

---

# Matrix product state approach for a multi-lead Anderson model

Andreas Michael Holzner

---



München 2006



---

# Matrix product state approach for a multi-lead Anderson model

Andreas Michael Holzner

---

Diplomarbeit  
an der Fakultät für Physik  
der Ludwig–Maximilians–Universität  
München

vorgelegt von  
Andreas Michael Holzner  
aus München

München, den 22.09.2006

Erstgutachter: Prof. Dr. Jan von Delft  
Zweitgutachter: Prof. Dr. Ignacio Cirac

# Contents

<b>Introduction</b>	<b>1</b>
<b>1 Model</b>	<b>3</b>
1.1 Anderson impurity model	3
1.1.1 Single impurity Hamiltonian	3
1.1.2 Multiple dot levels and leads	3
1.2 NRG transformation	5
1.2.1 Basic idea of NRG	5
1.2.2 Application of NRG formalism	6
1.2.3 NRG solution	8
<b>2 Matrix product state method</b>	<b>11</b>
2.1 Representation of states	11
2.1.1 Hamiltonian with fixed chain length	11
2.1.2 Construction of matrix product states	12
2.1.3 Useful techniques	18
2.1.4 Interpretation of the $A$ -matrices	20
2.1.5 Orthonormal basis states	21
2.1.6 Operator representation	22
2.1.7 Evaluation of an operator	23
2.1.8 Expectation values	24
2.2 Hilbert space truncation	27
2.2.1 Construction of the density matrix	27
2.2.2 Singular value decomposition	27
2.2.3 Orthonormalisation & truncation scheme	28
2.2.4 Orthonormalisation towards other indices	30
2.2.5 Matrix dimensions	30
2.3 Variational optimisation scheme	30
2.3.1 Minimisation problem	31
2.3.2 Site optimisation of the $A_k$	32
2.3.3 Solving the eigenvalue problem	32
2.3.4 Sweeping procedure	34
2.3.5 Numerical costs	35

---

2.3.6	Bond optimisation . . . . .	36
2.4	Determining the ground state in a nutshell . . . . .	38
<b>3</b>	<b>First results</b>	<b>41</b>
3.1	Consistency checks . . . . .	41
3.1.1	Matrix dimensions . . . . .	41
3.1.2	Flow diagram . . . . .	42
3.2	Occupation of the quantum dot . . . . .	44
3.2.1	Two-level dot . . . . .	44
3.2.2	Four-level dot . . . . .	46
<b>4</b>	<b>Conclusions and Outlook</b>	<b>49</b>
<b>A</b>	<b>Further details</b>	<b>51</b>
A.1	Jordan-Wigner Transformation . . . . .	51
A.1.1	Definition . . . . .	51
A.1.2	Commutator Relations . . . . .	51
A.1.3	Application . . . . .	52
A.2	Orthonormal basis states . . . . .	54
A.2.1	Derivation of the MPS orthonormality condition . . . . .	54
A.2.2	Orthonormalisation conditions for dot matrices . . . . .	56
	<b>List of Figures</b>	<b>59</b>
	<b>Bibliography</b>	<b>61</b>
	<b>Acknowledgements</b>	<b>63</b>

# Introduction

One very important model in the field of strongly correlated electrons is the Anderson impurity model [1]. Due to its many-body nature it cannot be solved exactly, but a powerful numerical method, the NRG, has been developed [2, 4] with which an accurate treatment of the model is possible. NRG has successfully been used to solve the Anderson model among other impurity models. Also thermodynamic and dynamical properties of such models are accessible with NRG. However, for a more complex impurity with  $m$  levels, the Hilbert space dimension scales with a factor of  $2^m$ . This makes it very difficult to obtain good numerical solutions of a more complex Anderson model with NRG.

Another well established method in this field is the density matrix renormalisation group (DMRG) [6, 7, 8] that allows for successful treatment of tight-binding models like the Hubbard model where NRG fails.

It has been shown that both methods, NRG and DMRG, can be formulated using matrix product states (MPS) [9, 10, 13, 14, 15]. DMRG is in many respects equivalent to the MPS approach but even there using MPS language can improve the performance of the computation compared to traditional DMRG formulation. Also the states created by NRG have by construction the structure of MPS [17]. This makes it possible to further improve NRG results by variationally optimising upon NRG generated MPS. As it is possible to distribute numerical resources more efficiently within the MPS formulation, models impossible to treat with NRG become feasible by using the MPS approach.

In this work we will take advantage from that fact and use the framework of matrix product states to combine ideas of both NRG and DMRG into an variational optimisation approach for the Anderson model. This way we can overcome the main weaknesses of NRG, high computational complexity for complex models and non optimal scheme for truncating the effective Hilbert space used to describe the system, and solve more complex models than possible with NRG. We will apply our method to the extended Anderson model for a quantum dot with several dot levels coupled to two electronic leads.

For a small system with two dot levels we checked our results obtained with the MPS approach with NRG data. For such systems NRG is known to work very well and the results of both methods show good agreement. We calculated first results for a spinful four-level model which is intractable with NRG. These results are very promising for future applications of this method to time-dependent problems. Also in prospect of steady state problems the results so far achieved are promising for further developments.

## Outline

This thesis is organised as follows. In chapter 1 we introduce the extended Anderson model, which we use to describe a multi-level quantum dot coupled to several leads. We also summarise the main ideas of NRG and show how NRG is applied to the Anderson model, which leads us to the Hamiltonian we will later treat numerically.

In chapter 2 we introduce the matrix product state approach for our model and we will present all the necessary techniques for solving our model within this formalism (like calculating scalar products, expectation values, ...). We will then show how to incorporate the DMRG idea of Hilbert space truncation into MPS language. Having provided all necessary technical tools, we present the variational optimisation scheme for determining the ground state of our system. We will discuss two variants for optimising the ground state and finally provide a cost analysis for the algorithm we introduced.

In chapter 3 we present first data from our implementation of a generalised Anderson model with two leads and up to four spinful dot levels. For the case with two dot levels we compare our results with NRG.

# Chapter 1

## Model

### 1.1 Anderson impurity model

Here we study a well known impurity model. It was first introduced by Anderson [1] in 1961 to describe the effect of iron-group ions, which carry a magnetic moment, dissolved in metals. Today this model is commonly used to describe quantum dot systems.

#### 1.1.1 Single impurity Hamiltonian

The Anderson Hamiltonian describes an atomic impurity coupled to a conduction band where the impurity is represented by a spinful one-level system. The density of states around the Fermi energy is assumed to be constant. In second quantization the Hamiltonian has the form<sup>1</sup>

$$H = \sum_{\mathbf{k},s} \epsilon_{\mathbf{k}} c_{\mathbf{k}s}^{\dagger} c_{\mathbf{k}s} + \sum_s \epsilon_d d_s^{\dagger} d_s + U d_{\uparrow}^{\dagger} d_{\uparrow} d_{\downarrow}^{\dagger} d_{\downarrow} + \sum_{\mathbf{k},s} \left( V_{\mathbf{k}} d_s^{\dagger} c_{\mathbf{k}s} + V_{\mathbf{k}}^* c_{\mathbf{k}s}^{\dagger} d_s \right). \quad (1.1)$$

Here,  $c_{\mathbf{k}s}^{\dagger}$  denotes the creation operator for a conduction band electron with momentum  $\mathbf{k}$  and spin  $s$ . Accordingly,  $d_s^{\dagger}$  denotes the creation operator for the impurity level with spin  $s$ . The  $U$  term represents the Coulomb interaction between electrons on the impurity, the coupling of the conduction band to the impurity is denoted by the last sum in the Hamiltonian. For a sketch see Figure 1.1.

We will apply the Anderson model to a quantum dot system. Thus we will refer to the impurity level as 'dot level' instead of the d-shell of a transition metal as in the original work of Anderson. A commonly used approximation is to assume the coupling strength to be real (for no magnetic field) and energy independent,  $V_{\mathbf{k}} = V_{\mathbf{k}}^* = V$ .

#### 1.1.2 Multiple dot levels and leads

For our studies of a quantum dot coupled to several leads in equilibrium we need to generalise the single impurity Anderson model. We consider  $m$  dot levels and  $N$  leads<sup>2</sup>

---

<sup>1</sup>Note that we use  $s$  as spin index here, because we will use  $\sigma$  for a different purpose later on.

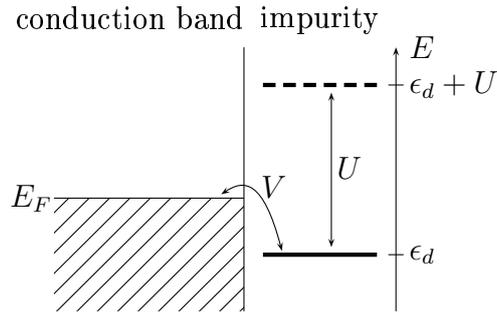


Figure 1.1: Anderson model for a single conduction band and one dot level.

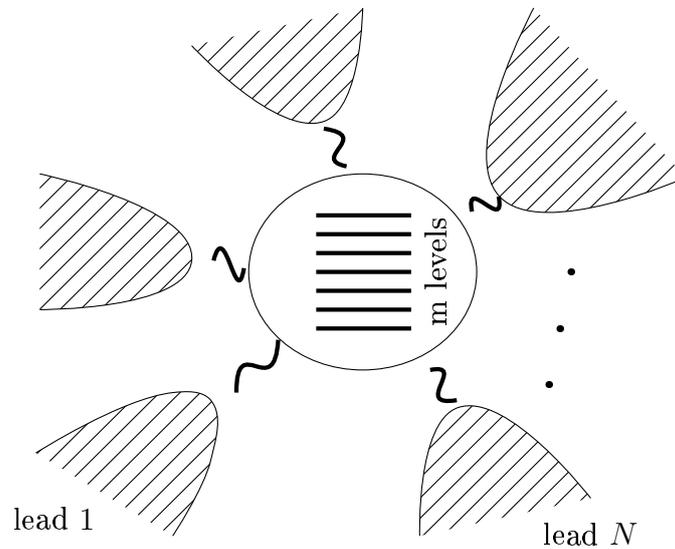


Figure 1.2: Sketch of a quantum dot coupled to several leads

that are coupled to the dot (see Figure 1.2). With these generalisations we arrive at the following parts of the Hamiltonian:

- Eigenenergies of the dot levels. Any magnetic field can be incorporated into the energies  $\epsilon_{is}$ .

$$H_{\text{dot}} = \sum_{i=1}^m \sum_{s=\downarrow, \uparrow} \epsilon_{is} d_{is}^{\dagger} d_{is} \quad (1.2)$$

<sup>2</sup>For our numerical method we will later distinguish between spin up and spin down leads, but this plays no important role in our current considerations.

- Coulomb interaction on the dot. By lifting the constraint of a constant  $U$ , thus introducing coefficients  $U_{ij}^{ss'}$ , one can easily model different dots coupled to the same leads.

$$H_{\text{int}} = \frac{U}{2} \sum_{(i,s) \neq (j,s')} d_{is}^\dagger d_{is} d_{js'}^\dagger d_{js'} \quad (1.3)$$

- Free Hamiltonian for  $N$  leads ( $\alpha = 1 \dots N$ ). We assume identical spectrum and density of states for all leads. Only the density of states near the Fermi edge<sup>3</sup> will be important<sup>4</sup>. Any differences of the density of states can be incorporated to the couplings of the leads to the dot.

$$H_{\text{leads}} = \sum_{\alpha \mathbf{k} s} \epsilon_{\mathbf{k}} c_{\alpha \mathbf{k} s}^\dagger c_{\alpha \mathbf{k} s} \quad (1.4)$$

- Coupling between the leads and the dot. The tunnel couplings may vary for different lead and level number:  $V_{i\alpha}$ ,  $i = 1 \dots m$ ,  $\alpha = 1 \dots N$ .

$$H_{\text{coupling}} = \sum_{\alpha i \mathbf{k} s} V_{i\alpha} \left( d_{is}^\dagger c_{\alpha \mathbf{k} s} + c_{\alpha \mathbf{k} s}^\dagger d_{is} \right) \quad (1.5)$$

The Hamiltonian for the whole system is just the sum of all these parts.

$$H_{\text{system}} = H_{\text{dot}} + H_{\text{int}} + H_{\text{leads}} + H_{\text{coupling}} \quad (1.6)$$

## 1.2 Numerical renormalization group formalism

In principle, one would like to calculate the properties of  $H_{\text{system}}$  from (1.6). Due to the infinite number of degrees of freedom, this problem cannot be solved neither analytically<sup>5</sup> nor numerically. For impurity problems like ours a very successful method, the numerical renormalization group (NRG), has been developed by Wilson [2] in 1975.

### 1.2.1 Basic idea of NRG

The main problem of solving the Anderson model comes from the fact, that the Anderson model is a true many-body problem, and that the conduction bands of the leads cannot be described by quasi-particles because of the presence of the impurity, as we will explain with the following short argument. Suppose we are in a regime when there is an odd number of electrons with overall spin down on the dot. Now, an electron with spin up from one of the leads can spin-flip scatter with the dot such that both the dot state and the scattering electron change spin, which leaves the dot in a spin up state. The next such spin-flip

<sup>3</sup>Since we study the system only in equilibrium  $E_F$  is the same for all leads.

<sup>4</sup>For more details see section 1.2.2 or [4].

<sup>5</sup>apart from some special cases, like  $U = 0$

scattering process can now only occur with a spin down electron of the leads. Therefore one cannot describe the leads coupled to the dot with a quasi-particle picture as with individual free leads. The main achievement of NRG is to provide a non-perturbative way to describe

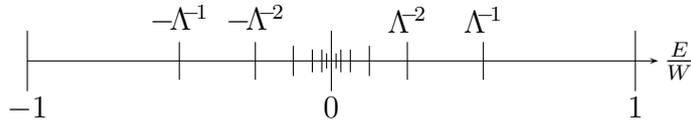


Figure 1.3: Logarithmic energy discretisation for a conduction band of a lead with bandwidth  $2W$  centered around  $E_F = 0$ .

the system. For this purpose a logarithmic discretisation of the conduction band around the Fermi energy with a discretisation parameter  $\Lambda > 1$  is introduced. The  $n$ -th state represents the energy interval proportional to the interval from  $\Lambda^{-(n+1)}$  to  $\Lambda^{-n}$ . After doing a Fourier transformation of the creation and annihilation operators of the conduction band electrons on these intervals and taking only their first Fourier components into account one can apply an unitary transformation to arrive at a tridiagonal Hamiltonian for the leads. With this the conduction bands of the leads transform to semi-infinite “Wilson chains” with nearest neighbour hopping and exponentially decaying coupling constants within the chains. Furthermore only the first element of each chain is coupled to the dot levels (for details see section 1.2.2). In NRG one then defines a series of Hamilton operators  $H_n$ , where  $H_0$  describes only the quantum dot,  $H_1$  describes the dot and the first sites of all the attached lead chains and so forth. One iteratively solves each Hamiltonian of this series and truncates the state space to a manageable size. One stops this iteration process once the spectrum no longer changes, which indicates that the considered chain is long enough to resolve even the smallest energy scale in the system i.e. to describe the ground state of the full system.

## 1.2.2 Application of NRG formalism

In this work we are interested in the ground state properties of the multi-level multiple-lead Anderson model at  $T = 0$ . In order to resolve low energy excitations we need high energy resolution near the Fermi energy. This is achieved by the logarithmic energy discretisation of NRG. We therefore apply the NRG formalism to our model Hamiltonian but will use a different method for solving it. Here we will follow the derivation and notation of [4] very closely and will cite only important steps of the calculation.

The first thing to note is, that  $H_{\text{dot}}$  (1.2) and  $H_{\text{int}}$  (1.3) are not affected by the NRG transformation, as it affects only the lead parts of the Hamiltonian. We assume the conduction band to be symmetric with respect to the Fermi energy and have bandwidth  $2W$ . So with the convention  $E_F = 0$  the conduction band extends from  $-W$  to  $W$ . Now we introduce a new variable  $\epsilon = \frac{E}{W}$ , note that  $\epsilon$  is called  $k$  in [4]. We also neglect the energy dependence of  $V_{i\alpha}$  and the density of states  $\rho$ , and replace both with their values at

the Fermi level<sup>6</sup>,  $\rho = \rho(E_F)$  and  $V_{i\alpha} = V_{i\alpha}(E_F)$ . As a further simplification, we consider only electrons with no angular momentum (s-wave states of the conduction band), thus we can label the electron states by energy ( $\epsilon$ ). We define new creation operators for the lead electrons  $a_{\alpha\epsilon s}^\dagger$  that create an electron in lead  $\alpha$  with energy  $\epsilon$  and spin  $s$ . They obey  $\{a_{\alpha\epsilon s}, a_{\alpha'\epsilon's'}^\dagger\} = \delta_{\alpha\alpha'}\delta(\epsilon - \epsilon')\delta_{ss'}$ . With the definition of

$$\Gamma_{i\alpha} = \pi\rho V_{i\alpha}^2, \quad (1.7)$$

we get for the affected parts of the Hamiltonian:

$$H_{\text{leads}} = W \sum_{\alpha s} \int_{-1}^1 d\epsilon \epsilon a_{\alpha\epsilon s}^\dagger a_{\alpha\epsilon s} \quad (1.8)$$

$$H_{\text{coupling}} = W \sum_{\alpha i s} \sqrt{\frac{\Gamma_{i\alpha}}{\pi W}} \int_{-1}^1 d\epsilon \left( d_{is}^\dagger a_{\alpha\epsilon s} + a_{\alpha\epsilon s}^\dagger d_{is} \right). \quad (1.9)$$

### Logarithmic discretisation

The next step is to introduce the logarithmic discretisation scheme and apply it to the Hamiltonian. One defines a discretisation parameter  $\Lambda > 1$  and divides the  $\epsilon$  domain into a series of intervals as shown in Figure 1.3 such that the  $n$ th interval extends from  $\Lambda^{-(n+1)}$  to  $\Lambda^n$  (for positive  $\epsilon$ ). By defining Fourier series on these intervals one can expand the  $a_{\alpha\epsilon s}$  in terms of two sets (for positive and negative  $\epsilon$ ) of independent and discrete operators  $a_{\alpha n p s}$  and  $b_{\alpha n p s}$ , which obey standard anticommutation rules. As a further approximation one neglects all higher harmonics<sup>7</sup> in the expansion of the  $a_{\alpha k s}$ , i.e. one drops all terms with  $a_{\alpha n p s}$  and  $b_{\alpha n p s}$  where  $p \neq 0$ . Thus we will drop the subscript  $p$  of  $a$  and  $b$ . With the definition

$$f_{\alpha 0 s} = \sqrt{\frac{1}{2}(1 - \Lambda^{-1})} \sum_{n=0}^{\infty} \Lambda^{-\frac{n}{2}} (a_{\alpha n s} + b_{\alpha n s}) \quad (1.10)$$

one arrives at the following Hamiltonians:

$$H_{\text{leads}} = W \sum_{\alpha s} \frac{1}{2} (1 + \Lambda^{-1}) \sum_{n=0}^{\infty} \Lambda^{-n} (a_{\alpha n s}^\dagger a_{\alpha n s} - b_{\alpha n s}^\dagger b_{\alpha n s}) \quad (1.11)$$

$$H_{\text{coupling}} = W \sum_{\alpha i s} \sqrt{\frac{2\Gamma_{i\alpha}}{\pi W}} \left( f_{\alpha 0 s}^\dagger d_{is} + d_{is}^\dagger f_{\alpha 0 s} \right). \quad (1.12)$$

### Nearest neighbour hopping Hamiltonian

So far one already has achieved that only the  $f_{\alpha 0 s}$  operators couple to the dot levels. In the final step the lead Hamiltonian gets transformed to a nearest neighbour hopping

<sup>6</sup>Note that we will use the same symbols.

<sup>7</sup>This turns out to be a surprisingly good approximation even for  $\Lambda$  being as big as 3.

Hamiltonian. Therefore one performs an unitary transformation on the  $a$  and  $b$  operators to get a new complete set of orthonormal operators  $f$ . As the Hamiltonian in (1.11) is diagonal in the  $a$  and  $b$  operators the best one can achieve is to get only nearest neighbour coupling in terms of the  $f$  operators, i. e. a tridiagonal Hamiltonian. The exact definition of the  $f_{\alpha ns}$  can be found in [4], but is not important for our further discussions. Defining new coefficients

$$\xi_n = (1 - \Lambda^{-n-1}) (1 - \Lambda^{-2n-1})^{-\frac{1}{2}} (1 - \Lambda^{-2n-3})^{-\frac{1}{2}} \quad (1.13)$$

the lead Hamiltonian is given by

$$H_{\text{leads}} = W \sum_{\alpha s} \frac{1}{2} (1 + \Lambda^{-1}) \sum_{n=0}^{\infty} \Lambda^{-\frac{n}{2}} \xi_n \left( f_{\alpha ns}^\dagger f_{\alpha(n+1)s} + f_{\alpha(n+1)s}^\dagger f_{\alpha ns} \right). \quad (1.14)$$

Putting the pieces together we get the Hamiltonian for the whole system after applying the NRG formalism. Any further uses of  $H_{\text{coupling}}$  and  $H_{\text{leads}}$  will refer to (1.12) and (1.14). Note that the hopping constant in (1.14) decays exponentially with  $\Lambda^{-\frac{n}{2}}$  as the  $\xi_n$  are of order  $\mathcal{O}(1)$ .

