

Variationele Renormalisatie Groep methodes in de veeldeeltjesfysica

Frédéric Peyskens

Promotor: prof. dr. Dimitri Van Neck

Begeleiders: Brecht Verstichel, Matthias Degroote, ir. Sebastian Wouters

Masterproef ingediend tot het behalen van de academische graad van
Master in de ingenieurswetenschappen: toegepaste natuurkunde

Vakgroep Fysica en Sterrenkunde

Voorzitter: prof. dr. Dirk Ryckbosch

Faculteit Ingenieurswetenschappen en Architectuur

Academiejaar 2010-2011



Variationele Renormalisatie Groep methodes in de veeldeeltjesfysica

Frédéric Peyskens

Promotor: prof. dr. Dimitri Van Neck

Begeleiders: Brecht Verstichel, Matthias Degroote, ir. Sebastian Wouters

Masterproef ingediend tot het behalen van de academische graad van
Master in de ingenieurswetenschappen: toegepaste natuurkunde

Vakgroep Fysica en Sterrenkunde

Voorzitter: prof. dr. Dirk Ryckbosch

Faculteit Ingenieurswetenschappen en Architectuur

Academiejaar 2010-2011



Woord Vooraf

Toen ik op het einde van mijn eerste bachelor voor de keuze stond om een specialisatierichting te kiezen werd er mij door heel wat vrienden aangeraden om *geen* Toegepaste Natuurkunde te kiezen. *Dat was immers veel te moeilijk en ik zou amper nog tijd hebben voor iets anders.* Nu, 4 jaar later, kan ik met gerust hart zeggen dat ik geen enkel moment spijt heb gehad van mijn blijkbaar niet zo voor de hand liggende keuze. Toen ik het cursusoverzicht van de Toegepaste Natuurkunde bekeek, stond ik vooral versteld van het ruime aanbod aan disciplines. En dat was precies wat ik wou: de wereld bekijken vanop een fundamenteel niveau en proberen een inzicht te verkrijgen in de meest uiteenlopende fysische processen. Gaandeweg raakte ik meer geïnteresseerd in de wereld van de kwantummechanica. De manier waarop een mathematisch formalisme in staat is om de meest fundamentele zaken te voorspellen met een grote accuraatheid was iets dat mij enorm fascineerde. En hoewel ook andere vakken zeer interessant waren, besloot ik toch om mijn thesis te kiezen in deze meer fundamentele tak van de fysica. De kwantumwereld heeft iets mysterieus en er is nog zoveel dat ontdekt moet worden op het kleinste niveau en het is precies het mysterieuze en fundamentele dat mij zo enorm wist te bekoren.

Bovendien zou ik graag nog een aantal mensen wens te bedanken. In de eerste plaats mijn promotor Professor Dimitri Van Neck. Hij stond ten alle tijde paraat om mij bij te staan en te helpen de theoretische concepten te doorgronden. Bovendien gaf hij mij ook voldoende vrijheid om aan de thesis te werken en kwam hij geregeld langs om te vragen hoever ik stond. Daarnaast dien ik ook mijn dank te betuigen aan mijn begeleiders Sebastian Wouters, Brecht Verstichel en Matthias Degroote. Vele keren ging ik bij hen aankloppen en moesten zij hun werk onderbreken. Zij stonden echter steeds met evenveel enthousiasme paraat om mij bij te staan met al mijn vragen. Bovendien wens ik ook de andere medewerkers van *de molmod* te bedanken voor het creëren van een aangename werksfeer.

Natuurlijk wil ik ook mijn ouders bedanken voor hun onvoorwaardelijke steun, het vele geduld en de goede zorgen. Zij hebben mij de kans gegeven om verder te studeren en stimuleren en ondersteunen mij in alles wat ik doe. Ik kan steeds bij hen terecht en ze zijn altijd in de weer om het voor mij zo aangenaam mogelijk te maken.

Toelating tot bruikleen

De auteur geeft de toelating deze masterproef voor consultatie beschikbaar te stellen en delen van de masterproef te kopiëren voor persoonlijk gebruik. Elk ander gebruik valt onder de beperkingen van het auteursrecht, in het bijzonder met betrekking tot de verplichting de bron uitdrukkelijk te vermelden bij het aanhalen van resultaten uit deze masterproef.

Frédéric Peyskens

31 mei 2011

Overzicht

Titel

Variationele Renormalisatie Groep methodes in de veeldeeltjesfysica

Auteur

Frédéric Peyskens

Promotor

prof. dr. Dimitri Van Neck

Begeleiders

Brecht Verstichel, ir. Matthias Degroote, ir. Sebastian Wouters

Algemeen

Masterproef ingediend tot het behalen van de academische graad van
Master in de ingenieurswetenschappen: toegepaste natuurkunde

Vakgroep Fysica en Sterrenkunde

Voorzitter: prof. dr. Dirk Ryckbosch

Faculteit Ingenieurswetenschappen en Architectuur

Academiejaar 2010-2011

Samenvatting

Matrix Product States (MPSs) zijn specifieke golf functie Ansätze die gebruikt kunnen worden om een (ééndimensionaal) veeldeeltjesysteem te karakteriseren. Door de matrixdimensie van de MPSs te beperken kan een snel algoritme ontwikkeld worden om de energie van een dergelijk systeem te bepalen. Bovendien kan men de verwachtingswaarde van elke andere operator (dus niet alleen de Hamiltoniaan) uitrekenen van zodra de MPS bepaald is. We stellen een specifieke blokvorm voor om de energie met behulp van een iteratieve procedure te berekenen. In eerste instantie onderzoeken we ééndimensionale (1D) spinsystemen waarvan de energie berekend wordt met behulp van het sweep algoritme (waar de matrices site per site geoptimaliseerd worden tot er convergentie optreedt). Voor alle geteste modellen vinden we een uitstekende overeenkomst met analytische uitdrukkingen. Vervolgens bekijken we de verstrengelingsentropie S_E , die een maat is voor de kwantumcorrelatie tussen 2 systemen. Het blijkt immers dat wanneer we een

gegeven accuraatheid E_{acc} wensen te bereiken, de efficiënte truncatie van de matrixdimensie (om die E_{acc} te bekomen) in verband staat met de grootte van S_E . De methode wordt daarna uitgebreid om ook fermionische systemen (zoals het 1D Hubbard model) te bestuderen. Additionele fasefactoren worden ingevoerd om rekening te houden met de fermionische anti-commutatierelaties. Daarnaast wordt een kostfunctie ingevoerd om expliciet het deeltjesaantal op te leggen. Zonder kostfunctie zal het algoritme het deeltjesaantal verminderen bij sterke repulsie om op die manier de energie te doen dalen. Tot slot wordt het algoritme uitgebreid om ook moleculaire systemen te bestuderen. In deze thesis beperken we ons (als eerste kennismaking) tot de H_2 molecule met elektronen in een $1s$ -orbitaal. Het algoritme dient opnieuw aangepast te worden om de elektron-elektron repulsie efficiënt te beschrijven. Deze bevat immers 4 operatoren (in tweede kwantisatie) wat resulteert in een groot aantal mogelijke contributies tot de energie. Om te vermijden dat de rekestijd enorm oploopt (als gevolg van de vele termen) wordt een cut-off, die het aantal termen beperkt, ingebouwd.

Trefwoorden

Matrix Product States, , verstrengelingsentropie, 1D spin systemen, 1D Hubbard model, H_2 molecule

Variational Renormalization Group Methods in Many Body Physics

Frédéric Peyskens

Supervisor(s): Prof. dr. Dimitri Van Neck, Brecht Verstichel, ir. Matthias Degroote, ir. Sebastian Wouters

Abstract— We investigate the low energy sector of many body systems by means of a specific wavefunction Ansatz that originates from quantum information theory: the Matrix Product States (MPSs). These MPSs are particularly suited for the description of one-dimensional (1D) chains and the matrixdimension can be related to the concept of entanglement entropy. We will describe spin (e.g. 1D Heisenberg model) as well as fermionic (e.g. 1D Hubbard model) systems and as a last topic the method is extended to calculate the energy-curve of an H_2 molecule.

Keywords— Matrix Product States, entanglement entropy, 1D spin chains, 1D Hubbard model, H_2 molecule.

I. INTRODUCTION

ONE of the most challenging tasks in many body physics is the efficient simulation of systems in a reasonable amount of time (and with acceptable memory usage). The biggest problem is the exponential growth of the associated Hilbert space which results in a very high time-to-accuracy-ratio. Renormalization group (RG) methods might be able to tackle this problem for a large set of Hamiltonians. The applied methods were initially developed for quantum spin systems but can be modified to describe fermionic systems as well.

For quantum spin systems it seems that all physical accessible states live on a tiny submanifold of the complete Hilbert space [1]. In many body physics one is mainly interested in the low energy region; as a result the possible set of wavefunctions is constrained. The main idea is then to find a class of variational wavefunctions which contain all the relevant low energy physics and it can be shown that such a parametrization exists (by means of quantum information theory). In the Density Matrix Renormalization Group (DMRG) method the Hilbert space is truncated and the most significant parts of the density matrix are kept. The VMPS method that we will describe is a variational reformulation of DMRG [2]. Both DMRG and VMPS are RG methods.

First the Matrix Product States (MPSs) are introduced for the description of 1D chains. We briefly discuss the concept of the sweep algorithm, which allows for an accurate and reasonably fast calculation of the energy. In a second part we present the results of some calculations on spin systems and explain why entanglement entropy is related to the dimension of the associated Hilbert space. Then the method is extended to describe fermionic systems (such as the Hubbard model) as well. In this case it is necessary to incorporate additional phase factors and we explicitly need to impose a particle number. Finally, we modify the algorithm to simulate a H_2 molecule.

II. THEORETICAL BACKGROUND

A. Matrix Product States

In this discussion it is assumed that we work with quantum systems on a 1D chain with length L . Each site is then described by a d -dimensional local space $\{\sigma_i\}$, $i = 1 \dots L$ and the wavefunction of such a chain can be written as

$$|\psi\rangle = \sum_{\sigma_1, \dots, \sigma_L} c_{\sigma_1, \dots, \sigma_L} |\sigma_1, \dots, \sigma_L\rangle. \quad (1)$$

By consecutive QR-decompositions it can be shown [2] that

$$c_{\sigma_1, \dots, \sigma_L} = \sum_{a_1, \dots, a_{L-1}} A_{1, a_1}^{\sigma_1} A_{a_1, a_2}^{\sigma_2} \dots A_{a_{L-2}, a_{L-1}}^{\sigma_{L-1}} A_{a_{L-1}, 1}^{\sigma_L}. \quad (2)$$

This form of the wavefunction is called a Matrix Product State (MPS) as the coefficients are written as products of matrices. In [2] it is also shown that the dimension of the matrices A^{σ_i} is given by (d^{i-1}, d^i) , $\forall i = 1 \dots \lfloor \frac{L}{2} \rfloor$ and by (d^{L-i+1}, d^{L-i}) , $\forall i = (\lfloor \frac{L}{2} \rfloor + 1) \dots L$. However for large L this quite rapidly exceeds the permissible dimensions for numerical simulations. Therefore an artificial truncation dimension D is introduced to limit this growth and $\forall n : d^n > D$, d^n is truncated at a value D . As a result the dimension of the associated Hilbert space remains limited and it turns out that this parametrization is indeed capable of describing the low energy sector of 1D quantum chains very efficiently.

B. Sweep algorithm

When we write $\langle \psi | \hat{H} | \psi \rangle$ in terms of (1) it turns out that the energy can be written in a specific block-form which divides the chain into a left block (containing sites $1 \dots l-1$), a right block (containing sites $l+1 \dots L$) and one site l . Due to the specific nature of the blocks, they can be iteratively constructed out of each other when we vary l . The matrices are optimized site by site which means that we optimize the matrices associated to site l while keeping the other $L-1$ fixed. Once the new set of matrices A^{σ_l} is found, one needs to construct new blocks before optimizing site $l+1$ (right sweep) or $l-1$ (left sweep). Once the end of the chain is reached the sweep direction is reversed and the complete cycle then looks like $(1 \rightarrow L \rightarrow 1)$. This procedure (the *sweep algorithm*) continues until the energy converges and the method is commonly called the VMPS method (Variational Matrix Product State).

III. SPIN SYSTEMS

We simulate several important spin systems and compare the numerical data with analytical results. The effect of the

truncation dimension is also investigated. In this section we will consider spin Hamiltonians of the form

$$\hat{H} = \sum_{i<j} J_{ij} \hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_j - \beta \sum_{i=1}^L \hat{S}_i^z \quad (3)$$

where $\hat{\mathbf{S}}_i$ is the spin operator that works on site i and β is a measure for the magnetic field strength.

A. Isotropic Heisenberg model

We calculate the energy as a function of the chain length L ($L = 10 \rightarrow 100$) for the isotropic Heisenberg model ($J_{ij} = J\delta_{j,i+1}, J \in \mathbb{R}$) with $D = 8$. The results are in excellent agreement with the Bethe Ansatz [3] (even for small L ($L = 10$) despite the fact that the Bethe Ansatz holds for large L ($L \rightarrow \infty$)). For $L = 100$ an exact treatment would require a Hilbert space of dimension $D = 2^{50} \approx 10^{15}$ while here $D = 8$ is sufficient to obtain already an accurate result.

B. $J_1 - J_2$ model and entanglement entropy

The $J_1 - J_2$ model ($J_{ij} = J_1\delta_{j,i+1} + J_2\delta_{j,i+2}, J_{1,2} \in \mathbb{R}$) is exactly solvable when $J_2 = J_1/2$ (the Majumdar-Ghosh chain [4]) and the energy is then given by $E_{MG} = -3J_1L/8$ when L is even. We simulate this case (with $D = 2$) for several $L = 2n, n \in \mathbb{R}$ and find an excellent agreement (for odd L the results deviate from E_{MG}). The reason why this method works well for modest values of D ($D = 2$ is sufficient for even L) is related to the topic of entanglement entropy. Entanglement entropy S_E describes the quantum correlation between two systems [5]. Strongly correlated systems will have high S_E values while uncorrelated systems will have $S_E = 0$. For $J_2 = J_1/2$ and $L = 2n, n \in \mathbb{R}$ a dimerization process will take place in which neighboring spins will couple to a singlet state. These pairs of spins are then decoupled from the rest of the chain (which results in $S_E = 0$ for blocks with an even number of sites).

C. Convergence

One can show that $\log(E_D - E_{conv}) \cong -\kappa \log(D)^2, \kappa \in \mathbb{R}$, with E_D the energy for a given D and E_{conv} the energy at convergence [6]. The larger κ , the faster the convergence will be. We illustrate this for the $J_1 - J_2$ model for different ratios of $J_2/J_1, J_1 = 4$ in figure (1). For $J_2/J_1 = 1/2$ the energy converges immediately (for $D = 2$). This is a result of the dimerization for even L . For other ratios the slope slightly depends on the value of J_2/J_1 (κ ranges from -2.35 to -2.92 if we look at the tail of all curves with $J_2/J_1 \neq 1/2$), which means that the speed of convergence (κ) depends on the parameters of the model. However, one can see that for $D = 16$ the difference $E_D - E_{conv}$ is of the order 10^{-5} , which is already quite good, since an exact diagonalization would require $D = 1024$.

IV. FERMIONIC SYSTEMS

The fermionic systems we consider in this section are all of the form

$$\hat{H} = - \sum_{s \in (\uparrow, \downarrow)} \sum_{i<j} T_{ij} \left(\hat{a}_{i,s}^\dagger \hat{a}_{j,s} + \hat{a}_{j,s}^\dagger \hat{a}_{i,s} \right) + U \sum_{i=1}^L \hat{n}_{i,\uparrow} \hat{n}_{i,\downarrow}. \quad (4)$$

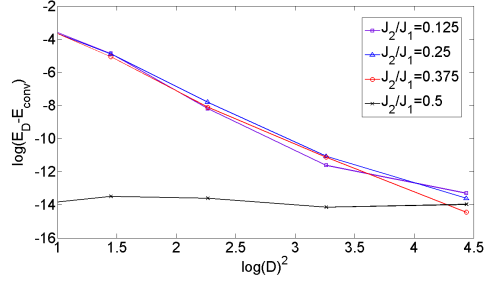


Fig. 1. Convergence of the $J_1 - J_2$ model, $L = 20, \beta = 0$.

with $\hat{n}_{i,s} = \hat{a}_{i,s}^\dagger \hat{a}_{i,s}$ the number operator and $U > 0$. While spin operators on different sites satisfy commutation relations ($[\hat{\mathbf{S}}_i, \hat{\mathbf{S}}_j] = 0$), fermionic creation and annihilation operators anti-commute. This requires an adaptation when long-range fermion systems are considered. First we need to define a local basis, which in our case is $\{|0\rangle, |\uparrow\rangle, |\downarrow\rangle, |\uparrow\downarrow\rangle\}$ (consistent with the Pauli principle). The local matrices for an operator \hat{O} are then constructed for each $\hat{O}_{i,s} \hat{O}_{i+1,s}$ combination while matrices for $\hat{O}_{i,s} \hat{O}_{i+j,s}, j > 1$ will obtain an additional phase of (-1) when an odd number of particles is crossed and $+1$ when the number of particles between i and $i + j$ is even. This is implemented by multiplying the matrices (for each site between i and $i + j$) associated with $\{|\uparrow\rangle, |\downarrow\rangle\}$ with (-1) while the matrices for $\{|0\rangle, |\uparrow\downarrow\rangle\}$ are not modified (or multiplied with $(+1)$). Operators which contain an even number of fermionic operators (like $\hat{n}_{i,s}$) do not require this phase factor because each fermionic operator will give a phase (-1) , but as the number of operators itself is even, it will always result in $(+1)$.

For large values of $U > 0$ the system will tend to reduce the number of particles in order to lower the energy. However, we want to control the number of particles in the system and impose that it equals some constant N . Therefore a quadratic cost function is added to the Hamiltonian: $\hat{H}_N = \hat{H} + \lambda(\hat{N} - N)^2$ with $\hat{N} = \sum_{s \in (\uparrow, \downarrow)} \sum_{i=1}^L \hat{n}_{i,s}$ and $\lambda > 0$. Because of this term the energy will increase when the particle number deviates from the desired number N . As a rule of thumb one must choose λ at least equal to $10 \max(|T|, |U|)$, to make sure that the increase in energy is large enough (which in turn will result in the desired N as for this specific value the positive contribution of $\lambda(\hat{N} - N)^2$ will vanish).

We test our fermionic algorithm with the 1D Hubbard model ($T_{ij} = T\delta_{j,i+1}, T \in \mathbb{R}$) for which an analytical result is known in the limit $L \rightarrow \infty$ and $N = L$ [7]. We take $U = 0$ since for this value exact results can be obtained by diagonalization of an $(L \times L)$ matrix. The results are listed in table (I). The VMPS result for $D = 64$ is in good agreement with the exact results and we see that the solution tends to the analytical limit if we increase L . The 1D $t_1 - t_2$ model ($T_{ij} = T_1\delta_{j,i+1} + T_2\delta_{j,i+2}, T_{1,2} \in \mathbb{R}$) is also tested and we find the transition to a ferromagnetic state for some critical $U_c \approx 7$ for a chain of length $L = 4$ and $N = 2, T_1 = 1, T_2 = -0.2$ ($D = 16, \lambda = 100$). This is in good agreement with the literature [8].

L	$E_{exact}/L(*)$	E_{VMPS}/L
6	-1.16465306914	-1.16465306914
10	-1.20533483667	-1.20533346501
20	-1.23814899997	-1.23812247319
30	-1.24957737436	-1.24950078441
40	-1.25538985558	-1.25525451485
$\rightarrow \infty$	-1.27323764229	—

TABLE I

$\frac{E}{L}$ AS A FUNCTION OF L FOR $U = 0, \lambda = 0$ ($D = 64, \hbar = 1$).

(*) EXACT VALUES WERE OBTAINED FROM S. WOUTERS.

V. H_2 MOLECULE

The ultimate goal is to combine the speed and accuracy of the VMPS method to describe molecular systems. We will elaborate on this task by considering an H_2 molecule with the electrons in a $1s$ -orbital. However, the fermionic algorithm is not directly applicable because the two-body operator \hat{V} , which describes the electron-electron repulsion, consists of four operators [9]. Therefore we rewrite the interaction in terms of operators that act on one, two, three and four sites respectively ($V^{1,2,3,4}$). For H_2 we only need V^1 and V^2 . Besides that, the coupling constants (T_{ij} in Hubbard) are two-center spatial integrals ($\langle ij | \hat{V} | kl \rangle$) (depending on 4 indices) [9]. Remarks concerning the phase factors and the quadratic cost function still need to be taken into account. We test our algorithm in the STO-6G basis and calculate the energy (and dissociation energy) of a H_2 molecule (Fig. 2) for which we find an equilibrium distance of 0.73503102 \AA and a minimal energy of $E_{min} = -1.145977$ Hartree which is consistent with FCI calculations. Better basis sets will allow for a more efficient calculation (in order to reproduce experimental results).

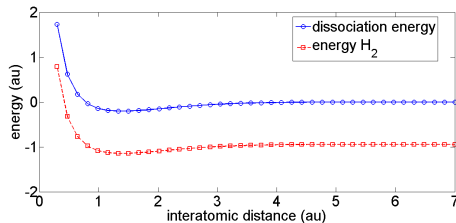


Fig. 2. Energy curves for H_2 molecule, $D = 4, \lambda = 7$.

VI. CONCLUSIONS

We can describe 1D many body systems with the help of Matrix Product States (MPSs), which allow for an efficient parametrization of the many particle wavefunction and describe the low energy sector of Hilbert space very well (although excited states can be considered too). Since the energy can be calculated iteratively and the matrix dimension can be truncated (related to the behaviour of the entanglement entropy), we end up with a sufficiently good time-to-accuracy-ratio. The method was applied to spin systems as well as fermionic systems and results were in (very) good agreement with analytical expressions. Finally we extended the algorithm to calculate the energy of an H_2 molecule. Results were in excellent agreement with FCI calculations.

ACKNOWLEDGMENTS

The author would like to acknowledge the support of prof. dr. D. Van Neck, ir. S. Wouters, B. Verstichel and ir. M. Degroote and thank them for the many stimulating conversations.

REFERENCES

- [1] F. Verstraete, V. Murg, J.I. Cirac, *Matrix product states, projected entangled pair states, and variational renormalization group methods for quantum spin systems*, Advances in Physics, 2008.
- [2] U. Schollwöck, *The density-matrix renormalization group in the age of matrix product states*, arXiv:1008.3477v1, 2010.
- [3] M. Karbach, K. Hu, G. Müller, *Introduction to the Bethe Ansatz II*, arXiv:cond-mat/9809163 v1, 1998.
- [4] R.W. Chhajlany, P. Tomczak, A. Wojcik, J. Richter, *Entanglement in the Majumdar-Ghosh model*, Physical Review, 2007.
- [5] J. Preskill, *Quantum Computation*, Caltech university course, 2011.
- [6] J. Hachmann, W. Cardoen, G.K.L. Chan *Multireference correlation in long molecules with the quadratic scaling density matrix renormalization group*, J. Chem. Phys., 2006.
- [7] E.H. Lieb, F.Y. Wu, *The one-dimensional Hubbard model: A reminiscence*, arXiv:cond-mat/0207529v2 [cond-mat.str-el], 2002.
- [8] S. Daul, R.M. Noack, *Ferromagnetic transition and phase diagram of the one-dimensional Hubbard model with next-nearest-neighbor hopping*, arXiv:cond-mat/9712310v1 [cond-mat.str-el], 1997.
- [9] W.H. Dickhoff, D. Van Neck, *Many-Body Theory Exposed! Propagator Description of Quantum Mechanics in Many-Body Systems, 2nd Edition*, World Scientific Publishing, 2008.

The laws of nature are constructed in such a way as to make the universe as interesting as possible.

Freeman Dyson

Inhoudsopgave

1	Renormalisatie	1
1.1	Inleiding	1
1.2	Renormalisatie	2
1.2.1	Renormalisatiegroep theorie	2
1.3	Renormalisatie en spinsystemen	5
1.3.1	De NRG-methode van Wilson	5
1.3.2	DMRG	5
1.3.3	Matrix Product States en de truncatiedimensie	7
1.4	Een vooruitblik	7
2	De VMPS methode	9
2.1	Inleiding en doel	9
2.2	Matrix Product State of MPS	9
2.2.1	Overgang naar MPS	10
2.2.2	Canonische vormen van een MPS	12
2.3	MPS-vorm van verwachtingswaarden	13
2.3.1	Bepalen van de verwachtingswaarde $\langle \psi \hat{H} \psi \rangle$	14
2.3.2	Diagrammatische voorstelling	18
2.3.3	De interactierange η	20
2.4	De X/Y -blokstructuur van $\langle \psi \hat{H} \psi \rangle$	21
2.4.1	Constructie van de Y^L/X^L -blokken	21
2.4.2	Constructie van de Y^R/X^R -blokken	26
2.4.3	$\langle \psi \hat{H} \psi \rangle$ in X/Y -vorm	26
2.5	Variationele optimalisatie en de effectieve Hamiltoniaan	30
2.6	Verwachtingswaarde van fysische grootheden	34
2.7	Verstrengelingsentropie	34
2.8	Convergentie van de VMPS methode	36
2.8.1	Dichtste-nabuur Hamiltonianen	37
2.8.2	Het $J_1 - J_2$ model	39
2.8.3	Conclusies met betrekking tot de convergentie	40
3	Spinsystemen	42
3.1	Het isotroop Heisenbergmodel	42
3.1.1	Het isotroop Heisenbergmodel voor $L = 2$	42

3.1.2	Het isotroop Heisenbergmodel voor $L > 2$	44
3.2	De $J_1 - J_2$ Hamiltoniaan	47
3.2.1	Het exact oplosbare model	47
3.2.2	Verstrengelingsentropie van de $J_1 - J_2$ keten	49
3.2.3	Magnetisatie van een $J_1 - J_2$ ketting	52
3.3	Het bilinear biquadratic model	54
4	Fermionsystemen	56
4.1	Fundamentele aspecten van de veeldeeltjesfysica	56
4.1.1	Inleiding	56
4.1.2	De Fock-ruimte	57
4.1.3	Creatie- en annihilatie-operatoren	57
4.1.4	Ééndeeltjesoperatoren in de Fock-ruimte	58
4.1.5	Tweedeeltjesoperatoren in de Fock-ruimte	58
4.2	Van spin-roosters naar fermion-roosters	59
4.2.1	Opleggen van het deeltjesaantal: de kwadratische kostfunctie	61
4.2.2	Long-range fermionsystemen	63
4.3	Het Hubbard-model	64
4.3.1	Invloed van de kostparameter λ	65
4.4	Het $t_1 - t_2$ model	66
5	H_2 molecule	68
5.1	Hamiltoniaan van een moleculair systeem	68
5.2	Bepaling van de ruimte-integralen	70
5.2.1	Kinetische energie	70
5.2.2	Attractieve elektron-kern interactie	71
5.2.3	Repulsieve elektron-elektron interactie	71
5.3	De tweedeeltjesinteractie \hat{V}	72
5.4	Energie en dissociatiecurve van een H_2 molecule	73
5.5	Theoretische beschouwingen voor long-range systemen	74
5.5.1	X/Y -blokken voor V^3 en V^4	75
5.5.2	Constructie van de lokale matrices	77
5.5.3	Energie van een algemene H-chain	80
6	Conclusies	81
A	Numerieke implementatie van de VMPS methode	83
A.1	Implementatie	83
A.1.1	Het sweep algoritme	83
A.1.2	Generatie en normalisatie van een MPS	84
A.1.3	Implementatie van de Hamiltoniaan	85
A.1.4	Initialisatie van de blokken	86
A.1.5	De sweep-cycli	87

B	Additionele uitleg bij de MATLAB-code	94
B.1	De functie contract.m	94
B.2	De functie construct Yblocks	95
B.2.1	Constructie van de Y^L -blokken	95
B.2.2	Constructie van de Y^R -blokken	96
B.3	De functie construct Xblocks	98
B.3.1	Constructie van de X^L -blokken	98
B.3.2	Constructie van de X^R -blokken	99
C	Input-files van gebruikte Hamiltonianen	101
C.1	Het Heisenbergmodel	101
C.2	Het $J_1 - J_2$ model	102
D	Exact oplosbaar $J_1 - J_2$ model	104
E	Tweedeeltjesinteractie voor de H_2 molecule	108
	Bibliografie	111

Hoofdstuk 1

Renormalisatie

1.1 Inleiding

Deze thesis heeft als doel ééndimensionale veeldeeltjessystemen accuraat te beschrijven met behulp van variationele renormalisatiegroep methodes. Het is tot op de dag van vandaag nog steeds een grote uitdaging om veeldeeltjessystemen accuraat te simuleren omdat de geassocieerde Hilbertruimte exponentieel groeit met het deeltjesaantal. Reeds vele methodes zijn ontwikkeld om veeldeeltjessystemen goed te beschrijven. Hartree-Fock maakt gebruik van een gemiddeld veld benadering en is dus niet in staat om de correlatie (als gevolg van de Coulomb-repulsie) accuraat te behandelen. Density Functional Theory (DFT) probeert dit enigszins te verbeteren, al moet men daarbij vaak gebruik maken van (semi)-empirische functionalen die gefit zijn aan moleculaire sets of geconstrueerd zijn vanuit theoretische limieten waaraan de functionaal moet voldoen ([1]). Coupled-cluster theory behandelt de dynamische correlaties via een exponentiële Ansatz die de exacte niet-relativistische golffunctie schrijft als $\exp(\hat{T})|\phi_0\rangle$ met \hat{T} een welbepaalde operator ([2]). De onbekende operator \hat{T} kan dan orde per orde bepaald worden door substitutie van de Ansatz in de Schrödingervergelijking. Vermits je natuurlijk nooit oneindig veel ordes (als gevolg van de exponentiële functie) kan bepalen, is deze methode ook een benadering van het exacte resultaat. De Full Configuration Interaction (FCI) methode genereert een exacte oplossing voor het probleem (in een gegeven basisset). De enige fout is dus afkomstig van het feit dat de basisset niet compleet is (de fout is afhankelijk van de keuze van het *eindige* basisset) ([3]). Zulke berekeningen zijn enkel mogelijk voor voldoende kleine systemen, omdat de rekentijd anders enorm oploopt. Daarnaast is er ook nog de klasse van variationele renormalisatiegroep (RG) methodes waaronder de NRG-methode van Wilson of de de DMRG-methode¹. Beide zullen we verderop kort bespreken. Ook de VMPS methode, die we in deze thesis zullen bestuderen, behoort tot de klasse van RG methodes. De VMPS methode is een mathematische herformulering van DMRG tot een variationele Ansatz ([4]). Deze methode maakt gebruik van Matrix Product States (die we grondig zullen bespreken in hoofdstuk 2), die het eenvoudigste geval vormen van een meer algemene klasse zogenaamde Tensor Network States (TNS). Voor de beschrijving van tweedimensionale systemen wordt bijvoorbeeld gebruik gemaakt van Projected Entangled Pair States (PEPS) ([5]). In wat volgt zullen we iets dieper ingaan op het concept van

¹Uiteraard zijn er nog andere methodes die we hier niet vermelden. Deze inleiding heeft enkel tot doel een aantal van de bestaande technieken op te sommen.

renormalisatie en we overlopen tot slot wat we in de volgende hoofdstukken zullen behandelen.

