonian

```
function [Hstorage]=initHstorage(mps,hset,d,i0)
% initialization of storage

 [M,N]=size(hset);
 Hstorage=cell(M,N+1);
   for m=1:M, Hstorage{m,1}=1; Hstorage{m,N+1}=1; end
    for j=N:-1:2
     for m=1:M
      %takes hset elements seperately for each operator
      if j==i0
       h=reshape(hset{m,j},[1,1,2*d,2*d]);
      else
       h=reshape(hset{m,j},[1,1,d,d]);
      end
      Hstorage{m,j}=updateCright(Hstorage{m,j+1},mps{j},h,mps{j});
    end
   end
```

Formation of effective Hamiltonian, minimization of energy and updating the local MPS tensor for the chain decomposition.

Listing C.11: minimize energy

```
function [A,E]=minimizeE_onesite(hsetj,Hleft,Hright,P)
  global precision

 DAl=size(Hleft{1},1);   % left block dimension
 DAr=size(Hright{1},1);  % right block dimension
 d=size(hsetj{1},1);
 % calculation of Heff
 M=size(hsetj,1);
 Heff=0;
  for m=1:M
    Heffm=contracttensors(Hleft{m},3,2,Hright{m},3,2);
    Heffm=contracttensors(Heffm,5,5,hsetj{m},3,3);
    Heffm=permute(Heffm,[1,3,5,2,4,6]);
    Heffm=reshape(Heffm,[DAl*DAr*d,DAl*DAr*d]);
    Heff=Heff+Heffm;
  end

 % projection on orthogonal subspace
 if ~isempty(P), Heff=P'*Heff*P; end

  % optimization
 options.disp=0;
 options.maxit=2000;
 options.tol=precision;
 [A,E]=eigs(Heff,1,'SR',options);
 %[A,E]=eig(Heff);
 %A=A(1:end,1);
 %E=E(1,1);
 if ~isempty(P), A=P*A; end
 A=reshape(A,[DAl,DAr,d]);
```

Listing C.12: SVD for a specific chain decomposition

```
function [B,U,DB]=prepare_onesite(A,direction)
 %D1 row, D2 columns, d many matrices
 [D1,D2,d]=size(A); % row, column, number of cells

 switch direction

  case 'lr'
    A=permute(A,[3,1,2]); A=reshape(A,[d*D1,D2]);
    [B,S,U]=svd(A,'econ'); U=U'; DB=size(S,1);
    B=reshape(B,[d,D1,DB]); B=permute(B,[2,3,1]);
    U=S*U;
  case 'rl'
    %creates a matrix from the local tensor whose columns is of d
    A=permute(A,[1,3,2]); A=reshape(A,[D1,d*D2]);
    [U,S,B]=svd(A,'econ'); B=B'; DB=size(S,1); %rank of the singular ↩
        value matrix
    B=reshape(B,[DB,d,D2]); B=permute(B,[1,3,2]);
    U=U*S;

 end


 % reshape(B,[DB,d,D2]) makes the matrix DB row, d column and D2 cell
```

Listing C.13: tensor contraction

```
function [X,numindX]=contracttensors(X,numindX,indX,Y,numindY,indY)


Xsize=ones(1,numindX); Xsize(1:length(size(X)))=size(X);
Ysize=ones(1,numindY); Ysize(1:length(size(Y)))=size(Y);

indXl=1:numindX; indXl(indX)=[];
indYr=1:numindY; indYr(indY)=[];

sizeXl=Xsize(indXl);
sizeX=Xsize(indX);
sizeYr=Ysize(indYr);
sizeY=Ysize(indY);

  if prod(sizeX)~=prod(sizeY)
    error('indX and indY are not of same dimension.');
  end

%if all the indices will be contracted
  if isempty(indYr)
    if isempty(indXl)
      X=permute(X,[indX]);
      X=reshape(X,[1,prod(sizeX)]);
      Y=permute(Y,[indY]);
      Y=reshape(Y,[prod(sizeY),1]);
      X=X*Y;   % a scalar
      Xsize=1;
    return;
    else
      X=permute(X,[indXl,indX]);
      X=reshape(X,[prod(sizeXl),prod(sizeX)]);
      Y=permute(Y,[indY]);
      Y=reshape(Y,[prod(sizeY),1]);
```

```
34        X=X*Y;
          Xsize=Xsize(indXl);
36        X=reshape(X,[Xsize,1]);
       end
38   end

40  X=permute(X,[indXl,indX]);
    X=reshape(X,[prod(sizeXl),prod(sizeX)]);
42  Y=permute(Y,[indY,indYr]);
    Y=reshape(Y,[prod(sizeY),prod(sizeYr)]);
44  X=X*Y;
    Xsize=[Xsize(indXl),Ysize(indYr)];
46  numindX=length(Xsize);
    X=reshape(X,[Xsize,1]);
```