### 1.2.3 NRG solution

We will now briefly describe how this Hamiltonian is solved by NRG. We introduce the notation of  $H_{\text{leads}}^k$  as the Hamiltonian of the lead chains including  $k$  sites for every chain. For the first NRG step one solves the system

$$H_0 = H_{\text{dot}} + H_{\text{int}} + H_{\text{coupling}} + H_{\text{leads}}^{k_0}, \quad (1.15)$$

where  $k_0$  is chosen such that  $H_0$  can directly be solved by exact diagonalisation. As a result one knows the spectrum and the eigenstates of  $H_0$ .  $H_n$  is defined similarly by

$$H_n = H_{\text{dot}} + H_{\text{int}} + H_{\text{coupling}} + H_{\text{leads}}^{k_0+n}. \quad (1.16)$$

Now starts the NRG iteration process, so we will describe the tasks of the  $n$ th iteration step. One keeps the  $D$  eigenstates<sup>8</sup> with the lowest energies from the set of eigenstates obtained in the previous iteration, and projects  $H_{n-1}$  onto the space spanned by the kept states. Now, the next site of each chain is added and the new Hamiltonian  $H_n$  is solved in the enlarged Hilbert space consisting of the tensor product of the  $D$ -dimensional truncated Hilbert space of the previous iteration with the local state spaces of the newly added chain sites, thus yielding the spectrum and eigenstates of  $H_n$ . This iteration step is repeated until the rescaled<sup>9</sup> spectrum converges. This typically takes place at chain lengths up to 60. Let  $|\sigma_n\rangle$  denote the local states of site  $n$  and  $|s_n\rangle$  the states we keep to describe the effective Hilbert space of  $H_n$ . Then we can express (see Figure 1.4 for a graphical

<sup>8</sup>Typical values for  $D$  vary between 500 and 2000.

<sup>9</sup>As the couplings in  $H_{\text{leads}}$  decay exponentially, one has to rescale the spectrum after each step with  $\Lambda^{1/2}$ .

representation) the states  $|s_n\rangle$  in terms of the local states  $|\sigma_n\rangle$  and the states kept from the previous iteration step  $|s_{n-1}\rangle$

$$|s_n\rangle = \sum_{\sigma_n s_{n-1}} |\sigma_n\rangle \otimes |s_{n-1}\rangle A_{s_{n-1}s_n}^{[\sigma_n]}, \quad (1.17)$$

where the coefficients  $A_{s_{n-1}s_n}^{[\sigma_n]}$  determine which linear superpositions are kept for the next iteration. By viewing these  $A$ -tensors as generalised matrices, the result of a repeated application of (1.17) can be denoted as a “matrix product state”. This structure arises naturally by the NRG construction. We will not elaborate on matrix product states in NRG context, more information on this can be found in [19].

$$|s_{n-1}\rangle - \boxed{A_n} - |s_n\rangle$$

$$\quad \quad \quad \downarrow$$

$$\quad \quad \quad |\sigma_n\rangle$$

Figure 1.4: Iterative construction of NRG state space  $|s_n\rangle$ .

## Discussion

The reason for this procedure to work so well is the energy scale separation between the different iteration steps that is guaranteed by the exponential decay of the coupling along the chains. This provides very good energy resolution at low excitation energies near  $E_F$  and bad energy resolution at high energies. So one basically treats a different energy scale in every step. In order to maintain this energy scale separation also for a multiple-chain model, one has to include the  $n$ th site of every chain at the same iteration step. Assuming a local state space dimension for each site and spin of  $d = 2$  (fermionic chains), in each step the Hilbert space dimension increases by a factor of  $d^{2N}$  (because of spin degree of freedom) and is then reduced again to  $D$ . This is known to work quite well for models with  $N = 2$ , but as soon as  $d^{2N}$  becomes larger than roughly 50, it is unclear how reliable this method works. An other problem of NRG is the truncation scheme that simply keeps these  $D$  eigenstates that have the lowest eigenenergies. This is the most simple but not the best possible approach<sup>10</sup>. While this works very well for impurity models, where the coupling constant decays for the mapped system, it fails for chain models without decaying coupling constants and real-space methods [5].

## Outlook

In the following chapter, we will show how these two limitations of NRG can be overcome using the matrix product state approach. The key idea is to variationally determine the ground state by minimising the expectation value  $\langle\psi|H|\psi\rangle$  within the space of all matrix

<sup>10</sup>We will show how to improve the truncation scheme in section 2.2.

product states, with the elements of all  $A$ -matrices as variational parameters. For implementing this idea, we have to introduce a number of technical steps. Therefore we will first show how a matrix product state is constructed and introduce an intuitive graphical representation for it. Then we will develop techniques to calculate scalar products  $\langle\psi'|\psi\rangle$ , reduced density matrices  $\rho_k$ , and expectation values  $\langle\psi|\hat{H}|\psi\rangle$ . Having provided these techniques, we will use them to implement a more accurate Hilbert space truncation scheme that follows the DMRG idea, and finally present our variational optimisation scheme for the ground state of the system.

## Chapter 2

# Matrix product state method

We will now introduce the matrix product state method. The main idea is to write down an ansatz for the groundstate—a so called matrix product state—and then improve on it with a variational optimisation scheme.

## 2.1 Representation of states

### 2.1.1 Hamiltonian with fixed chain length

As we don't use an iterative method like NRG, we have to fix the length  $L$  of the lead chains from the start. Apart from this the lead Hamiltonian is the same as in Wilson's NRG (1.14), so we will often refer to the lead chains as “Wilson chains”. Typically, we choose  $L = 60$  to  $80$ . From now on, we use for the bandwidth the convention  $W = 1$ , so the Hamiltonian of the whole systems is

$$\begin{aligned}
 H_{\text{system}} &= \sum_{i=1}^m \sum_{s=\downarrow,\uparrow} \epsilon_{is} d_{is}^\dagger d_{is} \\
 &+ \frac{U}{2} \sum_{\substack{i,j=1\dots m \\ s,s'=\downarrow,\uparrow \\ (i,s)\neq(j,s')}} d_{is}^\dagger d_{is} d_{js'}^\dagger d_{js'} \\
 &+ \sum_{s=\downarrow,\uparrow} \sum_{\alpha=1}^N \frac{1}{2} (1 + \Lambda^{-1}) \sum_{n=0}^L \Lambda^{-\frac{n}{2}} \xi_n \left( f_{\alpha ns}^\dagger f_{\alpha(n+1)s} + f_{\alpha(n+1)s}^\dagger f_{\alpha ns} \right) \\
 &+ \sum_{s=\downarrow,\uparrow} \sum_{\alpha=1}^N \sum_{i=1}^m \sqrt{\frac{2\Gamma_{i\alpha}}{\pi}} \left( f_{\alpha 0s}^\dagger d_{is} + d_{is}^\dagger f_{\alpha 0s} \right).
 \end{aligned} \tag{2.1}$$

One important point to note is that electrons with different spin only interact via the Coulomb interaction  $U$ . So it is convenient to interpret the lead term of the Hamiltonian as a sum over  $2N$  Wilson chains,  $N$  for each spin. However, one has to be very careful

about interchanging operators and states<sup>1</sup>, as we are dealing with Fermions. In order to avoid this ordering problem we apply the Jordan-Wigner transformation, which we describe in detail in appendix A.1, to the Hamiltonian (2.1). As a result we get almost bosonic behaviour of the former fermionic operators, and we can exchange operators and states for different sites<sup>2</sup> without generating any factors of  $-1$ . Because there are apart from few local corrections on the dot no further consequences applying the transformation, we will call the transformed  $f$  operators  $c$  from now on and assume that all corrections are absorbed into them. The Jordan-Wigner transformed Hamiltonian (2.1) will be our starting point for the treatment with the matrix product state method.

### 2.1.2 Construction of matrix product states

The full Hilbert space of the whole system is the product of the local state spaces of all chain sites and the dot state space. It is clear that its dimension is far too big in order to operate on it directly. Thus we need a description for the ground state—that is what we want to calculate here—that refers to an effective Hilbert space. This description will be provided by the matrix product state. We will now explain, how the matrix product description arises naturally, considering single chain sites<sup>3</sup> as fundamental building blocks of our system, by building an effective Hilbert space site by site.

In order to illustrate this, we restrict ourselves to only a single chain of length  $L$ . But this scheme can easily be generalised to the complex system of (2.1). We define the notion of the basis of an inner<sup>4</sup> state space  $|i_k\rangle$  (ISS), outer state space  $|o_k\rangle$  (OSS) and local state space  $|\sigma_k\rangle$  (LSS) with respect to a certain site  $k$ . So the outer basis  $|o_k\rangle$  is a basis of the effective Hilbert space covering all sites  $k' > k$ . Thus the dot is always contained in the ISS for every chain site. With the separation of spin up and down parts of the chains, our chain sites are simple spinless fermionic sites, thus the LSS basis consists of the states “occupied” and “un-occupied”:

$$|\sigma_k\rangle = \{|0_k\rangle, |1_k\rangle\}. \quad (2.2)$$

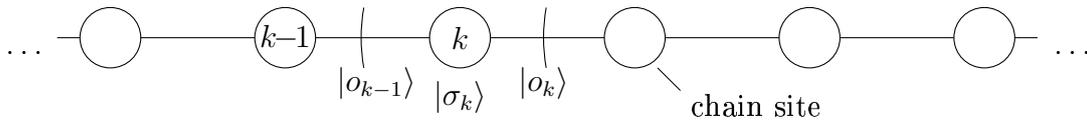


Figure 2.1: Part of a chain to illustrate how to determine an OSS basis  $|o_{k-1}\rangle$  of site  $k - 1$  from the outer and local bases of site  $k$ .

<sup>1</sup>States can always be written as some combination of creation operators applied to the vacuum state. Thus they cannot be interchanged without respecting the anticommutation relations, too.

<sup>2</sup>In particular we will often use the possibility of exchanging states of different parts of the system.

<sup>3</sup>The dot levels can as well be viewed as a combination of single sites (for each dot level and spin one site). The fact, that the couplings differ from a nearest neighbour hopping chain, imposes no restriction to our argument.

<sup>4</sup>Here outer refers to that direction in the chain that points to the end of the chain, inner refers to the direction leading to the dot.

It follows directly from the chain structure (Figure 2.1) that for any given site  $k$  the outer and local state spaces can be combined to an effective OSS for the next inward site<sup>5</sup>. The coefficients used for the linear combination are encoded in  $A$ .

$$|o_{k-1}\rangle = \sum_{\{o_k\},\{\sigma_k\}} A_{o_{k-1},o_k}^{[\sigma_k]} |o_k\rangle |\sigma_k\rangle \quad (2.3)$$

For brevity we will drop the sum symbols and imply summation over repeated<sup>6</sup> super-/subscripts, further we drop all but one  $k$ -subscript whenever possible. We will also use the notion of the basis and its associated state space synonymously<sup>7</sup>.

$$|o_{k-1}\rangle = |o'\rangle = A_{o'o}^{[\sigma_k]} |o\rangle |\sigma\rangle \quad (2.4)$$

This holds for every site  $k$  except for the end of the chain, as there are no more outer sites and thus no effective OSS. Keeping this in mind, one can repeatedly apply (2.4) to get a basis describing a whole chain.

$$|o_0\rangle = A_{o_0o_1}^{[\sigma_1]} A_{o_1o_2}^{[\sigma_2]} \dots A_{o_{L-1}o_L}^{[\sigma_L]} |\sigma_1\rangle |\sigma_2\rangle \dots |\sigma_{L-1}\rangle |\sigma_L\rangle \quad (2.5)$$

The Jordan-Wigner transformation allows us to place the local state space vectors last, to correspond to the ordering of the  $A$ -matrices. Moreover we use the convention that the first index of every chain site  $A$ -matrix refers to the inner and the second index to the outer basis<sup>8</sup>. The third index is the local basis index and is placed as a superscript, that carries also the chain index  $k$ . For historical reasons we use the term matrix also for tensors with arbitrary number of indices as long as they are used in the same context as the  $A$ s above. With that, we get an interpretation of the coefficients  $A_{o'o}^{[\sigma_k]}$  as matrices, and we can rely on the matrix product—that is where the name of this method, matrix product state method, is derived from—in order to write (2.5) more compactly:

$$|o\rangle = \sum_{\sigma} \left( \prod_{k=1}^L A^{[\sigma_k]} \right)_o |\sigma\rangle. \quad (2.6)$$

Additionally, we used the short-hand notation

$$|\sigma_k\rangle = |\sigma_k\rangle |\sigma_{k+1}\rangle \dots |\sigma_L\rangle \quad \text{and for } k=1 \quad |\sigma\rangle = |\sigma_1\rangle |\sigma_2\rangle \dots |\sigma_L\rangle. \quad (2.7)$$

The whole matrix product in (2.6) has only one index assigned. This comes from the fact that the  $A^{[\sigma_k]}$  have column-vector shape, as the  $L$ th site has no outer neighbour. So the

<sup>5</sup>Of course, this can be done for corresponding inner state space bases as well.

<sup>6</sup>Actually, we will even drop the summation symbol if the indices are just equivalent and no confusion is to be feared.

<sup>7</sup>So if we talk of a space  $|\sigma_k\rangle$ , we actually refer to the state space spanned by the basis  $|\sigma_k\rangle$ .

<sup>8</sup>But in the present context, while treating OSS bases, “inner” would refer to the “inner OSS” basis, e. g.  $|o_{k-1}\rangle$  in (2.4), and not to the basis spanning the inner state space of the current site. This convention allows us to use the standard nomenclature of “in” and “out”, regarding index directions, while still dealing only with outer (or, later on, inner) state space bases.

matrix product results actually in a vector (for every  $|\sigma\rangle$ ). We call (2.6) a matrix product state (MPS).

So far, we have done nothing to shrink the size of our Hilbert space, but the MPS structure provides an efficient tool for doing that. By setting some of the coefficients noted by the  $A$ -matrices to zero, i. e. limiting the size of the  $A$ -matrices, one can effectively truncate the described state space. The dimension of the effective Hilbert space is then given by the dimensions of the  $A$ -matrices. A very important question is how to truncate the Hilbert space in an efficient way so that one gets a reasonably good description of the ground state. We will address this question in section 2.2. Of course, neighbouring matrices must agree in their adjacent dimension. Thus we have an easily accessible parameter to control the effective state space size, the dimensions of the matrices.

### Graphical representation

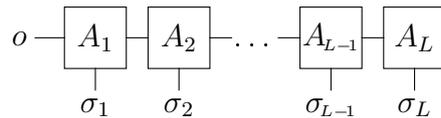


Figure 2.2: Graphical representation of the right-hand side of (2.6).

Matrix product states possess an intuitive graphical representation as shown in Figure 2.2. In this representation we depict the matrices by boxes and place the matrix identifier with just the site index  $k$  inside. All links from a box correspond to the indices of the matrix, where usually the left link corresponds to the “inner” index and the right one to the “outer” index<sup>9</sup>, and the link at the bottom of the boxes refers to the LSS index. Links that connect two boxes denote a contraction of the relevant indices, implying in most cases a matrix product. If we want to label the links explicitly, we place the index either inside the box, right next to the link, if the link is connected to another box, or at the open end of a link. There will be cases where we cannot use the matrix product notation and we have to use a general contraction. But by thinking of a contraction as a generalised matrix product, one can even stick with the same phrases.

To describe multiple chains, we reintroduce the additional indices  $\alpha = 1 \dots N$  and  $s = \uparrow, \downarrow$  to identify different chains

$$|o_{k-1,\alpha s}\rangle = \sum_{\sigma_{k\alpha s}} \left( \prod_{j=k}^L A^{[\sigma_{j\alpha s}]} \right)_{o_{k-1,\alpha s}} |\sigma_{k\alpha s}\rangle = P_{\alpha s}^{[\sigma_k]} |\sigma_{k\alpha s}\rangle, \quad (2.8)$$

<sup>9</sup>Context determines whether the indices refer to inner or outer state spaces.

where we used the following abbreviation<sup>10</sup>

$$P_{\alpha s}^k \equiv (P_{\alpha s}^{[\sigma_k]})_o \equiv P_{o_{\alpha s}}^{[\sigma_{k\alpha s}]} \equiv \left( \prod_{j=k}^L A^{[\sigma_{j\alpha s}]} \right)_o. \quad (2.9)$$

If  $k = 1$  in (2.9), we will drop the superscript<sup>12</sup>  $k$  of  $P$  completely, like in Figure 2.3, which

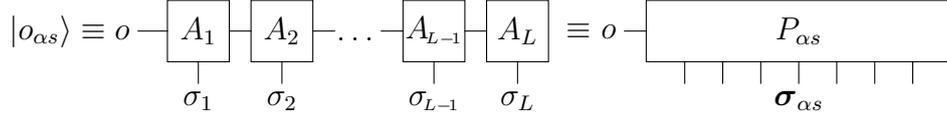


Figure 2.3: Graphical representation<sup>11</sup> of (2.8) for  $k = 1$ :  $|o_{\alpha s}\rangle = P_{\alpha s}|\sigma_{\alpha s}\rangle$ .

represents the equation  $|o_{\alpha s}\rangle = P_{\alpha s}|\sigma_{\alpha s}\rangle$ .

Similar to a chain site we introduce a coefficient “matrix” for the dot. But in contrast to a chain site, the effective Hilbert space at the dot is divided in  $2N$  effective chain state spaces  $|o_{\alpha s}\rangle$  (one for each chain and spin), which are independent from each other, and a LSS  $|\sigma_{\alpha s}\rangle$ , which typically is bigger than an ordinary chain site LSS for a single fermion. So the “dot-matrix”  $B$  has  $2N$  indices for the leads and a local index. With all these ingredients a state of the whole system can be written as a MPS:

$$\begin{aligned} |\psi\rangle &= P_{o_{1\uparrow}}^{[\sigma_{1\uparrow}]} P_{o_{2\uparrow}}^{[\sigma_{2\uparrow}]} \dots P_{o_{N\uparrow}}^{[\sigma_{N\uparrow}]} P_{o_{1\downarrow}}^{[\sigma_{1\downarrow}]} \dots P_{o_{N\downarrow}}^{[\sigma_{N\downarrow}]} B_{o_{1\uparrow}o_{2\uparrow}\dots o_{N\downarrow}}^{[\sigma_{0\uparrow}, \sigma_{0\downarrow}]} |\sigma_{0\uparrow}, \sigma_{0\downarrow}, \sigma_{1\uparrow}, \dots, \sigma_{N\downarrow}\rangle \\ &= B_{o_{1\uparrow}o_{2\uparrow}\dots o_{N\downarrow}}^{[\sigma_{0\uparrow}, \sigma_{0\downarrow}]} |o_{1\downarrow}\rangle |o_{2\uparrow}\rangle \dots |o_{N\uparrow}\rangle |o_{1\downarrow}\rangle \dots |o_{N\downarrow}\rangle |\sigma_{0\uparrow}, \sigma_{0\downarrow}\rangle, \end{aligned} \quad (2.10)$$

where  $|\sigma_{0s}\rangle = |\sigma_{d1s}, \sigma_{d2s}, \dots, \sigma_{dms}\rangle$  is the LSS of the dot composed of the state spaces of the  $m$  dot levels. Again, we use the fact that spin up and spin down electrons interact only via the Coulomb term to split the big  $B$ -matrix into two parts<sup>13</sup>. This turns out to be a huge numerical advantage (see section 2.3.5).

$$B_{o_{1\uparrow}\dots o_{N\uparrow}o_{1\downarrow}\dots o_{N\downarrow}}^{[\sigma_{0\uparrow}, \sigma_{0\downarrow}]} \equiv A_{o_{1\uparrow}\dots o_{N\uparrow}v}^{[\sigma_{0\uparrow}]} A_{o_{1\downarrow}\dots o_{N\downarrow}v}^{[\sigma_{0\downarrow}]} \quad (2.11)$$

Note the additional index  $v$  at both dot  $A$ -matrices that links spin up and spin down parts together (see Figure 2.4).

So far we have introduced a generic MPS for a multi-level, multi-lead Anderson model. Within this work we have implemented a spinful 2-lead Anderson model with 2 and 4 dot levels. So for clarity, we will from now on restrict our notation to the 2-level 2-lead case and use mostly the graphical representation (Figure 2.5), but the ensuing development can easily be generalised to the general  $N$ -lead case. Note that using the graphical representation is as precise as using an explicit notation like in (2.8) but far more concise. It is always

<sup>10</sup>We also introduce more and less detailed variants, which will be used as necessary.

<sup>12</sup>We will also drop the  $k$  subscript in  $|o_{k-1, \alpha s}\rangle$  for  $k = 1$ , i. e. instead of  $|o_{0\alpha s}\rangle$  we will just write  $|o_{\alpha s}\rangle$ .

<sup>13</sup>Actually this structure of the Hamiltonian is not a prerequisite for being able to split  $B$ , but because of it the splitting of  $B$  will turn out to be a very efficient choice.