1.2 Renormalisatie

Om deze thesis in een breder kader te plaatsen en te motiveren waarom de verderop besproken methodes gerechtvaardigd zijn, zullen we in de volgende sectie het concept van *renormalisatie* toelichten. Renormalisatie kent een rijke geschiedenis en reeds vele domeinen binnen de fysica konden profiteren van de renormalisatiegroep theorie. Dat deze theorie van groot belang is, mag blijken uit de Nobelprijs in de fysica uitgereikt in 1982 aan Kenneth G. Wilson voor de studie van kritische fenomenen in verband met fasetransities. In zijn Nobel Lecture geeft Wilson een motivatie voor het gebruik van de renormalisatiegroep ([6]) en we zullen dan ook de probleemstelling overnemen uit deze lezing.

Vele macroscopische fenomenen in de fysica hebben een microscopische achtergrond. In de simpelste gevallen kunnen de microscopische fluctuaties uitgemiddeld worden als men het systeem op een grotere afstandsschaal bekijkt. De uitgemiddelde grootheden voldoen dan aan klassieke continuümvergelijkingen; de hydrodynamica is hier een mooi voorbeeld van. Er is echter ook een klasse problemen waarbij fluctuaties bij grotere lengteschalen ook belangrijk zijn. Turbulentie, magnetische onzuiverheden in niet-magnetische metalen (het Kondo-probleem) en kritische fenomenen vormen slechts een greep uit deze klasse. Een kritisch punt is een speciaal geval van een fasetransitie. Als men bijvoorbeeld water bekijkt, dan verdwijnt het onderscheid tussen water en stoom bij het kritisch punt en het kookfenomeen verdwijnt. Bij dit kritisch punt krijgt men een mix van stoombellen en waterdruppels bij elke afstandsschaal. In het Kondo-effect zullen elektronen van alle golflengtes (van atomaire tot veel grotere golflengtes) die zich in de conductieband van een metaal bevinden, interageren met het magnetisch moment van elke onzuiverheid in het metaal. De grote moeilijkheid bij dergelijke problemen ligt in het grote aantal gekoppelde vrijheidsgraden. Om problemen met veel lengteschalen aan te pakken kan de renormalisatiegroep gebruikt worden. De benadering bestaat erin de fluctuaties uit te integreren startende met fluctuaties op de kleinste afstandsschaal. Een kleine afstandsschaal komt overeen met hoge energieën (momenta) volgens Heisenberg ($\Delta x \Delta p \geq \hbar$). Het falen van een theorie bij hogere energieën wordt dan toegeschreven aan het feit dat we een dergelijke (meer fundamentele) theorie nog niet kennen. Men kan dit probleem echter oplossen door gerenormaliseerde grootheden in te voeren. Wilson beschouwt in zijn lezing ([6]) de renormalisatiegroep als middel om de Landau-theorie, die het gedrag van een ferromagneet beschrijft, te verbeteren. Voor meer info omtrent de precieze uitwerking verwijzen we de geïnteresseerde lezer naar ([6]).

1.2.1 Renormalisatiegroep theorie

In ([7]) geeft Fisher een uiteenzetting van de historische ontwikkeling van de renormalisatiegroep (RG) theorie. We pogen dit beeld hier enigszins te reproduceren zonder echter in te gaan op de details. We zullen aan de hand van een specifiek voorbeeld aantonen hoe de renormalisatiegroep methode tussenbeide kan komen om een beter beeld te vormen van de fysische processen. Beschouw daartoe het concept van Kadanoff, zoals beschreven in Fisher ([7]). We starten van een spin $S = \frac{1}{2}$ Ising model; dit is de klassieke (niet-kwantummechanische) beschrijving van een

rooster met op elke site i een spin \mathbf{S}_i , interagerend via de Hamiltoniaan

$$H_{Ising} = -J \sum_{\mathbf{i}} \mathbf{S}_i \cdot \mathbf{S}_{\mathbf{i}+1}. \quad (1.1)$$

De roosterconstante is a en $J > 0$ (neem bijvoorbeeld een tweedimensionaal (2D) rooster; het aantal ruimtelijke dimensies noteren we met $d = 2$). Bij lage temperaturen zullen de spins dus bij voorkeur opgelijnd zijn met elkaar terwijl de ordening arbitrair wordt bij hogere temperaturen. De spontane magnetisatie bij lage temperaturen verdwijnt bij de kritische temperatuur T_c . Veronderstel dat we het 2D-rooster nu opdelen in $(L \times L)$ -blokken met $L = ba$ en $b > 1$. We associëren aan elk blok een spin S_L (elk blok bevat b^d spins) en herschalen de ruimtelijke dimensie \mathbf{x} als

$$\mathbf{x} \rightarrow \mathbf{x}' = \frac{\mathbf{x}}{b}. \quad (1.2)$$

Als men een dergelijke herschaling uitvoert en nu het systeem van blokspins S_L beschouwt moet er ook nog een verband gevonden worden tussen de koppelconstante J van het originele systeem en de gerenormaliseerde koppelconstante J' van het nieuwe herschaalde systeem. Bovendien moet er ook een verband bestaan tussen het oorspronkelijke magnetisch veld H en het geschaalde veld H' . Om dit te verwezenlijken veronderstelde Kadanoff dat b voldoende groot is maar echter kleiner dan $\frac{\xi}{a}$, waarbij $\xi(T)$ de correlatielengte is. Deze correlatielengte geeft aan over welke range de spins elkaar beïnvloeden en divergeert bij de kritische temperatuur T_c . Als gevolg daarvan kan b asymptotisch (dus bij T_c) arbitrair gekozen worden en stelde Kadanoff voor dat de koppeling van een magnetisch veld H aan een blok van spins equivalent is met de koppeling aan de gemiddelde spin van dat blok. We noteren de spin-renormalisatiefactor als $\zeta(b)$. Als men voor de koppelingsconstante (of analoog temperatuur T) ook een renormalisatiefactor $\theta(b)$ invoert dan vinden we als gerenormaliseerde grootheden

$$T' = \theta(b)T \quad H' = \zeta(b)H. \quad (1.3)$$

Uit de behandeling van Kadanoff volgde

$$\theta(b) = b^\lambda \quad \zeta(b) = b^{-\omega} \quad (1.4)$$

waarbij λ en ω het kritische punt karakteriseren. Het grote probleem in de theorie bestond uit de verklaring van deze exponentiële wetten (1.4) en in het bijzonder de verklaring van slechts 1 koppelconstante J' tussen de blokspins. Deze blokken bevatten oorspronkelijk sterk interagerende spins, terwijl we nu naar een uitgemiddeld gedrag kijken. Onder T_c is het gebruik van één J' toegestaan omdat de spins dan toch nagenoeg allemaal up of down staan, maar boven T_c zullen de spins op vele schalen fluctueren en de beschrijving met slechts één J' lijkt in dat geval niet voldoende. Het was Wilson die dit probleem oploste door zijn concepten van de renormalisatiegroep toe te passen.

In de theorie van Kadanoff wordt er verondersteld dat na herschaling (of renormalisatie) de gerenormaliseerde Hamiltoniaan dezelfde vorm heeft als de oorspronkelijke met als enige verschil dat er nu gerenormaliseerde parameters (zoals T' en H') in optreden. Om in te zien hoe de renormalisatiegroep ontstaat starten we van de partitiefunctie van Gibbs

$$Z_N(\bar{H}) = \text{Tr}_N^s \{ \exp \bar{H}(s) \} \quad (1.5)$$

waarbij het spoor staat voor een sommatie of integratie over alle mogelijke N spins in het systeem (gekaracteriseerd door $\bar{H}(s)$). Het uitrekenen van deze partitiefunctie is extreem ingewikkeld. We zullen daarom een alternatieve strategie ontwikkelen.

Stel dat we het systeem verdelen in 2 groepen. De eerste groep bevat N' spins (S_1) die we onaangeroerd laten en de tweede groep bevat de overige $N - N'$ spins (S_2) waarover gesommeerd of geïntegreerd wordt (deze verdwijnen dus uit het probleem). Dit partieel spoor levert uiteindelijk een effectieve Hamiltoniaan ($H_{eff}(S_1)$) op die enkel nog afhangt van de eerste groep spins en gedefinieerd is als

$$\exp H_{eff}(S_1) = \text{Tr}_{N-N'}^{S_2} \{ \exp \bar{H}(s) \}. \quad (1.6)$$

Passen we opnieuw een spatiale herschaling (1.2) toe dan bekommen we uiteindelijk $H'(s') \equiv H_{eff}(S_1)$. De renormalisatietransformatie wordt dan

$$H'(s') = \mathbb{R}_b \{ \bar{H}(s) \} \quad (1.7)$$

waarbij de RG-operator \mathbb{R}_b werd ingevoerd. Merk op dat als we nu het spoor nemen over de gerenormaliseerde spins we de gewenste partitiefunctie (1.5) terugvinden. Er is hierbij dus geen informatie verloren gegaan. Als men echter grondigere berekeningen uitvoert, dan vindt men dat H_{eff} niet dezelfde vorm heeft als de oorspronkelijke $\bar{H}(s)$, in tegenstelling tot de veronderstelling van Kadanoff. In het geval van het Ising model vindt men dat de effectieve Hamiltoniaan ook tweede-orde en hogere-orde interacties bevat tussen de spins. Vandaar dat de renormalisatiegroep een verbetering vormde op de bestaande theorie van Kadanoff. Bij een RG-transformatie zal men proberen eerst de microscopische variabelen die er het minst toe doen te elimineren. Een variant op deze ruimte-RG groep is momentum-RG waar men de hoge momenta (korte afstandsschaal volgens Heisenberg) uitintegreert.

De renormalisatiegroep heeft sinds zijn ontwikkeling de weg gevonden in vele takken van de fysica. Naast toepassingen in de condensed matter fysica wordt de RG-groep veel gebruikt in kwantumveldentheorieën zoals QED en QCD. In QED wordt het uitintegreren van hoge momenta gebruikt om de renormalisatie van de elektronlading uit te rekenen. Dit geeft aanleiding tot de zogenaamde *running coupling constant* waarbij de koppelconstante (\approx elektronlading) een functie wordt van de momentumschaal waarop men kijkt. In QCD wordt de renormalisatiegroep gekoppeld aan de beta-functie en krijgen we ook een *running coupling constant*.

De beschrijving door Wilson van de renormalisatiegroep en het verband met de Kadanoff theorie wordt uitvoeriger beschreven in ([8]). Meer info omtrent de RG-methode en toepassingen ervan in kwantumveldentheorie en andere takken van de fysica kan men vinden in ([9]) en een meer algemene review van het renormalisatieconcept wordt gegeven in ([10]). Deze referenties zijn enkel opgenomen voor lezers die zich wat meer wensen te verdiepen in het concept van renormalisatie. Ze vormen echter geen noodzakelijke literatuur voor de opbouw van de theorie die verderop wordt beschreven.

1.3 Renormalisatie en spinsystemen

Het doel van deze thesis is het bestuderen van een methode die in staat is om sterk gecorreleerde *kwantumsystemen* op een 1D-rooster te beschrijven. De methodes die we zullen bestuderen staan goed beschreven in ([5]) en ([4]). Wij zullen proberen om een zo handig mogelijke implementatie te vinden (om bijvoorbeeld chemische systemen te kunnen simuleren). Voorbeelden van sterk gecorreleerde kwantumsystemen zijn spinsystemen die een fysisch model kunnen zijn van bijvoorbeeld een ferromagneet. Daarnaast zullen we ook fermionische systemen beschrijven zoals het bekende Hubbard model. Ons eerste uitgangspunt is echter een algemeen spinsysteem op een rooster met L sites. Hierbij wordt aan elke site een spin geassocieerd en bij een veralgemening tot fermionen kan men aantonen dat een fermionsysteem in dezelfde functionele vorm geschreven kan worden als een algemene spin Hamiltoniaan. In deze sectie wensen we echter kort in te gaan op het verband tussen renormalisatie en de spin/fermion-systemen die we zullen beschrijven.

1.3.1 De NRG-methode van Wilson

In ([5]) worden een aantal algemene eigenschappen van veeldeeltjes-(spin)-systemen opgesomd, die we hieronder kort zullen vermelden. Eerst en vooral dient men op te merken dat de Hilbertruimte van een dergelijk veeldeeltjessysteem exponentieel groeit met het aantal spins. Daarnaast lijken interacties altijd lokaal te zijn, ze werken dus enkel tussen een beperkt aantal naburen. Het komt er dus op neer dat alle fysische toestanden in een kleine submanifold van de Hilbertruimte leven. Een voorbeeld in ([5]) maakt duidelijk dat bijna alle punten in de Hilbertruimte van een veeldeeltjessysteem niet fysisch zijn omdat ze gewoonweg niet toegankelijk zijn. Alle fysische toestanden (die kunnen gecreëerd worden) leven dus op een kleine submanifold. De vraag die zich dan onmiddellijk stelt is of er een efficiënte parametrisatie van zo'n manifold bestaat. Bovendien is men in de veeldeeltjesfysica hoofdzakelijk geïnteresseerd in de laag-energetische sector van een lokale Hamiltoniaan, wat extra beperkingen oplegt aan de toegelaten golffuncties. Via kwantuminformatietheorie kan men aantonen dat er inderdaad zo'n efficiënte parametrisatie van een manifold van toestanden bestaat. Het hoofdidee bestaat erin een klasse van variationele golffuncties te vinden die de laag-energetische fysica van een lokaal spinsysteem in zich vervat heeft. Dit gaf oorspronkelijk aanleiding tot de numerieke RG-methode (NRG) van Wilson ([5]). De NRG-methode is een recursieve methode voor het vinden van het laag-energetisch spectrum van een klasse Hamiltonianen die zeer accurate resultaten oplevert als er een duidelijke separatie is tussen de energieën (wat bijvoorbeeld tot uiting komt in een exponentieel verval van de koppelingen). De Hamiltoniaan wordt dan recursief gediagonaliseerd van hoge naar lage energieën. Bij elke iteratie wordt er een tensorproduct gemaakt van de hogere energiemodes met de lagere en vervolgens wordt dit geprojecteerd op een subruimte van lage-energie modes van het gecombineerde systeem. Daarna wordt de Hamiltoniaan herschaald. In deze discussie vinden we een analogie met de aanpak van Wilson voor het Kadanoff-probleem in de vorige sectie.

1.3.2 DMRG

In deze sectie geven we iets meer toelichting bij de density-matrix-renormalization-group methode of DMRG methode (deze discussie ontleen we aan ([4])). Dit zal ons toelaten om

een gefundeerd argument te ontwikkelen dat de truncatie van de Hilbertruimte rechtvaardigt. Deze truncatie zullen we verderop toepassen in de VMPS-methode. Meer informatie over DMRG is ook te vinden in het review artikel van Schollwöck ([11]). Een belangrijk concept hierbij is de zogenaamde verstrengelingsentropie die afkomstig is uit de kwantuminformatietheorie en een maat is voor de kwantumcorrelatie tussen twee systemen. Er blijkt namelijk een verband te bestaan tussen deze entropie en de efficiënte truncatie van de Hilbertruimte (daarover later meer).

Verder wordt er een onderscheid gemaakt tussen infinite-system DMRG en finite-system DMRG. We zullen in de volgende discussie steeds uitgaan van een 1D-ketting met L sites waarbij op elke site een deeltje met een bepaalde spin aanwezig is. Bij definitie stellen we dat de methode convergeert indien

$$|E_{n+1} - E_n| < E_{acc} \quad (1.8)$$

waarbij E_n de energie is in de n -de iteratiestap en E_{acc} een vooraf ingestelde accuraatheid.

Infinite-system DMRG

In infinite-system DMRG introduceert men een linkerblok A en een rechterblok B die in de eerste stap elk één site bevatten. In een volgende stap voegt men twee sites $\bullet\bullet$ toe tussen A en B zodat we een langer systeem bekomen. Daarna worden deze twee sites geabsorbeerd in A en B en voegt men opnieuw twee sites toe. Op die manier groeit het aantal sites binnen een blok. De ketting heeft dan altijd de volgende structuur: $A \bullet\bullet B$. Stel dat elke site een fysische dimensie d heeft (met bijvoorbeeld $d = 2$ voor een spin- $\frac{1}{2}$) dan groeit de dimensie van de Hilbertruimte exponentieel met de blok grootte. Daarom voeren we een truncatie D in en stellen dat de dimensie nooit groter mag worden dan deze D . Stel dat het A -blok reeds de dimensie D heeft bereikt dan zal in een volgende stap de basisdimensie groeien tot dD . We wensen dit nu terug te reduceren tot D basistoestanden. Daartoe beschouwt men in ([4]) de gereduceerde dichtheidsmatrix van $A\bullet$

$$\hat{\rho}_{A\bullet} = \text{Tr}_{\bullet B} |\psi\rangle \langle \psi| \quad (1.9)$$

waabij $|\psi\rangle$ de huidige benadering van de grondtoestand is. De eigentoestanden van $\hat{\rho}_{A\bullet}$ worden dan bepaald door diagonalisatie en men behoudt de D eigentoestanden met de grootste eigenwaarde als nieuwe basisvectoren voor het blok $A\bullet$. Deze vaststelling volgt door op te leggen dat

$$\| |\psi\rangle - |\psi\rangle_{\text{trunc}} \| \quad (1.10)$$

minimaal moet zijn. Hierbij is $|\psi\rangle_{\text{trunc}}$ de projectie van $|\psi\rangle$ op de getrunceerde basis met dimensie D . Het succes van de DMRG steunt op het feit dat zelfs voor vrij kleine D de niet-beschouwde eigentoestanden toch geen grote gewichtsfactor hebben met als gevolg dat de truncatiefout klein blijft.

Finite-system DMRG

In finite-system DMRG start men van de eindtoestand $A \bullet\bullet B$ van de infinite-system DMRG en gaat men bijvoorbeeld blok B laten groeien (door een site in blok B op te slorpen) terwijl blok

A kleiner wordt (met als gevolg dat een site die eerst in A zat nu één van de twee sites tussen blok A en B is). Eens dit gebeurd is, gaat men volledig analoog tewerk als in infinite-system DMRG: berekenen van de grondtoestand van het superblok $A \bullet \bullet B$ en vervolgens de D meest relevante eigentoestanden bepalen voor het nieuwe B blok. Dit gaat verder tot blok A nog maar één site bevat en vervolgens wordt de richting omgedraaid: nu laat men A groeien terwijl B kleiner wordt. Dit sweepen door het systeem gaat verder tot de energie convergeert binnen een vooraf opgelegde drempel.

In beide gevallen wordt dus naar de grondtoestand van een systeem gezocht, maar in het geval van infinite-system DMRG start men daarbij van de thermodynamische limiet ($L \rightarrow \infty$) terwijl in het andere geval een eindige lengte L gekozen wordt. In vele gevallen is het echter aangewezen om eerst een infinite-system DMRG uit te voeren gevolgd door een finite-system DMRG (waarbij als startpunt dan de resultaten van de infinite DMRG gebruikt worden).

1.3.3 Matrix Product States en de truncatiedimensie

In het volgende hoofdstuk zullen we in detail de zogenaamde Matrix Product States (MPSs) bespreken. We zullen aantonen dat een ééndimensionaal veeldeeltjessysteem gekarakteriseerd kan worden door een product van matrices (die na vermenigvuldiging scalair opleveren die we gebruiken als expansiecoëfficiënten van de golf functie). Maar ook hier zal het probleem van de exponentiële groei van de Hilbertruimte optreden. Om dit te vermijden gaat men in de NRG-methode de Hamiltoniaan projecteren op een nieuwe D -dimensionale eigenruimte corresponderend met de meest relevante eigenwaarden ([5]). De methode die we zullen bestuderen in het volgende hoofdstuk steunt ook op het trunceren van de matrixdimensie. Vandaar dat we op een bepaald moment in de afleiding zullen stellen dat de maximale matrixdimensie ($D \times D$) is. In de DMRG methode, die we hiervoor besproken hebben, wordt immers aangetoond dat de eigenwaarden die men moet behouden de meest significante eigenwaarden van een blokdichtheidsmatrix zijn ([4] en [12]). Dit geeft ons een rechtvaardiging voor het trunceren van de matrixdimensie aangezien de methode die wij zullen gebruiken een variationele herformulering van DMRG is, die mathematisch compleet equivalent is, maar waarbij er algoritmisch verschillen zijn ten opzichte van DMRG ([4]).

1.4 Een vooruitblik

In hoofdstuk 2 zullen we naast een grondige bespreking van de MPSs ook het concept van verstrengelingsentropie invoeren. De accuraatheid van het resultaat hangt samen met de truncatiedimensie D en de grootte van de verstrengelingsentropie laat toe om een bovengrens voor D te bepalen (voor een gegeven accuraatheid). We besluiten het hoofdstuk door de convergentie van de methode uit te testen voor een aantal spinsystemen.

In hoofdstuk 3 testen we de geïmplementeerde code uit voor een aantal spinsystemen. We vergelijken de resultaten met analytische resultaten om zo de accuraatheid van het programma te testen. Als toepassing bestuderen we het effect van magnetische velden op de magnetisatie van grote spinroosters. Voor een aantal systemen zullen we ook de verstrengelingsentropie

bestuderen.

In hoofdstuk 4 maken we de overgang naar fermionsystemen. Het algoritme is dan niet meer rechtstreeks toepasbaar omdat de beschouwde spin-operatoren op verschillende sites met elkaar commuteren terwijl fermion-operatoren (of dus creatie- en annihilatie-operatoren) anti-commuteren. Als gevolg daarvan moeten we extra fasefactoren invoeren en we zullen bespreken welke aanpassingen nodig zijn om het algoritme ook in dat geval efficiënt te laten werken. Bovendien moeten we bij de simulatie van fermionsystemen ook een deeltjesaantal opleggen. Als we dit niet expliciet doen, dan kan het algoritme het deeltjesaantal zelf kiezen (om bijvoorbeeld de energie te doen dalen als er repulsieve interacties aanwezig zijn). Het opleggen van een gewenst deeltjesaantal gebeurt via een kwadratische kostfunctie die de energie doet stijgen als het deeltjesaantal afwijkt van de gewenste waarde.

In Hoofdstuk 5 maken we de overstap naar moleculaire systemen. Als testmodel beschouwen we de H_2 molecule waarbij elke site een waterstofatoom met een $1s$ -orbitaal bevat. Als gevolg van de tweedeeltjesinteractie moeten we echter 2 grote aanpassingen doorvoeren. Enerzijds zijn er correlaties mogelijk tussen 4 verschillende sites (terwijl we in de overige hoofdstukken steeds werken met correlaties tussen 2 sites) wat tot gevolg heeft dat we de interactie moeten herschrijven tot een transparante vorm die we kunnen implementeren. Aanderzijds zijn er een groot aantal mogelijke correlaties zodat we een drempel moeten inbouwen om ervoor te zorgen dat de rekentijd niet enorm oploopt. De methode wordt uitgetest door de resultaten voor een H_2 molecule te vergelijken met FCI-berekeningen.

De resultaten en bevindingen worden samengevat in hoofdstuk 6. Tot slot geven we aan dat het doel van deze thesis uit 3 delen bestaat. Een eerste luik bestaat erin vat te krijgen op de theorie van de MPSs en vervolgens een accuraat algoritme te implementeren om spinsystemen te beschrijven. Een tweede luik poogt het bestaande algoritme uit te breiden tot fermionsystemen. In een derde fase stellen we ons tot doel de ontwikkelde methodes te gebruiken om uiteindelijk de energie van een H_2 molecule te berekenen.

Hoofdstuk 2

De VMPS methode

2.1 Inleiding en doel

De theoretische opbouw van de VMPS methode steunt in grote mate op de behandeling in ([4]). We zullen de opbouw van de Matrix Product States overnemen uit dit artikel, maar voor de concrete uitwerking van de energie gaan we op een andere manier te werk. De manier die we hieronder illustreren is meer intuïtief en de structuur van de verschillende energiebijdragen komt duidelijker naar voor. Om het concept van de Matrix Product States in te voeren vertrekken we steeds van een ééndimensionale (1D) spin-ketting met lengte L . Spinroosters worden reeds lang bestudeerd en analytische oplossingen zijn vaak niet voorhanden wegens de grote complexiteit. Het ontwikkelen van een efficiënte methode om dergelijke systemen te beschrijven is dan ook van enorm belang omdat spinsystemen een bijzondere rol spelen in de magnetische eigenschappen van materialen; ook in de verklaring van het gedrag van supergeleiders is spin belangrijk. Hoewel we starten vanuit het standpunt van spinsystemen, zal later blijken dat dit geen beperking vormt om ook fermionische systemen te beschrijven (mits enkele aanpassingen uiteraard). Van zodra we een efficiënt algoritme voor fermionische roosters hebben, kunnen we de overgang maken naar moleculaire ketens (waarbij we ons beperken tot de H_2 molecule; dit vormt tevens het uiteindelijke doel van deze thesis). Daartoe plaatsen we orbitalen op het rooster en pogen daarbij de Coulomb-interacties accuraat te beschrijven. De variationele Ansatz die we in dit hoofdstuk zullen invoeren is in staat om zeer accurate voorspellingen te geven voor spinsystemen (zonder de rekentijd exponentieel te laten oplopen). We zullen dan ook proberen deze efficiëntie verder te trekken naar moleculaire systemen, al zal verderop blijken dat ook hier problemen zullen optreden.

2.2 Matrix Product State of MPS

De theoretische afleidingen in deze sectie zijn in grote mate overgenomen uit ([4]). Meer in het bijzonder zijn de definities en afleidingen uit secties 2.2.1 en 2.2.2 overgenomen uit ([4]). Ook het concept van de linker-en rechterblokken (zoals ingevoerd in sectie 2.3.1) en de diagrammatische voorstelling (sectie 2.3.2) is te vinden in ([4]). De manier waarop de verwachtingswaarde van de Hamiltoniaan wordt uitgerekend via de X/Y -blokstructuur (sectie 2.4) is echter nieuw. In sectie 2.5 maken we gebruik van een variationele procedure die ook in ([4]) beschreven wordt,

maar de expliciete uitdrukkingen zijn wel afhankelijk van de X/Y -blokken.

2.2.1 Overgang naar MPS

Om de gedachten te vestigen gaan we uit van een rooster met L sites waarbij elke site $i = 1 \dots L$ beschreven wordt door een d -dimensionale ruimte $\{\sigma_i\}$. Stel bijvoorbeeld dat elke site beschreven wordt door een spin- $\frac{1}{2}$, dan is $d = 2$ overeenkomstig de 2 spinprojecties $\pm \frac{\hbar}{2}$. De meest algemene vorm van de golffunctie op dit rooster is dan ([4])

$$|\psi\rangle = \sum_{\sigma_1, \dots, \sigma_L} c_{\sigma_1, \dots, \sigma_L} |\sigma_1, \dots, \sigma_L\rangle. \quad (2.1)$$

We vormen de $c_{\sigma_1, \dots, \sigma_L}$ coëfficiënt met d^L componenten nu om tot een $(d \times d^{L-1})$ matrix $\Psi_{\sigma_1, (\sigma_2, \dots, \sigma_L)}$ volgens het voorschrift

$$\Psi_{\sigma_1, (\sigma_2, \dots, \sigma_L)} = c_{\sigma_1, \dots, \sigma_L}. \quad (2.2)$$

Op deze matrix zullen we een QR-decompositie toepassen. De lineaire algebra leert ons dat een willekeurige $(m \times n)$ matrix M ontbonden kan worden in een $(m \times m)$ unitaire matrix Q ($Q^\dagger Q = I = QQ^\dagger$ met I de eenheidsmatrix) en een $(m \times n)$ bovendriehoeksmatrix R . Passen we dit nu toe op $\Psi_{\sigma_1, (\sigma_2, \dots, \sigma_L)}$, dan vinden we

$$\Psi_{\sigma_1, (\sigma_2, \dots, \sigma_L)} = \sum_{a_1=1}^{r_1} Q_{\sigma_1, a_1} R_{a_1, (\sigma_2, \dots, \sigma_L)}. \quad (2.3)$$

Hierbij is r_1 het aantal niet-nul rijen in de bovendriehoeksmatrix R . We zullen Q_{σ_1, a_1} nu omvormen tot een rij van d ($1 \times r_1$) vectoren $A_{a_1}^{\sigma_1}$ en herschrijven $R_{a_1, (\sigma_2, \dots, \sigma_L)}$ op dezelfde manier als we voor $\Psi_{\sigma_1, (\sigma_2, \dots, \sigma_L)}$ gedaan hebben. We voeren dus opnieuw een transformatie

$$R_{a_1, (\sigma_2, \dots, \sigma_L)} \rightarrow \Psi_{(a_1 \sigma_2), (\sigma_3, \dots, \sigma_L)} \quad (2.4)$$

uit en passen op de resulterende $(r_1 d \times d^{L-2})$ matrix een QR-decompositie toe. We vinden dan

$$c_{\sigma_1, \dots, \sigma_L} = \sum_{a_1=1}^{r_1} \sum_{a_2=1}^{r_2} A_{a_1}^{\sigma_1} Q_{(a_1 \sigma_2), a_2} R_{a_2, (\sigma_3, \dots, \sigma_L)} \quad (2.5)$$

waarbij dezelfde notaties gehanteerd worden als bij de eerste transformatie. Hierbij wordt $Q_{(a_1 \sigma_2), a_2}$ omgezet naar een set van d ($r_1 \times r_2$) matrices. Zetten we dit proces door tot we het einde van de ketting bereiken, dan kan men makkelijk aantonen dat de $c_{\sigma_1, \dots, \sigma_L}$ coëfficiënt uiteindelijk equivalent is met ([4])

$$\begin{aligned} c_{\sigma_1, \dots, \sigma_L} &= \sum_{a_1, \dots, a_{L-1}} A_{1, a_1}^{\sigma_1} A_{a_1, a_2}^{\sigma_2} \dots A_{a_{L-2}, a_{L-1}}^{\sigma_{L-1}} A_{a_{L-1}, 1}^{\sigma_L} \\ &= A^{\sigma_1} A^{\sigma_2} \dots A^{\sigma_{L-1}} A^{\sigma_L}. \end{aligned}$$

De totale golffunctie kan dus ook geschreven worden als

$$|\psi\rangle = \sum_{\sigma_1, \dots, \sigma_L} A^{\sigma_1} A^{\sigma_2} \dots A^{\sigma_{L-1}} A^{\sigma_L} |\sigma_1, \dots, \sigma_L\rangle. \quad (2.6)$$

Op dit moment zijn we in staat om het concept MPS of Matrix Product State in te voeren. In het vorige hebben we immers aangetoond dat de totale golffunctie van ons 1D veeldeeltjessysteem (waarvan de Hilbertruimte dimensie d^L heeft) geschreven kan worden in termen van *producten* van *matrices* A^{σ_i} die geassocieerd zijn aan 1 specifieke site i . Vandaar dat we zeggen dat de golffunctie als een MPS kan voorgesteld worden. Dit zal achteraf heel voordelig blijken qua implementatie: de golffunctie kan geoptimaliseerd worden door de matrices A^{σ_i} site per site aan te passen. Merk op dat een MPS dus een Ansatz is (net als een Slaterdeterminant) die we zo optimaal mogelijk moeten maken om een accurate beschrijving van het systeem te bekomen. Enkel bij een voldoende grote truncatiedimensie zal een dergelijke Ansatz ook het exacte resultaat opleveren (zie hieronder). De MPS kan dus een exacte golffunctie voorstellen, maar zal in het algemeen een benadering zijn (omdat we de truncatiedimensie uiteraard niet onbeperkt kunnen laten toenemen).