Listing C.14: update left side basis

```
    function [Cleft]=updateCleft(Cleft,B,X,A)
2
     d=size(A,3);
4    if isempty(X), X=reshape(eye(size(B,3)),[1,1,d,d]); end

6    Cleft=contracttensors(A,3,1,Cleft,3,3);
     Cleft=contracttensors(X,4,[1,4],Cleft,4,[4,2]);
8    Cleft=contracttensors(conj(B),3,[1,3],Cleft,4,[4,2]);
```

Listing C.15: update right side basis

```
1   function [Cright]=updateCright(Cright,B,X,A)

3    d=size(A,3);

5    if isempty(X), X=reshape(eye(size(B,3)),[1,1,d,d]); end

7    Cright=contracttensors(A,3,2,Cright,3,3);
    %c1= size(Cright)
9    Cright=contracttensors(X,4,[2,4],Cright,4,[4,2]);
    %c2= size(Cright)
11   Cright=contracttensors(conj(B),3,[2,3],Cright,4,[4,2]);
    %c3=size(Cright)
```

Listing C.16: the scalar product of two MPSs for each decomposition of the chain

```
1   function [Cstorage]=initCstorage(mpsB,mpoX,mpsA,N)

3    Cstorage=cell(1,N+1);
     Cstorage{1}=1;
5    Cstorage{N+1}=1;
       for i=N:-1:2
7        if isempty(mpoX), X=[]; else X=mpoX{i}; end
         %size(X);
9        %size(mpsB{i});
         %size(mpsA{i});
11       Cstorage{i}=updateCright(Cstorage{i+1},mpsB{i},X,mpsA{i});
       end
```

Calculating projection operator for the excited state calculation is done by this function.

Listing C.17: projection operator

```
function [P]=calcprojector_onesite(B,Cleft,Cright)

 y=contracttensors(Cleft,3,3,B,3,1);
 y=contracttensors(y,4,[2,3],Cright,3,[2,3]);
 y=permute(y,[1,3,2]);
 y=reshape(y,[prod(size(y)),1]);
 Q=orth([y,eye(size(y,1))]);
 P=Q(:,2:end);
```

We calculate the expectation value of particle number operator.

Listing C.18: expectation values

```
function e=measurement(N,d,i0,Op, mps)

for i=1:N+1                          % the number of operators
  e(i)=0;
  em=1;
  for j=N:-1:1                       % the number of sites
    Opi=Op{i,j};                     % n{i,j} --> j: site number, i: ↩
        operator
    if j==i0
       Opi=reshape(Opi,[1,1,2*d,2*d]);
    else
       Opi=reshape(Opi,[1,1,d,d]);
    end
    em=updateCright(em, mps{j}, Opi, mps{j});
  end
  e(i)=e(i)+em;
end

norm=1;
X=eye(d); X=reshape(X, [1,1,d,d]);
X0=eye(2*d); X0=reshape(X0, [1,1,2*d,2*d]);

for i=N:-1:1
  if i==i0
      norm=updateCright(norm, mps{i}, X0, mps{i});
  else
      norm=updateCright(norm, mps{i}, X, mps{i});
  end
end

e=e/norm;
e=abs(e);
```

CURRICULUM VITAE

Name Surname: Nur Aslan

Place and Date of Birth: Istanbul, 25.09.1990

E-Mail: aslann@itu.edu.tr

B.Sc.: Istanbul Technical University (January, 2013)