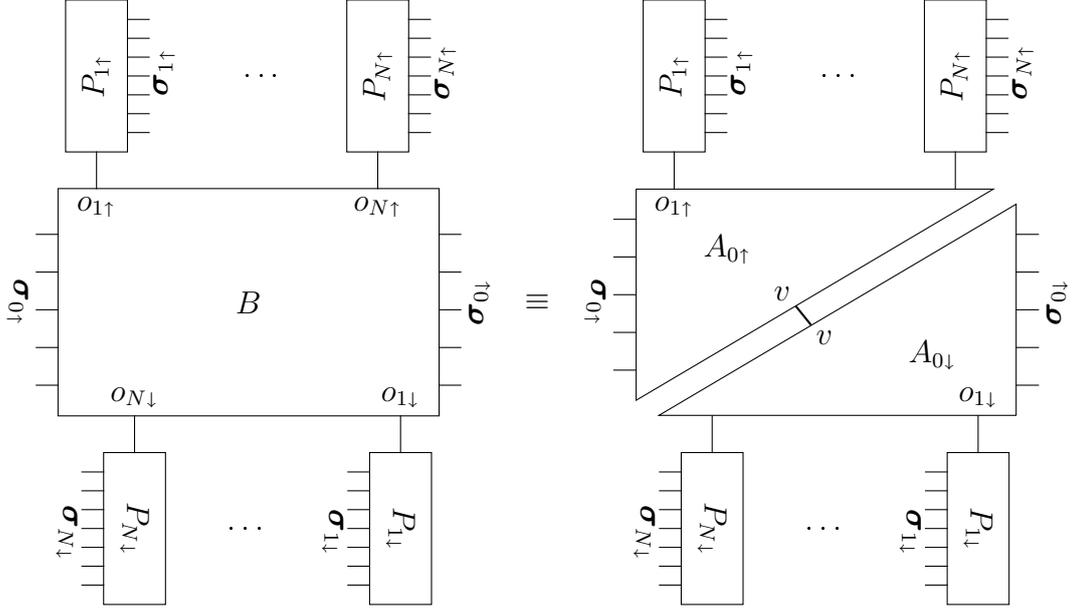


Figure 2.4: Graphical MPS representation for a multi-level multi-lead Anderson model.

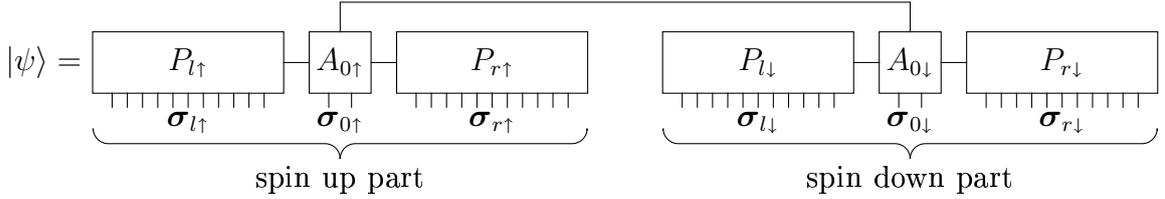


Figure 2.5: Graphical MPS representation of  $|\psi\rangle$  as in equation (2.10) with the dot matrix split in two for the 2-level 2-lead Anderson model, which we will use from now on.

possible to translate the graphics back to explicit formulas using the rules we defined so far.

Of course, there are numerical limitations for calculating properties of a multiple-lead model, and we will comment on this when we discuss algorithmic details. From now on, we refer to our leads as “left” ( $l$ ) and “right” ( $r$ ) lead and the chains are labelled accordingly as  $r\uparrow$ ,  $l\uparrow$ ,  $r\downarrow$ , and  $l\downarrow$ . If we do not explicitly label the various parts of a MPS in a graphical representation, we assume labels as in Figure 2.5. In cases where we do want to use the explicit notation of (2.10), we will use a simplified<sup>14</sup> symbolic variant to denote a MPS

$$|\psi\rangle = \left( \prod_k A^{[\sigma_k]} \right) |\sigma\rangle. \quad (2.12)$$

<sup>14</sup>Otherwise one gets easily lost in all the indices, which would appear in the equations but are not important at all.

### Inner state space basis

We have used the effective OSS bases for all previous statements. The construction of effective ISS is in principle analogous:

$$|i_{k+1}\rangle = A_{i_k i_{k+1}}^{[\sigma_k]} |i_k\rangle |\sigma_k\rangle \quad (2.13)$$

We illustrate our labelling conventions regarding ISS bases in Figure 2.6. There is only

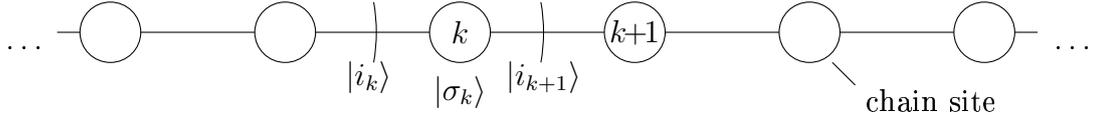


Figure 2.6: Part of a chain to illustrate how to determine an ISS basis  $|i_{k+1}\rangle$  of site  $k + 1$  from the inner and local bases of site  $k$ .

one difference, namely the beginning of the chain.

For building up effective OSS bases we started at the end of a certain Wilson chain, where there is no more site further out. That made it easy to build successive OSS bases starting from the end of the chain.

This is different for the ISS bases, which always have the dot (and the other chains attached to the dot) at the inner side of the first chain site. So in order to build successive ISS bases for a certain chain, we first need to determine the ISS basis  $|i_{1\alpha s}\rangle \equiv |i_{\alpha s}\rangle$  of the first chain site<sup>15</sup>. This basis provides an effective description for the dot and all the other chains coupled to the dot.

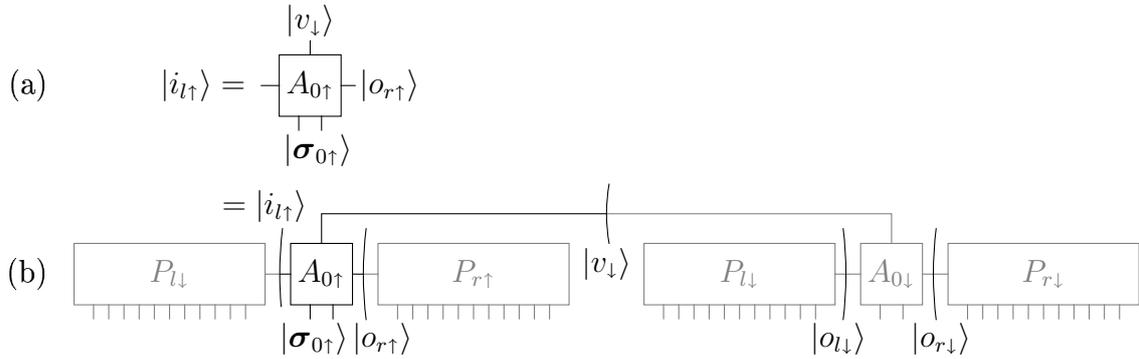


Figure 2.7: Diagrams for calculating the ISS bases for the first site of the left spin up chain. The gray parts in (b) indicate what the shorthand sketch (a) represents when written out explicitly.

To accomplish this, we have to obtain effective basis sets describing all the other chains, i. e. the effective OSS bases  $|o_{\alpha s}\rangle$  for their first sites. With these descriptions of the chains

<sup>15</sup>We drop the  $k$  index of  $|i_{k\alpha s}\rangle$  for  $k = 1$ .

combined with the local dot basis, we construct a new effective basis describing both the dot and the other chains (Figure 2.7):

$$|i_{l\uparrow}\rangle = A_{i_{l\uparrow}o_{r\uparrow}v_{l\downarrow}}^{[\sigma_{0\uparrow}]} |o_{r\uparrow}\rangle |v_{l\downarrow}\rangle |\sigma_{0\uparrow}\rangle \quad \text{with} \quad |v_{l\downarrow}\rangle = A_{o_{l\downarrow}o_{r\downarrow}v_{l\downarrow}}^{[\sigma_{0\downarrow}]} |o_{l\downarrow}\rangle |o_{r\downarrow}\rangle |\sigma_{0\downarrow}\rangle \quad (2.14)$$

This new effective basis  $|i\rangle$  can be used as starting point for calculating the other effective ISS bases along this chain, which works in complete analogy to the OSS case.

### 2.1.3 Useful techniques

Before we can go on and explain how to use matrix product states to get a good approximation for the groundstate, we need to introduce various techniques of handling matrix product states.

#### Scalar product

$$\langle \psi' | \psi \rangle = \langle \sigma' | \left( \prod_{k'} A'^{[\sigma'_{k'}]} \right) \left( \prod_k A^{[\sigma_k]} \right) | \sigma \rangle \quad (2.15)$$

The composed local state space basis is of course orthonormal:  $\langle \sigma_\alpha | \sigma_\beta \rangle = \delta_{\alpha\beta}$ . Thus we find that for calculating the scalar product of two MPS we “just”<sup>16</sup> need to contract every corresponding index pair as shown in (Figure 2.8).

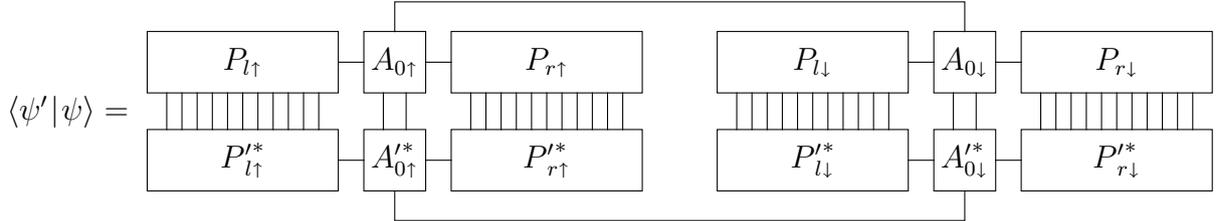


Figure 2.8: Scalar product  $\langle \psi' | \psi \rangle$  in MPS language.

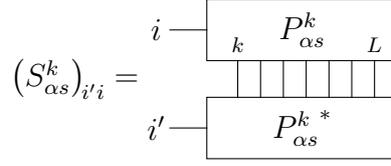
#### Partial product

The norm is defined as usual  $\|\psi\| = \langle \psi | \psi \rangle$ . It turns out to be useful to define a partial product (norm)  $S_{\alpha s}^k$  of a MPS<sup>17</sup> (see Figure 2.9 for MPS expression)

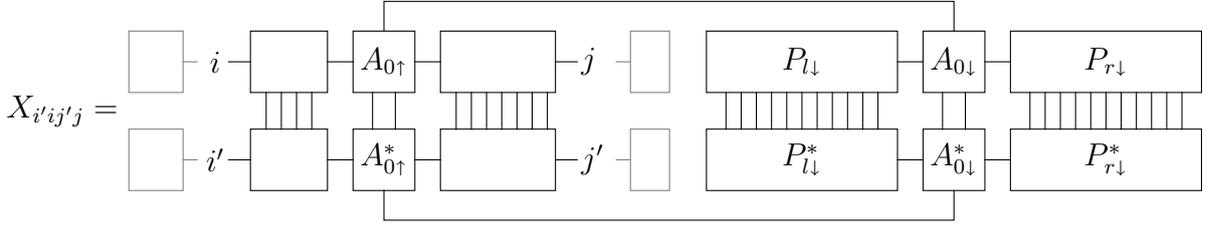
$$S_{\alpha s}^k \equiv (S_{\alpha s}^k)_{i'i} \equiv \sum_{\sigma_k} (P_{\alpha s}^{[\sigma_k]})_i (P_{\alpha s}^{[\sigma_k]*})_{i'}. \quad (2.16)$$

<sup>16</sup>While in principle the order in which we carry out the contractions is irrelevant. But we try to keep the size of intermediate results as small as possible, as the computational cost for contractions grows cubic with tensor dimensions and we might run into additional memory problems.

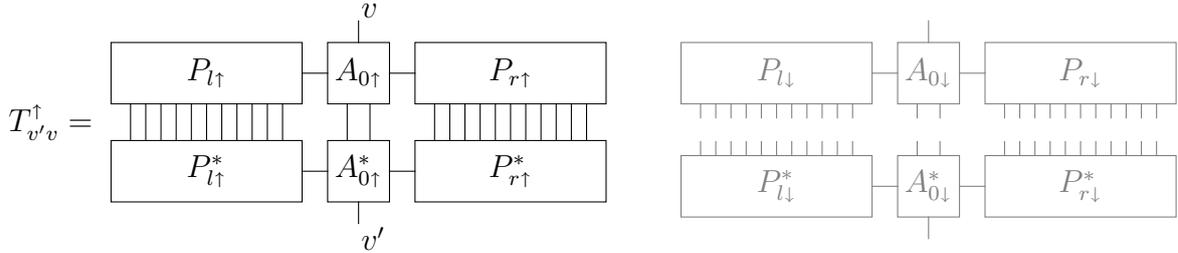
<sup>17</sup>Note that there is no summation over  $\alpha$  and  $s$  of the right hand side as these specify the lead


 Figure 2.9: MPS representation of the partial product  $(S_{\alpha s}^k)_{i'i}$ .

The definition of a partial product (2.16) starts with the contraction always at the end of a chain, as it is based on the definition of the  $P_{\alpha s}^k$ . But this restriction can easily be lifted and one can define arbitrary index ranges (even beyond single chains) whose local indices are to be contracted out. The result of such a procedure is an object (Figure 2.10) with four remaining indices  $X_{i'ij'j}$


 Figure 2.10: Generalised partial product  $X_{i'ij'j}$ . The parts of the MPS that are not contracted out are put in gray.

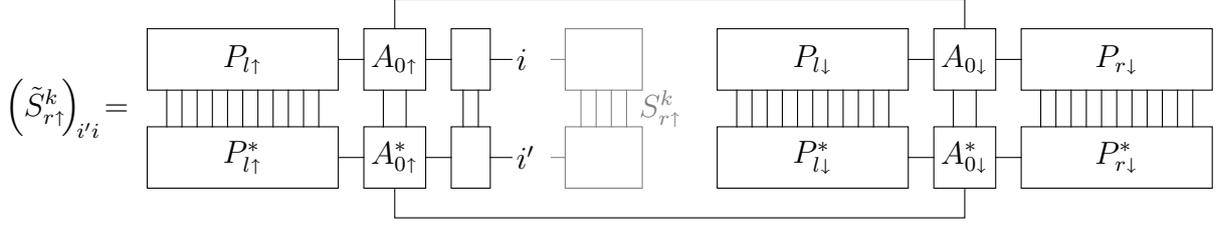
Other useful partial products are the contraction of a whole spin subsystem  $T_{v'v}^s$  (Figure 2.11) and  $\tilde{S}_{\alpha s}^k$ , the “complement” of  $S_{\alpha s}^k$  (Figure 2.12).


 Figure 2.11: Product of spin up subsystem  $T_{v'v}^{\uparrow}$ .

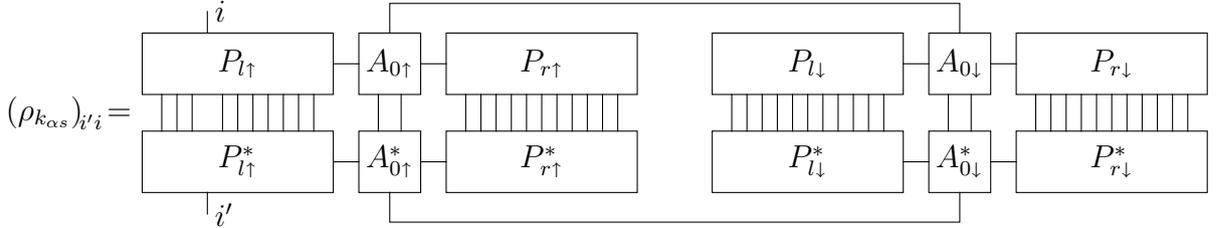
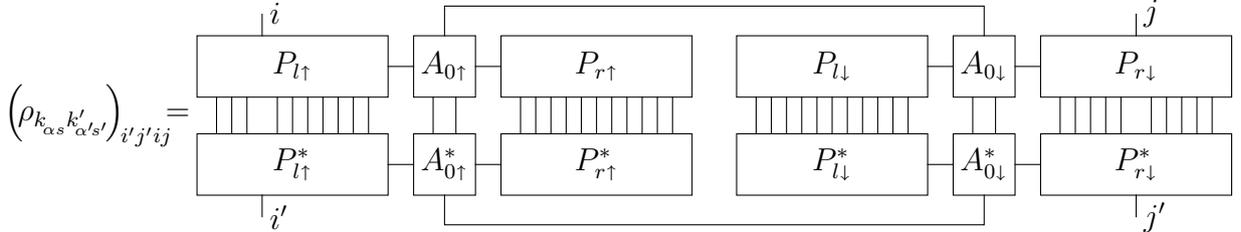
## Density matrix

One can easily build a density matrix out of a MPS:

$$\rho = |\psi\rangle\langle\psi| = \left( \prod_{k'} A_{[\sigma_{k'}]}^{[\sigma_{k'}]^*} \right) \left( \prod_k A_{[\sigma_k]} \right) |\sigma'\rangle\langle\sigma|. \quad (2.17)$$

Figure 2.12: Complementary partial product  $(\tilde{S}_{r\uparrow}^k)_{i'i}$ .

The reduced density matrix  $\rho_{k_{\alpha s}}$  for site  $k_{\alpha s}$  can be obtained by tracing out all the other local indices (Figure 2.13). We call  $\rho_{k_{\alpha s}}$  also single-site density matrix, to distinguish it from the two-site density matrix  $\rho_{k_{\alpha s} k'_{\alpha' s'}}$ , which is the reduced density matrix for sites  $k_{\alpha s}$  and  $k'_{\alpha' s'}$  (Figure 2.14).

Figure 2.13: One-site density matrix  $(\rho_{k_{\alpha s}})_{i'i}$ .Figure 2.14: Two-site density matrix  $(\rho_{k_{\alpha s} k'_{\alpha' s'}})_{i'j'ij}$ .

### 2.1.4 Interpretation of the $A$ -matrices

So far, we have shown in section 2.1.2 how to construct an MPS and how the state of the whole system can be described by a product of all the  $A$ -matrices. In this picture, we have introduced an interpretation for the  $A$ -matrices as a basis transformation from site to site in a very general form<sup>18</sup>. We also have shown how to calculate effective bases for the inner

<sup>18</sup>The  $A$ s may represent any linear transformation, we did not demand unitarity of the  $A$ -matrices, so far.

and outer state space of a certain site. With the knowledge of these bases together with the local basis, we can represent<sup>19</sup> a state of the whole system in a local manner

$$|\psi\rangle = A_{io}^{[\sigma_k]} |i\rangle |o\rangle |\sigma_k\rangle. \quad (2.18)$$

This description holds for every site  $k$ . We thus obtain a second interpretation for the  $A_k$ , as coefficients needed to construct a system state  $|\psi\rangle$  from the inner, outer and local basis states. (2.18) is really just a different interpretation of (2.10). If  $|o\rangle$  and  $|i\rangle$  are replaced with their explicit forms similar to (2.6), but different index regimes for  $k$ , (2.10) is recovered. The product of all the other matrices is just split in parts and hidden in the effective inner and outer basis states.

### 2.1.5 Orthonormal basis states

For every site we can construct effective basis states for the inner and outer parts of the system in such a way, that together with the local basis they span the effective Hilbert space for the whole system. We call this special site, for which we use this description, the “current site” and  $k$  ( $k_{\alpha s}$ ) the “current index”. A priori only the local basis states  $|\sigma_k\rangle$  are guaranteed to be orthonormal. Now, it will be convenient to demand that the basis states of the inner and outer bases of the current site  $|\sigma_k\rangle$ , namely  $|i\rangle$  and  $|o\rangle$ , are orthonormal, too, i. e. that they obey

$$\langle o|o'\rangle = \delta_{oo'} \quad (2.19a)$$

$$\langle i|i'\rangle = \delta_{ii'}. \quad (2.19b)$$

From this requirement we derive a condition on the MPS:

$$S_{\alpha s}^k = \mathbb{1} \quad \text{by (2.19a)} \quad (2.20a)$$

$$\tilde{S}_{\alpha s}^{k-1} = \mathbb{1} \quad \text{by (2.19b)}. \quad (2.20b)$$

These conditions have different implications for the  $A$ 's from different parts of the system. Let us focus for the moment<sup>20</sup> on  $A$ -matrices from the outer part. (2.20a) is fulfilled if we demand every  $A$ -matrix in the outer part to be “row unitary”

$$\sum_{\sigma_{k'}} A^{[\sigma_{k'}]} A^{[\sigma_{k'}]\dagger} = \mathbb{1} \quad \text{for } k' > k, \quad (2.21)$$

because then  $A^{[\sigma_{k'}]}$  converts one orthonormal basis,  $|o_{k'}\rangle$ , into another,  $|o_{k'-1}\rangle$ . In order to better understand, what this condition states, we rearrange the indices of  $A$  and introduce

<sup>19</sup>As we applied the Jordan-Wigner transformation to our Hamiltonian, we are free to choose any ordering of the state vectors. We place the non local state vectors in the same order as their corresponding indices of  $A$  are ordered, the local state vectors are placed last.

<sup>20</sup>The same reasoning applies to all other matrices, too, there are mainly technical differences. We will provide the details in appendix A.2.1.

a “superindex”  $x = (o, \sigma)$ . This rearranging of  $A$  does not change  $A$  at all. But it enables us to use standard terminology and well known definitions for two-dimensional matrices. With that we can rewrite the original three-dimensional  $A$  as follows:

$$A_{io}^{[\sigma_{k'}]} = A_{i,(o,\sigma)}^{k'} = A_{ix}^{k'}. \quad (2.22)$$

We use still the same symbol  $A$ , because we change only the notation but not the object itself. Now (2.21) can be written in ordinary matrix form (for a graphical representation see Figure 2.15a)

$$AA^\dagger = \mathbb{1}. \quad (2.23)$$

This is, what we call “row unitary”.  $A$  is not a normal unitary matrix as  $A$  is usually not a square matrix. However, it is subunitary, as the row vectors of  $A_{ix}$  are orthonormal. Note that the partial product of (2.16) reduces to

$$(S_{\alpha s}^k)_{i'i} \equiv \delta_{i'i} \quad (2.24)$$

if all  $A$ -matrices involved in it are now row unitary.