De truncatie dimensie D

In de vorige discussie hebben we zeer algemeen gewerkt met r_i . De maximale rang is steeds $r_i = d^i$ voor alle $i \leq \lfloor \frac{L}{2} \rfloor$ (de floor-functie dient enkel om problemen te vermijden bij oneven L). Bij de reductie tot een MPS-vorm wordt de dimensie van de matrix geassocieerd aan $\Psi_{(a_{i-1}\sigma_i),(\sigma_{i+1},\dots,\sigma_L)}$ gegeven door $(d^i \times d^{L-i})$. De matrices nemen dus steeds toe in dimensie tot op het punt waar $i = L - i$ of dus $i = \lfloor \frac{L}{2} \rfloor$. De sequentie van matrices die we op die manier genereren is dus

$$(1 \times d)(d \times d^2) \dots (d^{\lfloor \frac{L}{2} \rfloor - 1}, d^{\lfloor \frac{L}{2} \rfloor})(d^{\lfloor \frac{L}{2} \rfloor}, d^{\lfloor \frac{L}{2} \rfloor - 1}) \dots (d^2 \times d)(d \times 1). \quad (2.7)$$

Deze schrijfwijze toont meteen het grootste knelpunt van de methode. Als we de matrixdimensies laten toenemen tot $\lfloor \frac{L}{2} \rfloor$ dan is dit equivalent met de exponentiële groei van de Hilbertruimte. Het moge duidelijk zijn dat dit computationeel niet haalbaar is (behalve voor voldoende kleine systemen, dus kleine L). Daarom zullen we zelf de matrixdimensie trunceren op een bepaalde waarde D . Het komt er op neer dat voor elke n waarvoor geldt dat $d^n > D$ we d^n vervangen door D of dus

$$\forall n : d^n > D \Rightarrow d^n \rightarrow D \quad (2.8)$$

waardoor de matrixsequentie uiteindelijk

$$(1 \times d)(d \times d^2) \dots (d^{n-1}, D)(D, D)(D, D) \dots (D, D)(D, D)(D, d^{n-1}) \dots (d^2 \times d)(d \times 1) \quad (2.9)$$

wordt. Op die manier beperken we de volledige Hilbertruimte tot een kleinere ruimte waardoor het geheel computationeel handelbaar wordt ([4]). Uit deze behandeling is het ook duidelijk dat de methode exacte resultaten oplevert als $D = d^{\lfloor \frac{L}{2} \rfloor}$. Verderop bespreken we de convergentie van de VMPS-methode en zullen we nagaan welk effect D heeft op de energie.

Nu we weten wat een MPS is en hoe we de link moeten leggen met een golffunctie is het belangrijk om een aantal elementaire eigenschappen van deze matrices onder de loep te nemen: de zogenaamde canonische vormen van een MPS ([4]).

2.2.2 Canonische vormen van een MPS

We vertrekken van de algemene vorm van een MPS

$$|\psi\rangle = \sum_{\sigma_1, \dots, \sigma_L} \sum_{a_1, \dots, a_{L-1}} M_{1,a_1}^{\sigma_1} M_{a_1,a_2}^{\sigma_2} \dots M_{a_{L-2},a_{L-1}}^{\sigma_{L-1}} M_{a_{L-1},1}^{\sigma_L} |\sigma_1, \dots, \sigma_L\rangle \quad (2.10)$$

en vormen de $M_{1,a_1}^{\sigma_1}$ om tot één matrix $M_{(\sigma_1,1),a_1}$ waarop we een QR-decompositie toepassen:

$$|\psi\rangle = \sum_{\sigma_1, \dots, \sigma_L} \sum_{a_1, \dots, a_{L-1}} \left(\sum_{r_1} A_{(\sigma_1,1),r_1} R_{r_1,a_1} \right) M_{a_1,a_2}^{\sigma_2} \dots M_{a_{L-2},a_{L-1}}^{\sigma_{L-1}} M_{a_{L-1},1}^{\sigma_L} |\sigma_1, \dots, \sigma_L\rangle. \quad (2.11)$$

De matrix R_{r_1,a_1} vermenigvuldigen we vervolgens met elke matrix $M_{a_1,a_2}^{\sigma_2}$ ($\sigma_2 = 1 \dots d$) alvorens op de nieuwe $M_{a_1,a_2}^{\sigma_2}$ ook een QR-decompositie toe te passen. Deze procedure herhalen we voor alle $M_{a_{i-1},a_i}^{\sigma_i}$ met $i = 1 \dots L$ ($a_0 = a_L = 1$) en de matrices $A_{(\sigma_i,a_{i-1}),r_i}$ worden daarbij steeds terug omgevormd naar een set van d matrices $A_{a_{i-1},r_i}^{\sigma_i}$. Deze procedure levert uiteindelijk de volgende vorm voor de golffunctie:

$$|\psi\rangle = \sum_{\sigma_1, \dots, \sigma_L} A^{\sigma_1} A^{\sigma_2} \dots A^{\sigma_{L-1}} A^{\sigma_L} |\sigma_1, \dots, \sigma_L\rangle. \quad (2.12)$$

Wegens de successieve QR-decomposities geldt er voor elke $A_{(\sigma_i,a_{i-1}),r_i} \rightarrow \tilde{A}$ matrix dat

$$\tilde{A}^\dagger \tilde{A} = I_{(m,m)} \quad (2.13)$$

met $I_{(m,m)} = I$ de $(m \times m)$ eenheidsmatrix waarbij m het aantal kolommen van \tilde{A} is (dit is een gevolg van de QR-decompositie). Hieruit volgt

$$\begin{aligned} \delta_{r_i, r'_i} &= \sum_{(\sigma_i, a_{i-1})} \tilde{A}_{r_i, (\sigma_i, a_{i-1})}^\dagger \tilde{A}_{(\sigma_i, a_{i-1}), r'_i} \\ &= \sum_{(\sigma_i, a_{i-1})} A_{r_i, a_{i-1}}^{\sigma_i \dagger} A_{a_{i-1}, r'_i}^{\sigma_i} \\ &= \sum_{\sigma_i} \left(A^{\sigma_i \dagger} A^{\sigma_i} \right)_{r_i, r'_i}. \end{aligned}$$

We vinden dat de A -matrices voldoen aan

$$\sum_{\sigma_i} A^{\sigma_i \dagger} A^{\sigma_i} = I. \quad (2.14)$$

Matrices die aan de voorwaarde (2.14) voldoen zijn links-genormaliseerde matrices ([4]). Een golffunctie die enkel opgebouwd is uit links-genormaliseerde matrices wordt *links-canonisch* genoemd. Als we starten van de rechterkant (bij de matrix $M_{a_{L-1},1}^{\sigma_L}$) en LQ-decomposities toepassen op de verschillende matrices dan kunnen we de golffunctie schrijven als

$$|\psi\rangle = \sum_{\sigma_1, \dots, \sigma_L} \sum_{a_1, \dots, a_{L-1}} M_{1,a_1}^{\sigma_1} M_{a_1,a_2}^{\sigma_2} \dots M_{a_{L-2},a_{L-1}}^{\sigma_{L-1}} \left(\sum_{r_{L-1}} L_{a_{L-1}, r_{L-1}} B_{r_{L-1}, (\sigma_L, 1)} \right) |\sigma_1, \dots, \sigma_L\rangle. \quad (2.15)$$

Als we dit voor alle sets matrices herhalen dan kunnen we op volledig analoge manier aantonen dat de golffunctie geschreven kan worden als

$$|\psi\rangle = \sum_{\sigma_1, \dots, \sigma_L} B^{\sigma_1} B^{\sigma_2} \dots B^{\sigma_{L-1}} B^{\sigma_L} |\sigma_1, \dots, \sigma_L\rangle \quad (2.16)$$

waarbij de B -matrices voldoen aan

$$\sum_{\sigma_i} B^{\sigma_i} B^{\sigma_i \dagger} = I. \quad (2.17)$$

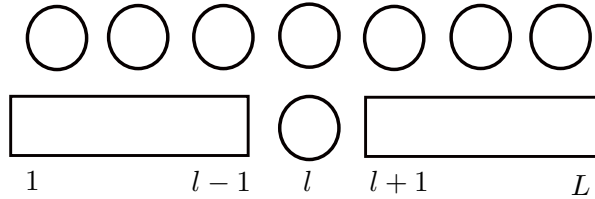
Dergelijke matrices worden rechts-genormaliseerde matrices genoemd en een golffunctie die enkel opgebouwd is uit dergelijke matrices wordt *rechts-canonisch* genoemd ([4]). Hiermee hebben we 2 canonische vormen van de golffunctie ingevoerd. Daarnaast kunnen we ook een gemengde vorm¹ bekijken waarbij alle matrices links van site l links-genormaliseerd zijn en alle matrices rechts van site l rechts-genormaliseerd. De normalisatie van de set matrices op site l (M^{σ_l}) doet er niet toe, aangezien later zal blijken dat we in elke iteratiestap van het algoritme precies één set matrices (geassocieerd aan site l) zullen optimaliseren. In een dergelijke iteratiestap doet het er dan niet toe hoe de matrices op site l genormeerd zijn. Een dergelijke gemengde canonische golffunctie ([4]) heeft de vorm

$$|\psi\rangle = \sum_{\sigma_1, \dots, \sigma_L} A^{\sigma_1} \dots A^{\sigma_{l-1}} M^{\sigma_l} B^{\sigma_{l+1}} \dots B^{\sigma_L} |\sigma_1, \dots, \sigma_L\rangle \quad (2.18)$$

waarbij de A^{σ_n} matrices links-genormaliseerd en de B^{σ_n} matrices rechts-genormaliseerd zijn terwijl de M^{σ_l} matrices geen specifieke normalisatie hoeven te hebben. Dergelijke canonische vormen zijn belangrijk voor het vervolg van de theoretische behandeling.

2.3 MPS-vorm van verwachtingswaarden

We zullen in wat volgt ééndimensionale kettingen met lengte L bestuderen. Deze kunnen interagerende spinsystemen voorstellen, maar verder zal blijken dat dergelijke kettingen ook gekoppeld kunnen worden aan orbitalen om bijvoorbeeld waterstofkettingen te beschrijven. Een dergelijk systeem wordt schematisch voorgesteld zoals in figuur (2.1). We wensen in eerste



Figuur 2.1: Schematische voorstelling van een ketting met lengte L .

instantie de grondtoestandsenergie van zo'n systeem te berekenen. Aangezien het hier over een ingewikkeld veeldeeltjesprobleem gaat (dat sterk gecorreleerd kan zijn), moeten we onze toevlucht nemen tot een numeriek schema. Daartoe zullen we de ketting verdelen in een site l die we optimaliseren (deze site schuift elke iteratie op) en een linker- en rechterblok zoals ook te zien is op de figuur. De blokken bevatten verwachtingswaarden van de Hamiltoniaan, uitgerekend met de MPS Ansatz. Op zich bevat de blokstructuur dus alle termen die bijdragen tot de energie, zodat we niets aan het probleem veranderd hebben door het invoeren van dergelijke blokken. Het zal echter blijken dat naarmate de blokken groeien of kleiner worden (als de

¹Er bestaan veel meer vormen dan links-canonische, rechts-canonisch of gemengd-canonische, maar we zullen verderop enkel gebruik maken van de 3 vormen die in deze sectie opgelijst zijn.

site l zich verplaatst om op die manier alle L sites te optimaliseren) we deze iteratief kunnen construeren uit de voorgaande blokken. Computationeel is dit dan ook zeer voordelig en in de hierop volgende discussie zullen we de verwachtingswaarde van de Hamiltoniaan schrijven in functie van de verschillende blokken. We vertrekken hierbij steeds van een Hamiltoniaan van de volgende vorm:

$$\hat{H} = \sum_{l=1}^L \hat{O}_l + \sum_{n=1}^{\delta} \sum_{(l,m;l < m)} \hat{P}_l^n \otimes \hat{P}_m^n. \quad (2.19)$$

Hierbij is \hat{O}_l een lokale operator op site l , wat betekent dat deze operator enkel werkt op een toestand op site l : $\hat{O}_l |\sigma_m\rangle = \delta_{ml} \hat{O}_l |\sigma_m\rangle$. De operator $\hat{P}_l^n \otimes \hat{P}_m^n$ is een interactie-operator tussen site l en site m (opnieuw zijn de lokale operatoren \hat{P}_l^n en \hat{P}_m^n operatoren die enkel op site l respectievelijk m werken). De som over n loopt over alle mogelijke δ interactietermen tussen l en m . Hierbij dienen we er steeds op te letten dat $l < m$ om te vermijden dat we 2 keer dezelfde interacties in rekening brengen. We merken op dat we vertrekken van spinsystemen, maar dat een Hamiltoniaan van de vorm (2.19) ook gebruikt kan worden voor de beschrijving van fermionische systemen. Voor moleculaire systemen zullen we een iets algemenere vorm nodig hebben, maar nog steeds zullen sommige termen in de moleculaire Hamiltoniaan van de vorm (2.19) zijn.

Om de gedachten te vestigen, geven we een voorbeeld (van een spinsysteem) om te verduidelijken wat de verschillende termen concreet inhouden. Daartoe beschouwen we de isotrope Heisenberg Hamiltoniaan ([13]), die enkel de dichtste-nabuur interacties tussen spins in rekening brengt, van een spinketen in een magnetisch veld β (deze Hamiltoniaan zullen we verder uitvoerig bespreken).

$$\hat{H} = \frac{J}{2} \sum_{i=1}^{L-1} \left(\hat{S}_i^+ \hat{S}_{i+1}^- + \hat{S}_i^- \hat{S}_{i+1}^+ + 2\hat{S}_i^z \hat{S}_{i+1}^z \right) - \beta \sum_{i=1}^L \hat{S}_i^z \quad (2.20)$$

Identificatie leert dat de lokale operatoren \hat{O}_i hier de matrices $-\beta \hat{S}_i^z$ zijn en dat er $\delta = 3$ interactietermen zijn voor elke $(i, i+1)$ -combinatie. Voor sites i is het duidelijk dat $\hat{P}_i^n = \hat{S}_i^n$ met $n = \{+, -, z\}$ en voor de corresponderende sites $i+1$ geldt $\hat{P}_{i+1}^n = \alpha \hat{S}_{i+1}^n$ met $n = \{-, +, z\}$ (bemerkt dat α gelijk is aan $J/2$ of J en let ook op de volgorde van de $+$ en $-$).

2.3.1 Bepalen van de verwachtingswaarde $\langle \psi | \hat{H} | \psi \rangle$

De stationaire Schrödingervergelijking van een willekeurig fysisch systeem met Hamiltoniaan \hat{H} wordt gegeven door

$$\hat{H} |\psi\rangle = E |\psi\rangle \quad (2.21)$$

waarbij E de energie voorstelt en $|\psi\rangle$ de golfunctie die het systeem beschrijft. We stellen ons tot doel de grondtoestandsenergie E_0 van een 1D veeldeeltjessysteem, beschreven door (2.19), accuraat én snel te bepalen. We zullen daarbij vertrekken van een bestaand algoritme en trachten dit zo efficiënt mogelijk te implementeren. Daartoe dienen we de volgende verwachtingswaarde te bepalen

$$E_0 = \frac{\langle \psi | \hat{H} | \psi \rangle}{\langle \psi | \psi \rangle} \quad (2.22)$$

waarbij $|\psi\rangle$ de corresponderende grondtoestandsgolffunctie voorstelt. Zoals hiervoor reeds vermeld zullen we de Hamiltoniaan van ons systeem steeds schrijven in functie van de linker (L)- en rechterblokken (R). Vertrekkende van de vorm (2.19) is het duidelijk dat deze Hamiltoniaan equivalent is met

$$\hat{H} = \sum_{i=1}^{l-1} \hat{O}_i + \hat{O}_l + \sum_{i=l+1}^L \hat{O}_i + \sum_{n=1}^{\delta} \left(\sum_{i,j \in L} \hat{P}_i^n \otimes \hat{P}_j^n + \sum_{i,j \in R} \hat{P}_i^n \otimes \hat{P}_j^n + \sum_{i \in L, j \in R} \hat{P}_i^n \otimes \hat{P}_j^n + \sum_{i \in L} \hat{P}_i^n \otimes \hat{P}_l^n + \sum_{j \in R} \hat{P}_l^n \otimes \hat{P}_j^n \right). \quad (2.23)$$

Hierbij werd de bijdrage van site l steeds afgezonderd en werden ook alle mogelijke interacties in rekening gebracht. We merken nogmaals op dat de site l volkomen arbitrair is en in het uiteindelijke algoritme zal variëren over de ganse ketting. We zullen nu $\langle \psi | \hat{H} | \psi \rangle$ zodanig schrijven dat alle bijdragen van site l duidelijk naar voor komen. Daartoe is het nodig elke term uit (2.23) apart te beschouwen en steeds de verwachtingswaarde in $|\psi\rangle$ te herschrijven naar verwachtingswaarden in de $\{|\sigma_l\rangle, l = 1 \dots L\}$ basis omdat elke lokale operator (die in ons geval steeds een spin-operator zal zijn²) een gekende matrixvoorstelling heeft in deze spin-basis. Als meest algemene gedaante van de golffunctie van het systeem gebruiken we de gemengde canonische golffunctie (2.18) waarbij de matrices de onbekenden zijn die moeten bepaald worden:

$$|\psi\rangle = \sum_{\sigma_1, \dots, \sigma_L} A^{\sigma_1} \dots A^{\sigma_{l-1}} M^{\sigma_l} B^{\sigma_{l+1}} \dots B^{\sigma_L} |\sigma_1, \dots, \sigma_L\rangle \quad (2.24)$$

Voor de eenvoud zullen we ook nog de volgende notatie hanteren

$$|\sigma_1, \dots, \sigma_L\rangle \equiv |\sigma\rangle. \quad (2.25)$$

De verschillende bijdragen tot de energie worden hieronder opgelijst.

- **De term** $\langle \psi | \sum_{i=1}^{l-1} \hat{O}_i | \psi \rangle$

Het is duidelijk dat \hat{O}_i enkel inwerkt op de lokale basis op site i ($\{|\sigma_i\rangle\}$) en de andere termen $\{|\sigma_j\rangle, j \neq i\}$ ongemoeid laat. Daarom geldt

$$\langle \sigma | \sum_{i=1}^{l-1} \hat{O}_i | \sigma' \rangle = \sum_{i=1}^{l-1} \langle \sigma_i | \hat{O}_i | \sigma'_i \rangle. \quad (2.26)$$

Uitschrijven van de verwachtingswaarde $\langle \psi | \sum_{i=1}^{l-1} \hat{O}_i | \psi \rangle$ in de lokale basis levert

$$\sum_{i=1}^{l-1} \sum_{\sigma \sigma'} \sum_{\{C\}} (A^{\sigma_1} \dots A^{\sigma_{i-1}})_{a_{i-1}, 1}^\dagger A_{a_i, a_{i-1}}^{\sigma_i \dagger} (A^{\sigma_{i+1}} \dots A^{\sigma_{l-1}})_{a_{l-1}, a_i}^\dagger \langle \sigma_i | \hat{O}_i | \sigma'_i \rangle (A^{\sigma_1} \dots A^{\sigma_{i-1}})_{1, a'_{i-1}} A_{a'_{i-1}, a'_i}^{\sigma'_i} (A^{\sigma_{i+1}} \dots A^{\sigma_{l-1}})_{a'_i, a'_{l-1}} M_{a_l, a_{l-1}}^{\sigma_l \dagger} M_{a'_{l-1}, a'_l}^{\sigma_l} \delta_{a_i, a'_i} \quad (2.27)$$

²Zoals vermeld in de inleiding zullen we in eerste instantie Hamiltonianen van spinsystemen beschouwen, die uitgedrukt kunnen worden met behulp van spin-operatoren. We zullen echter verderop zien dat een veel algemenere klasse van Hamiltonianen dezelfde functionele vorm heeft als de beschouwde spin Hamiltoniaan. Vandaar dat deze onderstelling geen afbreuk doet aan de algemeenheid van de methode.

waarbij we gebruik gemaakt hebben van het feit dat de B -matrices rechts-genormaliseerd zijn (zie (2.17)) zodat de som over σ zich beperkt tot de σ_k waarvoor $k = 1 \dots l$. Daarnaast is $\{C\} \equiv \{a_{l-1}, a'_{l-1}, a_l, a'_l, a_{i-1}, a'_{i-1}, a_i, a'_i\}$. We kunnen bovenstaande uitdrukking nog verder vereenvoudigen door in te zien dat

$$\sum_{\sigma_1 \dots \sigma_{i-1}} \left(A^{\sigma_{i-1} \dagger} \dots A^{\sigma_1 \dagger} A^{\sigma_1} \dots A^{\sigma_{i-1}} \right)_{a_{i-1}, a'_{i-1}} = \delta_{a_{i-1}, a'_{i-1}} \quad (2.28)$$

omdat de A -matrices links-genormaliseerd zijn (zie (2.14)). We vinden uiteindelijk

$$\sum_{i=1}^{l-1} \sum_{\sigma'_i} \sum_{\sigma_j, i \leq j \leq l} \sum_{\{C\}} A_{a_i, a_{i-1}}^{\sigma_i \dagger} \langle \sigma_i | \hat{O}_i | \sigma'_i \rangle A_{a_{i-1}, a'_i}^{\sigma'_i} (A^{\sigma_{i+1}} \dots A^{\sigma_{l-1}})_{a_{l-1}, a_i}^\dagger (A^{\sigma_{i+1}} \dots A^{\sigma_{l-1}})_{a'_i, a'_{l-1}} M_{a_l, a_{l-1}}^{\sigma_i \dagger} M_{a'_{l-1}, a_l}^{\sigma_i} \quad (2.29)$$

waarbij $\{C\} \equiv \{a_{l-1}, a'_{l-1}, a_l, a_{i-1}, a_i, a'_i\}$.

Op een volledig analoge manier kan men ook de andere termen uitrekenen in de lokale basis (steeds gebruik makend van de genormaliseerde A - en B -matrices en het feit dat lokale operatoren geassocieerd aan site i ook enkel inwerken op site i). Zonder de tussenberekeringen te vermelden, vatten we hieronder alle resultaten samen.

- De term $\langle \psi | \hat{O}_l | \psi \rangle$

$$\sum_{\sigma_l, \sigma'_l} \sum_{a_{l-1}, a_l} \langle \sigma_l | \hat{O}_l | \sigma'_l \rangle M_{a_l, a_{l-1}}^{\sigma_l \dagger} M_{a_{l-1}, a_l}^{\sigma'_l} \quad (2.30)$$

- De term $\langle \psi | \sum_{i=l+1}^L \hat{O}_i | \psi \rangle$

$$\sum_{i=l+1}^L \sum_{\sigma'_i} \sum_{\{\sigma_j, l \leq j \leq i\}} \sum_{\{C\}} B_{a_i, a_{i-1}}^{\sigma_i \dagger} \langle \sigma_i | \hat{O}_i | \sigma'_i \rangle B_{a'_{i-1}, a_i}^{\sigma'_i} (B^{\sigma_{l+1}} \dots B^{\sigma_{i-1}})_{a_{i-1}, a_i}^\dagger (B^{\sigma_{l+1}} \dots B^{\sigma_{i-1}})_{a'_i, a'_{l-1}} M_{a_l, a_{l-1}}^{\sigma_i \dagger} M_{a_{l-1}, a'_i}^{\sigma_i} \quad (2.31)$$

waarbij $\{C\} \equiv \{a_{l-1}, a_l, a'_l, a_{i-1}, a_i, a'_{i-1}\}$.

- De term $\langle \psi | \sum_{n=1}^{\delta} \sum_{i,j \in L} \hat{P}_i^n \otimes \hat{P}_j^n | \psi \rangle$

$$\sum_{n=1}^{\delta} \sum_{i < j; (i,j) \in L} \sum_{\sigma'_i \sigma'_j} \sum_{\{\sigma_k, i \leq k \leq l\}} \sum_{\{C\}} A_{a_i, a_{i-1}}^{\sigma_i \dagger} \langle \sigma_i | \hat{P}_i^n | \sigma'_i \rangle A_{a_{i-1}, a'_i}^{\sigma'_i} A_{a_j, a_{j-1}}^{\sigma_j \dagger} \langle \sigma_j | \hat{P}_j^n | \sigma'_j \rangle A_{a'_{j-1}, a'_j}^{\sigma'_j} (A^{\sigma_{i+1}} \dots A^{\sigma_{j-1}})_{a_{j-1}, a_i}^\dagger (A^{\sigma_{i+1}} \dots A^{\sigma_{j-1}})_{a'_i, a'_{j-1}} (A^{\sigma_{j+1}} \dots A^{\sigma_{l-1}})_{a_{l-1}, a_j}^\dagger (A^{\sigma_{j+1}} \dots A^{\sigma_{l-1}})_{a'_j, a'_{l-1}} M_{a_l, a_{l-1}}^{\sigma_i \dagger} M_{a'_{l-1}, a_l}^{\sigma_i} \quad (2.32)$$

waarbij $\{C\} \equiv \{a_{l-1}, a'_{l-1}, a_l, a_{i-1}, a_i, a'_i, a_{j-1}, a'_{j-1}, a_j, a'_j\}$.

- De term $\langle \psi | \sum_{n=1}^{\delta} \sum_{i,j \in R} \hat{P}_i^n \otimes \hat{P}_j^n | \psi \rangle$

$$\begin{aligned}
& \sum_{n=1}^{\delta} \sum_{i < j; (i,j) \in R} \sum_{\sigma'_i \sigma'_j} \sum_{\{\sigma_k: l \leq k \leq j\}} \sum_{\{C\}} B_{a_i, a_{i-1}}^{\sigma_i \dagger} \langle \sigma_i | \hat{P}_i^n | \sigma'_i \rangle B_{a'_{i-1}, a'_i}^{\sigma'_i} B_{a_j, a_{j-1}}^{\sigma_j \dagger} \langle \sigma_j | \hat{P}_j^n | \sigma'_j \rangle B_{a'_{j-1}, a_j}^{\sigma'_j} \\
& \quad (B^{\sigma_{l+1}} \dots B^{\sigma_{i-1}})_{a_{i-1}, a_l}^{\dagger} (B^{\sigma_{l+1}} \dots B^{\sigma_{i-1}})_{a'_l, a'_{i-1}} (B^{\sigma_{i+1}} \dots B^{\sigma_{j-1}})_{a_{j-1}, a_i}^{\dagger} \\
& \quad (B^{\sigma_{i+1}} \dots B^{\sigma_{j-1}})_{a'_i, a'_{j-1}} M_{a_l, a_{l-1}}^{\sigma_l \dagger} M_{a_{l-1}, a'_l}^{\sigma_l}. \quad (2.33)
\end{aligned}$$

waarbij $\{C\} \equiv \{a_{l-1}, a_l, a'_l, a_{i-1}, a'_{i-1}, a_i, a'_i, a_{j-1}, a'_{j-1}, a_j\}$.

- De term $\langle \psi | \sum_{n=1}^{\delta} \sum_{i \in L} \hat{P}_i^n \otimes \hat{P}_l^n | \psi \rangle$

$$\begin{aligned}
& \sum_{n=1}^{\delta} \sum_{i \in L} \sum_{\sigma'_i \sigma'_l} \sum_{\{\sigma_k: i \leq k \leq l\}} \sum_{\{C\}} A_{a_i, a_{i-1}}^{\sigma_i \dagger} \langle \sigma_i | \hat{P}_i^n | \sigma'_i \rangle A_{a_{i-1}, a'_i}^{\sigma'_i} \langle \sigma_l | \hat{P}_l^n | \sigma'_l \rangle \\
& \quad (A^{\sigma_{i+1}} \dots A^{\sigma_{l-1}})_{a_{l-1}, a_i}^{\dagger} (A^{\sigma_{i+1}} \dots A^{\sigma_{l-1}})_{a'_i, a'_{l-1}} M_{a_l, a_{l-1}}^{\sigma_l \dagger} M_{a'_{l-1}, a'_l}^{\sigma_l}. \quad (2.34)
\end{aligned}$$

waarbij $\{C\} \equiv \{a_{l-1}, a'_{l-1}, a_l, a_{i-1}, a_i, a'_i\}$.

- De term $\langle \psi | \sum_{n=1}^{\delta} \sum_{j \in R} \hat{P}_l^n \otimes \hat{P}_j^n | \psi \rangle$

$$\begin{aligned}
& \sum_{n=1}^{\delta} \sum_{j \in R} \sum_{\sigma'_j \sigma'_l} \sum_{\{\sigma_k: l \leq k \leq j\}} \sum_{\{C\}} B_{a_j, a_{j-1}}^{\sigma_j \dagger} \langle \sigma_j | \hat{P}_j^n | \sigma'_j \rangle B_{a'_{j-1}, a_j}^{\sigma'_j} \langle \sigma_l | \hat{P}_l^n | \sigma'_l \rangle \\
& \quad (B^{\sigma_{l+1}} \dots B^{\sigma_{j-1}})_{a_{j-1}, a_l}^{\dagger} (B^{\sigma_{l+1}} \dots A^{\sigma_{j-1}})_{a'_l, a'_{j-1}} M_{a_l, a_{l-1}}^{\sigma_l \dagger} M_{a'_{l-1}, a'_l}^{\sigma_l}. \quad (2.35)
\end{aligned}$$

waarbij $\{C\} \equiv \{a_{l-1}, a_l, a'_l, a_{j-1}, a'_{j-1}, a_j\}$.

- De term $\langle \psi | \sum_{n=1}^{\delta} \sum_{i \in L, j \in R} \hat{P}_i^n \otimes \hat{P}_j^n | \psi \rangle$

$$\begin{aligned}
& \sum_{n=1}^{\delta} \sum_{i \in L, j \in R} \sum_{\sigma'_i \sigma'_j} \sum_{\sigma_k: i \leq k \leq j} \sum_{\{C\}} A_{a_i, a_{i-1}}^{\sigma_i \dagger} \langle \sigma_i | \hat{P}_i^n | \sigma'_i \rangle A_{a_{i-1}, a'_i}^{\sigma'_i} B_{a_j, a_{j-1}}^{\sigma_j \dagger} \langle \sigma_j | \hat{P}_j^n | \sigma'_j \rangle B_{a'_{j-1}, a_j}^{\sigma'_j} \\
& \quad (A^{\sigma_{i+1}} \dots A^{\sigma_{l-1}})_{a_{l-1}, a_i}^{\dagger} (A^{\sigma_{i+1}} \dots A^{\sigma_{l-1}})_{a'_i, a'_{l-1}} (B^{\sigma_{l+1}} \dots B^{\sigma_{j-1}})_{a_{j-1}, a_l}^{\dagger} \\
& \quad (B^{\sigma_{l+1}} \dots B^{\sigma_{j-1}})_{a'_l, a'_{j-1}} M_{a_l, a_{l-1}}^{\sigma_l \dagger} M_{a'_{l-1}, a'_l}^{\sigma_l}. \quad (2.36)
\end{aligned}$$

waarbij $\{C\} \equiv \{a_{l-1}, a'_{l-1}, a_l, a'_l, a_{i-1}, a_i, a'_i, a_{j-1}, a'_{j-1}, a_j\}$.

Vermits deze berekeningen zeer omvangrijk zijn kan men gemakkelijk fouten maken in de indices. Analoog aan Feynmandiagrammen bestaan er voor deze berekeningen dan ook grafische weergaves die zeer helder weergeven over welke indices wordt gesommeerd en welke interacties er beschouwd worden. We zullen in de volgende sectie dit grafisch formalisme uitwerken (waarbij we opnieuw steunen op ([4])).

2.3.2 Diagrammatische voorstelling

We hebben aangetoond dat een golffunctie geschreven kan worden in termen van dL matrices M^{σ_l} . We kunnen M^{σ_l} ook interpreteren als een tensor gekarakteriseerd door 3 indices die d matrices ($\sigma_l = 1 \dots d$) voorstelt, elk met een dimensie ($i \times j$). Er zijn dus 3 *connectiepunten* die vertrekken vanuit de MPS-vertex en we kunnen dit grafisch voorstellen als in figuur (2.2). De hermitisch toegevoegde wordt op exact dezelfde manier voorgesteld met als enige verschil dat de lijn die de fysische index weergeeft nu naar boven staat (zoals ook te zien is in figuur (2.2)). We voeren nu een set regels in die toelaten om grafisch de verwachtingswaarde van de Hamiltoniaan voor te stellen.

Regel 1

De grafische voorstelling van een set van d matrices ($\sigma_l = 1 \dots d$) met dimensie ($i \times j$) is zoals in figuur (2.2). Links van de MPS-vertex (voorgesteld door een cirkel) bevindt zich de lijn-index i die de rijen voorstelt van de matrices en rechts van de MPS-vertex bevindt zich de lijn-index j die de kolommen voorstelt. De fysische index $\sigma_l = 1 \dots d$ wordt voorgesteld door een verticale lijn (naar boven of naar onder al naargelang we de hermitisch toegevoegde of niet beschouwen)



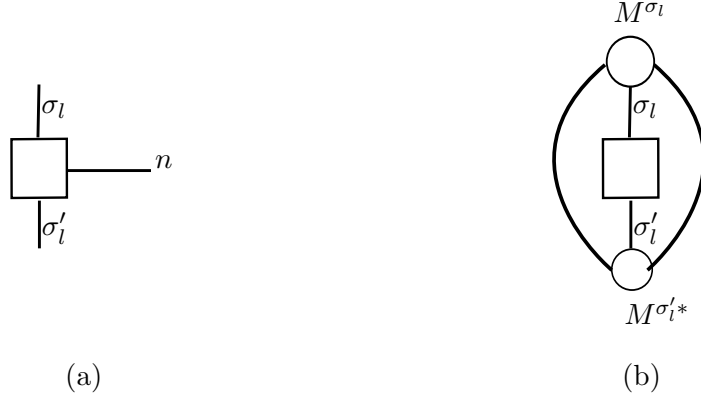
Figuur 2.2: Diagrammatische voorstelling van een MPS.

Regel 2

Lokale operatoren geassocieerd aan site l worden voorgesteld door een vierkant met (in het meest algemene geval) 3 benen (de operator-vertex). Van deze benen zijn er 2 die de de fysische dimensie σ_l voorstellen. Deze benen vertrekken verticaal naar boven en naar beneden uit de operator-vertex. Voor een spin- $\frac{1}{2}$ betekent dit dus dat de 2 verticale benen elk een index voorstellen die kan lopen van 1 tot $d = 2$. Het derde been vertrekt horizontaal uit de operator-vertex en stelt de index voor die loopt over het aantal interactie-termen (hiervoor genoteerd als δ , dus de horizontale index loopt van $n = 1 \dots \delta$). Dit alles wordt samengevat in figuur (2.3)(a).

Regel 3

Contracties zijn matrixvermenigvuldigingen ofwel sommaties over de lijn-indices (deze lijn-indices zijn dus ofwel de fysische indices σ_l , de matrix-indices i, j of de index behorend bij 1 van de δ interactietermen, dus de index n). We spreken af dat dergelijke contracties voorgesteld worden door geconnecteerde benen. Dit betekent dat we 2 benen met elkaar verbinden, wat zoveel betekent als het sommeren over alle waarden van de index die door die 2 benen voorgesteld wordt. Het is dus duidelijk dat je nooit 2 benen die over een verschillende index lopen mag verbinden. Je mag nooit een σ_l -been verbinden met een been van een matrix-index i, j . Contractie van



Figuur 2.3: Diagrammatische voorstelling van een algemene operator-vertex (a) en een voorbeeld van een contractie (b).

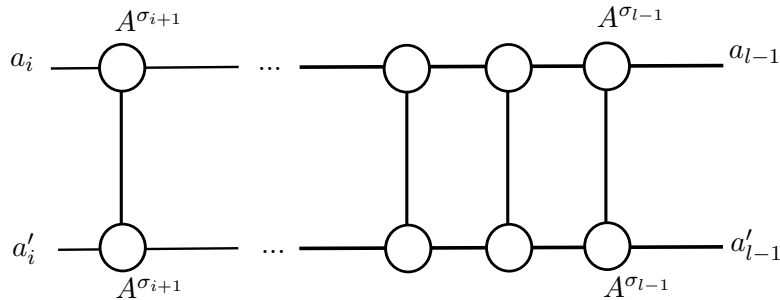
een fysische index σ_l wordt dus voorgesteld door een verticale lijn, terwijl de overige contracties voorgesteld worden door een horizontale lijn. De som over $n = 1 \dots \delta$ wordt voorgesteld door een contractie tussen 2 horizontale benen (één been afkomstig van de operator op site p en het andere afkomstig van de operator op site q).

Regel 4

Daarnaast komen er in de uitdrukking van $\langle \psi | \hat{H} | \psi \rangle$ ook regelmatig termen voor van de vorm

$$\sum_{\sigma_k, i+1 \leq k \leq l-1} (A^{\sigma_{i+1}} \dots A^{\sigma_{l-1}})_{a_{l-1}, a_i}^\dagger (A^{\sigma_{i+1}} \dots A^{\sigma_{l-1}})_{a'_i, a'_{l-1}}. \quad (2.37)$$

Intern zijn er nog vele matrixvermenigvuldigingen tussen de A^{σ_i} matrices die allemaal voorgesteld worden door een (horizontale) connectie van 2 lijnen tussen elke vertex. De som over de σ_l vormt nog een extra (verticale) verbinding tussen de vertices. Er blijven ook 4 vrije indices over, zodat een dergelijke term diagrammatisch kan voorgesteld worden als in figuur (2.4). We spreken dus af dat alle open lijnen een niet-geconnecteerde index voorstellen en dat een dergelijke structuur dus een matrix/tensor voorstelt. De niet-geconnecteerde indices worden bij de open lijnen genoteerd.



Figuur 2.4: Diagrammatische voorstelling van opeenvolgende contracties.

Regel 5

De matrices op het einde van de ketting zijn steeds zo gedimensioneerd dat de volledige contractie (van alle matrices) steeds een scalair is. De eerste en laatste matrix hebben immers een

respectieve dimensie $(1 \times d)$ en $(d \times 1)$ en de 1-index impliceert dus onmiddellijk dat het geheel een scalar voorstelt. Diagrammatisch zullen we dit voorstellen door een boog aan het begin of op het einde van een ketting toe te voegen. Dit betekent dat er steeds voldaan is aan (2.14) of (2.17) en het invoeren van een boog stemt dan inderdaad overeen met de identiteit.

Voorbeeld

Stel dat we bijvoorbeeld de verwachtingswaarde (2.30) willen uitdrukken in een diagram, dan is het duidelijk dat we 4 contracties hebben. Er zijn 2 contracties tussen de matrices M^{σ_l} en $M^{\sigma'_l}$ en telkens 1 contractie tussen de operator \hat{O}_l en de respectieve matrices M^{σ_l} en $M^{\sigma'_l}$. De operator \hat{O}_l heeft 2 indices σ_l en σ'_l en wordt voorgesteld door een vierkant met 2 lijnen vertrekkend uit deze vertex. Diagrammatisch geven we dit weer zoals in figuur (2.3)(b).

2.3.3 De interactierange η

Nu we de basisdiagrammen hebben ingevoerd kunnen we een techniek ontwikkelen die ons toelaat de formules (2.29) tot (2.36) op een iteratieve manier op te bouwen. We zullen dit hieronder in detail uitwerken met behulp van het grafisch formalisme. Daarnaast dienen we op te merken dat de hierboven opgestelde diagrammen in staat zijn om de formules exact weer te geven (net als Feynmandiagrammen dat doen). Het zal blijken dat dit vaker veel overzichtelijker is dan de nogal lijvige formules. Alvorens we deze iteratieve methode uitwerken, voeren we nog een belangrijk begrip in dat we verderop nog regelmatig zullen tegenkomen en dat ook van belang is bij de iteratieve constructie, de zogenaamde *interactierange*. Merk daarvoor op dat we de algemene Hamiltoniaan (2.19) nog kunnen schrijven als

$$\hat{H} = \sum_{l=1}^L \hat{O}_l + \sum_{n=1}^{\delta} \sum_{l=1}^{L-1} \sum_{m=l+1}^{l+\eta} \hat{P}_l^n \otimes \hat{P}_m^n, \quad \forall l : l + \eta \leq L. \quad (2.38)$$

Als alle sites met elkaar kunnen interageren dan is de interactierange uiteraard $\eta = L - 1$ omdat site 1 dan ook nog met site L kan interageren. Wanneer we echter vooropstellen dat site 1 slechts kan interageren met site 2 tot en met site $\eta + 1$ dan is de *interactierange* η . Aangezien de matrices \hat{P}_l^n en \hat{P}_l^n (alsook \hat{O}_l) steeds kunnen voorgesteld worden door een bepaalde spin-operator \hat{S}_l^n , eventueel vermenigvuldigd met een koppelingsconstante $J \in \mathbb{R}$ (die voor elke m bij een gegeven l kan verschillen), kunnen we de Hamiltoniaan nog schrijven als

$$\hat{H} = \beta \sum_{l=1}^L \hat{S}_l^p + \sum_{n=1}^{\delta} \sum_{l=1}^{L-1} \sum_{m=l+1}^{l+\eta} J_{m-l} \hat{S}_l^n \otimes \hat{S}_m^n, \quad \forall l : l + \eta \leq L, \quad \beta \in \mathbb{R} \quad (2.39)$$

waarbij $\hat{S}_l^{p/n}$ een willekeurige spinoperator op site l is. Concreet betekent dit dat de interactie tussen dichtste naburen ($l = p, m = p + 1$) een sterkte J_1 heeft, de interactie tussen de op één na dichtste naburen ($l = p, m = p + 2$) een sterkte J_2 , enzovoort. We zullen nu grafisch alle termen van $\langle \psi | \hat{H} | \psi \rangle$ voorstellen. Als voorbeeld beschouwen we een ketting met lengte $L = 5$ en $\eta = 2$ en beschouwen alle termen die bijdragen tot $\langle \psi | \hat{H} | \psi \rangle$. We geven dit weer in figuur (2.5) voor de opeenvolgende gevallen $l = 2, 3, 4$ en baseren ons daarbij steeds op de formules (2.29) tot (2.36). Dit betekent dat we steeds verschillende linker- en rechterblokken hebben afhankelijk van de waarde van l . In de figuur geven we de site steeds weer met het corresponderende site-nummer

zonder steeds de matrices M^{σ_l} te vermelden om het geheel overzichtelijk te houden. Daarnaast zullen we de lokale operator ook als \hat{O}_l noteren terwijl we de interactie-matrices zullen noteren met \hat{S}_l^n . De operator die vergezeld is van de extra factor J_{m-l} zullen we enkel noteren door de J_{m-l} te vermelden, het nummer van de site geeft immers duidelijk aan welke operator erbij hoort. Let ook op dat er een extra horizontale lijn is tussen de vierkante blokken die de matrices \hat{S}_l^n (dit is een matrix $\forall n$) bevatten omdat er in de Hamiltoniaan ook nog een som over de n -index voorkomt. Door zorgvuldige studie van figuur (2.5) kunnen we verbanden vinden tussen de termen voor opeenvolgende l . Stel dat we bijvoorbeeld l laten toenemen van links naar rechts (dus van site 1 tot site 5) dan kunnen we het blok links van l construeren door gebruik te maken van het blok dat links van $l - 1$ zat. Op analoge manier kunnen we de rechterblokken iteratief construeren als we l van site 5 tot site 1 laten lopen. Het onderwerp van de volgende sectie is dan ook deze verbanden af te leiden met de bedoeling het geheel in een compacte vorm weer te geven.

2.4 De X/Y -blokstructuur van $\langle \psi | \hat{H} | \psi \rangle$

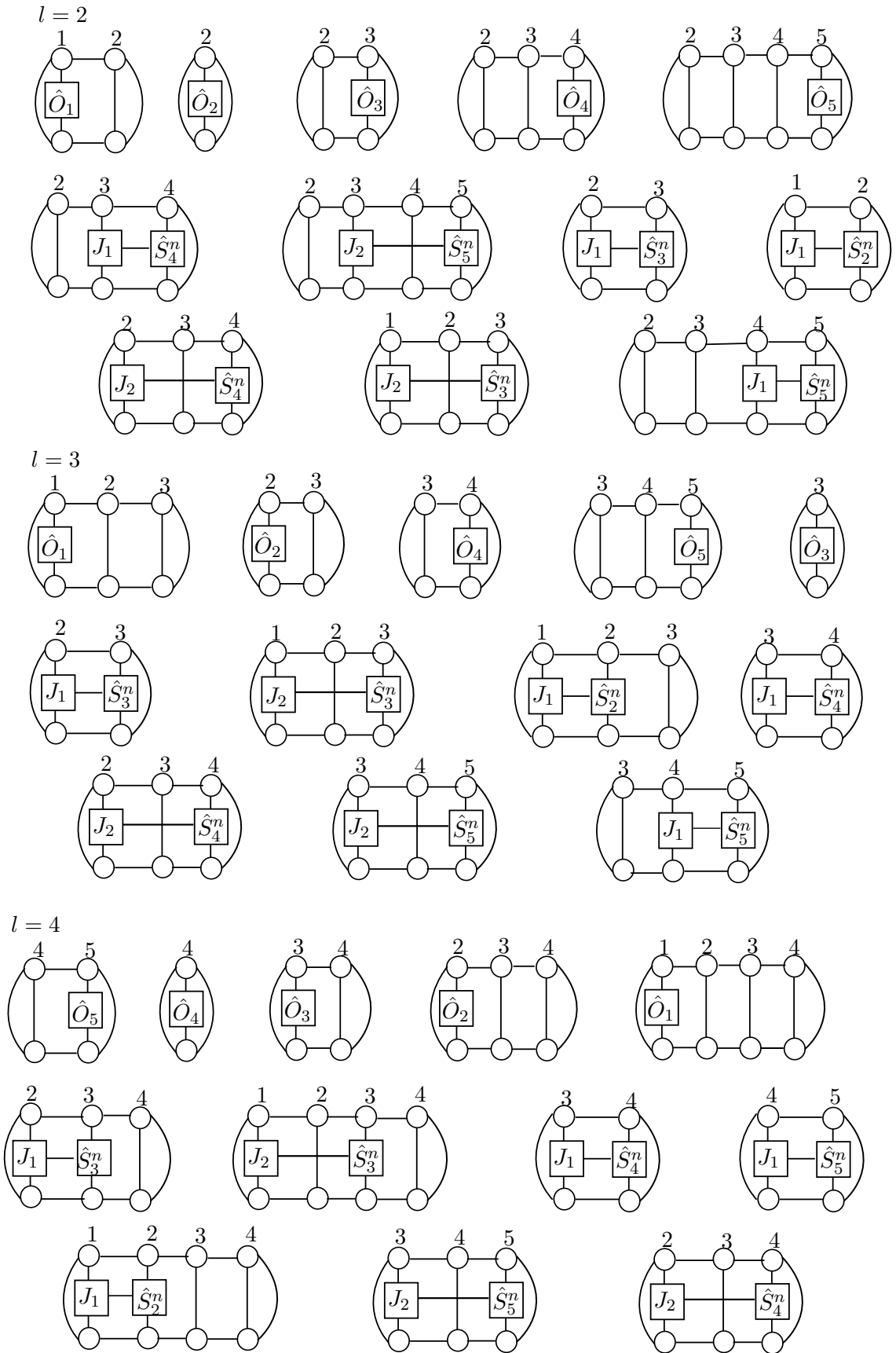
In deze sectie wijken we af van ([4]) en introduceren we een blokvorm om de energie voor te stellen. We voeren eerst de volgende notaties in die we in het vervolg zullen gebruiken in de diagrammen

$$\begin{aligned} C_l^n &\triangleq \hat{S}_l^n \\ (D_l^n)_\alpha &\triangleq J_\alpha \hat{S}_l^n. \end{aligned} \tag{2.40}$$

Er zijn dus η operatoren $(D_l^n)_\alpha$ ($\alpha = 1 \dots \eta$) waarbij elke $(D_l^n)_\alpha$ de δ operatoren ($n = 1 \dots \delta$) op site l bevat, vermenigvuldigd met de koppelingsconstante J_α . Als we opnieuw figuur (2.5) bekijken dan zien we dat er fundamenteel twee verschillende linker- en rechterblokken zijn. Enerzijds zijn er (zowel links als rechts van l) blokken waarbij alle contracties tussen operatoren (dus horizontale lijnen tussen de vierkanten) opgenomen zijn in het linker- en rechterblok zelf. Anderzijds zijn er ook termen waarbij er een koppeling is tussen de blokken: er loopt dus een horizontale contractie tussen de linker- en rechterkant. Blokken waarbij alle horizontale contracties binnen het blok blijven noemen we X -blokken (X^L en X^R voor de respectievelijke linker- en rechterblokken), terwijl blokken die nog een externe horizontale lijn hebben (die contraheert met de externe lijn van het andere blok of met site l) Y -blokken zijn (Y^L en Y^R). De blokken Y_l^L en X_l^L bevatten alle sites 1 tot en met l terwijl de blokken Y_l^R en X_l^R alle sites l tot en met L bevatten. De volledige fysica zit dus vervat in de blokken Y_{l-1}^L , X_{l-1}^L , Y_{l+1}^R , X_{l+1}^R en site l .

2.4.1 Constructie van de Y^L/X^L -blokken

Stel dat we site l variëren van 1 tot L , dan is het duidelijk dat het linkerblok steeds groter wordt en het rechterblok kleiner. In deze sectie bestuderen we de constructie van de Y^L - en X^L -blokken. We zullen daarbij gebruik maken van het grafisch formalisme om aan te tonen hoe de blokken groeien. Om de gedachten te vestigen kan het voor de lezer handig zijn om terug te grijpen naar figuur (2.5).



Figuur 2.5: Voorbeeld van een aantal termen in $\langle \psi | \hat{H} | \psi \rangle$.

We beginnen met de Y^L blokken en voeren voor elke site η $(Y_l^L)_\alpha$ blokken in (waarbij dus $l = 1 \dots L$ en $\alpha = 1 \dots \eta$). Verder heeft elke $(Y_l^L)_\alpha$ ($\forall l, \forall \alpha$) 3 indices: 1 index die loopt over alle δ interactie-operatoren (aangegeven door de index n) en voor elke operator $(Y_l^L)_\alpha^n$ zijn er ook nog 2 matrixindices (i, j) . Elk element dat vervat is in het Y^L -blok kan op een unieke manier genummerd worden met de volgende meest algemene notatie

$$((Y_l^L)_\alpha^n)_{i,j}.$$

Het diagram dat correspondeert met een Y^L -blok wordt weergegeven in figuur (2.6)(a). Nu we alle conventies ingevoerd hebben kunnen we via diagrammen weergeven hoe de blokken iteratief opgebouwd worden. Beschouw daartoe figuur (2.7) om te zien hoe we de opeenvolgende Y_l^L construeren. Gebruik makend van de figuur vinden we als algemene iteratieve formule voor een Y^L -blok (zie ook figuur (2.7)(b)):

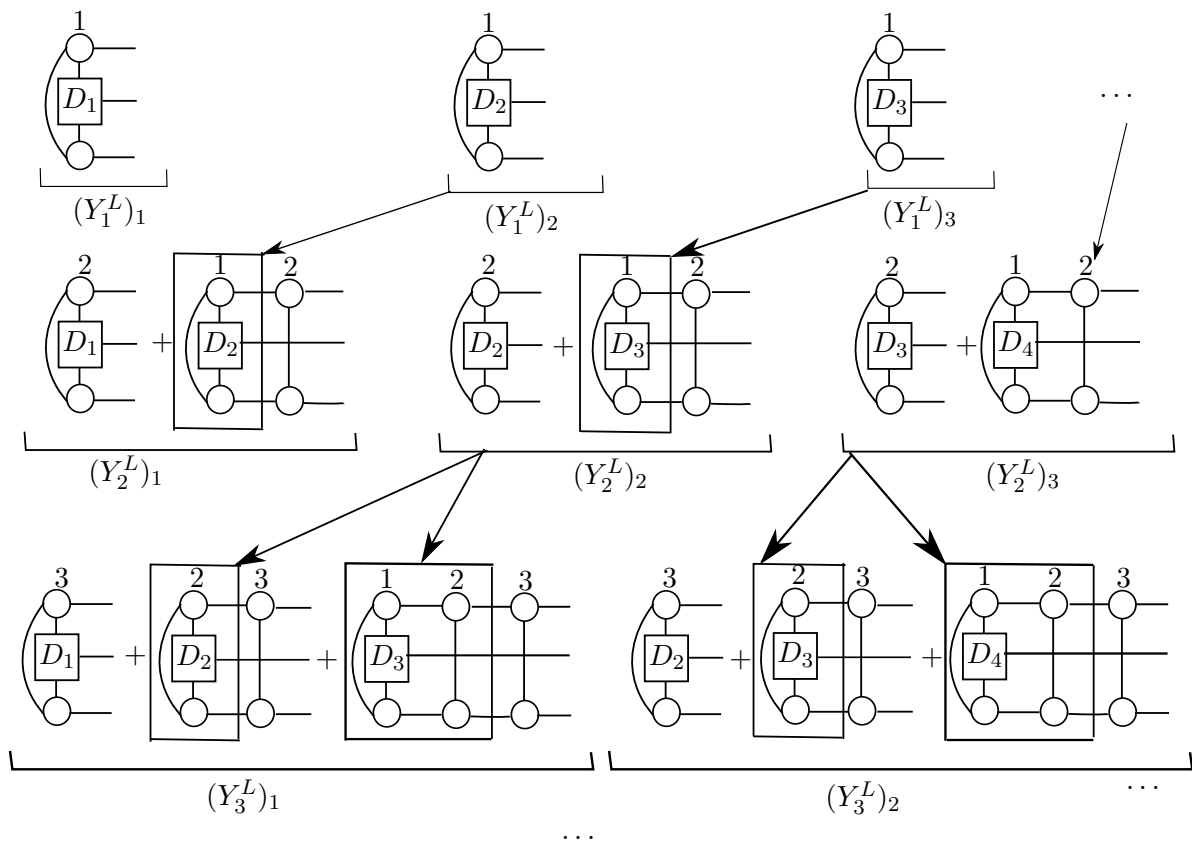
$$((Y_l^L)_\alpha^n)_{i,j} = \sum_{k,\sigma_l,\sigma'_l} M_{k,j}^{\sigma_l} M_{k,i}^{\sigma'_l*} ((D_l^n)_\alpha)_{\sigma_l,\sigma'_l} + \sum_{k,m,\sigma_l} ((Y_{l-1}^L)_\alpha^n)_{k,m} M_{m,j}^{\sigma_l} M_{k,i}^{\sigma_l*} \quad (2.41)$$

met $l = 1, 2, \dots, L - 1$ en $\alpha = 1, 2, \dots, \eta$ en waarbij $(Y_{l-1}^L)_\alpha^n = 0$ als $l = 1$ of $\alpha = \eta$ (waarbij 0 hier, en ook verderop in analoge randvoorwaarden, de nul-matrix voorstelt). Op het eerste zicht lijkt deze definitie en constructie van de Y^L -blokken wat uit de lucht gevallen, maar we zullen verderop zien dat deze definitie ervoor zorgt dat we alle interacties tussen het linkerblok en site l of het rechterblok mooi in rekening kunnen brengen.

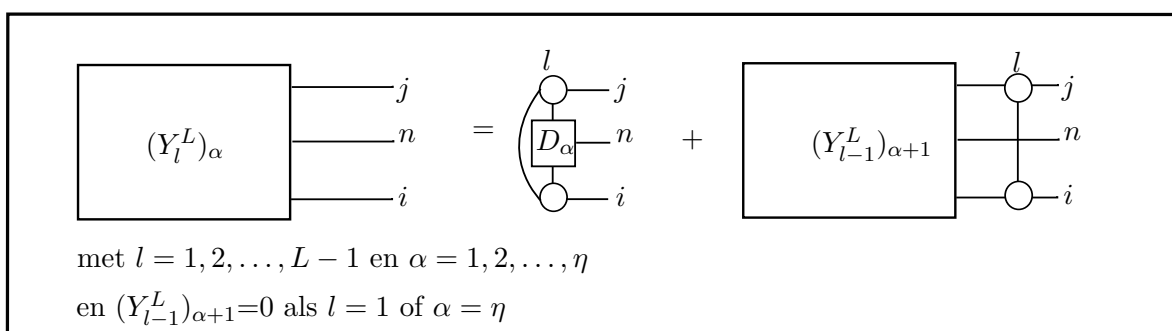
Nu we weten hoe we het Y^L -blok moeten construeren gaan we over naar het X^L -blok (zie figuur (2.6)(b)). We weten dat bij conventie het X_l^L -blok alle interne contracties van het linkerblok (links van site $l + 1$) bevat. Dat betekent dat alle mogelijke interacties tussen de sites 1 tot en met l in dit blok vervat moeten zitten. Door gebruik te maken van het $(Y_{l-1}^L)_{\alpha=1}$ blok zullen we zien dat we X_l^L onmiddellijk kunnen neerschrijven in functie van de linkerblokken voor $l - 1$. Bovendien zullen ook de lokale operatoren geassocieerd aan de sites 1 tot en met l in X_l^L worden opgenomen, omdat die ook geen externe contracties bevatten met site $l + 1$ of het rechterblok. We verwijzen naar figuur (2.5) en vatten in figuur (2.8) samen welke termen moeten opgenomen worden voor opeenvolgende waarden van l . In de figuur gebruiken we de ingevoerde conventies (2.40), zonder echter expliciet het site-nummer en de index n te vermelden bij de C -operator. In deze figuur kunnen we zien dat X_l^L iteratief kan opgebouwd worden uit X_{l-1}^L en $(Y_{l-1}^L)_{\alpha=1}$.



Figuur 2.6: Diagram voor een Y^L -blok (a) en X^L -blok (b).