By analogous reasoning, the orthonormality of the inner basis, (2.19b), is guaranteed if the matrices for  $k' < k$  satisfy column unitarity (Figure 2.15b),

$$\sum_{\sigma_{k'}} A^{[\sigma_{k'}]} A^{[\sigma_{k'}]\dagger} = A^\dagger A = \mathbb{1}. \quad (2.25)$$



Figure 2.15: Orthonormality conditions for outer and inner  $A$ -matrices. Gray lines indicate orthogonal vectors.

### 2.1.6 Operator representation

We will now discuss how to obtain a representation of an operator in an effective basis. Usually we know the representation of an operator  $\hat{B}$  in the LSS basis<sup>21</sup>:

$$B_{\sigma'\sigma} = \langle \sigma' | \hat{B} | \sigma \rangle \quad (2.26)$$

We assume that  $\hat{B}$  acts locally on site  $k$  and the basis  $|o_k\rangle$  is orthonormal. If the  $|o_k\rangle$  are not orthonormal, we have to replace the  $\delta_{o'_k o_k}$ , which we used in order to arrive at (2.27),

<sup>21</sup>For clarity we focus here on operators that act only upon one site. But a generalisation to operators acting upon several sites straightforward.

by the partial scalar product  $(S^k)_{o'_k o_k}$ . See also Figure 2.17 for an example. To calculate the matrix elements of  $\hat{B}$  in the effective outer state space basis (see also Figure 2.16) of site  $k-1$  one uses (2.4) twice:

$$B_{o'_{k-1} o_{k-1}} = \langle o'_{k-1} | \hat{B} | o_{k-1} \rangle = A_{o_{k-1} o_k}^{[\sigma_k]} A_{o'_{k-1} o_k}^{[\sigma'_k]^*} B_{\sigma'_k \sigma_k}. \quad (2.27)$$

Similarly we get for an operator  $\hat{C}$  that is given in the outer basis of site  $k$

$$C_{o'_{k-1} o_{k-1}} = \langle o'_{k-1} | \hat{C} | o_{k-1} \rangle = A_{o_{k-1} o_k}^{[\sigma_k]} A_{o'_{k-1} o_k}^{[\sigma'_k]^*} C_{o'_k o_k}. \quad (2.28)$$

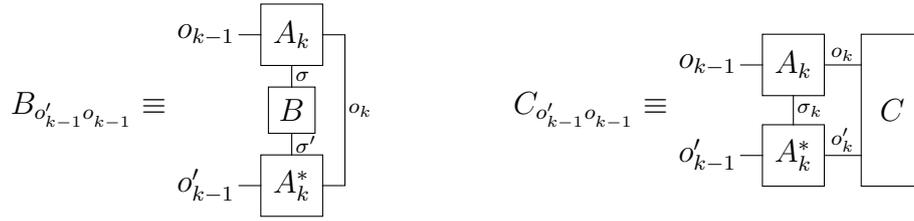


Figure 2.16: Basis transformation of operator representations

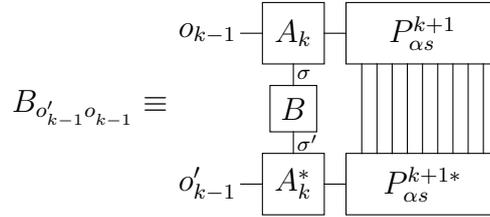


Figure 2.17: Example for a basis transformation of an operator in case the OSS basis is not orthonormal.

### 2.1.7 Evaluation of an operator

When dealing with the nearest neighbour hopping Hamiltonian we need to be able to calculate terms like  $c_k |\psi\rangle$  and  $c_k^\dagger c_{k+1} |\psi\rangle$ . There are two ways to look at this problem.

#### Global view

With this way we do not need to worry about orthonormalisation of the MPS. The operators are represented in their LSS basis,

$$\begin{aligned} \langle \sigma'_k | c_k^\dagger | \sigma_k \rangle &= \left( c_k^\dagger \right)_{\sigma'_k \sigma_k} \quad \text{and} \\ \langle \sigma'_{k+1} | c_{k+1} | \sigma_{k+1} \rangle &= \left( c_{k+1} \right)_{\sigma'_{k+1} \sigma_{k+1}}. \end{aligned} \quad (2.29)$$

Now these operators are contracted directly with the corresponding indices of the MPS (we use the symbolic notation of (2.12), see Figure 2.18 for graphical version)

$$c_k^\dagger c_{k+1} |\psi\rangle = \left( \prod_{k_1 < k} A^{[\sigma_{k_1}]} \right) (c_k^\dagger)_{\sigma_k \sigma'_k} A^{[\sigma'_k]} (c_{k+1})_{\sigma_{k+1} \sigma'_{k+1}} A^{[\sigma'_{k+1}]} \left( \prod_{k_2 > k+1} A^{[\sigma_{k_2}]} \right) |\sigma_{k_1}\rangle |\sigma_k, \sigma_{k+1}\rangle |\sigma_{k_2}\rangle. \quad (2.30)$$

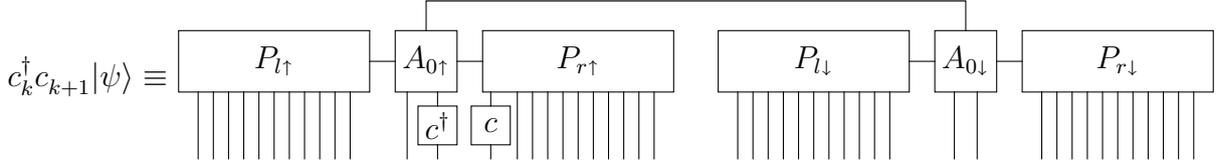


Figure 2.18: Global way for calculating  $c_k^\dagger c_{k+1} |\psi\rangle$ .

### Local view

This view uses the local interpretation of the  $A$ -matrices. Therefore we need to provide a representation of the operators with which we want to act on  $|\psi\rangle$  in the appropriate local effective bases. It is convenient to choose a  $A$ -matrix such that one operator can be represented via the local  $|\sigma_k\rangle$  space. In this example we choose  $k+1$  as current site index (see also Figure 2.19). If any of the effective bases are not orthonormal, this will only be reflected in the representation of the operators we use.

$$c_k^\dagger c_{k+1} |\psi\rangle = (c_k^\dagger)_{ii'} (c_{k+1})_{\sigma_{k+1} \sigma'_{k+1}} A_{i'o}^{[\sigma'_{k+1}]} |i\rangle |o\rangle |\sigma_{k+1}\rangle \quad (2.31)$$

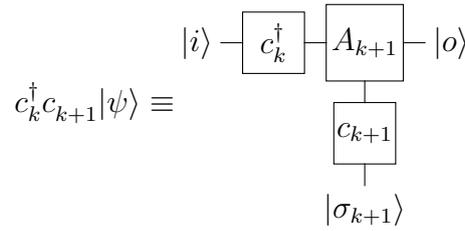


Figure 2.19: Local way for calculating  $c_k^\dagger c_{k+1} |\psi\rangle$ ; graphical representation of (2.31).

### 2.1.8 Expectation values

From the global view it is straightforward to see how expectation values have to be calculated. One has to calculate the scalar product of  $|\psi\rangle$  with  $c_k^\dagger c_{k+1} |\psi\rangle$  as shown in Figure 2.20.

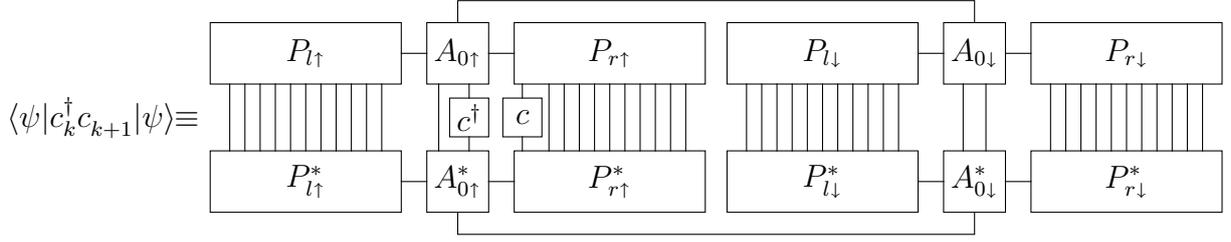


Figure 2.20: Global way for calculating the expectation value  $\langle \psi | c_k^\dagger c_{k+1} | \psi \rangle$ .

In case of the Hamilton operator, which acts nontrivially on all sites,  $\langle \psi | \hat{H} | \psi \rangle$  can be visualised like in Figure 2.21. However, it is still important that the Hamiltonian breaks up into simple (few-operator) local expressions for this contraction scheme to be doable in practice.

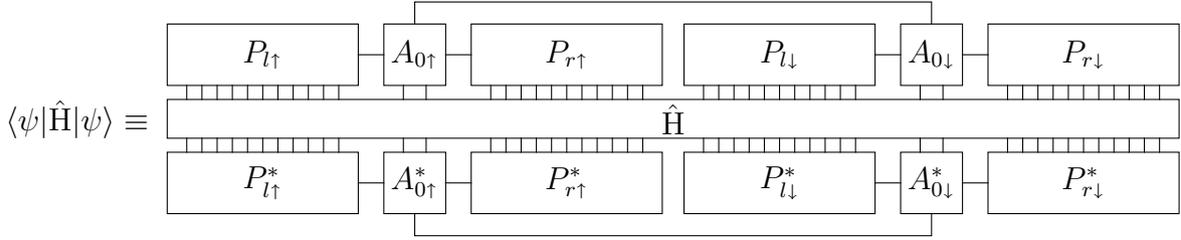


Figure 2.21: Global way for calculating the expectation of the Hamiltonian,  $\langle \psi | \hat{H} | \psi \rangle$ .

For calculating an expectation value the local way one needs to multiply (2.31) with  $\langle \psi |$  in the local picture. Assuming the local bases to be orthonormal<sup>22</sup>, we get (see Figure 2.22)

$$\langle \psi | c_k^\dagger c_{k+1} | \psi \rangle = A_{io}^{[\sigma_{k+1}]^*} \left( c_k^\dagger \right)_{ii'} (c_{k+1})_{\sigma_{k+1}\sigma'_{k+1}} A_{i'o}^{[\sigma'_{k+1}]}. \quad (2.32)$$

Since both methods of calculating expectation values are equivalent, the local variant is numerically far more efficient as it involves fewer matrix multiplications compared to the global method. Consequently, it is very important for an efficient method to work locally throughout keeping the relevant operator representations. This directly leads to an iterative method.

## Effects of Jordan-Wigner Transformation

In the above examples we did not care about fermionic ordering. This is fine for the linear portions of the chains, as we are using the Jordan-Wigner Transformation (JWT), see appendix A.1. For the dot region, however, a few extra thoughts are necessary. On

<sup>22</sup>Later on we will always have orthonormal bases. In cases where this assumption is not valid, one simply replaces Kronecker-deltas with the appropriate partial scalar products.

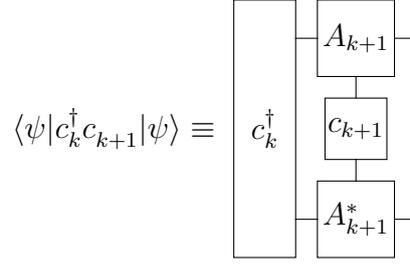


Figure 2.22: Local calculation of the expectation value  $\langle \psi | c_k^\dagger c_{k+1} | \psi \rangle$ .

the dot, the hopping terms to the leads are not necessarily nearest neighbour terms with respect to the JWT numbering scheme.

We will now calculate the term for the coupling of the first<sup>23</sup> dot level to the right chain in order to demonstrate the effects of the JWT, which occur in such a case. For pointing out the difference between the JWT operators and the original ones we denote them by  $\tilde{c}$ . Using  $k_1, k_2$  to denote the dot level one and two and assuming properly orthonormalised bases, we can write (see Figure 2.23)

$$\langle \psi | c_{k_1}^\dagger c_{k_2} | \psi \rangle = \langle \psi | \tilde{c}_{k_1}^\dagger p_{k_2} \tilde{c}_{k_2} | \psi \rangle = A_{o_l o_r v}^{[\sigma_{k_1}, \sigma_{k_2}]} \left( \tilde{c}_{k_1}^\dagger \right)_{\sigma_{k_1} \sigma'_{k_1}} (p_{k_2})_{\sigma_{k_2} \sigma'_{k_2}} \left( \tilde{c}_{k_2} \right)_{o_r o'_r} A_{o_l o'_l v}^{[\sigma'_{k_1}, \sigma'_{k_2}]^*}, \quad (2.33)$$

where  $p_k = 1 - 2c_k^\dagger c_k$  is the JWT correction term. The calculation of hoppings from the dot to the first site in the lead introduces extra  $p$  operators which are to be included for the sites intermediate in the JWT ordering.

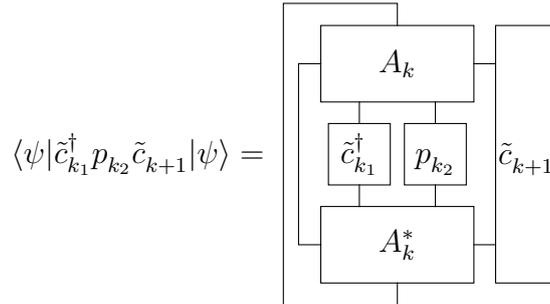


Figure 2.23: Local calculation of the expectation value  $\langle \psi | \tilde{c}_{k_1}^\dagger p_{k_2} \tilde{c}_{k_2} | \psi \rangle$ .

<sup>23</sup>Our site indices we use here cannot distinguish between the dot levels, but the graphical representation indicates which level we refer to. Of course, in the actual implementation one has to use a different indexing scheme, in order to correctly implement JWT, but this is merely bookkeeping.

## 2.2 Hilbert space truncation

We now show how to overcome the limitations of the NRG truncation scheme that keeps simply the  $D$  states with lowest energy after each iteration. For this we utilise the DMRG Hilbert space truncation scheme<sup>24</sup>, and choose which states to keep according to their weight in the density matrix. Our goal is to keep all these states in the effective Hilbert spaces, whose weight in the density matrix is larger than a certain threshold  $w_{\min}$ . We usually choose  $w_{\min} = 10^{-6}$ , which turns out to provide a good compromise between accuracy and manageable numerical resources. We will now show how to maintain this criterion while orthonormalising the effective basis states to obey (2.19)

### 2.2.1 Construction of the density matrix

To apply the DMRG truncation idea, we need an expression for the density matrix in the local picture of (2.18):

$$\rho = |\psi\rangle\langle\psi| = A_i^{[\sigma_k]} A_{i'o'}^{[\sigma'_k]*} |i\rangle\langle i'| |o\rangle\langle o'| |\sigma_k\rangle\langle\sigma'_k|. \quad (2.34)$$

Suppose we want to change the current site from  $k$  to  $k-1$ , therefore we need a new orthonormalised outer basis  $|o_{k-1}\rangle$ . To achieve this we need to replace  $A_k$  by a row unitary matrix  $\tilde{A}_k$  in order to fulfill (2.21). The density matrix  $\rho$  of (2.34) describes the whole system. If we trace over the inner system states  $|i\rangle$ , we end up with the reduced density matrix for the current site and the outer part of the system

$$\rho^{(i)} = \text{tr}_i \rho = A_i^{[\sigma_k]} A_{i'o'}^{[\sigma'_k]*} |o\rangle\langle o'| |\sigma_k\rangle\langle\sigma'_k|, \quad (2.35)$$

which is exactly what we want to describe with the new OSS basis  $|o_{k-1}\rangle$ . Once again we introduce a superindex  $x \equiv (o, \sigma_k)$  in order to use conventional matrices for notation

$$\rho^{(i)} = A_{ix} A_{ix'}^* |x\rangle\langle x| = A^\dagger A. \quad (2.36)$$

In principle one would now diagonalise  $\rho^{(i)}$ , determine the eigenstates  $|\tilde{x}\rangle$  with the biggest weights and finally choose a transformation that keeps exactly these states. It turns out that this can be accomplished without even calculating  $\rho^{(i)}$  explicitly using singular value decomposition (SVD).

### 2.2.2 Singular value decomposition

The SVD is a very robust decomposition method for rectangular matrices, see [20, 21] for details and proofs. Any  $n \times n'$  matrix  $A$  can be written in the following form, called SVD:

$$A = USV^\dagger, \quad (2.37)$$

---

<sup>24</sup>We will not elaborate on DMRG here, see [6, 7, 8] for details.

where  $S = \text{diag}(s_1, s_2, \dots, s_l)$  with  $s_1 \geq s_2 \geq \dots \geq s_l$  and  $l = \min(n, n')$  is a diagonal matrix that contains the descending ‘‘singular values’’ of  $A$ ,  $U$  and  $V^\dagger$  are column and row unitary matrices, respectively, and obey  $U^\dagger U = V^\dagger V = \mathbb{1}$ . If  $n \geq n'$ ,  $U$  is a  $n \times n'$  matrix and  $V^\dagger$  is  $n' \times n'$ . In the case  $n < n'$ ,  $U$  is a square matrix of size  $n \times n$  and  $V^\dagger$  is  $n \times n'$ , see Figure 2.24 for visualisation.

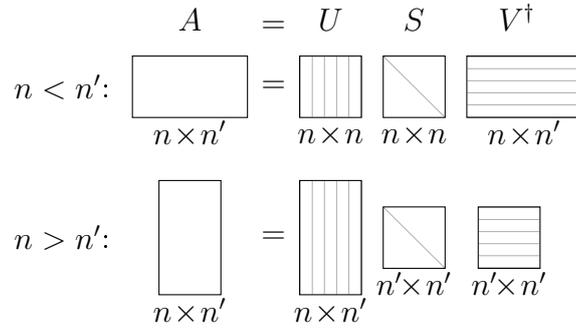


Figure 2.24: Singular value decomposition for rectangular matrices.

Constructing the reduced density matrix  $\rho^{(i)}$  out of  $A = A_{ix}$ , writing  $x \equiv (o, \sigma_k)$ , using (2.36), and applying the SVD to  $A$ , we obtain

$$\rho^{(i)} \equiv A^\dagger A = V S U^\dagger U S V^\dagger = V S^2 V^\dagger. \quad (2.38)$$

So by calculating the SVD of  $A$  we implicitly diagonalise  $\rho^{(i)}$  and from the rows of  $V^\dagger$  we obtain a set of orthonormal basis states that are ordered according to their weight in the density matrix.

To keep only states whose weight is larger than some threshold  $w_{\min}$ , we can simply set all singular values  $s_i = 0$  for  $s_i^2 < w_{\min}$ . Equivalently we can set the dimension  $n = D$  when we build new OSS basis for site  $k - 1$ , where  $s_D$  is the smallest singular value we would keep. In practice we work the other way round. We set  $n = D$  from the beginning and check for every site whether

$$s_D^2 < w_{\min} \quad (2.39)$$

holds. If the smallest singular value is too big, we need to increase  $D$ . Generally we try to keep  $D$  as small as possible while preserving (2.39), because the value of  $D$  has a big impact on the numerical performance of our optimisation scheme as the cost of the algorithm scales like  $\mathcal{O}(D^3)$  as we will show in section 2.3.5.

An OSS basis for site  $k - 1$  that is optimal in DMRG sense is thus given by

$$|o_{k-1}\rangle = V_{o_{k-1}, (o_k, \sigma_k)}^\dagger |o_k\rangle |\sigma_k\rangle. \quad (2.40)$$

### 2.2.3 Orthonormalisation & truncation scheme

We are now finally in the position to specify explicitly how to implement the orthonormalisation procedure of our MPS in a ‘‘DMRG-optimal’’ way. An important point to notice

is that, if we demand the effective bases of the current site in the local picture of (2.18) to be orthonormal, the matrix  $A_k$  from (2.18) will be the only one not to be orthonormal in any direction, as it carries the coefficients that connect all incoming basis states. All other  $A$ -matrices are part of the description of the effective bases for the inner and outer state spaces, or in case of a dot matrix, of the lead and the other spin state space. In a proper orthonormalised MPS all other matrices are orthonormal towards the current index  $k$ .

Assume that we want to change the current site from  $k$  to  $k - 1$ . This requires to rewrite the product  $A_{k-1}A_k$  into a different product  $\tilde{A}_{k-1}\tilde{A}_k$  that describes the same state yet ensures that the inner and outer bases are now orthonormal with respect to the new current site  $k - 1$ . To be explicit,  $\tilde{A}_k$  now needs to fulfill (2.23), which was not the case while  $k$  was the current index. On the other side we can lift the orthonormality condition for  $A_{k-1}$  and replace it with  $\tilde{A}_{k-1}$ , as now  $k - 1$  is the current index. To ensure that the overall state  $|\psi\rangle$  does not change, we get as a basic condition for our procedure (introducing the superindices  $x = (i, \sigma_{k-1})$  and  $y = (o, \sigma_k)$ ):

$$A_{i\alpha}^{[\sigma_{k-1}]} A_{\alpha o}^{[\sigma_k]} = A_{x\alpha}^{k-1} A_{\alpha y}^k = A_{k-1}A_k = \text{unchanged}, \quad (2.41)$$

where the sum over  $\alpha$  yields the usual matrix product.