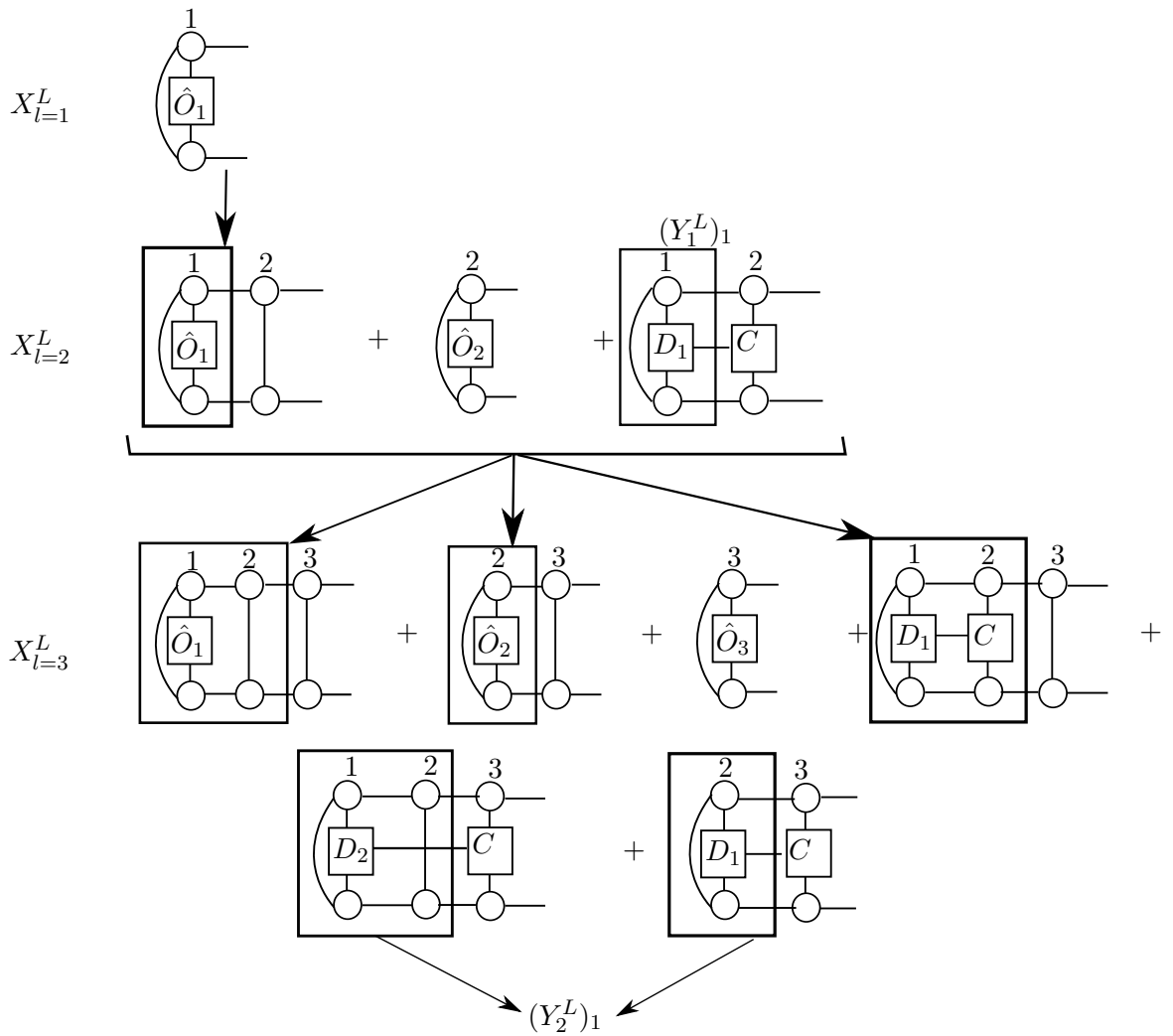


(a) Iteratieve constructie van de verschillende Y^L blokken

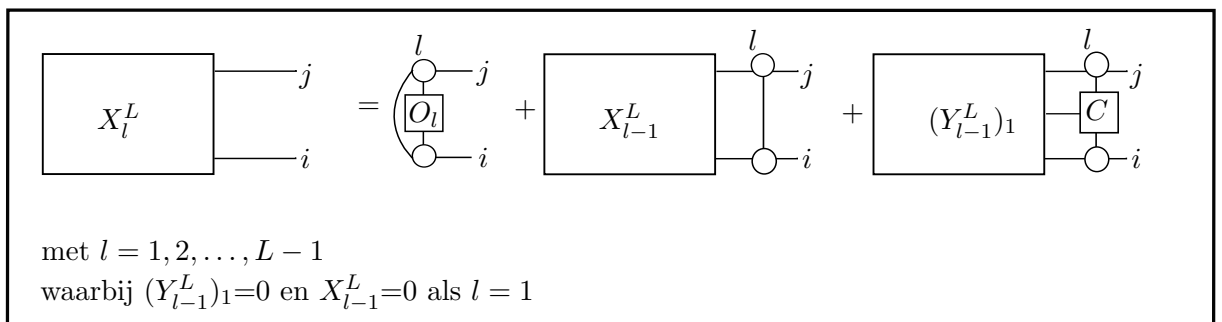


(b) Algemene formule voor $(Y_l^L)_\alpha$

Figuur 2.7: Iteratieve constructie van een Y^L -blok.



(a) Iteratieve constructie van de verschillende X^L blokken



(b) Algemene formule voor X_l^L

Figuur 2.8: Iteratieve constructie van een X^L -blok.

In formules uitgeschreven wordt dit

$$(X_l^L)_{i,j} = \sum_{k,\sigma_l,\sigma'_l} M_{k,j}^{\sigma_l} M_{k,i}^{\sigma'_l*} (\hat{O}_l)_{\sigma_l,\sigma'_l} + \sum_{k,m,\sigma_l} (X_{l-1}^L)_{k,m} M_{m,j}^{\sigma_l} M_{k,i}^{\sigma_l*} + \sum_{k,m,n,\sigma_l,\sigma'_l} ((Y_{l-1}^L)_1^n)_{k,m} M_{m,j}^{\sigma_l} M_{k,i}^{\sigma'_l*} (C_l^n)_{\sigma_l,\sigma'_l} \quad (2.42)$$

met $l = 1, 2, \dots, L-1$ en waarbij $X_{l-1}^L = 0$ en $(Y_{l-1}^L)_1 = 0$ als $l = 1$.

2.4.2 Constructie van de Y^R/X^R -blokken

Op een analoge manier zullen we de blokken voor het rechterdeel construeren. Toch verloopt de constructie niet geheel symmetrisch omdat we de koppelingsconstanten J_n opsorpen in het linkerdeel (bij de constructie van het X^R -blok zullen we hiervan echter moeten afwijken om alles in een nette vorm te kunnen schrijven). Daarom zullen we de Y^R -blokken op een aangepaste manier definiëren zoals weergegeven in figuur (2.9) waarbij de 0 slaat op de nul-matrix. Ook de Y^R -blokken worden dus iteratief opgebouwd:

$$((Y_l^R)_\alpha^n)_{i,j} = \sum_{k,\sigma_l,\sigma'_l} M_{j,k}^{\sigma_l} M_{i,k}^{\sigma'_l*} (C_l^n)_{\sigma_l,\sigma'_l} \delta_{\alpha,1} + \sum_{k,m,\sigma_l} ((Y_{l+1}^R)_\alpha^{n-1})_{k,m} M_{j,m}^{\sigma_l} M_{i,k}^{\sigma_l*} \quad (2.43)$$

met $l = L, L-1, \dots, 2$ en $\alpha = 1, 2, \dots, \eta$ waarbij $(Y_{l+1}^R)_{\alpha-1} = 0$ als $l = L$ of $\alpha = 1$.

Voor de opbouw van de X^R -blokken zullen we opnieuw gebruik maken van figuur (2.5) en kijken naar alle interne contracties en lokale operatoren rechts van site l . Vermits de Y^R -blokken enkel de operator C bevatten³ zullen we een som moeten beschouwen van contracties tussen de verschillende Y^R en de operator D_α . Dit alles is samengevat in figuur (2.10), waarbij de lokale operatoren \hat{O}_l gewoon genoteerd zijn als \hat{O} . De formule voor X^R is analoog aan deze voor X^L :

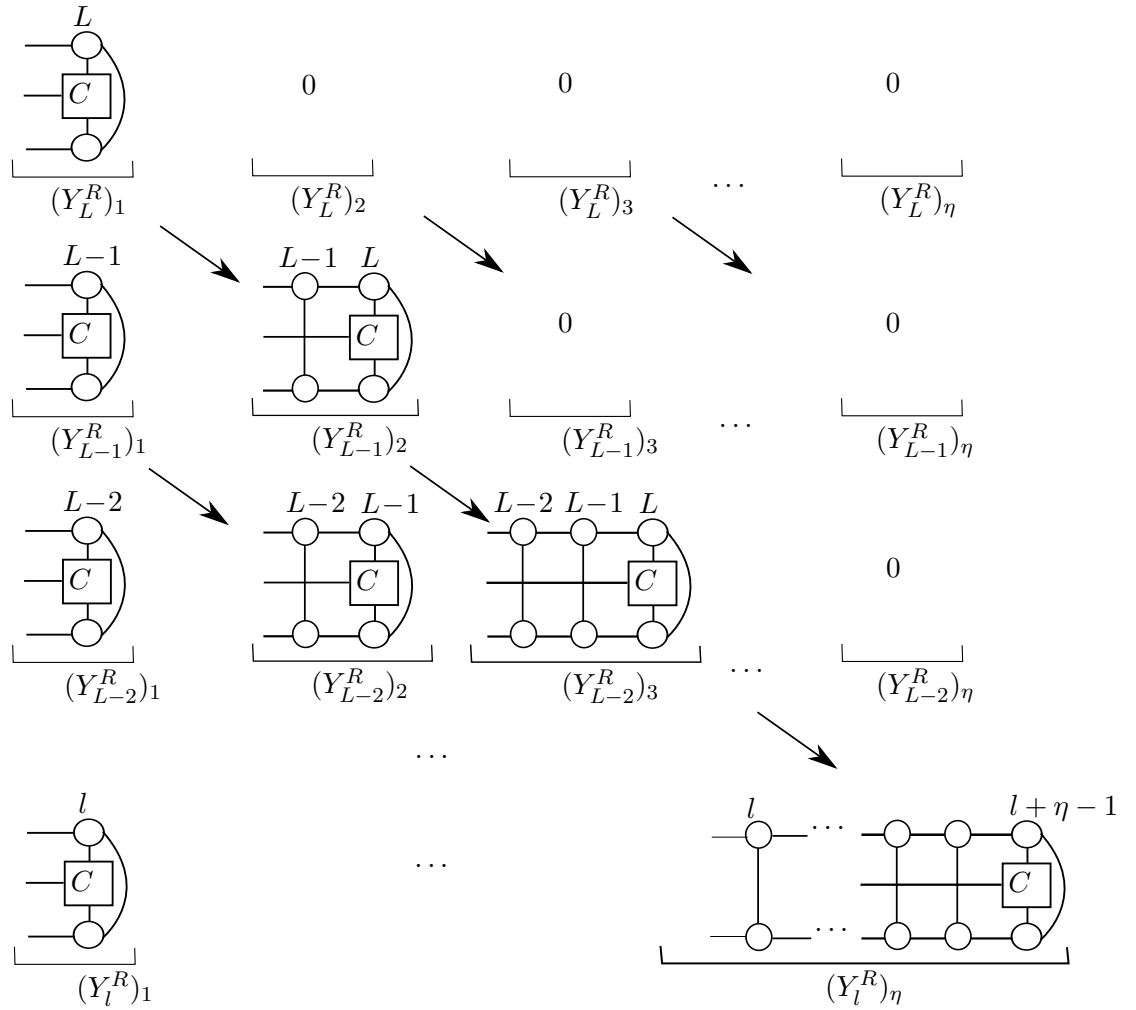
$$(X_l^R)_{i,j} = \sum_{k,\sigma_l,\sigma'_l} M_{j,k}^{\sigma_l} M_{i,k}^{\sigma'_l*} (\hat{O}_l)_{\sigma_l,\sigma'_l} + \sum_{k,m,\sigma_l} (X_{l+1}^R)_{k,m} M_{j,m}^{\sigma_l} M_{i,k}^{\sigma_l*} + \sum_{\alpha=1}^{\eta} \sum_{k,m,n,\sigma_l,\sigma'_l} ((Y_{l+1}^R)_\alpha^n)_{k,m} M_{j,m}^{\sigma_l} M_{i,k}^{\sigma'_l*} ((D_l^n)_\alpha)_{\sigma_l,\sigma'_l} \quad (2.44)$$

met $l = L, L-1, \dots, 2$ en waarbij $X_{l+1}^R = 0$ en $(Y_{l+1}^R)_\alpha = 0$ als $l = L$.

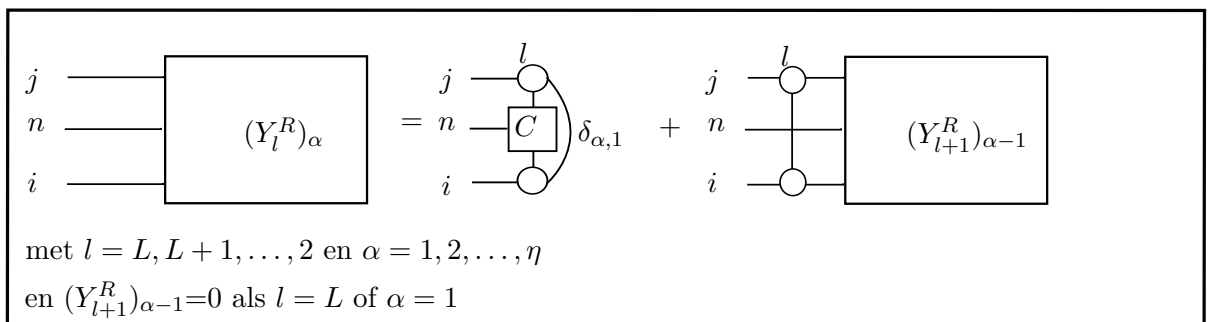
2.4.3 $\langle \psi | \hat{H} | \psi \rangle$ in X/Y -vorm

We hebben nu alle blokken geconstrueerd om de verwachtingswaarde $\langle \psi | \hat{H} | \psi \rangle$ uit te drukken in functie van de X - en Y -blokken. Alle interacties binnen een blok zijn reeds vervat in de X^L - en X^R -blokken, maar de interacties tussen de blokken en de interacties van elk blok met site l zijn nog niet in rekening gebracht. Het zal blijken dat de Y^L - en Y^R -blokken daartoe bijzonder geschikt zijn. Bij de hieropvolgende bespreking kan het nuttig zijn om af en toe terug te grijpen naar figuur (2.5) om in te zien welke interacties dit precies zijn. Daarnaast is het ook instructief om de figuren (2.7) en (2.9) bij de hand te nemen.

³We laten het hoedje bij de operatoren C en D ook hier weg voor de notationale eenvoud.

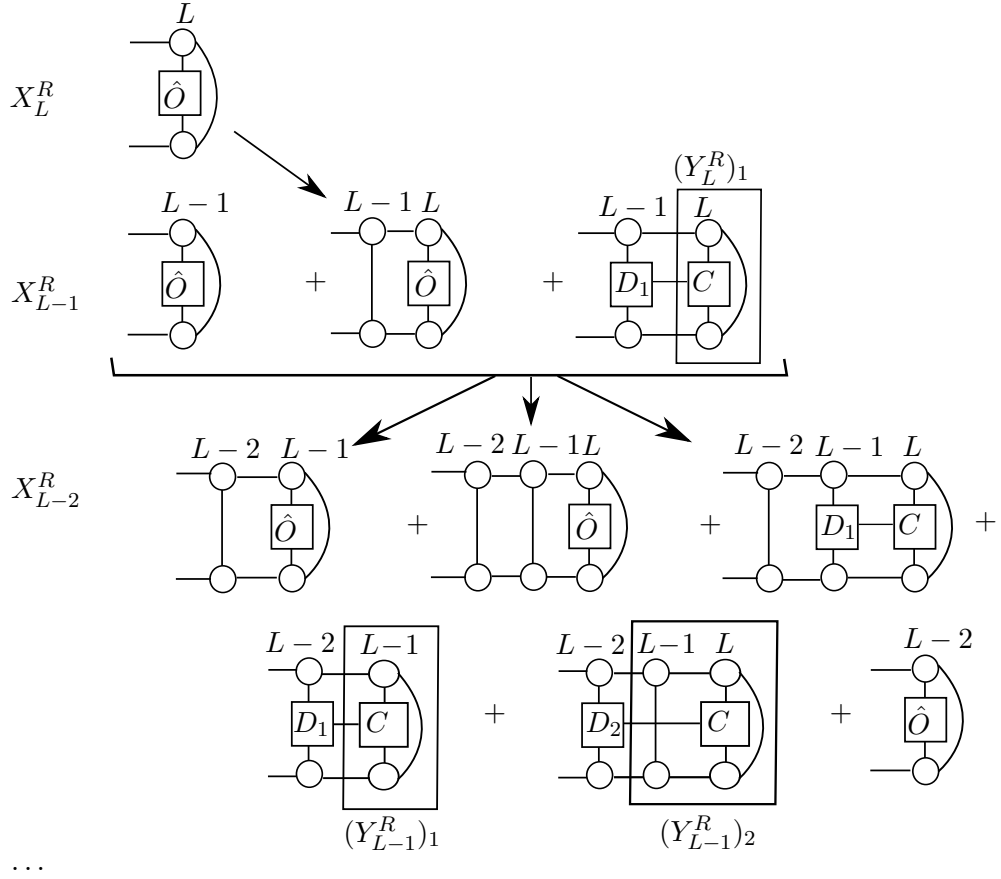


(a) Iteratieve constructie van de verschillende Y^R blokken

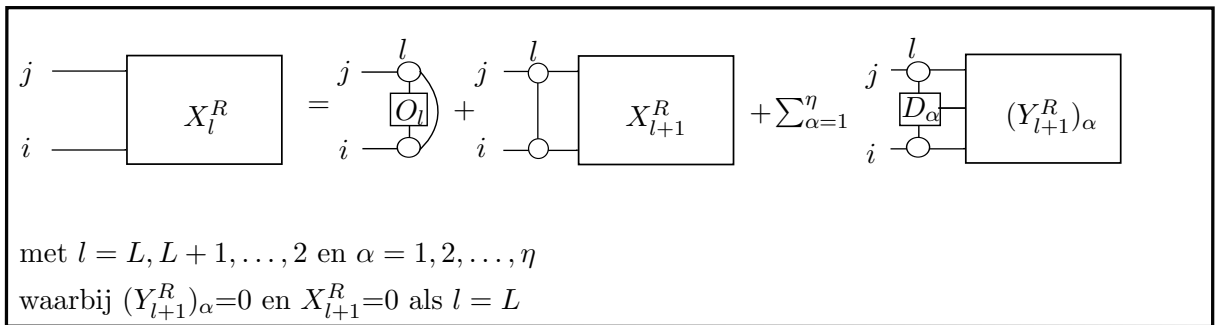


(b) Algemene formule voor $(Y_l^R)_\alpha$

Figuur 2.9: Iteratieve constructie van een Y^R -blok.



(a) Iteratieve constructie van de verschillende X^R blokken



(b) Algemene formule voor X_l^R

Figuur 2.10: Iteratieve constructie van een X^R -blok.

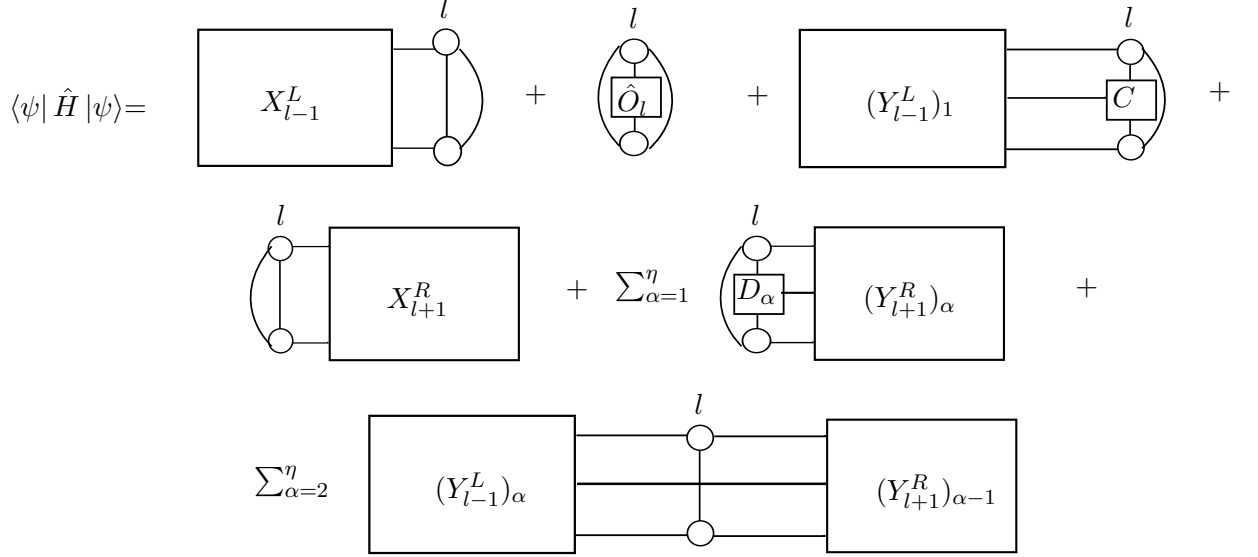
Om de gedachten te vestigen beschouwen we het geval $l = 3$ in een ketting met lengte $L = 5$ en een interactierange $\eta = 3$. Aangezien we $l = 3$ beschouwen moeten we de blokken X_2^L en Y_2^L (algemeen worden dit de blokken met index $l - 1$) gebruiken om het linkerdeel te beschrijven. We kijken nu wat de blokken $(Y_2^L)_\alpha$ ($\alpha = 1, 2, 3$) precies bevatten (zie figuur (2.7)):

- $(Y_{l-1=2}^L)_{\alpha=1}$ bevat alle termen die, indien ze gecontraheerd worden met een C -operator op site l (hier $l = 3$), de juiste interactie met site l geven. Als het aantal sites in het linkerblok groter wordt dan de interactierange η (dus als $l > \eta + 1$) dan zorgt de specifieke constructie van de Y_L ervoor dat sites die te ver liggen van l (dus verder dan η) niet meer zullen interageren met l .
- $(Y_{l-1=2}^L)_{\alpha=2}$ bevat alle termen die, indien ze gecontraheerd worden met een C -operator op site $l + 1$, de juiste interactie geven met site $l + 1$.
- $(Y_{l-1=2}^L)_{\alpha=3}$ bevat alle termen die, indien ze gecontraheerd worden met een C -operator op site $l + 2$, de juiste interactie geven met site $l + 2$.
- ...

Voor het rechterblok moeten we $(Y_{l+1=4}^R)_\alpha$ beschouwen (zie figuur 2.9):

- $(Y_{l+1=4}^R)_{\alpha=1}$ bevat de term die, indien gecontraheerd met een $(D_{\alpha=1})$ -operator op site l , de dichtste-nabuur interactie geeft met site l (dus tussen site l en $l + 1$).
- $(Y_{l+1=4}^R)_{\alpha=2}$ bevat de term die, indien gecontraheerd met een $(D_{\alpha=2})$ -operator op site l , de tweede orde (J_2) interactie geeft met site l (dus tussen site l en $l + 2$).
- ...

We hebben dus een stramien gevonden om de interacties met site l volledig op te nemen in het schema. Het enige dat ons nu nog rest is de interacties tussen de linkerkant en de rechterkant van site l in rekening brengen. We hebben hierboven gezien dat de contractie van $(Y_{l-1}^L)_\alpha$ met een C -operator op site $l + \alpha - 1$ de juiste contracties geeft met die site. De C -operator van een dergelijke site is precies wat er in $(Y_{l+1}^R)_{\alpha-1}$ ($\alpha \neq 1$) zit. Om de overblijvende interacties tussen het linker- en rechterdeel te beschrijven, moeten we dus de som nemen over α van 2 tot η waarbij elke term in de som de contractie bevat tussen $(Y_{l-1}^L)_\alpha$ en $(Y_{l+1}^R)_{\alpha-1}$ (met uiteraard nog de site l ertussen gecontraheerd zoals men kan zien in figuur (2.5)). Tot slot merken we op dat de lokale operator op site l (\hat{O}_l) ook nog in rekening moet gebracht worden om alle energie-bijdragen in rekening te brengen. Deze beschouwing laat ons nu toe de verwachtingswaarde $\langle \psi | \hat{H} | \psi \rangle$ volledig uit te drukken met behulp van de X - en Y -blokken. Dit wordt weergegeven in figuur (2.11).



Figuur 2.11: De verwachtingswaarde $\langle \psi | \hat{H} | \psi \rangle$ in X/Y-vorm.

We vermelden voor de volledigheid ook nog $\langle \psi | \hat{H} | \psi \rangle$ in formulevorm:

$$\begin{aligned}
\langle \psi | \hat{H} | \psi \rangle = & \sum_{k,m,\sigma_l,\sigma'_l} M_{m,k}^{\sigma_l} M_{m,k}^{\sigma'_l*} (\hat{O}_l)_{\sigma_l,\sigma'_l} + \sum_{k,m,p,\sigma_l} (X_{l-1}^L)_{k,m} M_{m,p}^{\sigma_l} M_{k,p}^{\sigma_l*} + \\
& \sum_{k,m,p,\sigma_l} (X_{l+1}^R)_{k,m} M_{p,m}^{\sigma_l} M_{p,k}^{\sigma_l*} + \sum_{k,m,p,n,\sigma_l,\sigma'_l} ((Y_{l-1}^L)_1^n)_{k,m} M_{m,p}^{\sigma_l} M_{k,p}^{\sigma'_l*} (C_l^n)_{\sigma_l,\sigma'_l} + \\
& \sum_{\alpha=1}^{\eta} \sum_{k,m,p,n,\sigma_l,\sigma'_l} ((Y_{l+1}^R)_\alpha^n)_{k,m} M_{p,m}^{\sigma_l} M_{p,k}^{\sigma'_l*} ((D_l^n)_\alpha)_{\sigma_l,\sigma'_l} + \\
& \sum_{\alpha=2}^{\eta} \sum_{k,m,p,q,n,\sigma_l} ((Y_{l-1}^L)_\alpha^n)_{k,m} M_{m,p}^{\sigma_l} M_{k,q}^{\sigma_l*} ((Y_{l+1}^R)_{\alpha-1}^n)_{q,p} \quad (2.45)
\end{aligned}$$

Op dit moment hebben we alle nodige objecten ingevoerd voor de constructie van een eigenwaardeprobleem, geassocieerd aan de Schrödingervergelijking, en het onderwerp van de volgende sectie is dan ook het afleiden van een dergelijk eigenwaardeprobleem voor een (sterk) gecorreleerd kwantumsysteem.

2.5 Variationale optimalisatie en de effectieve Hamiltoniaan

Vanaf nu maken we terug gebruik van technieken uit ([4]). Het is ons uiteindelijk doel om de grondtoestand van een kwantumsysteem te vinden (de methode kan echter uitgebreid worden om ook geëxciteerde toestanden te vinden). We zullen daartoe een variationeel principe toepassen op

$$E_0 = \frac{\langle \psi | \hat{H} | \psi \rangle}{\langle \psi | \psi \rangle} \quad (2.46)$$

waarbij we als golf functie $|\psi\rangle$ de gemengd canonische MPS gebruiken

$$|\psi\rangle = \sum_{\sigma_1, \dots, \sigma_L} A^{\sigma_1} \dots A^{\sigma_{l-1}} M^{\sigma_l} B^{\sigma_{l+1}} \dots B^{\sigma_L} |\sigma_1, \dots, \sigma_L\rangle. \quad (2.47)$$

We herinneren eraan dat de A -matrices hierbij links-genormaliseerd zijn en de B -matrices rechts-genormaliseerd (dit werd immers gebruikt bij de afleiding van formules (2.29) tot (2.36)) en dat we de verwachtingswaarde van de Hamiltoniaan $\langle \psi | \hat{H} | \psi \rangle$ kunnen uitdrukken in de X/Y -vorm (zie (2.45)). Om de grondtoestand van het systeem (met de corresponderende grondtoestandsenergie) te bepalen voeren we een Lagrange multiplicator λ in en extremeren

$$\langle \psi | \hat{H} | \psi \rangle - \lambda \langle \psi | \psi \rangle. \quad (2.48)$$

Het grote probleem echter is dat de onbekende toestanden M^{σ_i} in productvorm voorkomen (de Matrix *Product State*) en dat het bovenstaande probleem dus sterk niet-lineair is. Daarom zullen we een iteratieve procedure toepassen waarbij de matrices op alle sites behalve site l vastgehouden worden terwijl de matrices op site l geoptimaliseerd worden. We extremeren dus in elke iteratiestap naar één specifieke M^{σ_i} wat ons uiteindelijk een eigenwaardeprobleem zal opleveren. In de volgende stap gaan we naar site $l + 1$ (cyclus naar rechts) en herhalen deze procedure. Wanneer we bij het einde van de ketting komen (dus site L) zullen we bij de volgende iteratiestap terugkeren naar site $L - 1$ (cyclus naar links) tot we site 1 bereiken en we de rechtercyclus opnieuw opstarten. Dit alles gaat verder tot de energie convergeert binnen een vooraf ingestelde accuraatheid. Men noemt deze opeenvolging van cycli het *sweep algoritme*. In wat volgt tonen we aan dat deze iteratieve constructie aanleiding geeft tot een eigenwaardeprobleem waarbij λ uiteindelijk de grondtoestandsenergie zal zijn. Daartoe zullen we wel een beroep moeten doen op de Hamiltoniaan in blokform (zie formule (2.45)).

We extremeren (2.48) naar $M_{a'_{l-1}, a'_l}^{\sigma'_l}$

$$\frac{\partial}{\partial M_{a'_{l-1}, a'_l}^{\sigma'_l}} \left(\langle \psi | \hat{H} | \psi \rangle - \lambda \langle \psi | \psi \rangle \right) = 0 \quad (2.49)$$

en merken daarbij op dat

$$\begin{aligned} \langle \psi | \psi \rangle &= \sum_{\{S\}} \sum_{\sigma_1 \dots \sigma_{l-1}} \left(A^{\sigma_{l-1} \dagger} \dots A^{\sigma_1 \dagger} A^{\sigma_1} \dots A^{\sigma_{l-1}} \right)_{a_{l-1}, a'_{l-1}} \sum_{\sigma_l, \sigma'_l} M_{a_{l-1}, a_l}^{\sigma_l} M_{a'_{l-1}, a'_l}^{\sigma'_l} \\ &\quad \sum_{\sigma_{l+1} \dots \sigma_L} \left(B^{\sigma_{l+1}} \dots B^{\sigma_L} B^{\sigma_L \dagger} \dots B^{\sigma_{l+1} \dagger} \right)_{a_l, a'_l} \delta_{\sigma_l, \sigma'_l} \\ &= \sum_{\{S\}} \sum_{\sigma_l} M_{a_{l-1}, a_l}^{\sigma_l} M_{a'_{l-1}, a'_l}^{\sigma'_l} \left(\delta_{a_{l-1}, a'_{l-1}} \delta_{a_l, a'_l} \delta_{\sigma_l, \sigma'_l} \right) \end{aligned} \quad (2.50)$$

wegens de normalisatie van de A - en B -matrices en waarbij $\{S\} = \{a_{l-1}, a'_{l-1}, a_l, a'_l\}$. Bovendien geldt ook dat $\sigma_l = \sigma'_l$ zodat na extremeren de som over σ_l zal wegvallen en wegens de twee Kronecker delta functies zal ook de som over de verzameling $\{S\}$ volledig wegvallen (omdat de som door de delta functies op zich al beperkt werd tot twee indices, zodat na extremeren er geen enkele sommatie-index meer overblijft). We bekomen op die manier

$$\frac{\partial}{\partial M_{a'_{l-1}, a'_l}^{\sigma'_l}} (\langle \psi | \psi \rangle) = \sum_{\{S\}} M_{a_{l-1}, a_l}^{\sigma_l} \left(\delta_{a_{l-1}, a'_{l-1}} \delta_{a_l, a'_l} \delta_{\sigma_l, \sigma'_l} \right) = M_{a'_{l-1}, a'_l}^{\sigma'_l} \quad (2.51)$$

Voor het uitwerken van de andere termen (dus het extremeren van de termen in $\langle \psi | \hat{H} | \psi \rangle$) maken we handig gebruik van het feit dat sommige indices gelijk zijn (als gevolg van de normalisatie

van de A - of B -matrices) en we dus evengoed a_l kunnen vervangen door a'_l of dat we de ene matrix M een index σ_l kunnen geven en de andere een index σ'_l (wat dan weer het gevolg is van het feit dat een dergelijk blok geen operator bevat op site l en dus $\sigma_l = \sigma'_l$). We zullen van deze bevindingen expliciet gebruik maken om één van de termen in (2.45) te extremeren, maar laten voor de andere termen deze berekeningen achterwege omdat ze volledig analoog verlopen. Beschouw bijvoorbeeld

$$\begin{aligned}
& \frac{\partial}{\partial M_{a'_{l-1}, a'_l}^{\sigma'_l}} \left(\sum_{a'_{l-1}, a_{l-1}, a_l, \sigma_l} (X_{l-1}^L)_{a'_{l-1}, a_{l-1}} M_{a_{l-1}, a_l}^{\sigma_l} M_{a'_{l-1}, a_l}^{\sigma'_l} \right) \\
&= \frac{\partial}{\partial M_{a'_{l-1}, a'_l}^{\sigma'_l}} \left(\sum_{a'_{l-1}, a_{l-1}, a_l, \sigma_l} (X_{l-1}^L)_{a'_{l-1}, a_{l-1}} M_{a_{l-1}, a_l}^{\sigma_l} M_{a'_{l-1}, a'_l}^{\sigma'_l} \delta_{a_l, a'_l} \delta_{\sigma_l, \sigma'_l} \right) \\
&= \sum_{a_{l-1}, a_l, \sigma_l} (X_{l-1}^L)_{a'_{l-1}, a_{l-1}} M_{a_{l-1}, a_l}^{\sigma_l} \delta_{a_l, a'_l} \delta_{\sigma_l, \sigma'_l} \\
&= \sum_{a_{l-1}} (X_{l-1}^L)_{a'_{l-1}, a_{l-1}} M_{a_{l-1}, a'_l}^{\sigma'_l}. \tag{2.52}
\end{aligned}$$

Voor de andere termen vinden we dan achtereenvolgens:

$$\frac{\partial}{\partial M_{a'_{l-1}, a'_l}^{\sigma'_l}} \left(\sum_{a'_{l-1}, a'_l, \sigma_l, \sigma'_l} M_{a'_{l-1}, a'_l}^{\sigma_l} M_{a'_{l-1}, a'_l}^{\sigma'_l} (\hat{O}_l)_{\sigma_l, \sigma'_l} \right) = \sum_{\sigma_l} M_{a'_{l-1}, a'_l}^{\sigma_l} (\hat{O}_l)_{\sigma_l, \sigma'_l} \tag{2.53}$$

$$\frac{\partial}{\partial M_{a'_{l-1}, a'_l}^{\sigma'_l}} \left(\sum_{a_{l-1}, a_l, a'_l, \sigma_l} (X_{l+1}^R)_{a'_l, a_l} M_{a_{l-1}, a_l}^{\sigma_l} M_{a_{l-1}, a'_l}^{\sigma'_l} \right) = \sum_{a_l} (X_{l+1}^R)_{a'_l, a_l} M_{a_{l-1}, a_l}^{\sigma'_l}, \tag{2.54}$$

$$\begin{aligned}
& \frac{\partial}{\partial M_{a'_{l-1}, a'_l}^{\sigma'_l}} \left(\sum_{a_{l-1}, a'_l, a'_{l-1}, n, \sigma_l, \sigma'_l} ((Y_{l-1}^L)_1^n)_{a'_{l-1}, a_{l-1}} M_{a_{l-1}, a'_l}^{\sigma_l} M_{a'_{l-1}, a'_l}^{\sigma'_l} (C_l^n)_{\sigma_l, \sigma'_l} \right) \\
&= \sum_{a_{l-1}, n, \sigma_l} ((Y_{l-1}^L)_1^n)_{a'_{l-1}, a_{l-1}} M_{a_{l-1}, a'_l}^{\sigma_l} (C_l^n)_{\sigma_l, \sigma'_l}, \tag{2.55}
\end{aligned}$$

$$\begin{aligned}
& \frac{\partial}{\partial M_{a'_{l-1}, a'_l}^{\sigma'_l}} \left(\sum_{\alpha=1}^{\eta} \sum_{a'_{l-1}, a_l, a'_l, n, \sigma_l, \sigma'_l} ((Y_{l+1}^R)_\alpha^n)_{a'_l, a_l} M_{a'_{l-1}, a_l}^{\sigma_l} M_{a'_{l-1}, a'_l}^{\sigma'_l} ((D_l^n)_\alpha)_{\sigma_l, \sigma'_l} \right) \\
&= \sum_{\alpha=1}^{\eta} \sum_{a_l, n, \sigma_l} ((Y_{l+1}^R)_\alpha^n)_{a'_l, a_l} M_{a'_{l-1}, a_l}^{\sigma_l} ((D_l^n)_\alpha)_{\sigma_l, \sigma'_l}, \tag{2.56}
\end{aligned}$$

$$\begin{aligned}
& \frac{\partial}{\partial M_{a'_{l-1}, a'_l}^{\sigma'_l}} \left(\sum_{\alpha=2}^{\eta} \sum_{a_{l-1}, a_l, a'_{l-1}, a'_l, n, \sigma'_l} ((Y_{l-1}^L)_\alpha^n)_{a'_{l-1}, a_{l-1}} M_{a_{l-1}, a_l}^{\sigma'_l} M_{a'_{l-1}, a'_l}^{\sigma'_l} ((Y_{l+1}^R)_{\alpha-1})_{a'_l, a_l} \right) \\
&= \sum_{\alpha=2}^{\eta} \sum_{a_{l-1}, a_l, n} ((Y_{l-1}^L)_\alpha^n)_{a'_{l-1}, a_{l-1}} M_{a_{l-1}, a_l}^{\sigma'_l} ((Y_{l+1}^R)_{\alpha-1})_{a'_l, a_l}. \tag{2.57}
\end{aligned}$$

Zorgvuldige inspectie van de termen (2.51) tot en met (2.57) toont dat de uitkomst voor elke term nog slechts afhangt van 3 indices, namelijk $\sigma'_l, a'_{l-1}, a'_l$. De sommen zorgen er immers voor

dat er steeds een aantal indices weggesommeerd worden. Als gevolg van deze vaststelling kunnen we de som van al deze termen ((2.52) tot en met (2.57)) zeer algemeen schrijven als⁴

$$\sum_{a_{l-1}, a_l, \sigma_l} (H_{eff})_{\sigma'_l, a'_{l-1}, a'_l, \sigma_l, a_{l-1}, a_l} M_{a_{l-1}, a_l}^{\sigma_l} \quad (2.58)$$

waarbij $(H_{eff})_{\sigma'_l, a'_{l-1}, a'_l, \sigma_l, a_{l-1}, a_l}$ een zogenaamde *effectieve Hamiltoniaan* voorstelt. Deze term bevat immers alle operatoren en interactietermen en kan dus gezien worden als een equivalent van de fysische Hamiltoniaan. De effectieve Hamiltoniaan is afhankelijk van de ganse MPS zonder M^{σ_l} en verandert dus elke iteratiestap (omdat we in elke stap een andere M^{σ_l} beschouwen). Men kan het ook nog opvatten als de Hamiltoniaan die site l ondervindt als gevolg van de omgeving (dus de overige $L - 1$ sites). Op die manier kunnen we het extremum-probleem in de volgende compacte gedaante brengen:

$$\sum_{a_{l-1}, a_l, \sigma_l} (H_{eff})_{\sigma'_l, a'_{l-1}, a'_l, \sigma_l, a_{l-1}, a_l} M_{a_{l-1}, a_l}^{\sigma_l} - \lambda M_{a'_{l-1}, a'_l}^{\sigma'_l} = 0. \quad (2.59)$$

Door $M_{a'_{l-1}, a'_l}^{\sigma'_l}$ nu om te vormen tot een vector $\nu_{\sigma'_l, a'_{l-1}, a'_l}$ en de effectieve Hamiltoniaan tot een matrix, is het duidelijk dat het extremum-probleem zicht herleidt tot een gewoon eigenwaardeprobleem met operator \hat{H}_{eff} en corresponderende eigenvector $|\nu\rangle$:

$$\hat{H}_{eff} |\nu\rangle = \lambda |\nu\rangle. \quad (2.60)$$

Mochten we de A - en B -matrices niet genormaliseerd hebben dan zouden we een veralgemeend eigenwaardeprobleem bekomen hebben, dat er als volgt zou uitzien:

$$\hat{H}_{eff} |\nu\rangle = \lambda \hat{N} |\nu\rangle \quad (2.61)$$

met \hat{N} een welbepaalde matrix. Computationeel is het echter veel voordeliger om een gewoon eigenwaardeprobleem op te lossen, zodat de extra stappen (in het algoritme) om de matrices steeds te normaliseren zeker de moeite zijn. Het oplossen van dit eigenwaardeprobleem (samen met de overige numerieke details) wordt uitvoerig behandeld in Appendix A.

We zijn pas nu in staat om een verklaring te geven voor de naam *VMPS methode*. Hierboven hebben we immers aangetoond dat de grondtoestand bepaald wordt door een variationele optimalisatie toe te passen op de opeenvolgende $M_{a_{l-1}, a_l}^{\sigma_l}$. Deze $M_{a_{l-1}, a_l}^{\sigma_l}$ bouwen de golf functie op zodat we met recht kunnen spreken van een MPS. De **VMPS** methode zal dus **V**ariationeel de **M**atrix **P**roduct **S**tate optimaliseren en het algoritme stopt als de energie geconvergeerd is binnen een bepaalde grens van accuraatheid. Het aantal cycli ($1 \rightarrow L$ en $L \rightarrow 1$) is dus vooraf niet gekend, maar hangt volledig af van hoe snel de energie convergeert (de definitie van convergentie wordt ook in Appendix A gegeven).

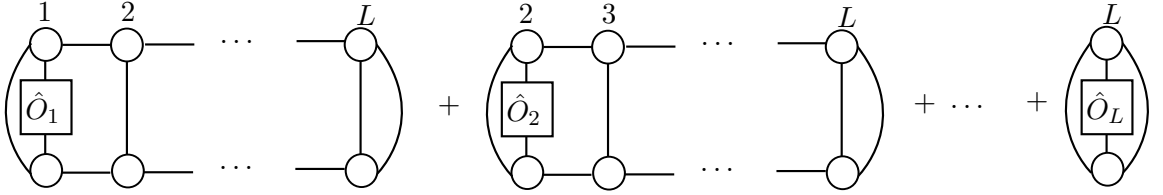
⁴Let op dat we H_{eff} vermenigvuldigen met $M_{a_{l-1}, a_l}^{\sigma_l}$. Dit is mogelijk omdat inspectie van alle termen leert dat sommige indices aan elkaar gelijk zijn (bv. $\sigma_l = \sigma'_l$) zodat we $M_{a_{l-1}, a_l}^{\sigma'_l}$ dan bijvoorbeeld kunnen vervangen door $M_{a_{l-1}, a_l}^{\sigma_l}$. Daarnaast is het ook mogelijk dat $M_{a_{l-1}, a_l}^{\sigma_l} \rightarrow M_{a_{l-1}, a_l}^{\sigma'_l}$ bekomen wordt door het kiezen van een andere dummy-index.

2.6 Verwachtingswaarde van fysische grootheden

In deze sectie beschrijven we hoe de verwachtingswaarde van een one-site operator \hat{O} kan uitgerekend worden. Een voorbeeld van een dergelijke operator is bijvoorbeeld de totale spinprojectie van een keten

$$\hat{S}^z = \sum_{i=1}^L \hat{S}_i^z. \quad (2.62)$$

De operator \hat{O} is dus opgebouwd uit lokale operatoren op elke site. Diagrammatisch kunnen we dit voorstellen als in figuur (2.12). Door de diagrammen te bestuderen kunnen we makkelijk een



Figuur 2.12: Diagrammatische voorstelling van een verwachtingswaarde

formule afleiden om een dergelijke operator te evalueren. Beschouw daarvoor eerst de tweede term uit de expansie van X^R in figuur (2.10). Deze reproduceert perfect het rechterstuk (rechts van de site waarop we de lokale operator beschouwen) als we als randconditie stellen dat $X_{L+1}^R = \mathbb{I}$ (de eenheidsmatrix). Deze gemodificeerde X^R (genoteerd als $X^{R,mod}$) wordt dus opgebouwd door als lokale operator de eenheidsmatrix te nemen op site L en vervolgens iteratief het tweede blok uit de expansie op te bouwen (de andere 2 termen uit de oorspronkelijke X^R worden dus niet meegenomen). Nu moeten we nog de lokale operator op site l in rekening brengen. Daartoe is het voldoende de eerste term uit de expansie van X^L (figuur (2.8)) in rekening te brengen, waarbij we als operator de lokale operator \hat{O}_l nemen. We noteren deze term met $X_l^{L,mod}$. Om de verwachtingswaarde uit te rekenen combineren we dus steeds een $X_l^{L,mod}$ met een $X_{l+1}^{R,mod}$. In formulevorm uitgeschreven wordt dit

$$\begin{aligned} \langle \psi | \sum_{l=1}^L \hat{O}_l | \psi \rangle &= \sum_{l=1}^L \sum_{a_l, a'_l} \left(X_l^{L,mod} \right)_{a'_l, a_l} \left(X_{l+1}^{R,mod} \right)_{a'_l, a_l} \\ &= \sum_{l=1}^L \sum_{a'_l} \left(X_l^{L,mod} \left(X_{l+1}^{R,mod} \right)^T \right)_{a'_l, a'_l} \\ &= \sum_{l=1}^L \text{Tr} \left(X_l^{L,mod} \left(X_{l+1}^{R,mod} \right)^T \right) \end{aligned}$$

met de randvoorwaarde $X_{L+1}^R = \mathbb{I}$. In het vervolg zullen we alle one-site operatoren op die manier uitwerken.

2.7 Verstrelingsentropie

Alvorens we de convergentie bestuderen, zullen we eerst nog een belangrijke grootheid invoeren: de verstrelingsentropie. Er blijkt namelijk een verband te bestaan tussen de grootte van de

verstrelingsentropie en de grootte van de truncatiedimensie D .

Verstreling komt tot uiting in kwantumcorrelaties tussen gescheiden systemen. Wij zijn in het bijzonder geïnteresseerd in hoe een subrooster met l sites ($l < L$) verstrengeld is met de overige $L - l$ sites van een spinrooster met L spins⁵. Een maat voor de verstreling tussen deze twee systemen wordt gegeven door de von Neumann entropie of verstrelingsentropie ([14]):

$$S = -\text{Tr}_l(\rho_l \log_2 \rho_l). \quad (2.63)$$

Hierbij is ρ_l de dichtheidsmatrix van het blok met l spins:

$$\rho_l = \text{Tr}_R |\psi\rangle \langle \psi|$$

waarbij het spoor Tr_l genomen wordt over alle toestanden $\{|\sigma_i\rangle, i = 1 \dots l\}$ van het subblok met l spins (als de $\{|\sigma_l\rangle\}$ een lokale basis vormen). Het spoor Tr_R wordt dan genomen over het complementaire blok met de overige $L - l$ spins $\{|\sigma_i\rangle, i = l + 1 \dots L\}$. De golf functie $|\psi\rangle$ is de totale golf functie van het *volledige* spinrooster. De verstrelingsentropie zal dus nul worden als er geen kwantumcorrelatie bestaat tussen de twee subsystemen en maximaal worden als de kwantumcorrelatie maximaal is. In het algemeen convergeert de verstrelingsentropie naar een eindige waarde als de lengte van het subsysteem vergroot. Er treedt echter divergent gedrag op als het systeem een kritisch punt nadert ([15]). In hoofdstuk 1 hebben we reeds vermeld dat de correlatielengte divergeert bij de kritische temperatuur. Op dat moment zal de entropie logaritmisches toenemen als functie van de lengte van het subsysteem ([15]). We zullen voor 2 spinsystemen (isotroop Heisenberg model en $J_1 - J_2$ model) de verstrelingsentropie bestuderen en eventuele implicaties vermelden. We moeten echter een manier vinden om de entropie te berekenen uit de MPS; hiervoor gebruiken we de methode zoals beschreven ([4]). Eerst en vooral moeten we een SVD-decompositie toepassen op de golf functie. Een SVD-decompositie van een $(m \times n)$ -matrix M ontbindt de matrix M volgens:

$$M = USV^\dagger \quad (2.64)$$

waarbij U een $(m \times \min(m, n))$ -matrix is met $U^\dagger U = I$ (U is unitair als $m < n$) en S een $(\min(m, n) \times \min(m, n))$ diagonaal matrix met r niet-negatieve diagonaalelementen (de singuliere waarden); r is de rang van de matrix M . Er wordt verondersteld dat de diagonaalelementen gerangschikt worden in dalende volgorde: $s_1 \geq s_2 \geq \dots \geq s_r > 0$. Tot slot is V^\dagger een $(\min(m, n) \times n)$ -matrix met $V^\dagger V = I$ (V is unitair als $m > n$). Met behulp van een SVD kunnen we de golf functie nu in een gemengde canonische vorm brengen. Stel dat we een SVD-decompositie hebben toegepast op de eerste l sites, dan wordt de golf functie:

$$|\psi\rangle = \sum_{\sigma} \sum_{a_l} (A^{\sigma_1} \dots A^{\sigma_l})_{1, a_l} S_{a_l, a_l} (V^\dagger)_{a_l, (\sigma_{l+1} \dots \sigma_L)} |\sigma_1 \dots \sigma_L\rangle.$$

Vervolgens herschalen we V^\dagger tot $\Psi_{(a_l \sigma_{l+1} \dots \sigma_{L-1}), \sigma_L}$ en passen we opnieuw SVD's toe, maar nu vanaf rechts. Op die manier bekomen we

$$V_{a_l, (\sigma_{l+1} \dots \sigma_L)}^\dagger = \sum_{a_{l+1}, \dots, a_{L-1}} B_{a_l, a_{l+1}}^{\sigma_{l+1}} \dots B_{a_{L-1}, 1}^{\sigma_L}.$$

⁵De verstrelingsentropie kan echter ook berekend worden voor fermionische systemen.

Als gevolg van de SVD-decompositie kunnen we nu 2 orthonormale toestanden

$$|a_l\rangle_l = \sum_{\sigma_1 \dots \sigma_l} (A^{\sigma_1} \dots A^{\sigma_l})_{1,a_l} |\sigma_1 \dots \sigma_l\rangle$$

$$|a_l\rangle_R = \sum_{\sigma_{l+1} \dots \sigma_L} (B^{\sigma_{l+1}} \dots B^{\sigma_L})_{a_l,1} |\sigma_{l+1} \dots \sigma_L\rangle$$

invoeren om de golffunctie uiteindelijk te schrijven als

$$|\psi\rangle = \sum_{a_l} S_{a_l, a_l} |a_l\rangle_l |a_l\rangle_R.$$

Op die manier kunnen we de dichtheidsmatrix ρ_l van het blok met l spins makkelijk bepalen. We sommeren alle toestanden in het rechterdeel R weg en vinden⁶

$$\rho_l = \sum_{a_l} S_{a_l, a_l}^2 |a_l\rangle \langle a_l|.$$

Vermits ρ_l diagonaal is in de $\{|a_l\rangle\}$ basis wordt de entropie (2.63) uiteindelijk gegeven door

$$S = - \sum_{a_l} S_{a_l, a_l}^2 \log_2 S_{a_l, a_l}^2.$$

Op die manier kunnen we uit de MPS de verstrengelingsentropie berekenen. We vermelden voor de volledigheid dat indien we zouden vertrekken van een golffunctie die volledig uit rechts-genormaliseerde matrices M^{σ_i} bestaat

$$|\psi\rangle = \sum_{\sigma} M^{\sigma_1} \dots M^{\sigma_L} |\sigma_1 \dots \sigma_L\rangle$$

we voor de verstrengelingsentropie ook een gesloten gedaante vinden, namelijk:

$$S_l = - \sum_{\sigma_1 \dots \sigma_l} \Lambda_{\sigma_1 \dots \sigma_l} \log_2 \Lambda_{\sigma_1 \dots \sigma_l}$$

met

$$\Lambda_{\sigma_1 \dots \sigma_l} = M^{\sigma_1} \dots M^{\sigma_l} M^{\sigma_l \dagger} \dots M^{\sigma_1 \dagger} \in \mathbb{R}.$$

Het aantal termen dat we echter moeten berekenen stijgt exponentieel, immers als we l $|\sigma_i\rangle$ hebben, dan zijn er d^l combinaties (met d de fysische dimensie). Dit is dus niet haalbaar als we de entropie van grotere systemen willen bekijken. In wat volgt zullen we dus steeds de formule bekomen via de SVD gebruiken om de entropie te berekenen. Meer informatie over verstrengelingsentropie in veeldeeltjessystemen kan gevonden worden in de volgende review-artikels ([16],[17]).

2.8 Convergentie van de VMPS methode

Tot slot bestuderen we de convergentie van de VMPS methode voor verschillende systemen. We zullen alle systemen hier enkel definiëren zonder verdere uitleg aangezien ze uitvoerig zullen behandeld worden in volgend hoofdstuk. We starten eerst met een algemene opmerking omtrent

⁶De buitenindex l in $|a_l\rangle_l$ wordt ook weggelaten.

de keuze van de truncatiedimensie D . Stel daarvoor dat $|\psi_{exact}\rangle$ de exacte grondtoestand is van ons systeem met lengte $2L$ (let op de 2) en $|\psi_D\rangle$ de golf functie bij een gegeven waarde D . Kies daarnaast een waarde ϵ_0 onafhankelijk van L . In ([5]) wordt dan aangetoond dat de minimale waarde van D die men nodig heeft om

$$\| |\psi_{exact}\rangle \| - \| |\psi_D\rangle \| \leq \frac{\epsilon_0}{L} \quad (2.65)$$

te bekomen, gegeven wordt door

$$D_L \leq \beta L^\gamma \quad (2.66)$$

waarbij β en $\gamma \in \mathbb{R}$ een functie zijn van onder andere ϵ_0 maar niet van L . De macht γ is evenredig met 2^{S^α} waarbij S^α de Renyi entropie voorstelt (dit is een veralgemening van de von Neumann entropie $S = \lim_{\alpha \rightarrow 1} S^\alpha$ ([18])). Een hogere entropie eist dus een grotere waarde van D voor een gegeven ϵ_0 . In het volgend hoofdstuk zullen we ook zien dat zwak verstrengelde systemen (gekaracteriseerd door een kleine verstrengelingsentropie) reeds efficiënt gesimuleerd kunnen worden met een kleine D . In heel speciale gevallen zal een zeer kleine D volstaan om zeer grote systemen te beschrijven. Wanneer de verstrengeling toeneemt, dan zal een grotere D noodzakelijk zijn. Verder impliceert (2.66) dat het voldoende is om D polynomiaal te schalen om de accuraatheid constant te houden. Het is dus niet nodig om $D = d^L$ te stellen aangezien we reeds bij een polynoom van L een goed resultaat kunnen bekomen. Daarnaast vinden we in ([19]) dat het volgend verband bestaat tussen het energieverval $E_D - E_{conv}$ (met E_D de energie bij een gegeven D en E_{conv} de energie bij convergentie) en D

$$\log(E_D - E_{conv}) \cong -\kappa (\log D)^2, \quad \kappa \in \mathbb{R}. \quad (2.67)$$

Een maat voor de snelheid van convergentie is dan de richtingscoëfficiënt κ . Hoe groter κ , hoe sneller er convergentie zal optreden. We zullen nu een aantal modelsystemen en hun convergentie bestuderen. We merken nog op dat we met log het logaritme met basis 10 bedoelen (dus \log_{10}).

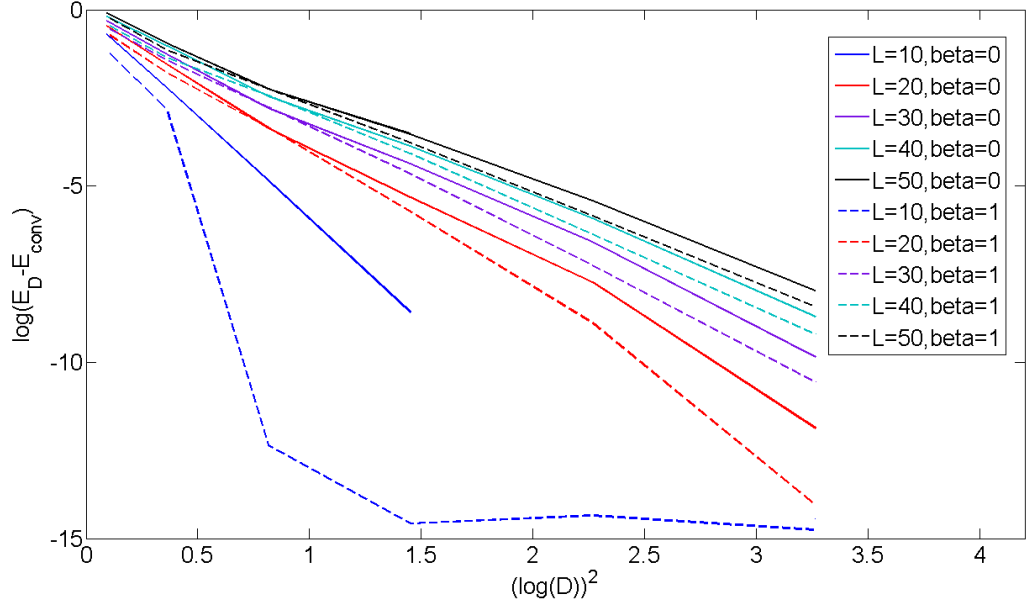
2.8.1 Dichtste-nabuur Hamiltonianen

Het Heisenbergmodel

We bestuderen eerst het (isotroop) Heisenbergmodel (met $J = 1$ en fysische dimensie $d = 2$) ([13])

$$\hat{H}_{Heisenberg} = J \sum_{i=1}^{L-1} \hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_{i+1} - \beta \sum_{i=1}^L \hat{S}_i^z \quad (2.68)$$

voor 5 verschillende lengtes L gaande van 10 tot en met 50 in stapjes van 10 en dit zowel zonder magnetisch veld ($\beta = 0$) als met een magnetisch veld ($\beta = 1$). We geven het energieverval $\log(E_D - E_{conv})$ weer als functie van $\log(D)^2$. Het resultaat is te vinden in figuur (2.13). Het is duidelijk dat het verschil $E_D - E_{conv}$ zeer snel daalt als functie van D (zowel zonder als met magnetisch veld). De fout is voor alle geteste lengtes $L = 10 \rightarrow 50$ reeds kleiner dan $O(10^{-5})$ als $D = 64$. Daarnaast merken we op dat reeds bij $D = 8$ een vrij accuraat resultaat kan bekomen worden voor $L = 10 \rightarrow 50$. Als we de gevallen met $L = 10$ buiten beschouwing laten (zeer snelle convergentie als gevolg van de initieel zeer steile helling) dan vinden we dat κ ongeveer gelegen is in het interval [2.4, 4.5]. Deze waarden werden bepaald door de staart van elke curve



Figuur 2.13: Convergentiestudie van het isotroop Heisenbergmodel, $J = 1$, energie in eenheden $\hbar = 1$.

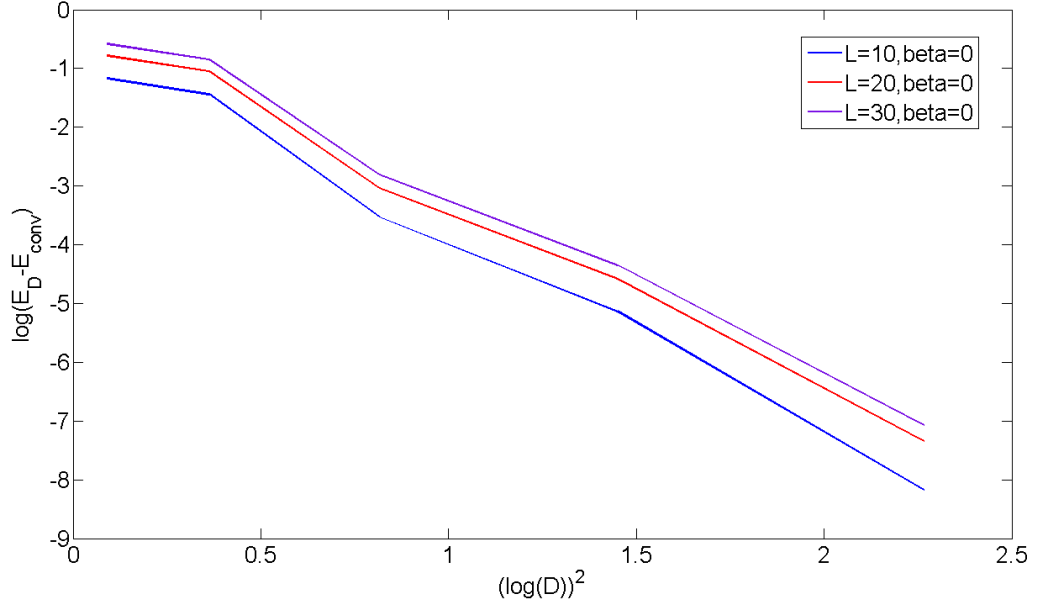
(overeenkomend met de grote waarden van D) te bekijken vermits die lineair is (zoals voorspeld door formule (2.67)). Als we enkel $L = 20$ en $L = 30$ (met $\beta = 0$) bekijken (om later te vergelijken met andere modellen) dan vinden we het interval $[2.9, 4.25]$.