Our demand for a ‘‘DMRG-optimal’’ truncation already determines  $\tilde{A}_k$  through (2.40):

$$\tilde{A}_k = V_{i,(o,\sigma_k)}^\dagger, \quad (2.42)$$

where  $A_k = USV^\dagger$  is the SVD of  $A_{i,(o,\sigma_k)}^k$ . The condition for an unchanged overall state then determines  $\tilde{A}_{k-1}$  (see also Figure 2.25):

$$\tilde{A}_{k-1} = A_{k-1}US. \quad (2.43)$$

With this choice we obviously obtain an new ‘‘DMRG-optimal’’ orthonormal OSS basis as in Figure 2.25 and keep the overall state  $|\psi\rangle$  unchanged.

$$A_{k-1}A_k = \begin{array}{|c|} \hline \text{A}_{k-1} \\ \hline \end{array} \begin{array}{|c|} \hline \text{A}_k \\ \hline \end{array} = \begin{array}{|c|} \hline \text{A}_{k-1} \\ \hline \end{array} \begin{array}{|c|} \hline US \\ \hline \end{array} \begin{array}{|c|} \hline V^\dagger \\ \hline \end{array} = \begin{array}{|c|} \hline \tilde{\text{A}}_{k-1} \\ \hline \end{array} \begin{array}{|c|} \hline \tilde{\text{A}}_k \\ \hline \end{array} = \tilde{A}_{k-1}\tilde{A}_k$$

Figure 2.25: Orthonormalisation of  $A_k$  towards the inner direction. The matrices that are not orthonormalised in any direction are printed with gray background. The gray lines within the boxes indicate whether the row or column vectors are orthonormal.

Let  $D_k$ ,  $D_{k+1}$  and  $d$  be the dimensions of the inner, outer and local state spaces of  $A_k$ , so  $A_k$  has the size  $D_k \times D_{k+1} \times d$ . The LSS ( $d = 2$ ) is fixed<sup>25</sup>, but we are free to choose  $D_k$  and  $D_{k+1}$ . So by choosing  $D_k < D_{k+1}d$  we automatically truncate the OSS for site  $k - 1$ . Our particular choice (2.42) and (2.43) for orthonormalising  $A_k$  guarantees that we keep

<sup>25</sup>We implement the dot matrices to have  $m$  local indices of dimension  $d = 2$ , where  $m$  is the number of dot levels.

all states with finite weights in the density matrix. Note that this is somewhat different to conventional DMRG, where by addressing two neighbouring sites at the same time there is explicit truncation indeed. In our case, however, SVD just provides a convenient and an exact way of orthonormalising the  $A$ -matrices without changing the overall state.

### 2.2.4 Orthonormalisation towards other indices

The above procedure, which led to a “DMRG-optimal” OSS basis for site  $k - 1$ , can easily be generalised for an orthonormalisation towards any index. The main idea for orthonormalising towards index  $g$  is to trace out all indices of the density matrix except for index  $g$ . All the other indices can be combined into a superindex  $x$ . The last step is to apply the SVD to  $A_{gx}$  (or  $A_{xg}$ , depending on the ordering of the indices) to get the transformation equations. Since everything works out in complete analogy, we just state the result for creating a new ISS basis for site  $k + 1$  starting from site  $k$ :

$$\tilde{A}_k = U, \quad \tilde{A}_{k+1} = SV^\dagger A_{k+1}, \quad (2.44)$$

where the SVD of  $A_{(i,\sigma_k),o}$  is given by  $A_k = USV^\dagger$  and the condition for the overall state  $|\psi\rangle$  to remain unchanged is

$$A_k A_{k+1} = \tilde{A}_k \tilde{A}_{k+1} = \text{unchanged}. \quad (2.45)$$

### 2.2.5 Matrix dimensions

At the ends of the terminated Wilson chains there are no outer states. So starting from the end of the chains<sup>26</sup> the state space dimension  $D_k$  will take the values  $d, d^2, d^3, \dots = 2, 4, 8, \dots$  until we start to truncate the effective Hilbert space. At the sites where  $D_k = D_{k+1}d$  we do not truncate, so we choose  $A_{i,(o,\sigma_k)} = \mathbb{1}_{i,(o,\sigma_k)}$ . It turns out that, because there is only Coulomb interaction between the two dot matrices but no particle exchange, we can choose a small  $D_v$  between the spin up and down dot sites compared to the  $D_k$  along the chains. Usually we choose the same matrix dimensions in all four chains. Figure 2.26 shows an example.

## 2.3 Variational optimisation scheme

So far we have provided the MPS description for system states, calculation techniques and an efficient way to truncate our state space according to DMRG ideas. In this section we will devise a variational approach for finding the optimal MPS representation of the system ground state.

---

<sup>26</sup>The dimension  $D_L$  is actually given by  $D_L = 1$ , which means that the last chain matrix has also an outer index. But as its dimension  $D_L = 1$  this does not have any influence and actually enables us to use the same structure for all chain matrices.

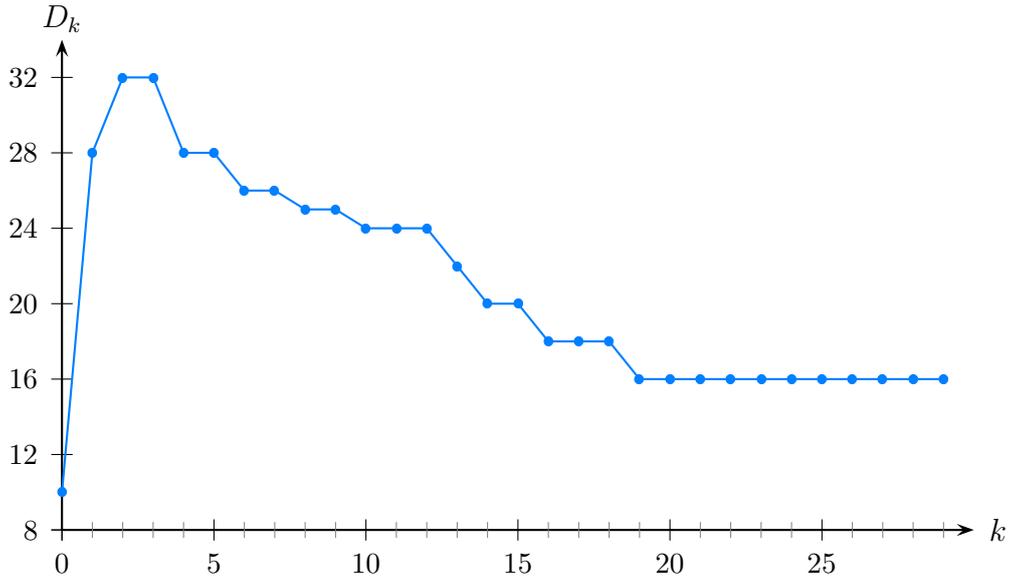


Figure 2.26: Example for a choice of matrix dimensions along the first part of a Wilson chain. The value for  $k = 0$  is the dimension  $D_v$  between the two dot matrices.

### 2.3.1 Minimisation problem

In order to find the ground state of the system in matrix product description we need to calculate the MPS  $|\psi\rangle$  that minimises the energy  $E = \langle\psi|H|\psi\rangle$  with the constraint of keeping the norm of  $|\psi\rangle$  constant [17]. Using  $\lambda$  as a Lagrange multiplier for normalisation we arrive at the following minimisation problem:

$$\min_{|\psi\rangle} (\langle\psi|H|\psi\rangle - \lambda\langle\psi|\psi\rangle). \quad (2.46)$$

The key idea of the variational MPS optimisation is to optimise every single  $A$ -matrix of  $|\psi\rangle$  separately until the ground state energy has converged. Therefore we need to derive a condition from (2.46) for a single  $A$ . Inserting the local MPS description  $|\psi\rangle = A_{io}^{[\sigma_k]}|i\rangle|o\rangle|\sigma_k\rangle$  from (2.18) into (2.46) yields (see Figure 2.27 for graphical representation)

$$\min_{A_k} \left( A_{i'o'}^{[\sigma'_k]*} H_{(i'o'\sigma'_k),(io\sigma_k)} A_{io}^{[\sigma_k]} - \lambda A_{io}^{[\sigma_k]*} A_{io}^{[\sigma_k]} \right), \quad (2.47)$$

where  $H_{(i'o'\sigma'_k),(io\sigma_k)}$  are the matrix elements of the Hamiltonian in the current local bases

$$H_{(i'o'\sigma'_k),(io\sigma_k)} = \langle\sigma'_k|\langle o'|\langle i'|H|i\rangle|o\rangle|\sigma_k\rangle. \quad (2.48)$$

What we actually have done using the local description of  $|\psi\rangle$  is to treat one  $A_k$  of the multi-dimensional minimisation problem at a time keeping all other  $A_{k'}$  fixed. For general problems this can be a very bad approach as one can get stuck in a local minimum during the optimisation. This must be kept in the back of ones mind. However, it has proven to work reliable when the site site coupling varies smoothly and monotonously. In our case the Hamiltonian has only nearest neighbour interactions and there are no long-range correlations in the system. This leads to the reliable behaviour of our approach.

$$A_{i'o'}^{[\sigma'_k]^*} H_{(i'o'\sigma'_k), (io\sigma_k)} A_{io}^{[\sigma_k]} - \lambda A_{io}^{[\sigma_k]^*} A_{io}^{[\sigma_k]} \equiv \begin{array}{c} \boxed{A_k} \\ \text{---} \\ \boxed{H} \\ \text{---} \\ \boxed{A_k^*} \end{array} - \lambda \begin{array}{c} \boxed{A_k} \\ \text{---} \\ \boxed{A_k^*} \end{array}$$

Figure 2.27: Graphical representation of the ground state minimisation problem.

### 2.3.2 Site optimisation of the $A_k$

We solve (2.47) for the optimal solution in the usual<sup>27</sup> way:

$$\frac{\partial}{\partial A_{i'o'}^{[\sigma'_k]^*}} \left( A_{i'o'}^{[\sigma'_k]^*} H_{(i'o'\sigma'_k), (io\sigma_k)} A_{io}^{[\sigma_k]} - \lambda A_{io}^{[\sigma_k]^*} A_{io}^{[\sigma_k]} \right) = H_{(i'o'\sigma'_k), (io\sigma_k)} A_{io}^{[\sigma_k]} - \lambda A_{i'o'}^{[\sigma'_k]} = 0. \quad (2.49)$$

Switching from coefficient notation to operator notation and replacing  $\lambda$  with  $E_0$  results in the eigenvalue problem

$$H A_{io}^{[\sigma_k]} |i\rangle|o\rangle|\sigma_k\rangle = E_0 A_{io}^{[\sigma_k]} |i\rangle|o\rangle|\sigma_k\rangle. \quad (2.50)$$

The eigenvector of the smallest eigenvalue is the solution to our minimisation problem. So after solving this eigenvalue problem for the current site we replace the  $A_k$  with the coefficients of the eigenvector corresponding to the smallest eigenvalue and move on to the next site.

### 2.3.3 Solving the eigenvalue problem

The main work we need to do in order to find the ground state is to solve the eigenvalue problem (2.50):  $H|\psi\rangle = E_0|\psi\rangle$ . This is a well known problem from linear algebra and there exist efficient algorithms to solve it. The effective Hilbert space dimension is  $D_i D_o d$  which can become big easily. However, we are not interested in the whole spectrum of  $H$  but only in the smallest eigenvalue and its eigenvector, so we apply standard Lanczos method<sup>28</sup> to solve it. For applying this method we only need to be able to calculate  $H|\psi\rangle$ , which means first of all we need to calculate the matrix elements  $H_{(i'o'\sigma'_k), (io\sigma_k)}$ .

Using the fact that the Hamiltonian contains, in terms of matrix site indices, only “nearest neighbour” coupling terms<sup>29</sup>, we can write for every site<sup>30</sup>  $k$ :

$$\begin{aligned} \langle \sigma'_k | \langle o' | \langle i' | H^{(k)} | i \rangle | o \rangle | \sigma_k \rangle &= \langle i' | H_i^{(k)} | i \rangle + \langle \sigma'_k | \langle i' | H_{i\bullet}^{(k)} | i \rangle | \sigma_k \rangle + \langle \sigma'_k | H_{\bullet o}^{(k)} | \sigma_k \rangle \\ &+ \langle \sigma'_k | \langle o' | H_{\bullet o}^{(k)} | o \rangle | \sigma_k \rangle + \langle o' | H_o^{(k)} | o \rangle. \end{aligned} \quad (2.51)$$

<sup>27</sup>Because the states are elements of a Hilbert space with a hermitian scalar product we need to take the derivative of (2.47) with respect to the complex conjugate  $A_k^*$ .

<sup>28</sup>See [21] for details.

<sup>29</sup>As as the dot levels are described by one matrix per spin, this holds also for the couplings from the leads to the dot levels.

<sup>30</sup>We will drop the additional site superscripts of (2.51) whenever possible.

For a graphical representation of (2.51) see Figure 2.28. Here,  $H_i$  and  $H_o$  denote the parts

$$\langle \sigma'_k | \langle o' | \langle i' | H^{(k)} | i \rangle | o \rangle | \sigma_k \rangle \equiv \begin{array}{c} \begin{array}{|c|c|c|} \hline i & \sigma_k & o \\ \hline \end{array} \\ \boxed{H} \\ \begin{array}{|c|c|c|} \hline i' & \sigma'_k & o' \\ \hline \end{array} \end{array} = \begin{array}{c} \begin{array}{|c|} \hline i \\ \hline \end{array} \\ \boxed{H_i} \\ \begin{array}{|c|} \hline i' \\ \hline \end{array} \end{array} + \begin{array}{c} \begin{array}{|c|c|} \hline i & \sigma_k \\ \hline \end{array} \\ \boxed{H_{i\bullet}} \\ \begin{array}{|c|c|} \hline i' & \sigma'_k \\ \hline \end{array} \end{array} + \begin{array}{c} \begin{array}{|c|} \hline \sigma_k \\ \hline \end{array} \\ \boxed{H_{\bullet}} \\ \begin{array}{|c|} \hline \sigma'_k \\ \hline \end{array} \end{array} + \begin{array}{c} \begin{array}{|c|c|} \hline \sigma_k & o \\ \hline \end{array} \\ \boxed{H_{\bullet o}} \\ \begin{array}{|c|c|} \hline \sigma'_k & o' \\ \hline \end{array} \end{array} + \begin{array}{c} \begin{array}{|c|} \hline o \\ \hline \end{array} \\ \boxed{H_o} \\ \begin{array}{|c|} \hline o' \\ \hline \end{array} \end{array}$$

Figure 2.28: Partitioning of the Hamiltonian into smaller parts.

of the Hamiltonian that act on the ISS/OSS only<sup>31</sup>,  $H_{i\bullet}$  and  $H_{\bullet o}$  contain the hoppings<sup>32</sup> between the sites  $k$  and  $k \mp 1$ , respectively.  $H_{\bullet}$ , which is only present at the dot, consists of the pure local part of the Hamiltonian, i. e. the level energy terms.

Calculating the matrix elements of  $H_{\bullet}$ ,  $H_{i\bullet}$  and  $H_{\bullet o}$  is straight forward using the techniques we introduced in section 2.1.8. For  $H_i$  and  $H_o$  we can use the recursion formulas

$$H_i^{(k)} = H_i^{(k-1)} + H_{i\bullet}^{(k-1)} \quad \text{and} \quad (2.52a)$$

$$H_o^{(k)} = H_o^{(k+1)} + H_{\bullet o}^{(k+1)}. \quad (2.52b)$$

The formula (2.52a) for  $H_i$  translates into the MPS equation Figure 2.29.

$$\begin{array}{c} \boxed{H_i^{(k-1)}} \\ \begin{array}{|c|} \hline \end{array} \end{array} \begin{array}{c} \boxed{A_{k-1}} \\ \begin{array}{|c|} \hline i \end{array} \end{array} \begin{array}{c} \boxed{A_{k-1}^*} \\ \begin{array}{|c|} \hline i' \end{array} \end{array} + \begin{array}{c} \boxed{H_{i\bullet}^{(k-1)}} \\ \begin{array}{|c|} \hline \end{array} \end{array} \begin{array}{c} \boxed{A_{k-1}} \\ \begin{array}{|c|} \hline i \end{array} \end{array} \begin{array}{c} \boxed{A_{k-1}^*} \\ \begin{array}{|c|} \hline i' \end{array} \end{array} = \left( H_i^{(k)} \right)_{i'i}$$

Figure 2.29: Recursion formula for the inner part of the Hamiltonian  $\left( H_i^{(k)} \right)_{i'i}$ .

We can calculate  $H_o$  starting from the end of every chain. There by construction we have the initial conditions

$$H_o^{(L)} = H_{\bullet o}^{(L)} = 0, \quad H_o^{(L-1)} = 0, \quad H_{\bullet o}^{(L-1)} = \Lambda^{\frac{L-1}{2}} \xi_{L-1} \left( c_L^\dagger c_{L-1} + h.c. \right). \quad (2.53)$$

For calculating  $H_i$  we proceed in the same way as for orthonormalising the ISS bases. We first calculate all  $H_o$  for all the other chains, use them to get an effective Hamiltonian describing also the dot in addition to the other chains. From there we can start using the recursion formula (2.52a) for  $H_i$ .

<sup>31</sup>At the dot there is a third term for the other spin part of the system

<sup>32</sup>With respect to hopping to or from the dot, all dot levels of course have to be taken with the proper Jordan-Wigner corrections into account

### 2.3.4 Sweeping procedure

The whole optimisation process is put together as follows. First we initialise the whole MPS randomly except those sites at the very end of the chain where we never truncate the Hilbert space. There we choose the adequate unity matrix as in section 2.2.5. In addition we do not optimise these sites as there is no need to, since no Hilbert space truncation takes place. As a starting point for the optimisation process we chose site  $k_0 = L - 1$  of the left spin up chain.

So we orthonormalise the MPS towards  $k_0$  and prepare the matrix elements of the Hamiltonian accordingly to (2.51). The next step is to solve the minimisation problem for  $A_{k_0}$  and replace it by the newly found ground state of  $H$ . After that we iterate and go on to the next site until after having passed all sites in the system we arrive at  $k_0$  again. This is what we call a sweep, optimising the MPS back and forth once, see Figure 2.30

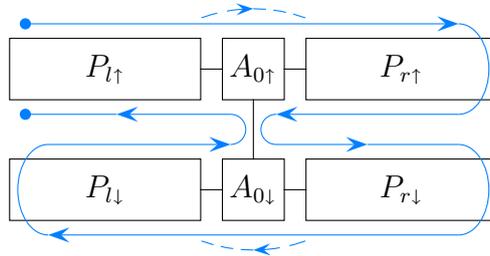


Figure 2.30: Sweeping sequence. For clarity we place the spin up and spin down parts on top of each other. The solid blue line depicts standard procedure. Following the dashed sequence the dot sites are skipped when switching from one lead to the other of the same spin. As a consequence the dot sites get optimised only twice in a sweep, too. In this case the sweeping time decreases, but to achieve as good energy convergence as with the standard sequence we need to perform more sweeps. So the overall computational effort is comparable for both sweeping sequences.

for sweeping sequence. During one sweep every chain matrix gets optimised twice, the dot matrices three times or also twice (see Figure 2.30). We repeat the sweeps until the MPS has converged, which we test by calculating the overlap of the states before  $|\psi_{N-1}\rangle$  and after the  $N$ -th sweep  $|\psi_N\rangle$ . If we demand that the change

$$1 - |\langle\psi_{N-1}|\psi_N\rangle| \leq \epsilon \quad (2.54)$$

is to be bounded by  $\epsilon = 10^{-3}$  we typically need between 10 and 15 sweeps. This depends crucially on the system parameters. As we iterate site by site through the MPS we can exploit that the system has changed only on one side of the current site thus enabling us to reuse former calculated data for the other, unchanged side. To exploit this advantage we save  $H_i^{(k)}$  while the sweeping steps are directed outwards and  $H_o^{(k)}$  while they are directed inwards for every  $k$ .

Due to the strictly variational nature of our optimisation strategy we can only improve the ground state description on every step, so the approximated ground state energy will decrease until it converges. Let  $E_k$  be the lowest eigenvalue of  $H$  found at optimising site  $k$ .

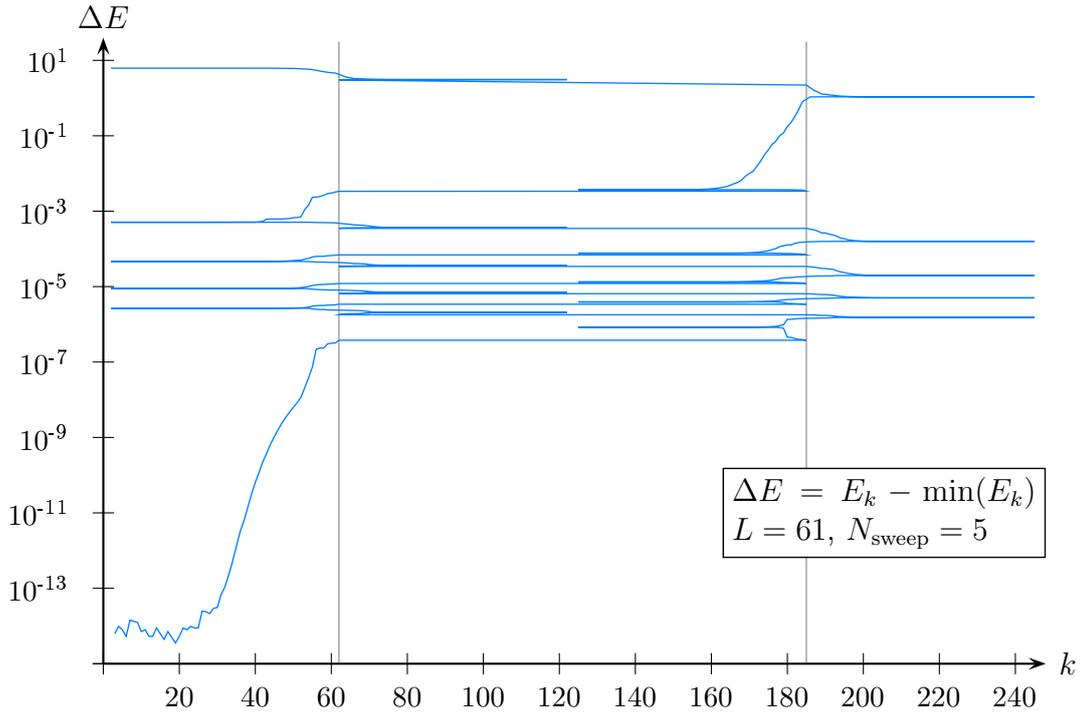


Figure 2.31: Energy convergence plot for 5 sweeps.  $k$  is the matrix site index with Jordan-Wigner ordering except that the dot matrices have only one index assigned. The dot sites are marked with the gray lines.

By plotting  $\Delta E = E_k - \min(E_k)$  versus the matrix site index  $k$  we can assess the energy convergence (see Figure 2.31). We can see that the biggest improvements on the ground state energy are made at the dot sites and their vicinity. At the chains the energy changes only little. As we do not know the exact ground state energy we use the lowest  $E_k$  as a reference for plotting. This leads to the apparent tremendous energy decrease at the end of the last sweep, but this means only that the energy hardly changes in the outer half of the chain.

### 2.3.5 Numerical costs

Let us now discuss the numerical costs of our algorithm. The calculation of  $H|\psi\rangle$  has the main impact on the computational time needed, thus we focus our analysis on this term whose calculation basically consists of elementary matrix multiplications. The Lanczos method, which we chose for solving the eigenvalue problem (2.50), is an iterative method. For every ‘‘Lanczos iteration’’ the term  $H|\psi\rangle$  is calculated once. We cannot completely control the number of Lanczos iterations  $l$  performed but it will be usually less than 100 per site. We will now discuss the cost of a single Lanczos iteration, i. e. the multiplication  $H|\psi\rangle$ , in further detail.

The computational costs for every matrix multiplication is proportional to the size of the resulting matrix times the dimension of the index that was contracted out. We will provide a detailed analysis of the term  $H_{i\bullet}|\psi\rangle$  in the middle of a chain, where

$$H_{i\bullet} \propto c_{k-1}^\dagger c_k + h.c. \quad (2.55)$$

We need the matrix elements  $\langle i'|c_{k-1}^\dagger|i\rangle$ . There are  $D_k \times D_k$  of them and they are constant for the Lanczos iterations of the current site, so we only calculate them once for the site under consideration.  $c_k$  is a  $2 \times 2$  matrix in the LSS. Both of them need to be contracted to the current  $A_k$  of size  $D_k D_{k+1} d$ . Thus the costs  $C_{i\bullet}$  for calculating this term is

$$C_{i\bullet} = \mathcal{O}(D^2(D+d)d), \quad (2.56)$$

where we used  $D = D_k = D_{k+1}$  for simplicity as  $\mathcal{O}(D_k) = \mathcal{O}(D_{k+1})$ . By similar arguments we find  $C_i = C_o = \mathcal{O}(D^3 d)$  and as the total costs for one Lanczos iteration in the chains

$$C_c = \mathcal{O}(D^2(D+d)d). \quad (2.57)$$

Using the same reasoning the result for the costs of an iteration step at a dot site ( $m$  dot levels,  $D = D_1$ ) is

$$C_d = \mathcal{O}(D^2 D_v d^m (D + D_v + md)), \quad (2.58)$$

note that  $\mathcal{O}(D) = \mathcal{O}(D_v)$  is not necessarily true, but depends on the system parameters. The total cost for the optimisation process is then given by

$$C_s = N_{\text{sw}} l (4LC_c + C_d), \quad (2.59)$$

where  $N_{\text{sw}}$ , the number of sweeps performed,  $l$  the number of Lanczos iterations, and  $L$  the Wilson chain length.

So it is obvious that the size of the dot matrices is the main limiting factor for the performance of this algorithm. Its dimension is of course given by the problem, but we can choose  $D$  and  $D_v$  as close as possible to the required minimum for keeping all states with more weight in the density matrix than  $w_{\text{min}}$ . Space complexity is of no concern as long as we manage to fit several dot matrices into the available computer memory.

### 2.3.6 Bond optimisation

The temporal complexity of the above introduced site optimisation method scales exponentially with the number of dot levels  $m$ , which is bad if one would like to increase the number of levels. The reason for this is the fast growing size of the dot Hilbert space as  $D^3 d^m$ , which is a direct subject to the Lanczos iterations. If it is possible to decrease the size of the objects which are optimised, we could gain a huge advantage for the computational time needed for the Lanczos iterations. We will now present an alternative to the site optimisation method, the bond optimisation method, where the computational cost of every Lanczos iteration step is  $\mathcal{O}(D^3)$ . This is effectively a bond optimisation, so the current site reduces to the current bond.

It is very similar to the site optimisation procedure. In principle we only perform the optimisation step during the orthonormalisation process. Suppose the current site is  $k$  with proper orthonormalised inner and outer state space bases  $|i_k\rangle$  and  $|o_k\rangle$  and we want to sweep towards site  $k+1$ . Therefore we perform a SVD on site  $k$  as in section 2.2.4, but we do not yet change  $A_{k+1}$ , see Figure 2.32. Performing a SVD on  $A_k$  yields  $A_k = USV^\dagger$ .

$$A_k A_{k+1} = \left[ \text{gray box } A_k \right] \left[ \text{gray box } A_{k+1} \right] = \left[ \text{gray box } \tilde{A}_k \right] \left[ \text{gray box } SV^\dagger \right] \left[ \text{gray box } A_{k+1} \right] = \tilde{A}_k \Psi A_{k+1}$$

$|i_{k+1}\rangle$        $|o_k\rangle$

Figure 2.32: Orthonormalisation of  $A_k$ . The gray lines indicate orthonormal vectors, non unitary matrices are printed with gray background

As usual we set  $\tilde{A}_k = U$ , but we introduce  $\Psi = SV^\dagger$  instead of multiplying the remainder of the SVD onto  $A_{k+1}$ . This way we get a description of the whole system through  $\Psi$  in terms of the ISS  $|i_{k+1}\rangle$  and the OSS  $|o_k\rangle$

$$|\psi\rangle = \Psi_{i_{k+1}o_k} |i_{k+1}\rangle |o_k\rangle. \quad (2.60)$$

Using this description for a system state we can adopt the whole reasoning of sections 2.3.1 and 2.3.2 and apply the optimisation algorithm to  $\Psi$ . The only difference is that the Hamiltonian now splits into three parts

$$\langle o' | \langle i' | H | i \rangle | o \rangle = \langle i' | H_i | i \rangle + \langle o' | H_o | o \rangle + \langle o' | \langle i' | H_{io} | i \rangle | o \rangle, \quad (2.61)$$

which can all be calculated using the same arguments as in section 2.3.3. To finish the current sweeping step after having  $\Psi$  optimised we restore the original MPS structure of  $|\psi\rangle$  by setting

$$\tilde{A}_{io}^{[\sigma_{k+1}]} = \Psi_{io'} A_{o'o}^{[\sigma_{k+1}]}. \quad (2.62)$$

By analysing the temporal complexity of this approach we see that the costs  $C_b$  for performing one Lanczos iteration is now independent on the current position and does no longer depend on  $m$  at all:

$$C_b = \mathcal{O}(D^3) \quad (2.63)$$

So as long as we are interested in ground state properties only, we can use this bond optimisation scheme, which is far more efficient than the site optimisation scheme and can efficiently be expanded for more dot levels keeping memory restrictions in mind. The big advantage of the bond optimisation is to solve the actual minimisation problem on smaller configuration spaces compared with the site optimisation. Optimising the dot site ( $C_d = \mathcal{O}(D^4 d^m)$ ) is replaced by three “small” optimisations on the bonds only ( $C_b = \mathcal{O}(D^3)$ ) for optimising the same parameter space ( $D^3 d^m$ ). This directly implies the necessity of more sweeps.

However, this is also the downside of this method. Considering finite size spectra (see section 3.1.2) it turns out that the excitation spectrum does not converge as well as with

the site optimisation approach. So in order to reach an equally well converged spectrum we need to increase the number of sweeps we perform by a considerable amount, which will reduce the performance advantage of the bond optimisation scheme.

There is another point one has to be very careful about when using the bond optimisation method. If one is calculating a series of ground states while varying a certain parameter, e. g. the position of the dot levels compared to the Fermi energy of the leads, it is tempting to use an already calculated ground state from a different parameter value as starting point for the actual calculation. This works fine when doing site optimisation. But due to the slow convergence of the bond optimisation scheme the sweep truncation criterion (2.54) may be fulfilled after very few sweeps, causing the algorithm to stop sweeping prematurely, although the current state is still very different from the real ground state. This behaviour can be improved by demanding a minimum number of sweeps, but our experience shows that it is best to start the calculation for every parameter value with a randomly initialised MPS.

## 2.4 Determining the ground state in a nutshell

In the previous sections we have introduced the MPS description for system states and provided the tools to handle such states efficiently. Furthermore we applied the DMRG Hilbert space truncation scheme to the MPS representation and derived a variational algorithm to determine the system ground state. Now we will put all the pieces together and present the whole algorithm for determining the ground state in condensed form (we use Matlab pseudo code notation).

First we need to create the initial MPS and prepare the Hamiltonian.

```

1  % initialise MPS mps
2  mps = MPState(L, D, d, ...);
3  k0 = 2; k_end = 4*L+1
4  orthonormalise(mps, k0);

```

For normalising the MPS we formally just need to create the ISS  $|i_{L+1}\rangle$  for the whole system to ensure  $\|\psi\| = 1$ . Of course we need to use the Jordan-Wigner numbering scheme for implementing the algorithm (see section A.1.3).

```

5  % prepare Hamiltonian h for sweeping
6  h = hamiltonian(eps_dot, Gamma, U, ...);
7  prepareOps(h, mps); % calculate  $H_i^{(k)}$  and  $H_o^{(k)} \forall k$ 

```

After we calculated all effective Hamilton operators  $H_i^{(k)}$  and  $H_o^{(k)}$ , we can start with the sweeping.

```

8  for n = 1:maxSweeps
9      mps_old = mps;
10     for k = [k0:k_end, k_end:-1:k0] % back and forth through the system
11         A = mps{k};

```

```

12     A2 = LanczosGroundState(A,h); % A2 is the new optimised one

```

This requires a method for calculating  $H|\psi\rangle$ , which we need to supply. This is also the function with the highest impact on computational performance as it is called very often.

```

13     [u,s,v] = svd(A2); % obtain singular value decomposition of A2
14     mps{k} = u; % provides orthonormal basis
15     mps{k+1} = s*v'*mps{k+1} % preserves original MPS
16     updateHamiltonAndOps(h,mps{k}) % calculate new  $H_o^{(k+1)}$  or  $H_i^{(k+1)}$ 
17     end

```

These inner and outer Hamilton operators are required by the Lanczos method for calculating the ground state with respect to the current site (bond). We neglected here the consequences of a more complicated sweeping order, but this is only bookkeeping.

```

18     % check for stopping criterion
19     change = 1 - abs(scalarproduct(mps, mps_old));
20     if change < sweepEps
21         break; % stop the sweeping if the state hardly changed
22     end
23 end % sweeping loop
24 % now we got ground state mps ready to use

```

This kind of stopping criterion works fine for site optimisation. In case of bond optimisation we need a more sophisticated one to avoid stopping the sweeping process prematurely.

In principle this is the whole sweeping procedure. We did not show all the functions needed for handling a MPS (like initialising, orthonormalising, scalarproducts, ...), as they are straightforward to implement using the formulas we provided.

In the next chapter we will show how to check whether our choices of the  $D_k$  and  $L$ , the chain length, were sufficient. Additionally we will present first results we obtained using this optimisation approach and compare our results with NRG for parameters where NRG is still feasible.



# Chapter 3

## First results

Within this work we successfully implemented a two-lead Anderson model with two and four dot levels. We will now present first data we obtained from this method.

### 3.1 Consistency checks

#### 3.1.1 Matrix dimensions

After the calculation of a ground state it is important to check whether the choice of the matrix dimensions was sufficient to describe the result well enough. Therefore we calculate the reduced density matrix  $\rho^{(i)} = A_k^\dagger A_k = \rho_k$  from (2.38) for every site  $k$ . We then can determine the minimal  $D_{k,\min}$  necessary for keeping all states with a corresponding weight bigger than  $w_{\min}$  by just counting the eigenvalues of  $\rho_k$  bigger than  $w_{\min}$ .

We compare this  $D_{k,\min}$  with what we call bond entropy  $S_k$ . We define the bond entropy as the Shannon entropy of the reduced density matrix [8]

$$S_k = -\rho_k \ln \rho_k. \quad (3.1)$$

With the name bond entropy we emphasise the fact that we used the reduced density matrix, which describes the system as a bipartite system with respect to one specific bond, i. e. an index in the MPS to be summed over. In plot Figure 3.1 we actually plot  $e^{S_k}$ . Because in the case where  $\rho_k$  is a classical homogeneous mixture of  $N$  states, all the eigenvalues of  $\rho_k$  are equal to  $\frac{1}{N}$ , and so we find

$$e^{S_k} = N, \quad \text{with } \rho_k = \text{diag}_N\left(\frac{1}{N}, \dots, \frac{1}{N}\right). \quad (3.2)$$

If we interpret the bond entropy in an information theoretical sense, it provides a measure for the amount of information that links the two parts of the system. So we would expect for the  $D_{k,\min}$  to show similar characteristics along the chains.

For the comparison plot we scale  $e^{S_k}$  by a linear factor so that it nearly overlaps with  $D_{k,\min}$  at the far ends of the chains. We notice that both curves show similar behaviour but

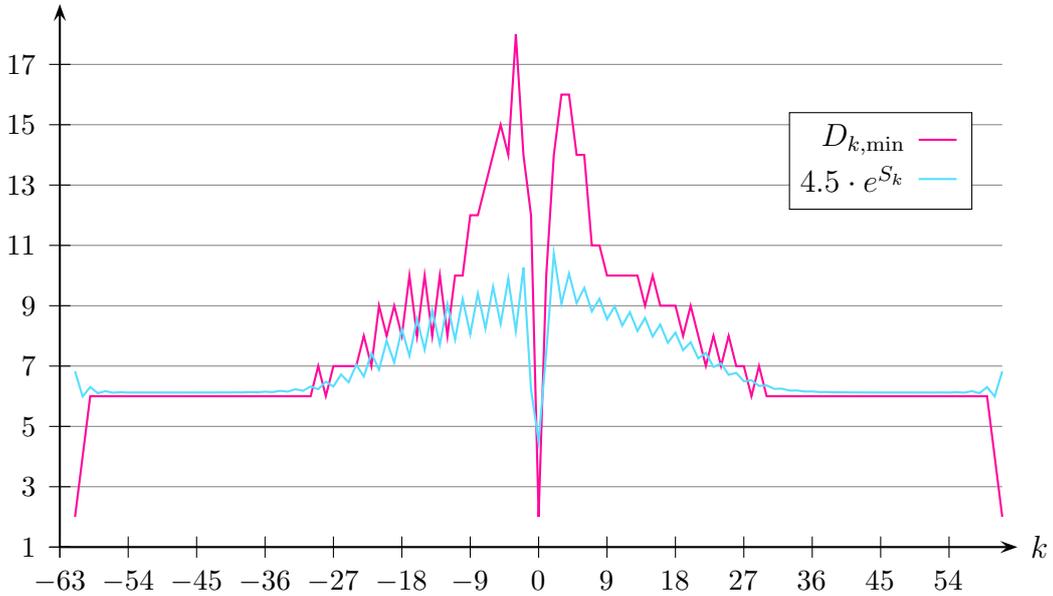


Figure 3.1: bond entropy

especially near the dot the rescaled bond entropy does not rise as high as  $D_{k,\min}$ . These even-odd oscillations that appear in both curves are frequently observed with NRG-type methods and can be linked to singlet formation.

The significant criteria for judging whether the chosen  $D_k$  were sufficient are the  $D_{k,\min}$ . Since per construction

$$D_{k,\min} \leq D_k, \quad (3.3)$$

we need to check the smallest eigenvalue<sup>1</sup> of  $\rho_k$  explicitly if it is smaller than  $w_{\min}$ . If it is not our choice for  $D_k$  was too small and we need to increase it.

### 3.1.2 Flow diagram

The second parameter to check is the Wilson chain length  $L$ . This can be done by considering energy flow diagrams, also called finite size spectra.

#### Single chain flow diagram

Given our model with four chains attached to the dot, we can obtain an energy flow diagram in two ways. For the first way we consider only one chain<sup>2</sup>. We start with the current position at the dot and proceed out till the end of the chain. Along the way, we calculate a sequence of inner Hamiltonians  $H_i^{(k)}$  for all  $k$  along this chain just the same way as during

<sup>1</sup>Everything can be done using the singular values of the proper reshaped  $A_k$ . There is no need to calculate the reduced density matrix explicitly.

<sup>2</sup>Usually, we use the upper right chain for this purpose, but this choice has no big consequences, especially for symmetric couplings to the leads.

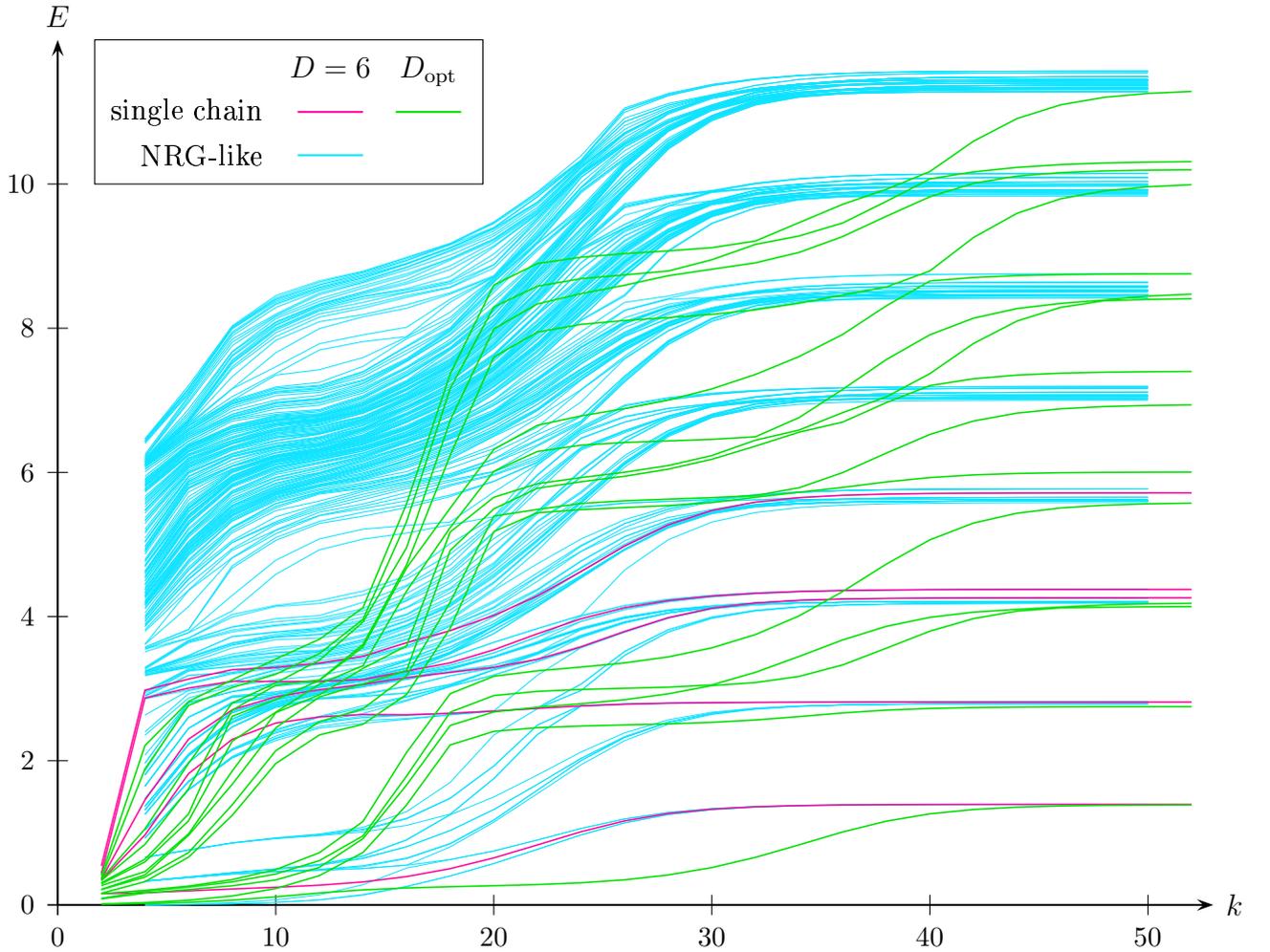


Figure 3.2: Flow diagram for the model parameters of Figure 3.3 at  $\epsilon = 0.4U$ .

the sweeping process. We can then calculate the spectra  $\mathbf{E}^{(k)}$  of these Hamiltonians by exact diagonalisation. In order to undo the effects of the logarithmic energy discretisation of the NRG formalism we rescale the energy differences with respect to the ground state with  $\Lambda^{\frac{k}{2}}$ :

$$\tilde{\mathbf{E}}^{(k)} = \left( \mathbf{E}^{(k)} - E_0^{(k)} \right) \cdot \Lambda^{\frac{k}{2}} \quad (3.4)$$

We then plot these  $\tilde{\mathbf{E}}^{(k)}$  versus the chain index  $k$  to obtain the single chain flow diagram (see Figure 3.2) for even or odd site indices  $k$ . Note that this already contains the expectation value of the Hamiltonians for the remaining chains. Even-odd oscillations would again become visible if one would plot the spectrum for every  $k$ . To avoid this effect we only plot the spectra for even or odd site indices  $k$ . In plot Figure 3.2 we chose to plot the flow diagrams for even sites. By doing this we study how the system responds to a length change of one of the chains. If the rescaled spectrum no longer changes, the chain length is big enough to prevent finite chain length effects.