Het bilinear biquadratic model

Het bilinear biquadratic (BB) model ([20]) wordt beschreven door de volgende Hamiltoniaan:

$$H_{BB} = J \sum_i \left(\cos \theta \left(\hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_{i+1} \right) + \sin \theta \left(\hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_{i+1} \right)^2 \right) - \beta \sum_i \hat{S}_i^z \quad (2.69)$$

met $d = 3$ voor alle spins $i = 2 \dots L - 1$ en $d = 2$ voor de spins op site 1 en L (zie volgend hoofdstuk). Ook voor het BB model bekijken we de convergentie voor 3 lengtes van de kettingen ($L = 10, 20, 30$ en $J = 1$) en $\theta = \arctan\left(\frac{1}{3}\right) + 0.1$ (dus voor een waarde die afwijkt van de waarde waarvoor een exact resultaat gekend is, zie volgend hoofdstuk). Er werd geen magnetisch veld aangelegd. Het resultaat kan men vinden in figuur (2.14). We bepaalden ook hier het κ -interval (op dezelfde wijze als bij het Heisenbergmodel) en vonden $[3.43, 3.70]$. Als we ons beperken tot $L = 20$ en $L = 30$ dan vinden we één enkele κ waarde gelijk aan 3.43. De convergentie is dus sneller dan in het Heisenbergmodel voor $L = 30$, maar trager voor $L = 20$. We merken op dat deze convergentieplot geldt voor 1 specifieke θ . Andere waarden van θ zullen logischerwijs ander convergentiegedrag vertonen. Stel bijvoorbeeld dat $\theta = 0$, dan vinden we gewoonweg het isotroop Heisenbergmodel terug. In het volgend hoofdstuk zullen we zien dat voor $\theta = \arctan\left(\frac{1}{3}\right)$ het volstaat om $D = 2$ te kiezen. Naarmate θ afwijkt van dit punt, zal een andere waarde van D nodig zijn. De truncatiedimensie zal dus fluctueren in functie van θ .



Figuur 2.14: Convergentiestudie van het BB model, $\theta = \arctan\left(\frac{1}{3}\right) + 0.1$, $J = 1$, energie in eenheden $\hbar = 1$.

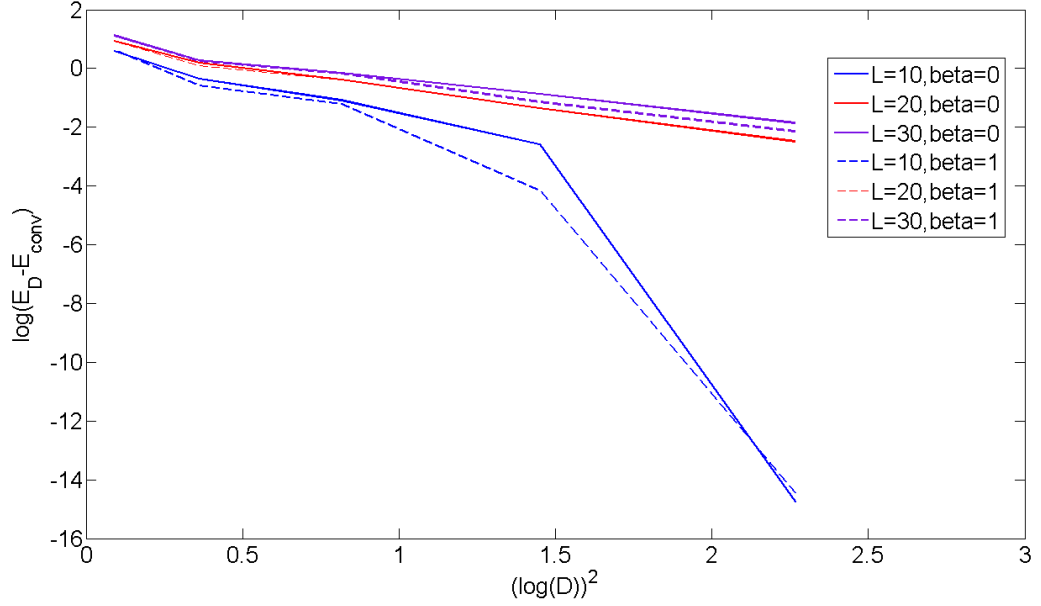
2.8.2 Het $J_1 - J_2$ model

We wensen nu ook te kijken wat de invloed is van de interactie-range η op de convergentie. Daartoe bestuderen we de convergentie (op dezelfde manier als we deden voor het isotroop Heisenbergmodel) van het $J_1 - J_2$ model (een model met tweede-orde interacties). Hogere-orde spinsystemen (met $\eta \geq 3$) werden niet bestudeerd. Het $J_1 - J_2$ model ([21]) wordt bepaald door de volgende Hamiltoniaan ($d = 2$):

$$\hat{H}_{J_1 - J_2} = J_1 \sum_{i=1}^{L-1} \hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_{i+1} + J_2 \sum_{i=1}^{L-2} \hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_{i+2} - \beta \sum_{i=1}^L \hat{S}_i^z. \quad (2.70)$$

Om het effect op de convergentie na te gaan stellen we $J_1 = 1$ en $J_2 = 4$ zodat het tweede-orde karakter domineert op het eerste-orde karakter. We bestuderen 3 kettingen met lengte $L = 10, 20, 30$ zonder magnetisch veld ($\beta = 0$) en met magnetisch veld ($\beta = 1$). Het resultaat is te vinden in figuur (2.15). Als we kijken naar de resultaten zonder magnetisch veld dan vinden we één enkele κ -waarde van 1.23. In figuur (2.16) bestuderen we de convergentie zonder magnetisch veld voor dezelfde lengtes maar dit keer met $J_1 = 4$ en $J_2 = 1$. Beperken we ons tot $L = 20$ en $L = 30$ (met $\beta = 0$) dan blijkt κ in het interval $[0.8334, 0.84]$ te liggen. In vergelijking met het geval $J_1 = 1$ en $J_2 = 4$ gebeurt de convergentie dus minder snel, want κ is kleiner. Daarnaast valt het ook op dat in beide gevallen κ kleiner is dan in het Heisenberg of BB model (voor de geteste L).

Bovendien moet men voorzichtig zijn in het kiezen van een bepaalde D . De parameters J_1 en J_2 spelen hierin een belangrijke rol (de parameter θ vervult een analoge rol in het BB model). De snelheid van convergentie is immers afhankelijk van de verhouding $\frac{J_2}{J_1}$ zoals te zien is in figuur (2.17) waar de convergentie is weergegeven voor verschillende $\frac{J_2}{J_1}$ verhoudingen (met verder

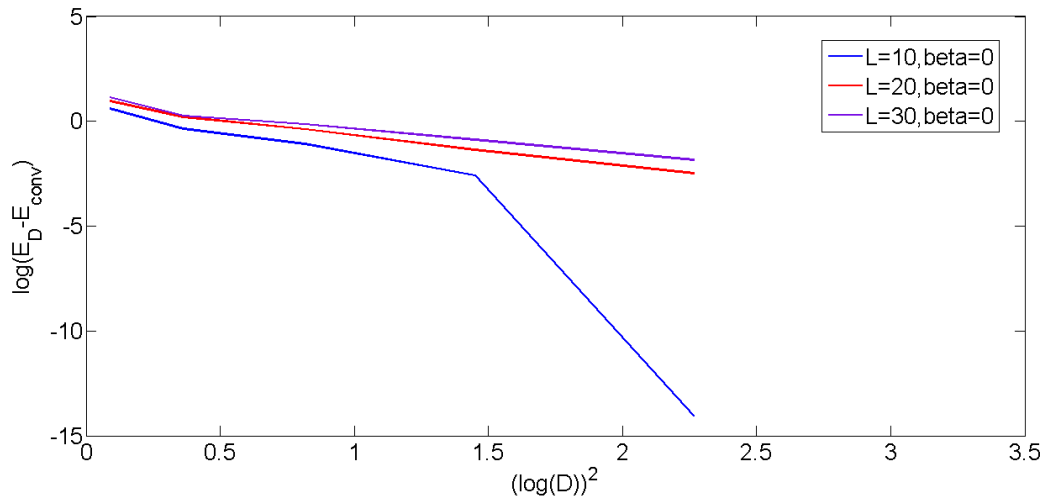


Figuur 2.15: Convergentiestudie van het $J_1 - J_2$ model, $J_1 = 1$ en $J_2 = 4$, energie in eenheden $\hbar = 1$.

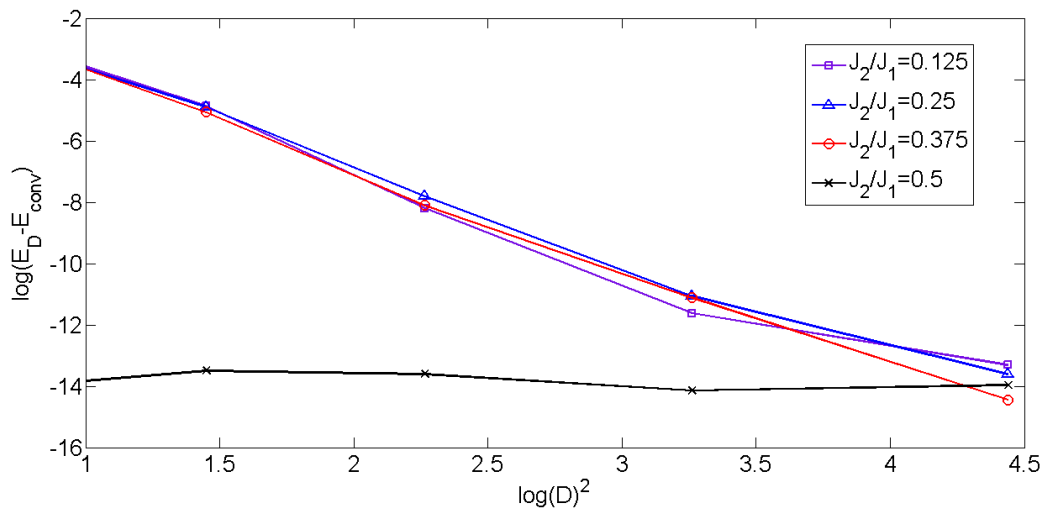
$L = 20$, $\beta = 0$, $J_1 = 4$). Het is duidelijk dat voor $\frac{J_2}{J_1} = \frac{1}{2}$ er al convergentie is voor $D = 2$ (de ietwat grillige vorm wijten we aan numerieke fluctuaties omdat de waarden reeds zeer klein zijn en dus in de buurt komen van de numerieke accuraatheid). Het geval $\frac{J_2}{J_1} = \frac{1}{2}$ wordt goed beschreven door $D = 2$ omdat er in dit geval een dimerisatie van de grondtoestand optreedt (zie hoofdstuk 3).

2.8.3 Conclusies met betrekking tot de convergentie

De Hamiltoniaan geeft aanleiding tot een bepaalde grondtoestand die gekarakteriseerd wordt door een zekere verstrengelingsentropie $S_E(l) \triangleq S_E$ (met l de lengte van een subblok, $l = 1 \dots L - 1$). De waarde van D die nodig is om een systeem efficiënt te simuleren hangt samen met deze S_E . Voor systemen met (zeer) kleine S_E volstaat een kleine waarde van D , zelfs al is de interactie full range (een aantal specifieke gevallen worden besproken in volgend hoofdstuk; bijvoorbeeld het exact oplosbaar $J_1 - J_2$ model). De parameters van het model (bijvoorbeeld J_1, J_2 voor het $J_1 - J_2$ model) spelen hierin een belangrijke rol, omdat voor bepaalde waarden van de parameters de entropie zeer klein wordt. Daarnaast wordt de snelheid van de convergentie bepaald door de richtingscoëfficiënt κ uit formule (2.67). Hoe groter κ , hoe sneller er convergentie optreedt.



Figuur 2.16: Convergentiestudie van het $J_1 - J_2$ model, $J_1 = 4$ en $J_2 = 1$, energie in eenheden $\hbar = 1$.



Figuur 2.17: Convergentiestudie van het $J_1 - J_2$ model, $L = 20$, $\beta = 0$, $J_1 = 4$ en variabele J_2 , energie in eenheden $\hbar = 1$.

Hoofdstuk 3

Spinsystemen

In dit hoofdstuk zullen we een aantal belangrijke spinsystemen bestuderen met behulp van het VMPS-algoritme en waar mogelijk de bekomen resultaten vergelijken met analytische uitdrukkingen om de accuraatheid van de methode te testen. Spinsystemen spelen een essentiële rol in vele fysische processen. De verklaring van magnetisatie vindt bijvoorbeeld zijn oorsprong in de spin en ook in supergeleiding speelt spin een belangrijke rol. In meer recent onderzoek is de spin ook een kandidaat informatiedrager in de kwantuminformatietheorie en kwantumcomputers.

3.1 Het isotroop Heisenbergmodel

3.1.1 Het isotroop Heisenbergmodel voor $L = 2$

In het geval we een interagerend systeem van slechts twee $\frac{1}{2}$ -spins bestuderen, kunnen we de resultaten vergelijken met een eenvoudige analytische uitdrukking (die we hieronder afleiden). We beschouwen daartoe een Hamiltoniaan van de vorm ([13])

$$\hat{H}_{L=2} = J\hat{\mathbf{S}}_1 \circ \hat{\mathbf{S}}_2 - \beta \left(\hat{S}_1^z + \hat{S}_2^z \right). \quad (3.1)$$

Dit is een isotrope Heisenberg Hamiltoniaan met koppelconstante J in een magnetisch veld β . Om een dergelijk systeem analytisch op te lossen gaan we over naar de gekoppelde representatie waarbij we de 2 spin-operatoren $\hat{\mathbf{S}}_1$ en $\hat{\mathbf{S}}_2$ koppelen tot 1 spin-operator $\hat{\mathbf{S}}$

$$\hat{\mathbf{S}} = \hat{\mathbf{S}}_1 + \hat{\mathbf{S}}_2. \quad (3.2)$$

De grootte van de spins van $\hat{\mathbf{S}}$ is dus $s = 0$ of $s = 1$ wegens de driehoeksrelatie

$$|s_1 - s_2| \leq s \leq |s_1 + s_2| \quad (3.3)$$

met $s_1 = s_2 = \frac{1}{2}$. In de gekoppelde representatie beschouwen we golffuncties van de vorm $|sm_s\rangle$ (m_s kan in dit geval de waarden $-1, 0, 1$ aannemen) met als eigenschappen

$$\begin{aligned} \hat{\mathbf{S}}^2 |sm_s\rangle &= \hbar^2 s(s+1) |sm_s\rangle \\ \hat{S}^z |sm_s\rangle &= \left(\hat{S}_1^z + \hat{S}_2^z \right) |sm_s\rangle = \hbar m_s |sm_s\rangle \end{aligned}$$

en analoog voor de operatoren $\hat{\mathbf{S}}_1$ en $\hat{\mathbf{S}}_2$. Daarnaast dienen we ook de Hamiltoniaan in een gekoppelde representatie te brengen, wat in dit geval eenvoudig gebeurt via

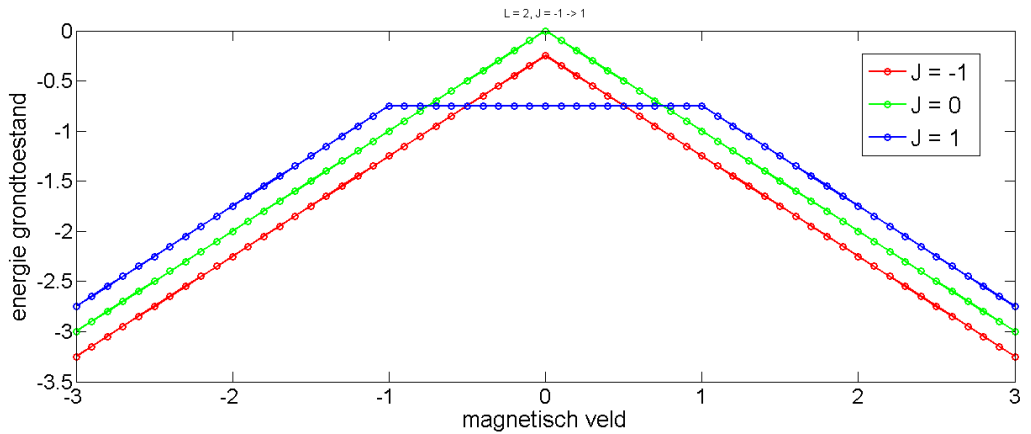
$$\hat{\mathbf{S}}_1 \circ \hat{\mathbf{S}}_2 = \frac{1}{2} \left(\hat{\mathbf{S}}^2 - \hat{\mathbf{S}}_1^2 - \hat{\mathbf{S}}_2^2 \right) \quad (3.4)$$

aangezien de operatoren $\hat{\mathbf{S}}_1$ en $\hat{\mathbf{S}}_2$ commuteren. We kunnen nu makkelijk de eigenwaarden afleiden uit volgende vergelijking

$$\begin{aligned}
\hat{H}_{L=2} |sm_s\rangle &= \left(\frac{J}{2} (\hat{\mathbf{S}}^2 - \hat{\mathbf{S}}_1^2 - \hat{\mathbf{S}}_2^2) - \hbar\beta m_s \right) |sm_s\rangle \\
&= \left(\frac{J\hbar^2}{2} (s(s+1) - s_1(s_1+1) - s_2(s_2+1)) - \hbar\beta m_s \right) |sm_s\rangle \\
&= \left(\frac{J\hbar^2}{2} \left(s(s+1) - \frac{3}{2} \right) - \hbar\beta m_s \right) |sm_s\rangle.
\end{aligned} \tag{3.5}$$

We bestuderen eerst het geval zonder magnetisch veld ($\beta = 0$). Voor $s = 0$ vinden we $E_{s=0} = -\frac{3J\hbar^2}{4}$ en voor $s = 1$ vinden we $E_{s=1} = \frac{J\hbar^2}{4}$. Hieruit kunnen we besluiten dat als $J > 0$ de energie behorend bij $s = 0$ bevoordeeld wordt terwijl bij $J < 0$ de toestand met $s = 1$ de laagste energie bezit. Fysisch gezien geeft $J > 0$ dus aanleiding tot een anti-parallelle ophijning van de spins terwijl bij $J < 0$ de spins parallel zullen ophijnen. Dit is logisch aangezien $J > 0$ betekent dat de termen $\hat{\mathbf{S}}_1$ en $\hat{\mathbf{S}}_2$ een tegengesteld teken moeten hebben om een negatieve contributie in de Hamiltoniaan te geven. Op zijn beurt betekent dit dat de spinoriëntaties tegengesteld zullen zijn. Dit model werd ook ingegeven in het VMPS-programma met als input $L = 2$, $d = 2$ en $D = 2$ (de methode moet dus in principe het exacte resultaat teruggeven). We simuleerden beide gevallen ($J > 0$ en $J < 0$) en vonden het exacte resultaat terug (zowel voor $J > 0$ als voor $J < 0$).

Vervolgens brengen we het systeem in een magnetisch veld en kijken welke invloed het veld uitoefent op de grondtoestandsenergie. Het resultaat van deze simulatie wordt weergegeven in figuur (3.1). Net als in de vorige simulatie nemen we $D = 2$. Als extra laten we het magnetisch veld variëren en kijken we naar 3 gevallen: $J = -1$, $J = 0$, $J = 1$. Merk op dat het programma de resultaten weergeeft in natuurlijke eenheden ($\hbar = 1$), zodat we in de hierop volgende bespreking ook $\hbar = 1$ zullen stellen. In een magnetisch veld wordt de grondtoestandsenergie voor een



Figuur 3.1: Grondtoestandsenergie als functie van het magnetisch veld, $\hbar = 1$.

anti-parallelle ($s = 0$) respectievelijk parallelle ($s = 1$) ophijning gegeven door

$$s = 0 : \quad -\frac{3J}{4} \quad (3.6)$$

$$s = 1 : \quad \frac{J}{4} - \beta m_s \quad (3.7)$$

In het geval dat $J < 0$ zal (3.6) steeds positief zijn wat betekent dat het nooit de grondtoestand E_0 van ons systeem kan zijn. In dat geval verloopt E_0 lineair in functie van het magnetisch veld. Zonder magneetveld is $E_0 = \frac{J}{4}$ en wanneer we nu een $\beta > 0$ aanleggen zal de energie dalen als het systeem in een $m_s = 1$ toestand zit (de spins lijnen zich dus op met het veld). Omgekeerd zal bij $\beta < 0$ de energie dalen als $m_s = -1$ (de spins volgen dus opnieuw het magnetisch veld). Indien $J = 0$ hebben we dezelfde situatie als hiervoor met dat verschil dat $E_0 = 0$ bij afwezigheid van een veld. Wanneer nu echter $J > 0$ dan is het duidelijk dat $s = 0$ een lagere energie heeft dan de $s = 1$ toestand bij afwezigheid van een veld. Stel dat we nu $\beta > 0$ aanleggen dan blijft het systeem in de toestand $E_0 = \frac{-3J}{4}$ zolang $\frac{J}{4} - \beta > -\frac{3J}{4}$ (indien $J > 0, \beta > 0$ zal enkel de toestand $m_s = 1$ de energie van de $s = 1$ toestand kunnen verlagen). Indien $\beta < 0$ dan blijft $s = 0$ stabiel zolang $\frac{J}{4} + \beta > -\frac{3J}{4}$. We verwachten dus dat zolang $-J < \beta < J$ de grondtoestandsenergie onafhankelijk zal zijn van β (immers, dan is (3.6) lager in energie dan (3.7) en (3.6) is onafhankelijk van β). Wanneer we één van de knikpunten $|\beta| = J$ bereiken zal E_0 wel lineair afhankelijk worden van β volgens (3.7). Vergelijken we de numerieke resultaten uit figuur (3.1) met deze bespreking dan vinden we opnieuw een perfecte overeenkomst.

3.1.2 Het isotroop Heisenbergmodel voor $L > 2$

In de vorige sectie hebben we reeds kennisgemaakt met dit model, maar we wensen nu de meer algemene gedaante van de Hamiltoniaan te bekijken. Het isotrope Heisenbergmodel voor een ketting met lengte L ([13]) wordt gegeven door

$$\begin{aligned} \hat{H} &= J \sum_{i=1}^{L-1} \hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_{i+1} - \beta \sum_{i=1}^L \hat{S}_i^z \\ &= J \sum_{i=1}^{L-1} \left(\hat{S}_i^x \hat{S}_{i+1}^x + \hat{S}_i^y \hat{S}_{i+1}^y + \hat{S}_i^z \hat{S}_{i+1}^z \right) - \beta \sum_{i=1}^L \hat{S}_i^z. \end{aligned} \quad (3.8)$$

We zullen deze Hamiltoniaan echter schrijven in functie van de ladderoperatoren \hat{S}_i^+ en \hat{S}_i^- (om met reële matrices te kunnen werken), die men bekomt door de volgende substituties door te voeren:

$$\begin{aligned} \hat{S}^x &= \frac{1}{2} \left(\hat{S}^+ + \hat{S}^- \right) \\ \hat{S}^y &= \frac{1}{2i} \left(\hat{S}^+ - \hat{S}^- \right). \end{aligned}$$

De ladderoperatoren hebben als eigenschap $\hat{S}^\pm |sm_s\rangle = \sqrt{(s \mp m_s)(s \pm m_s + 1)} |s(m_s \pm 1)\rangle$. Op die manier kunnen we de Hamiltoniaan in de algemene vorm brengen die we ook in de code zullen implementeren:

$$\hat{H} = \frac{J}{2} \sum_{i=1}^{L-1} \left(\hat{S}_i^+ \hat{S}_{i+1}^- + \hat{S}_i^- \hat{S}_{i+1}^+ + 2\hat{S}_i^z \hat{S}_{i+1}^z \right) - \beta \sum_{i=1}^L \hat{S}_i^z. \quad (3.9)$$

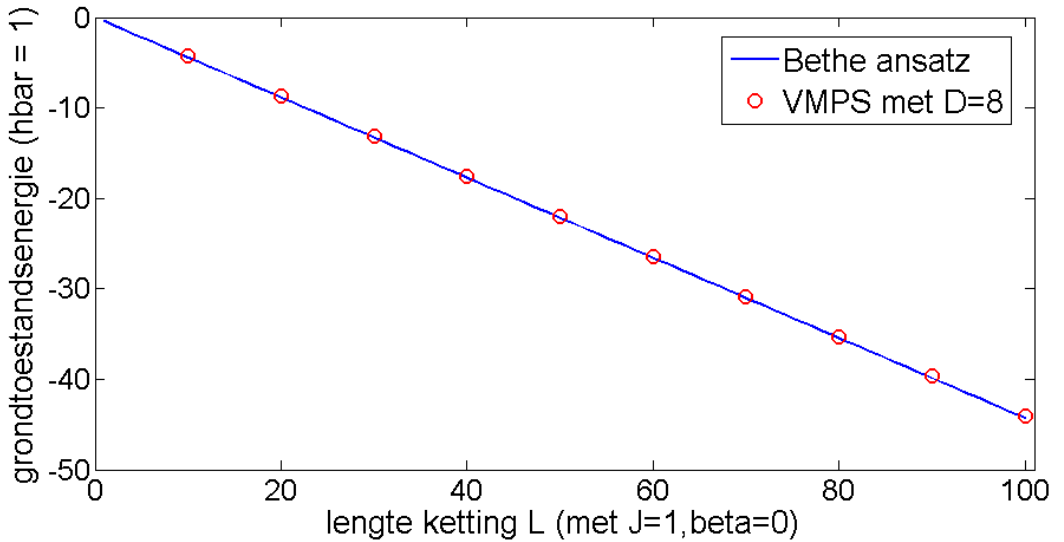
Hoe deze Hamiltoniaan in verband staat met de algemene vorm (2.19) werd reeds besproken in hoofdstuk 2 (de interactierange is hier $\eta = 1$). De inputfile wordt in Appendix C gegeven.

De Bethe ansatz

Het 1D Heisenbergmodel voor spin- $\frac{1}{2}$ systemen kan ook analytisch opgelost worden door gebruik van de Bethe ansatz ([22]). In ([22]) vinden we dat de grondtoestandsenergie (in de afwezigheid van een magnetisch veld) asymptotisch ($L \rightarrow \infty$) gegeven wordt door

$$E_L = JL \left(\frac{1}{4} - \ln 2 \right). \quad (3.10)$$

Deze uitdrukking vormt een uitermate handige benchmark voor het VMPS-programma. We simuleren een Heisenbergketen met $J = 1$, $\beta = 0$ en truncatiedimensie $D = 8$ en vergelijken de Bethe ansatz met de resultaten van het programma (waarin we L variëren van 10 tot 100 in stapjes van 10). Het resultaat wordt weergegeven in figuur (3.2). We zien dat beide in

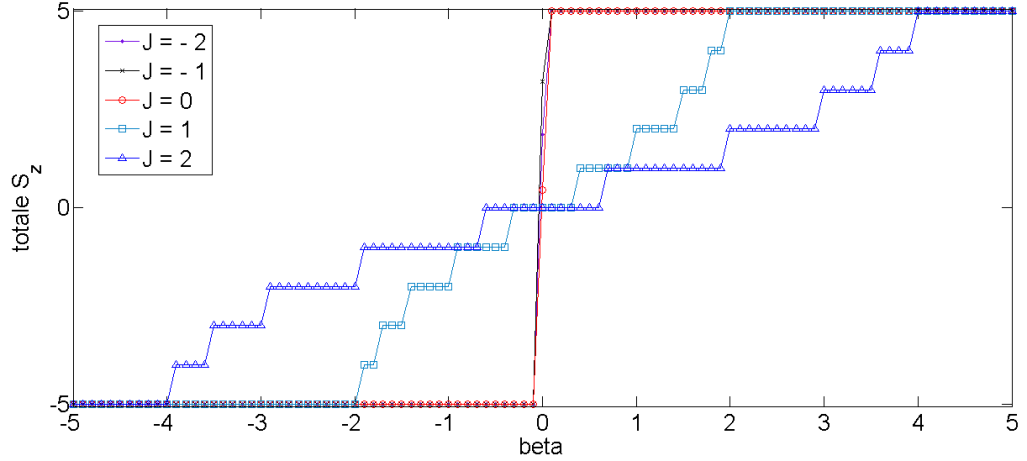


Figuur 3.2: Vergelijking tussen Bethe ansatz en VMPS voor $D = 8$, $J = 1$, $\beta = 0$.

zeer goede overeenkomst zijn maar dienen wel op te merken dat de analytische uitdrukking enkel asymptotisch (voor grote L) geldig is. Zo vinden we voor $L = 2$ een waarde $E_{Bethe} = 2J \left(\frac{1}{4} - \ln 2 \right) \approx -0.89J$ terwijl het exacte resultaat (uit angulaire-momentum algebra) $E_{exact} = \frac{-3J}{4}$ is. Voor $L = 2$ geeft de Bethe ansatz een grondtoestandsenergie die lager is dan de exacte ($-0.89J$ tegenover $-0.75J$). Meer informatie over de Bethe ansatz met betrekking tot het Heisenbergmodel is ook te vinden in ([13]) en ([23]).

Effect van een magnetisch veld

Daarnaast wensen we ook de invloed van een magnetisch veld na te gaan op de spinprojectie (voor $L = 10$). We verwachten (in lijn met het geval voor $L = 2$) dat steeds meer spins zullen oplijnen met het veld als we de veldsterkte verhogen. De resultaten van deze simulatie (met $D = 32$) worden weergegeven in figuur (3.3). Voor een voldoende sterk magnetisch veld



Figuur 3.3: Totale spinprojectie in functie van het magnetisch veld ($\hbar = 1$), $L = 10$, $D = 32$.

zullen alle spins (10 in totaal) ophijnen zodat we inderdaad een spinprojectie ± 5 verwachten (in eenheden $\hbar = 1$). Als er geen koppeling is tussen de spins ($J = 0$) dan zullen alle spins het veld onmiddellijk volgen wat duidelijk te zien is in de stapvorm van de functie bij $J = 0$. Als er positieve koppeling bestaat tussen de spins ($J > 0$) dan zal de spinprojectie niet onmiddellijk veranderen aangezien het veld een zekere barrière moet overwinnen om de interactie tussen de spins tegen te werken. Ook dit is duidelijk te zien op de figuur aangezien de spinprojectie stapsgewijs verandert. Op de figuur zien we ook dat van zodra $|\beta| = 2J$ alle spins het veld zullen volgen. We kunnen hieruit besluiten dat totale magnetisatie optreedt bij een magnetisch veld $|\beta| = 2J$. Als $J < 0$ dan zullen de spins bij voorkeur reeds opgelijnd zijn met elkaar (zelfs zonder magnetisch veld). Enkel de richting van de magnetisatie kan nog veranderd worden door het magnetisch veld. Ook dit vinden we terug op de figuur aangezien voor $J < 0$ de spinprojectie geen trapvorm vertoont, maar steeds in één van de 2 maximaal gemagnetiseerde toestanden terecht komt.