### NRG-like flow diagram

As an alternative to the single chain flow diagram we can calculate a flow diagram in analogy to NRG. There we consider a somewhat different series of Hamiltonians  $H^{(k)} = H_{\text{dot}} + H_{\text{coupling}} + H_{\text{leads}}^k$ , where  $H_{\text{leads}}^k$  describes the first  $k$  sites of all chains. So in this series we add step by step one site at the end of every chain. For calculating the spectra of the  $H^{(k)}$  we combine all ISS  $|i_{k\alpha s}\rangle$  for the same  $k$  to a single NRG-like ISS

$$|i_k^{\text{NRG}}\rangle = |i_{kl\uparrow}\rangle \otimes |i_{kr\uparrow}\rangle \otimes |i_{kl\downarrow}\rangle \otimes |i_{kr\downarrow}\rangle. \quad (3.5)$$

In principle the calculation of the spectra is as straightforward as for the single chain spectra, but there is the technical problem that by combining the  $|i_{k\alpha s}\rangle$  in a tensor product like fashion, the resulting Hilbert space, i. e. the one that has to be used in NRG context, is orders of magnitudes larger than the one in the underlying variational problem. Accordingly the Hamiltonians  $H^{(k)}$  have the size  $D^4 \times D^4$ . Because of the size of these objects we can at the moment only calculate the NRG-like flow diagram for  $D \leq 6$ . For every  $k$  we determine the spectrum for a system with chain length  $k$ , hence the name “finite size spectrum”.

We compare both types of flow diagrams for  $D = 6$  with a single chain flow diagram for an adapted choice of  $D_k$ , where  $D_k$  is 16 along the outer part of the chains. We note that the fixed points in the flow diagram agree quite well for the lowest lying excitation and show also for higher excitation energies similarities. The major source for the difference is that for the single chain flow diagram a larger  $D_k$  was taken ( $D_k = 16 > D = 6$ ) and thus can be considered more accurate. For  $D = D_k$  the fixed point spectra coincides.

## 3.2 Occupation of the quantum dot

Troughout this part we fix the Coulomb interaction  $U = 0.2W$ ,  $2W$  being the bandwidth, and use the convention  $W = 1$ . The relative positions of the single dot levels  $\epsilon = (\epsilon_1, \dots, \epsilon_m)$  are given with respect to the overall level position  $\epsilon$  corresponding to the onsite chemical potential. We label the dot levels with increasing level energy. For the occupation number plots we use  $\epsilon$  as abscissa. For the comparison of MPS and NRG results we used parameter sets from [16].

### 3.2.1 Two-level dot

#### Spinless system

For symmetric couplings of two spinless dot levels to two spinless leads, the resulting two-level Anderson model calculations are also easily feasible using NRG. First we calculated the occupation of two spinless dot levels (see Figure 3.3). By setting the Coulomb interaction  $U = 0$  between the spin up and down parts of the dot we can simulate a spinless system with our model, as there is no interaction between the two spin flavours left. In this case we can also set the matrix dimension between the two dot matrices  $D_v = 1$  and restrict the

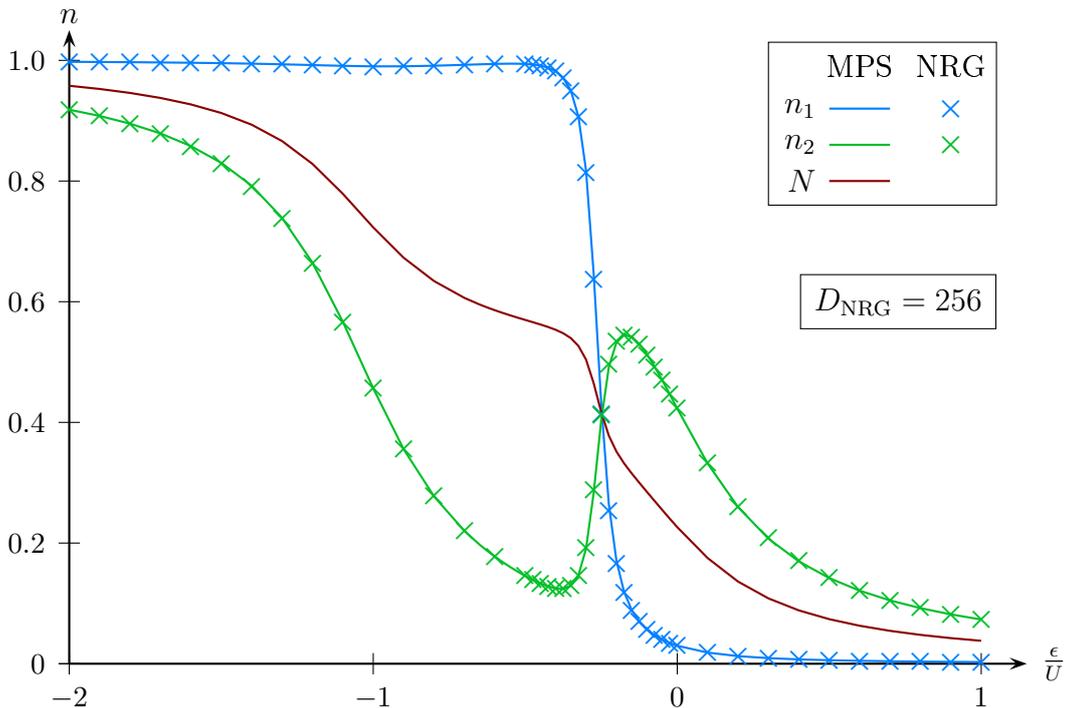


Figure 3.3: Dot level occupation for a spinless two-level system. For the symmetric level splitting and the couplings of the dot we used  $\epsilon_2 - \epsilon_1 = 0.1U$  and  $\Gamma_{1l} = \Gamma_{1r} = 0.005U$ ,  $\Gamma_{2l} = \Gamma_{2r} = 30\Gamma_{1l}$ .  $N = \frac{1}{2}(n_1 + n_2)$  is the rescaled overall dot occupation.

sweeping to one spin subsystem. For the plot of Figure 3.3 we find very good agreement in both the occupation of the dot levels and the groundstate energy. The relative energy difference of MPS and NRG data is of order  $10^{-5}$ . The NRG calculations were carried out by Andreas Weichselbaum using  $D = 256$ .

### Spinful system

We also compared with NRG results for a spinful two-level system (Figure 3.4), again with symmetric couplings of to the leads. Since we applied no magnetic field and therefore the spin up and spin down site for each level are degenerate, we plotted only the summed occupation for each dot level.

The NRG data calculated with  $D = 512$  is again from Andreas Weichselbaum. Note the deviation between NRG and MPS data in this case. Introducing spin into the system basically squares the dimension of the Hilbert space. This is starting to be a problem especially for NRG, because it is not possible to increase the kept Hilbert space to the same extent. However, it is possible for symmetric couplings to the leads to map the leads to a new set of leads such that each dot level only couples to one lead [16]. This transformation would enable NRG to describe the system in a better way, since only two instead of four “spinless” leads need to be taken into account.

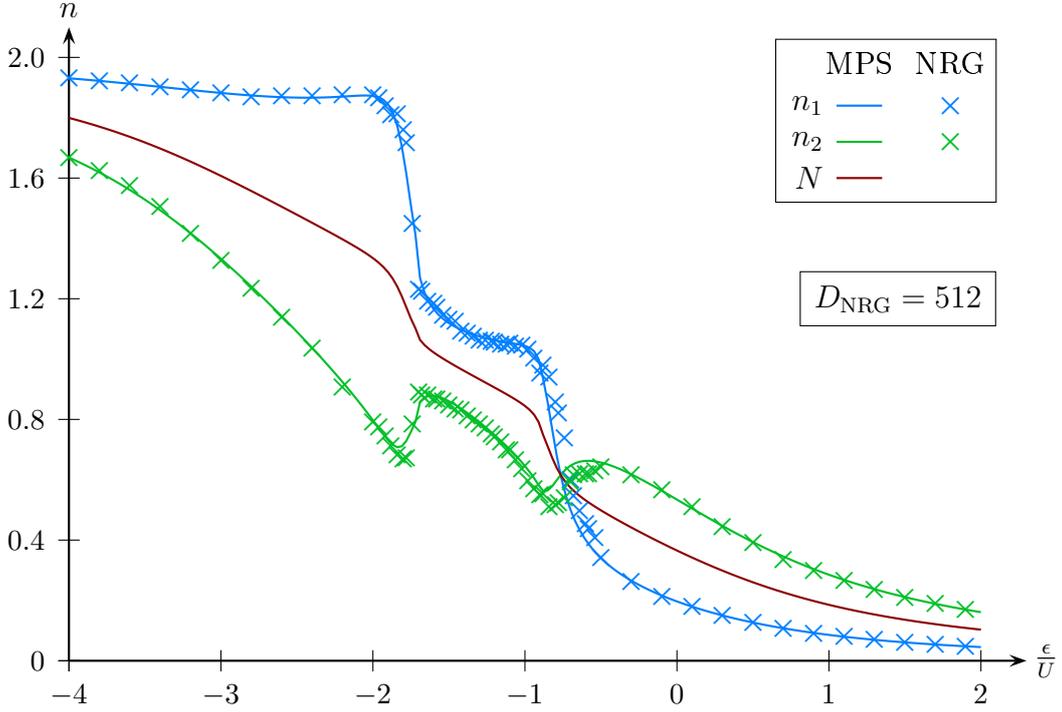


Figure 3.4: Dot level occupation for a spinful two-level system. For the symmetric level splitting and the couplings of the dot we used  $\epsilon_2 - \epsilon_1 = 0.2U$  and  $\Gamma_{1l} = \Gamma_{1r} = 0.1U$ ,  $\Gamma_{2l} = \Gamma_{2r} = 4\Gamma_{1l}$ .  $n_i = n_{i\uparrow} + n_{i\downarrow}$ ,  $N = \frac{1}{2}(n_1 + n_2)$  is the rescaled overall dot occupation.

### 3.2.2 Four-level dot

There was little time to explore the rich physics that is accessible with our method within the scope of this thesis. We will show only a demonstration plot of the occupation numbers of a spinful four-level dot coupled to two leads (see Figure 3.5). As parameters we used:  $U = 0.2W$  and  $\epsilon_i = (-0.1, -0.03, 0.07, 0.1) \cdot U + \epsilon$ , where  $\epsilon$  is the middle level position which is plotted along the abscissa. For the couplings of the lead to the dot we chose symmetric values up to a minus sign:  $\Gamma_{il} = \Gamma_i$ ,  $\Gamma_{ir} = s_i \Gamma_i$  with  $\Gamma_i = (0.5, 0.02, 1, 0.7) \cdot 0.2U$  and  $s_i = (+1, -1, -1, +1)$ . However, in principle our implementation imposes no restrictions at all on the values of  $\Gamma$ . For this plot we applied three different magnetic fields  $B_1 = 0.2U$ ,  $B_2 = 0.02U$  and  $B_3 = 0.002U$ , which has the effect to remove the energy degeneracy of spin up and spin down parts of the dot levels,  $\epsilon_{i\uparrow} = \epsilon_i + \frac{1}{2}B$  and  $\epsilon_{i\downarrow} = \epsilon_i - \frac{1}{2}B$ . The main features of this plot (Figure 3.5) are well understood. The levels with small coupling  $\Gamma$  to the leads exhibit steep rises of their occupation (especially level 2). The different magnetic fields mainly change the energy differences between spin up and spin down part of the levels, which is again best visible for level 2. We used the site optimisation method for calculating the data for this plot. We also applied the bond optimisation scheme to this model and we found it very hard to judge how well the state has already converged due to the slow convergence of that method, see also section 2.3.6. One possible improvement on

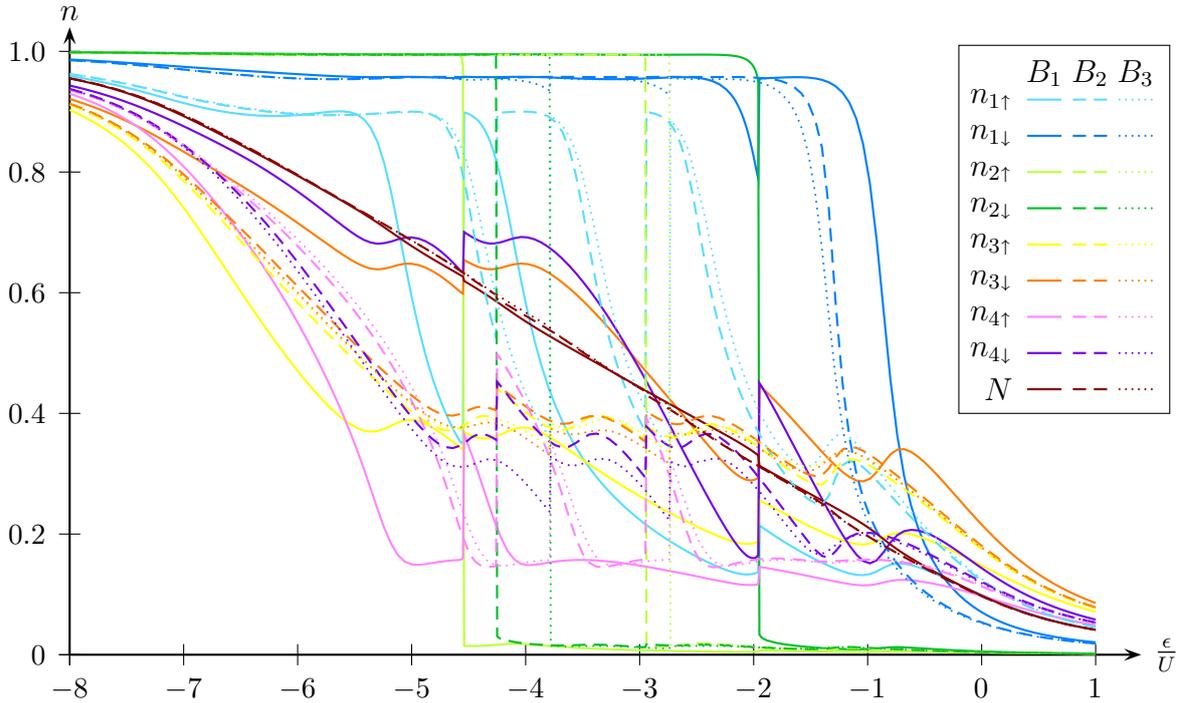


Figure 3.5: Occupation of a four-level dot. We used the site optimisation method to create this plot.

this situation may be to combine both sweeping methods to get a higher level of confidence in the results.

The method we have shown in this thesis so far allow us to study an interesting and broad field of physics. We demonstrated this by applying it to a four-level model, that in its full generality is highly challenging for conventional NRG calculations. It should be noted that whereas NRG already heavily relies on the exploitation of symmetries in the system such as total particle or spin conservation, the method presented in this thesis was built on a simple computational environment without symmetries except for the very important fact that on a variational basis the chains can be treated separately, which is absolutely impossible within the spirit of NRG.



## Chapter 4

# Conclusions and Outlook

The goal of this thesis was to develop the matrix product state approach for finding the ground state of an Anderson model with multiple leads and dot levels. We applied the NRG transformation to the Anderson Hamiltonian and treated in principle the same Hamiltonian numerically as NRG does, but we used the DMRG Hilbert space truncation idea. Additionally the MPS approach is a strictly variational method compared to the iterative solving scheme of NRG. The MPS ansatz enables us to distribute numerical resources differently along the system and provides methods for checking the quality of the chosen model parameters.

For a spinful two-lead Anderson model with two and four dot levels we successfully implemented two different sweeping methods (onsite and bond optimisation). Comparison with NRG results for setups where NRG is still feasible yields very good agreement. But with our approach we are able to calculate properties of more complex systems like the spinful four-level dot for zero temperature.

However, this is a rather young method and there is still some development to do.

- The calculation of spectral functions has already been demonstrated for one-dimensional systems [17] and should be straightforward to implement in our model.
- For finite temperature properties solving for the ground state is not enough. One needs to calculate the full density matrix  $\rho \propto e^{-\beta H}$ , which has been accomplished for NRG using MPS language [12, 19]. Therefore it should be possible to extend this method to our MPS approach.
- Time-dependent problems can be solved using time-dependent DMRG [8]. Also for NRG exists a method to study the time evolution of a system [18], but there are some limitations as the parameters may only change abruptly at  $t = 0$ . As DMRG can also be formulated in terms of MPS, it should also be applicable to our model.
- Steady state solutions of non-equilibrium problems can be described by a many-body Lippmann-Schwinger equation. Solving the Lippmann-Schwinger equation requires similar techniques as calculating spectral functions. So it should be possible to formulate such an approach within the MPS framework.



# Appendix A

## Further details

### A.1 Jordan-Wigner Transformation

We will now adopt the Jordan-Wigner transformation (JWT), which was first introduced by Jordan and Wigner in 1928 [3]. This method mainly targets one-dimensional systems and obtains almost bosonic behaviour of the transformed fermionic operators. This makes numerical treatment much easier, as operators of different sites commute, greatly simplifying the treatment of fermionic signs in the underlying physical system.

#### A.1.1 Definition

Let  $\{c_i\}$  be a set of fermionic operators with some ordering scheme, say  $i = 1, \dots, N$ . The JWT is defined as the following transformation:

$$\tilde{c}_i \equiv (-1)^{\sum_{k<i} c_k^\dagger c_k} c_i \equiv \left( \prod_{k<i} (-1)^{c_k^\dagger c_k} \right) c_i \equiv \left( \prod_{k<i} p_k \right) c_i \equiv P_i c_i, \quad (\text{A.1})$$

where we introduced the following sign factors

$$p_k = (-1)^{c_k^\dagger c_k} \quad \text{and} \quad P_i = \prod_{k<i} p_k, \quad (\text{A.2})$$

which is diagonal and can only take the values  $\pm 1$ . With  $\tilde{c}_k^\dagger \tilde{c}_k = P_k^2 c_k^\dagger c_k = c_k^\dagger c_k$ , the inverse transformation is

$$c_i = (-1)^{\sum_{k<i} \tilde{c}_k^\dagger \tilde{c}_k} \tilde{c}_i. \quad (\text{A.3})$$

#### A.1.2 Commutator Relations

The sign factor  $p_k$  can be rewritten:

$$p_k \equiv (-1)^{c_k^\dagger c_k} = 1 - 2c_k^\dagger c_k = \left[ c_k, c_k^\dagger \right]. \quad (\text{A.4})$$

Keeping the fermionic anticommutator relations for the  $c_i$  in mind, the commutators for the transformed operators (for  $i = j$ ) can be written down:

$$\begin{aligned} [\tilde{c}_i, \tilde{c}_i^\dagger] &= P_i^2 [c_i, c_i^\dagger] = p_i, \\ [\tilde{c}_i, \tilde{c}_i] &= P_i^2 [c_i, c_i] = 0, \\ [\tilde{c}_i^\dagger, \tilde{c}_i^\dagger] &= P_i^2 [c_i^\dagger, c_i^\dagger] = 0. \end{aligned} \quad (\text{A.5})$$

For  $i \neq j$  (we assume  $i < j$ , the argument for  $j < i$  works exactly the same) one finds using

$$P_{k-i} = \prod_{j=i+1}^{k-1} p_j : \quad (\text{A.6})$$

$$\begin{aligned} [\tilde{c}_i, \tilde{c}_j^\dagger] &= P_i^2 P_{j-i} [c_i, p_i c_j^\dagger] = P_{j-i} [c_i, (c_i c_i^\dagger - c_i^\dagger c_i) c_j^\dagger] \\ &= P_{j-i} (\underbrace{c_i c_i c_i^\dagger c_j^\dagger}_{=0} - c_i c_i^\dagger c_i c_j^\dagger - c_i c_i^\dagger c_j^\dagger c_i + \underbrace{c_i^\dagger c_i c_j^\dagger c_i}_{=0}) \\ &= P_{j-i} (-c_i c_i^\dagger c_i c_j^\dagger + c_i c_i^\dagger c_i c_j^\dagger) = 0, \\ [\tilde{c}_i, \tilde{c}_j] &= P_i^2 P_{j-i} [c_i, (c_i c_i^\dagger - c_i^\dagger c_i) c_j] = 0, \\ [\tilde{c}_i^\dagger, \tilde{c}_j^\dagger] &= P_i^2 P_{j-i} [c_i^\dagger, (c_i c_i^\dagger - c_i^\dagger c_i) c_j^\dagger] = 0. \end{aligned}$$

Summarising these results:

$$[\tilde{c}_i, \tilde{c}_j^\dagger] = (-1)^{c_i^\dagger c_i} \delta_{ij}, \quad [\tilde{c}_i, \tilde{c}_j] = [\tilde{c}_i^\dagger, \tilde{c}_j^\dagger] = 0. \quad (\text{A.7})$$

This implies bosonic relationship for  $i \neq j$ , while maintaining the fermionic locally for  $i = j$ . This is important as it prohibits double occupancy as  $\tilde{c}_i \tilde{c}_i = c_i c_i = 0$ , a consequence from dealing with Fermions. The ordering scheme itself is unimportant up to the point that hopping terms in the Hamiltonian should appear as nearest neighbour terms. This can easily be done in linear systems, but gives rise to more complex operators in the Hamiltonian if this is not possible.

### A.1.3 Application

#### Operators

Now we apply the JWT to our system Hamiltonian (2.1). There are only two types of terms we need to consider. The first one are the energy and Coulomb interaction terms on the dot, which consist of  $\hat{n}$  operators only. Using the definition (A.1) we immediately get

$$\tilde{n}_k = P_k^2 c_k^\dagger c_k = n_k. \quad (\text{A.8})$$

So we can directly replace all  $n$  operators with their JWT counterparts.

The second type of terms are the hopping terms. For nearest neighbour hopping we derive

$$\tilde{c}_k^\dagger \tilde{c}_{k+1} = P_k^2 c_k^\dagger p_k c_{k+1} = \left( c_k^\dagger c_k c_k^\dagger - c_k^\dagger c_k^\dagger c_k \right) c_{k+1} = c_k^\dagger c_{k+1}. \quad (\text{A.9})$$

For other hopping terms, which occur, e. g. for the hopping from the leads to the dot, we have

$$\tilde{c}_k^\dagger \tilde{c}_{k+l+1} = P_k^2 c_k^\dagger p_k \prod_{i=1}^l (p_{k+i}) c_{k+l+1} = c_k^\dagger \prod_{i=1}^l (p_{k+i}) c_{k+l+1}. \quad (\text{A.10})$$

So we see that, when using JWT operators in our Hamiltonian, only hopping terms other than nearest neighbour hopping are rewritten. These terms require appropriate corrections  $p_i$  that affect only the sites between those coupled by the hopping itself.

### Ordering scheme

As a last step for applying the JWT to our problem, we need to introduce a coherent ordering scheme for the whole system. For brevity of discussion we use the two-level two-lead model as an example (Figure A.1), but the principle applies to any number of levels and leads and can easily be generalised. We start labelling at the end of the left spin up

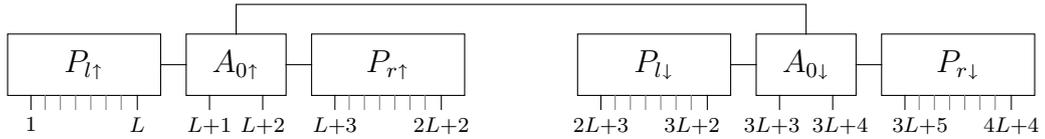


Figure A.1: Ordering scheme in the 2-level 2-lead Anderson model we use for the Jordan-Wigner transformation. Only the first and last index of the chains are labelled, the others are greyed out.

chain and continue to the right until we hit the end of the right spin up chain. The spin up dot sites are placed in the middle of the two spin up chains. Then we proceed at the end of the left spin down chain the same way to the end of the right spin down chain. Note that the site indices differ from the matrix indices, because one dot matrix describes more (in this case two) dot levels (sites). The generalisation to  $m$  dot levels is straight forward, all the dot sites are added in an arbitrary but fixed order. For more leads, however, we would have to add whole additional chains somewhere between the already existing chains in the labelling scheme. Assume we add another Wilson chain for a third lead, which also allows hopping onto the dot. Suppose we put it between the first Wilson chain and the dot with respect to the JWT ordering. The original nearest neighbour coupling between the first chain and the dot would transform to a hopping term over the distance  $L$  of the newly added third chain. This would generate correction terms affecting the whole third chain. This cannot be avoided since such a three chain model cannot be mapped to an one dimensional chain model keeping all couplings in nearest neighbour order<sup>1</sup>.

<sup>1</sup>JWT can be generalised to more than one dimension [11]. However, it is much more complex and beyond the scope of this thesis.

The reason for the JWT to work so well in our model lies in the fact that the coupling between the spin up and down part is only carried by Coulomb interactions. In this terms appear only  $\hat{n}$  operators, which do not acquire corrections due to JWT. So in terms of particle hopping we can view our system as two distinct one-dimensional systems. To these we can apply the JWT without complications.

By applying the JWT all but some of the coupling terms of the dot levels to the leads remain unchanged. To the changed terms simple additional operators ( $p_k$ ) are added between the original hopping sites. This can always be incorporated in the effective creation and annihilation operators of the dot or corrected locally on the dot itself, when the dot site is the current site. So we will suppress this corrections in our further discussions and just use the transformed operators with the same symbols. We will also neglect the difference between the ordering imposed by the JWT and our labelling developed in section 2.1.2, this difference matters only for the actual implementation of the algorithm, but for our discussion we will stick with the more convenient nomenclature of 2.1.2.

## A.2 Orthonormal basis states

### A.2.1 Derivation of the MPS orthonormality condition

#### Outer state space basis

We start at the end of a chain. The effective OSS basis for the second last chain site is given by

$$|o_{L-1}\rangle = A_{o_{L-1}}^{[\sigma_L]} |\sigma_L\rangle \quad (\text{A.11})$$

The orthonormalisation condition (2.19a) can be written in MPS form (Figure A.2), dropping superfluous subscripts.

$$\delta_{o'o} \equiv \langle o' | o \rangle \equiv \begin{array}{c} o \text{---} \boxed{A_L} \\ | \\ o' \text{---} \boxed{A_L^*} \end{array} \equiv \left. \begin{array}{l} o \\ o' \end{array} \right\}$$

Figure A.2: Orthonormalisation condition for  $|o\rangle = |o_{L-1}\rangle$

For an arbitrary chain position we may write the outer basis states  $|o_k\rangle$  as

$$|o_k\rangle = A_{o_k o_{k+1}}^{[\sigma_{k+1}]} |o_{k+1}\rangle |\sigma_{k+1}\rangle. \quad (\text{A.12})$$

We immediately derive the following orthonormalisation condition:

$$\langle o'_k | o_k \rangle = S_{o'_k o_k}^{k+1} = P_{o'_k}^{[\sigma]_{k+1}*} P_{o_k}^{[\sigma]_{k+1}} = \delta_{o'_k o_k}. \quad (\text{A.13})$$

Using the fact that the  $|o_{k+1}\rangle$  are already orthonormal, the orthonormality condition (2.19a) readily yields (see also Figure A.3)

$$A_{o_k o_{k+1}}^{[\sigma_{k+1}]} A_{o'_k o'_{k+1}}^{[\sigma_{k+1}]*} \delta_{o'_{k+1} o_{k+1}} = \delta_{o'_k o_k}. \quad (\text{A.14})$$

We can again reformulate this in terms of an ordinary two-dimensional matrix  $A_{o_k x}$ , if

$$A_{o_k o_{k+1}}^{[\sigma_{k+1}]} A_{o'_k o'_{k+1}}^{[\sigma_{k+1}]*} = \begin{array}{c} o_k \text{---} \boxed{A_{k+1}} \text{---} o_{k+1} \\ | \\ \sigma_{k+1} \\ | \\ o'_k \text{---} \boxed{A_{k+1}^*} \text{---} o'_{k+1} \end{array} = \begin{array}{c} o_k \\ | \\ o'_k \end{array} = \delta_{o'_k o_k}$$

Figure A.3: Orthonormalisation condition for  $A_{k+1}$

we introduce a superindex  $x = (o_{k+1}, \sigma_{k+1})$ . In this compact representation we obtain as orthonormalisation condition (for a graphical representation see Figure 2.15a)

$$AA^\dagger = \mathbb{1}, \quad (\text{A.15})$$

which states that the row vectors of  $A$  are orthonormal.

### Inner state space basis

In principle everything works the same for the inner bases, only the inner and outer indices need to be interchanged. Let us assume we already obtained an effective ISS basis  $|i_1\rangle$  for the first chain site, as we will describe in section A.2.2. For the inner basis states  $|i_k\rangle$  of site  $k$  the orthonormalisation condition is

$$\langle i'_k | i_k \rangle = \tilde{S}_{i'_k i_k}^k = \delta_{i'_k i_k}. \quad (\text{A.16})$$

Using the orthonormality of  $|i_{k-1}\rangle$ ,

$$\tilde{S}_{i'_{k-1} i_{k-1}}^{k-1} = \delta_{i'_{k-1} i_{k-1}}, \quad (\text{A.17})$$

we get again a condition (Figure A.4) on the  $A_{k-1}$ . Once again we introduce a superindex<sup>2</sup>

$$A_{i_{k-1} i_k}^{[\sigma_{k-1}]} A_{i'_{k-1} i'_k}^{[\sigma_{k-1}]*} = \begin{array}{c} \boxed{A_{k-1}} \text{---} i_k \\ | \\ \boxed{A_{k-1}^*} \text{---} i'_k \end{array} = \begin{array}{c} i_k \\ | \\ i'_k \end{array} = \delta_{i'_k i_k}$$

Figure A.4: Orthonormalisation condition for  $A_{k-1}$  regarding inner basis states

$x = (i_{k-1}, \sigma_{k-1})$  in order to write  $A$  as a two-dimensional matrix  $A_{xi_k}$ . In this representation we can formulate the condition of Figure A.4 very compactly as<sup>3</sup>:

$$A^\dagger A = \mathbb{1}. \quad (\text{A.18})$$

So in order to have orthonormal ISS bases, the  $A_{xi_i}$  must be column unitary, i.e. their column vectors need to be orthonormal.

The difference to the outer basis case is caused by the different position of the index in whose direction we build the basis. So by transposing this two-dimensional  $A$ -matrix we could transform (A.18) to (A.15).

## A.2.2 Orthonormalisation conditions for dot matrices

As we represent the quantum dot with two matrices, one for the spin up part, one for the spin down part, we perform orthonormalisation of the dot for either of the two matrices separately. As an example we show the resulting orthonormality conditions for constructing the inner basis for the beginning of the right spin up chain  $|i_{r\uparrow}\rangle$ . A necessary prerequisite is the construction of orthonormal effective outer state space bases for the first sites of all the other chains.

### Basis for complete spin subsystem

In the first step we need to orthonormalise the dot down matrix  $A_{0\downarrow}$ . Using the orthonormality of the adjacent effective chain bases  $|o_{l\downarrow}\rangle$  and  $|o_{r\downarrow}\rangle$  we obtain Figure A.5 as a condition on  $A_{0\downarrow}$ . The orthonormal basis  $|v\rangle$ , which is determined by  $A_{0\downarrow}$ ,

$$A_{o_{l\downarrow}o_{r\downarrow}v}^{[\sigma_{0\downarrow}]} A_{o_{l\downarrow}o_{r\downarrow}v'}^{[\sigma_{0\downarrow}]*} = o_{l\downarrow} \left[ \begin{array}{c} \boxed{A_{0\downarrow}} \\ \boxed{A_{0\downarrow}^*} \end{array} \right] o_{r\downarrow} = \begin{array}{c} \left[ \begin{array}{c} v \\ v' \end{array} \right] = \delta_{v'v}$$

Figure A.5: Orthonormalisation condition for the dot down matrix

$$|v\rangle = A_{o_{l\downarrow}o_{r\downarrow}v}^{[\sigma_{0\downarrow}]} |o_{l\downarrow}\rangle |o_{r\downarrow}\rangle |\sigma_{0\downarrow}\rangle, \quad (\text{A.19})$$

then describes the whole spin down part of the system.

<sup>2</sup>In order to group the inner and the local index of  $A$  we need to permute the local index with the outer one. As long as we do not interchange the inner and the outer index, this permutation does not influence our reasoning. Only when transforming  $A_{xi_k}$  back to its original form, one needs to undo this permutation.

<sup>3</sup>See also Figure 2.15b.

### Dot matrix orthonormalised towards a chain

The second step provides a condition (Figure A.6) on the dot up matrix  $A_{0\uparrow}$  assuming the the bases  $|v\rangle$  and  $|o_{l\uparrow}\rangle$  to be orthonormal. Given that  $A_{0\uparrow}$  satisfies this condition, the states

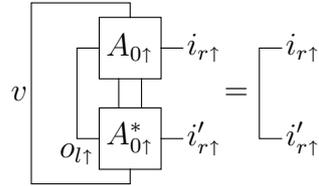


Figure A.6: Orthonormalisation condition for the dot up matrix with respect to the right chain index

$$|i_{r\uparrow}\rangle = A_{o_{l\uparrow}i_{r\uparrow}v}^{[\sigma_{0\uparrow}]} |o_{l\uparrow}\rangle |v\rangle |\sigma_{0\uparrow}\rangle \quad (\text{A.20})$$

constitute an orthonormal ISS basis for the first site of the right spin up chain.



# List of Figures

1.1	Anderson model for a single conduction band and one dot level. . . . .	4
1.2	Sketch of a quantum dot coupled to several leads . . . . .	4
1.3	Logarithmic energy discretisation . . . . .	6
1.4	Iterative construction of NRG state space . . . . .	9
2.1	Calculation of the effective outer state space basis . . . . .	12
2.2	Graphical representation of the right-hand site of (2.6). . . . .	14
2.3	Graphical equivalent of (2.8) for $k = 1$ . . . . .	15
2.4	Graphical MPS Representation for a general Anderson model . . . . .	16
2.5	Graphical MPS representation of th 2-level 2-lead Anderson model . . . . .	16
2.6	Calculation of the effective inner state space basis . . . . .	17
2.7	Diagrams for calculating the ISS bases for the beginning of a chain . . . . .	17
2.8	Scalar product $\langle \psi'   \psi \rangle$ in MPS language. . . . .	18
2.9	MPS representation of the partial product $(S_{\alpha s}^k)_{i' i}$ . . . . .	19
2.10	Generalised partial prduct . . . . .	19
2.11	Product of spin up subsystem . . . . .	19
2.12	Complementary partial product . . . . .	20
2.13	One-site density matrix . . . . .	20
2.14	Two-site density matrix . . . . .	20
2.15	Orthonormality conditions for outer and inner $A$ -matrices. . . . .	22
2.16	Basis transformation of operator representations . . . . .	23
2.17	Basis transformation for non orthonormal states . . . . .	23
2.18	Global way for the evaluation of an operator . . . . .	24
2.19	Local way for the evaluation of an operator . . . . .	24
2.20	Global way for calculating the expectation value . . . . .	25
2.21	Global way for calculating the expectation value of the Hamiltonian . . . . .	25
2.22	Local calculation of the expectation value . . . . .	26
2.23	Local expectation value calculation with Jordan-Wigner corrections . . . . .	26
2.24	Singular value decomposition for rectangular matrices. . . . .	28
2.25	Orthonormalisation of $A_k$ towards the inner direction . . . . .	29
2.26	Example of matrix dimensions along a chain . . . . .	31
2.27	Graphical representation of the ground state minimisation problem. . . . .	32
2.28	Partioning of the Hamiltonian into smaller parts. . . . .	33

2.29	Recursion formula for inner part of the Hamiltonian . . . . .	33
2.30	Sweeping sequence . . . . .	34
2.31	Energy convergence plot . . . . .	35
2.32	Orthonormalisation of $A_k$ . . . . .	37
3.1	bond entropy . . . . .	42
3.2	Flow diagram . . . . .	43
3.3	Spinless two-level system . . . . .	45
3.4	Spinful two-level system . . . . .	46
3.5	Occupation of a four-level dot . . . . .	47
A.1	Ordering scheme for the Jordan-Wigner transformation . . . . .	53
A.2	Orthonormalisation condition for $ o\rangle =  o_{L-1}\rangle$ . . . . .	54
A.3	Orthonormalisation condition for $A_{k+1}$ . . . . .	55
A.4	Orthonormalisation condition for $A_{k-1}$ . . . . .	55
A.5	Orthonormalisation condition for the dot down matrix . . . . .	56
A.6	Orthonormalisation condition for the dot up matrix . . . . .	57

# Bibliography

- [1] P. W. Anderson, *Localized Magnetic States in Metals*, Phys. Rev. **124** 41 (1961).
- [2] Kenneth G. Wilson, *The renormalization group: Critical phenomena and the Kondo problem*, Rev. Mod. Phys. **47** 773 (1975).
- [3] P. Jordan and E. Wigner, *Über das Paulische Äquivalenzverbot*, Z. Phys. **47** (1928), 631.
- [4] H. R. Krishna-murthy, J. W. Wilkins, and K. G. Wilson, *Renormalization-group approach to the Anderson model of dilute magnetic alloys. I. Static properties for the symmetric case*, Phys. Rev. B **21** 1003 (1980).
- [5] S. R. White and R. M. Noack, *Real-space quantum renormalization groups*, Phys. Rev. Lett. **68** 3487 (1992).
- [6] Steven R. White, *Density matrix formulation for quantum renormalization groups*, Phys. Rev. Lett. **69** 2863 (1992).
- [7] Steven R. White, *Density-matrix algorithms for quantum renormalization groups*, Phys. Rev. B **48** 10345 (1993).
- [8] U. Schollwöck, *The density-matrix renormalization group*, Rev. Mod. Phys. **77** 259 (2005).
- [9] M. Fannes, B. Nachtergaele, and R. F. Werner, *Finitely correlated states on quantum spin chains*, Comm. Math. Phys. **144** 443 (1992).
- [10] J. Dukelsky, M. A. Martín-Delgado, T. Nishino, and G. Sierra, *Equivalence of the variational matrix product method and the density matrix renormalization group applied to spin chains*, Europhys. Lett. **43** 457 (1998).
- [11] F. Verstraete and J. I. Cirac, *Mapping local Hamiltonians of fermions to local Hamiltonians of spins*, J. Stat. Mech P09012 (2005).
- [12] F. Verstraete, J. J. Garcia-Ripoll, and J. I. Cirac, *Matrix Product Density Operators: Simulation of Finite-Temperature and Dissipative Systems*, Phys. Rev. Lett. **93** 207204 (2004).

- [13] F. Verstraete, M. A. Martin-Delgado, and J. I. Cirac, *Diverging Entanglement Length in Gapped Quantum Spin Systems*, Phys. Rev. Lett. **92** 087201 (2004).
- [14] Stellan Östlund and Stefan Rommer, *Thermodynamic Limit of Density Matrix Renormalization*, Phys. Rev. Lett. **75** 3537 (1995).
- [15] F. Verstraete, D. Porras, and J. I. Cirac, *Density matrix Renormalization Group and Periodic Boundary Conditions: A Quantum Information Perspective*, Phys. Rev. Lett. **93** 227205 (2004).
- [16] Michael Sindel, Alessandro Silva, Yuval Oreg, and Jan von Delft, *Charge oscillations in quantum dots: Renormalization group and Hartree method calculations*, Phys. Rev. B **72** 125316 (2005).
- [17] F. Verstraete, A. Weichselbaum, U. Schollwöck, J. I. Cirac, and Jan von Delft, *Variational matrix product state approach to quantum impurity models*, cond-mat/0504305 (2005).
- [18] Frithjof B. Anders and Avraham Schiller, *Real-Time Dynamics in Quantum-Impurity Systems: A Time-Dependent Numerical Renormalization-Group Approach*, Phys. Rev. Lett. **95** 196801 (2005).
- [19] A. Weichselbaum and Jan von Delft, *Sum-rule-conserving Spectral Functions from the Numerical Renormalization Group*, cond-mat/0607497 (2006).
- [20] H. R. Schwarz, *Numerische Mathematik*, Stuttgart, 1997<sup>4</sup>.
- [21] James W. Demmel, *Applied numerical linear algebra*, Philadelphia, 1997.

# Acknowledgements

At the end of this thesis, I would like to thank all people who made this possible.

My parents Ina and Fritz Holzner for their enduring support over all the years of my studies.

Prof. Dr. Jan von Delft for the opportunity to join his group and for his advice and support.

A sincere “Vergelt’s Gott” to my advisor Andreas Weichselbaum, who was always willing to help. I learned a lot from the discussions with you.

All people at the Chair for Theoretical Condensed Matter Physics at the LMU, Prof. Dr. Stefan Kehrein, Florian, Michael, Ferdinand, Hamed, Theresa, Peter, Benjamin, Enrique, Afif and Stéphane. They have all been great company during this year and very helpful. I especially would like to thank Hamed for the interesting and helpful discussions about MPS and Theresa for the discussions about NRG and proofreading my thesis.

Ralf Simmler, Alexander Thomas and Philipp Frik for keeping the computers up and running.

Prof. Dr. Ignacio Cirac from the MPQ in Munich, who kindly agreed to co-report on this thesis.