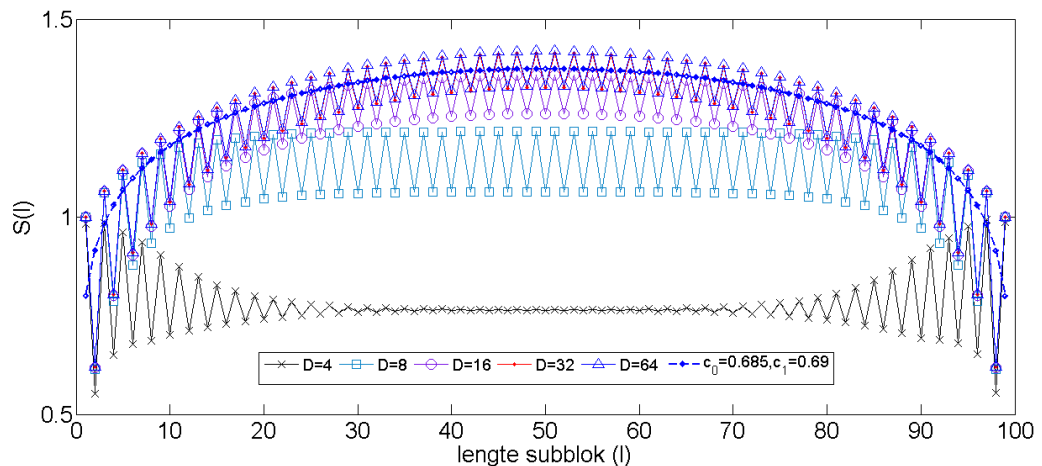
Verstrengelingsentropie in het Heisenbergmodel

In deze sectie bestuderen we de verstrengelingsentropie tussen 2 subsystemen die samen de Heisenbergketting opbouwen als functie van de lengte van het eerste subsysteem. Voor eindige systemen met L sites en open randvoorwaarden, verdeeld in 2 intervallen met respectievelijk l en $L - l$ sites, wordt de verstrengelingsentropie gegeven door de volgende formule ([15]):

$$S(l, L) = c_0 + \frac{c_1}{6} \log_2 \left(\frac{2L}{\pi} \sin \left(\frac{l\pi}{L} \right) \right) \quad (3.11)$$

waarbij c_0 en c_1 constanten zijn die afhangen van het model en de bijhorende veldentheorie ([15]). We proberen dit resultaat te reproduceren met het VMPS-programma. We berekenen de entropie als functie van de lengte van het subsysteem voor een ketting met totale lengte $L = 100$ zonder magnetisch veld en dit voor verschillende waarden van D . Het resultaat wordt weergegeven in figuur (3.4). Naast het resultaat bekomen uit het VMPS-programma hebben we ook de analytische relatie weergegeven met enerzijds $c_0 = 0.685$ en $c_1 = 0.69$ (deze waarden verder bepaald door $S(l, L)$ te fitten aan de numerieke resultaten). Eerst en vooral merken

we op dat de entropie satureert als functie van D . In dit geval blijkt $D = 32$ voldoende om de gesatureerde $S(l, L)$ waarde te bekomen. Kleine subblokken (kleine l) hebben een lagere entropie dan subblokken overeenkomend met grotere l . Zoals reeds vermeld in hoofdstuk 2 is γ in $D_L \leq \beta L^\gamma$ evenredig met 2^{S^α} (en S^α is de Renyi entropie). Dat betekent dat de lokale matrixdimensie voor kleine l niet zo groot moet zijn als deze bij grotere l (omdat de entropie bij kleinere l lager is). Vandaar dat onze MPS in staat is om het volledige systeem goed te beschrijven aangezien de matrixdimensie van een MPS ook toeneemt naarmate we naar het midden van de ketting gaan.



Figuur 3.4: Verstrelingsentropie S van een Heisenbergmodel als functie van de blok grootte bij een ketting met totale lengte $L = 100$, $J = 1$ en $\beta = 0$.

3.2 De $J_1 - J_2$ Hamiltoniaan

Een meer geavanceerd model dat naast dichtste-nabuur interacties ook tweede-orde interacties toelaat (d.w.z dat een spintoestand op site i kan interageren met zowel site $i + 1$ als met site $i + 2$) is het zogenaamde $J_1 - J_2$ model ([21]) beschreven door de volgende Hamiltoniaan:

$$\begin{aligned} \hat{H} &= J_1 \sum_{i=1}^{L-1} \hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_{i+1} + J_2 \sum_{i=1}^{L-2} \hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_{i+2} \\ &= \frac{J_1}{2} \sum_{i=1}^{L-1} \left(\hat{S}_i^+ \hat{S}_{i+1}^- + \hat{S}_i^- \hat{S}_{i+1}^+ + 2\hat{S}_i^z \hat{S}_{i+1}^z \right) + \frac{J_2}{2} \sum_{i=1}^{L-2} \left(\hat{S}_i^+ \hat{S}_{i+2}^- + \hat{S}_i^- \hat{S}_{i+2}^+ + 2\hat{S}_i^z \hat{S}_{i+2}^z \right). \end{aligned} \quad (3.12)$$

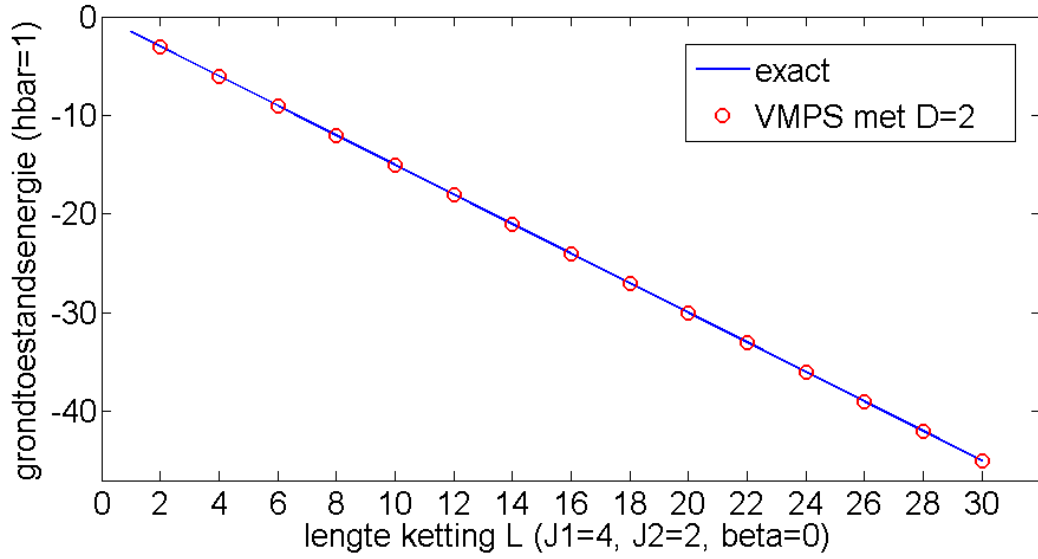
3.2.1 Het exact oplosbare model

Als $J_2 = \frac{J_1}{2}$ dan spreekt men over het Majumdar-Ghosh model en in dat specifieke geval kan de Hamiltoniaan geschreven worden als functie van de projectie-operator op de spin- $\frac{3}{2}$ subruimte van 3 gekoppelde spin- $\frac{1}{2}$, wat toelaat een analytische oplossing voor dit speciaal geval te bepalen. De geïnteresseerde lezer kan het bewijs terugvinden in Appendix D. We vermelden hier enkel het eindresultaat. De energie in de grondtoestand E_0 van een Majumdar-Ghosh ketting (met

$J_2 = \frac{J_1}{2}$) wordt in de limiet van een oneindig lange ketting gegeven door

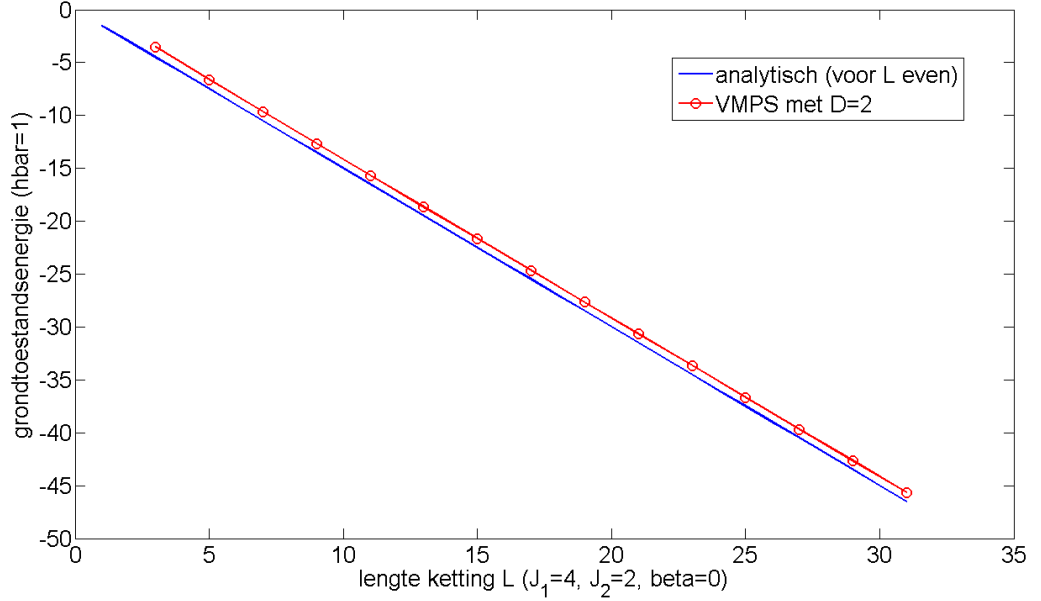
$$E_0 = -\frac{3}{8}J_1L. \quad (3.13)$$

Daarnaast is dit resultaat ook exact als we enkel kettingen met even L beschouwen wegens de dimerisatie van naburige spins (zie verder). Voor dit bijzondere geval kunnen we de resultaten bekomen met het VMPS-algoritme vergelijken met het analytische resultaat. Daartoe kiezen we $J_2 = \frac{J_1}{2}$ en nemen $J_1 = 4$ met $D = 2$ (zie verder waarom $D = 2$ volstaat voor even L). De grondtoestandsenergie (zonder magnetisch veld) als functie van de lengte van de ketting L wordt weergegeven in figuur (3.5). We vinden een perfecte overeenkomst, wat logisch is aangezien we



Figuur 3.5: Grondtoestandsenergie ($\hbar = 1$) als functie van L (even): vergelijking van het analytische resultaat met de VMPS-output bij $D = 2$.

enkel even L getest hebben. We kijken nu ook wat er gebeurt als we oneven L beschouwen. Het resultaat is weergegeven in figuur (3.6). Het is duidelijk dat de energie nu niet overeenkomt met de analytische curve. Voor grotere L (oneven) is de afwijking tussen het VMPS-resultaat en het analytische reeds kleiner (niet te zien op de figuur, maar we hebben dit wel getest voor grotere waarden van (oneven) L). In de limiet $L \rightarrow \infty$ zou het verschil moeten verdwijnen. Figuur (3.6) werd gemaakt met $D = 2$ maar voor oneven L zal dit niet meer het exacte resultaat geven omdat je de spins nu niet per 2 kan afkoppelen tot een singlet (de zogenaamde dimerisatie, zie verder); er is altijd nog één site die dan geen paar kan vormen met een andere site. Om een exact resultaat te bekomen moeten we dus $D = \lceil \frac{L}{2} \rceil$ stellen. We onderzochten de invloed op de energie voor één bepaalde waarde van L om te zien of we werkelijk een andere energie bekomen voor verschillende D (wat niet het geval is als we met even L werken, en uiteraard $J_2 = \frac{J_1}{2}$). Voor $L = 15$ worden de resultaten samengevat in tabel (3.1). Het is duidelijk dat de energie bij $D = 2$ afwijkt van de exacte waarde (hier bij $D = 2 \lceil \frac{15}{2} \rceil$). Voor oneven L moet men dus $D = 2 \lceil \frac{L}{2} \rceil$ stellen, zelfs al is $J_2 = \frac{J_1}{2}$, om het *exacte* resultaat te bekomen (al zal een kleinere waarde van D ook al volstaan om een voldoende accuraat resultaat te bekomen, zie vroeger).



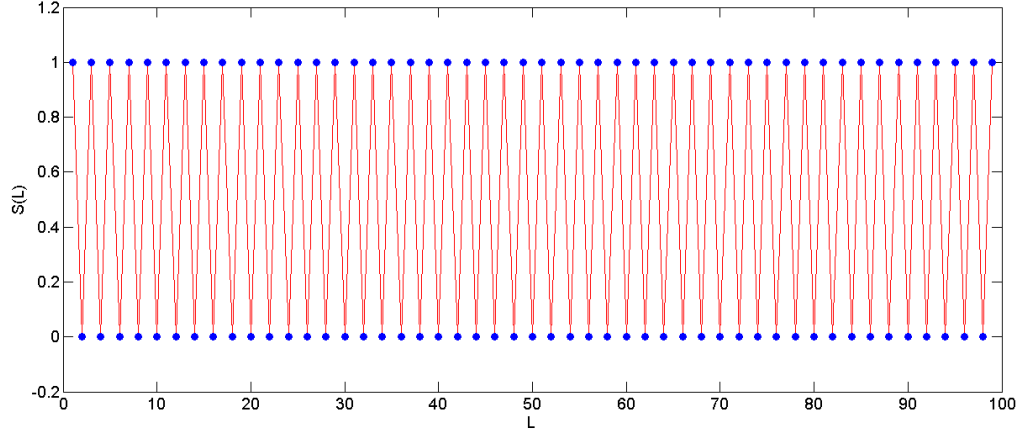
Figuur 3.6: Grondtoestandsenergie ($\hbar = 1$) als functie van L (oneven): vergelijking van het analytische resultaat met de VMPS-output bij $D = 2$.

D	$E(D), \hbar = 1$
2	-21.665350371485133
4	-21.935848120087606
8	-21.950742723138060
16	-21.950921112302268
32	-21.950921177099392
128	-21.950921429674700

Tabel 3.1: Energie voor $L = 15$ als functie van D , $J_1 = 4$, $J_2 = \frac{J_1}{2}$.

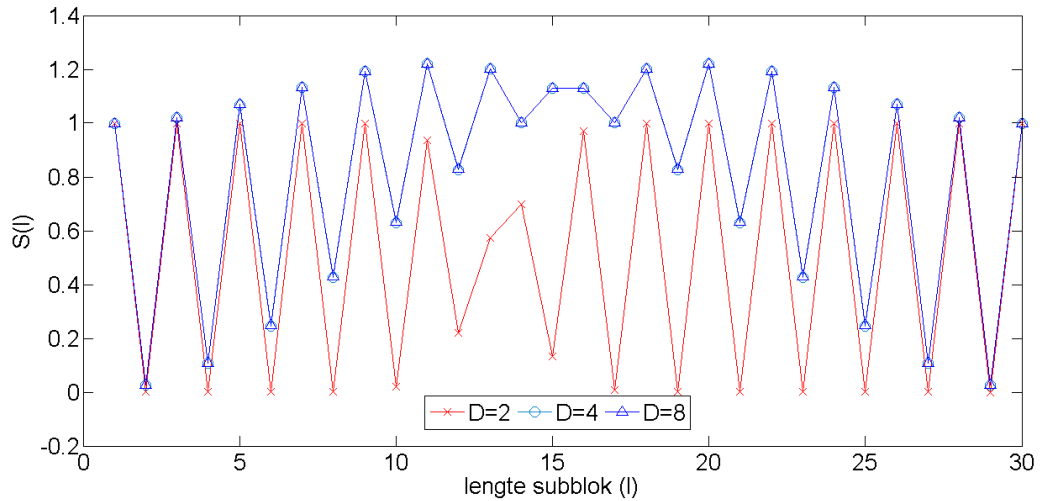
3.2.2 Verstrelingsentropie van de $J_1 - J_2$ keten

In deze sectie gaan we dieper in op een aantal verstrengingsaspecten van de $J_1 - J_2$ keten. We bekijken eerst het exact oplosbare geval $J_2 = \frac{J_1}{2}$ en nemen $J_1 = 4$. We nemen een ketting met $L = 100$ zonder magnetisch veld en bestuderen de verstrengingsentropie van de linker- en rechterblokken waarbij we de lengte variëren. Wanneer we figuur (3.7) bekijken, merken we iets heel bijzonders op. De entropie $S(l)$ is steeds 0 of 1 wat duidt op helemaal geen verstrengeling of juist wel verstrengeling (en dit voor alle waarden van D die we getest hebben: $D = [2, 4, 8, 16, 32]$). Een verstrengingsentropie die 0 is, wijst op 2 niet-gecorrleerde systemen. Zoals op de figuur te zien is, zullen subblokken met een even aantal spins steeds $S(2n) = 0, \forall n \in \mathbb{N}$ geven wat fysisch wijst op een dimerisatie. Dit betekent dat 2 naburige spins zullen afkoppelen tot een singlet toestand. De verstrengeling met de rest van de keten verdwijnt dan, consistent met $S(2n) = 0, \forall n \in \mathbb{N}$. Vandaar dat we in dit geval ook het exacte resultaat kunnen bekomen door $D = 2$ te stellen voor een willekeurige lengte van de ketting. De correlatie beperkt zich steeds tot 2 naburige spins en dit kan perfect beschreven worden met $D = 2$. Als



Figuur 3.7: Verstregelingsentropie S van een $J_1 - J_2$ keten als functie van de blokgrrootte, met totale lengte $L = 100$ en $J_2 = \frac{J_1}{2}$, $J_1 = 4$.

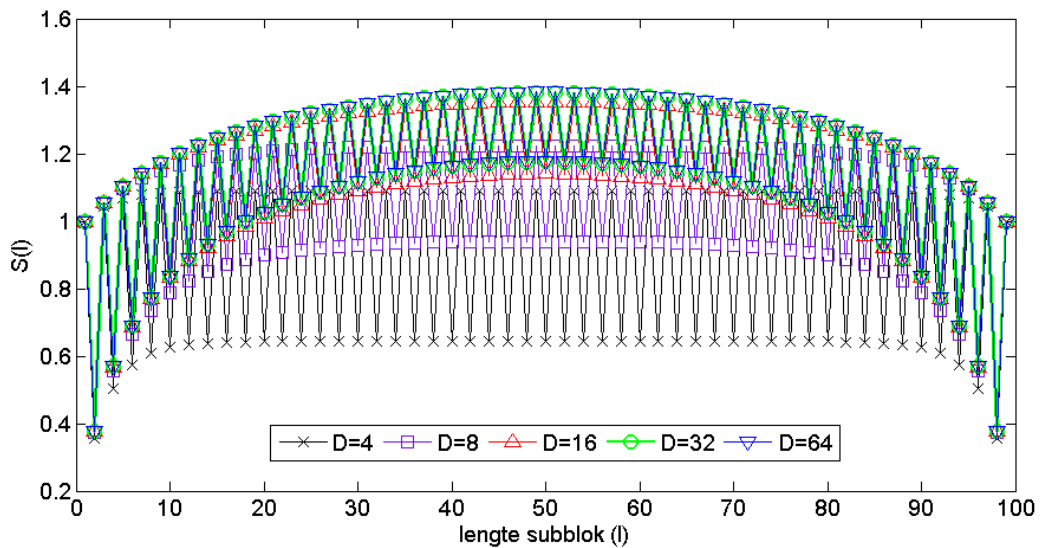
we een ketting met lengte $L = 2n, n \in \mathbb{N}$ nemen dan is de energie $E(L) = nE(2)$ (we hebben dit getest voor een reeks van even L -waarden), wat men kan interpreteren als de energie van n niet-gecorreleerde systemen, elk met energie $E(2)$. Dit is opnieuw een bevestiging dat de spins per twee zullen interageren en niet gecorreleerd zijn met de overige $L - 2$ spins van de ketting. Meer informatie over correlaties en entropie in het $J_1 - J_2$ model en over dimerisatie in het Majumdar Ghosh (MG) punt ($J_2 = \frac{J_1}{2}$) is te vinden in ([21]). Voor oneven L met $J_2 = \frac{J_1}{2}$ stijgt de entropie $S(l)$ tot aan het midden van de ketting (zie figuur (3.8)).



Figuur 3.8: Verstregelingsentropie S van een $J_1 - J_2$ keten als functie van de blokgrrootte, met totale lengte $L = 31$ en $J_2 = \frac{J_1}{2}$, $J_1 = 4$.

Vervolgens kijken we naar het geval $J_1 = 4, J_2 = 1$. We zien in figuur (3.9) een analoog gedrag als dat in het Heisenbergmodel. Als we het maximum van de verstregelingsentropie van een Heisenbergmodel (S_{max}^H) met $J = 4, L = 100$ (figuur (3.4) is voor $J = 1$, maar functioneel geeft

$J = 4$ precies dezelfde vorm als figuur (3.4)) vergelijken met het maximum van figuur (3.9) ($S_{max}^{J_1 J_2}$) dan merken we op dat $S_{max}^H > S_{max}^{J_1 J_2}$. Dit geldt bovendien ook voor de andere punten (dus $S^H > S^{J_1 J_2}$ voor die $J_1 - J_2$ combinatie). Dat betekent dat er meer correlatie is tussen de subblokken in het Heisenbergmodel dan in het $J_1 - J_2$ model (**voor de gegeven $J_1 - J_2$ combinatie**). We kunnen dit wijten aan het feit dat een $J_2 \neq 0$ en $J_2 \neq \frac{J_1}{2}$ aanleiding geeft tot een partiële dimerisatie, zodat de correlatie ietwat afgezwakt wordt in vergelijking met het Heisenbergmodel. Dit kan men kwantitatief nagaan door te kijken in hoeverre $E(2n) = nE(2)$ nog voldaan is voor $J_2 \neq \frac{J_1}{2}$. De resultaten worden samengevat in tabel (3.2). We zien inderdaad dat naarmate we dichter tot het MG punt naderen, we steeds beter aan de relatie $E(2n) = nE(2)$ voldoen. Ter vergelijking wordt in de laatste kolom ook het resultaat voor een Heisenbergmodel met $J = 4$ vermeld. Het is duidelijk dat de afwijking van $E(2n) = nE(2)$ in dit geval groter is als bij het $J_1 - J_2$ model (voor alle geteste combinaties van $J_1 - J_2$ met $J_1 = 4$). Voor $L = 2$ vinden we uiteraard hetzelfde resultaat voor beide gevallen vermits er dan nog geen mogelijkheid is tot tweede-orde interacties, zodat het $J_1 - J_2$ model zich herleidt tot het Heisenbergmodel (bemerkt dat $E = -3$ inderdaad overeenstemt met de exacte waarde voor $L = 2, J = 4$ volgens formule (3.6) die we afgeleid hebben voor een Heisenbergmodel met lengte $L = 2$).

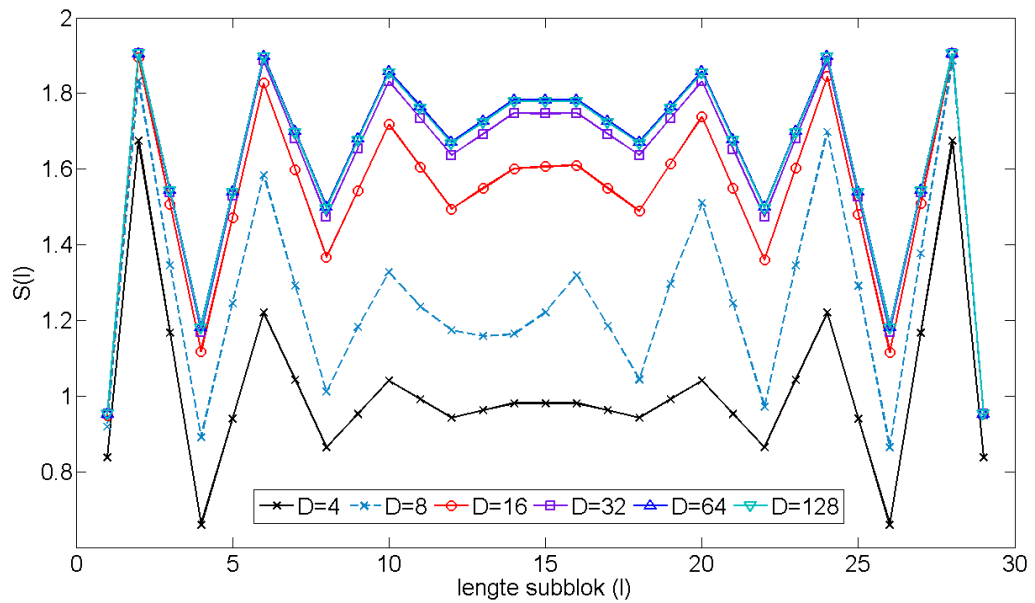


Figuur 3.9: Verstrelingsentropie S van een $J_1 - J_2$ keten als functie van de blokgrrootte, met totale lengte $L = 100$ en $J_1 = 4, J_2 = 1$.

L	$E, \frac{J_1}{J_2} = 2$	$E, \frac{J_1}{J_2} = 4$	$E, \frac{J_1}{J_2} = 2.5$	$E, \frac{J_1}{J_2} = 2.3$	$E_H, J = 4$
2	-3.0	-3.0	-3.0	-3.0	-3.0
4	-6.0	-6.15	-6.03	-6.01	-6.46
6	-8.99	-9.32	-9.06	-9.03	-9.97
8	-11.99	-12.51	-12.10	-12.04	-13.50
10	-14.99	-15.70	15.14	-15.06	-17.03

Tabel 3.2: Energie als functie van $\frac{J_1}{J_2}$ vergeleken met $E_{Heisenberg}, J = 4$ ($D = 32$)

Tot slot kijken we ook naar het geval waarbij $J_2 = 4$ en $J_1 = 0$ (zie figuur (3.10) voor een even $L = 30$ en figuur (3.11) voor oneven $L = 31$). Wat opvalt is dat de entropie in beide gevallen afvlakt (in tegenstelling tot de stijgende functie, in het interval $l = 1 \dots L/2$, die we kregen voor het geval $J_2 = 1$ en $J_1 = 4$). Bovendien is de range van subblokken met een hoge entropie ook vrij groot (vanaf $l = 5$ is de gemiddelde grootte van de entropie ongeveer 1.7 en dit voor zowel even als oneven L). De stijgende functie (in het andere geval) impliceert dat de correlatie het grootste is tussen 2 blokken met ongeveer dezelfde lengte $l = L/2$ terwijl we hier reeds voor kleinere l een grotere entropie bekomen. Daarnaast ligt de entropie bij de ketting met oneven L ook gemiddeld iets hoger dan in het geval van een even L (wat we wijten aan het verschil in partiële dimerisatie).



Figuur 3.10: Verstregelingsentropie S van een $J_1 - J_2$ keten als functie van de blokgruote, met totale lengte $L = 30$ en $J_1 = 0$, $J_2 = 4$.

3.2.3 Magnetisatie van een $J_1 - J_2$ ketting

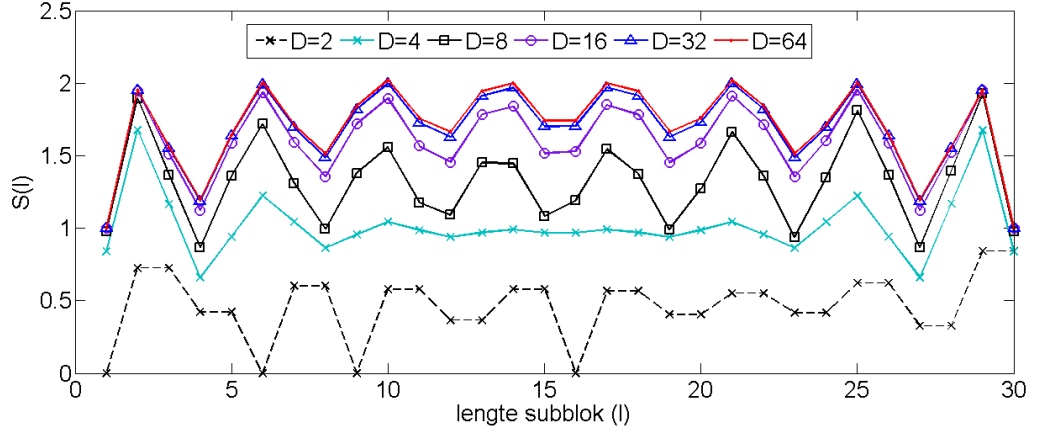
We onderzoeken nu de invloed van de koppelsterkte J_1 en het magnetische veld β op de magnetisatie van een $J_1 - J_2$ ketting. Daartoe voegen we een extra term toe aan de Hamiltoniaan om het effect van een magnetisch veld te bestuderen.

$$\hat{H} = J_1 \sum_{i=1}^{L-1} \hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_{i+1} + \frac{J_1}{2} \sum_{i=1}^{L-2} \hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_{i+2} - \beta \sum_{i=1}^L \hat{S}_i^z \quad (3.14)$$

De magnetisatie M van een spinketting met lengte L is gedefinieerd als

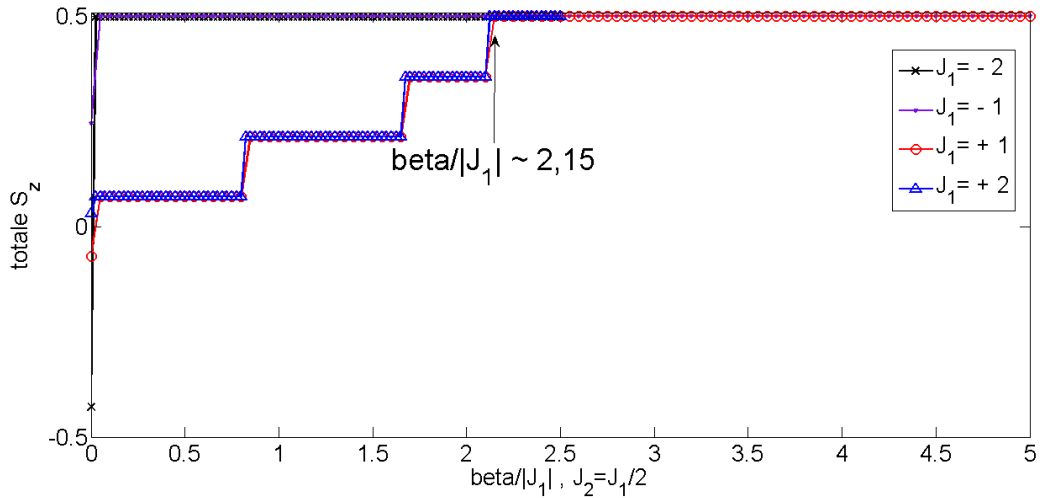
$$M = \frac{1}{L} \sum_{l=1}^L \langle \psi | S_l^z | \psi \rangle. \quad (3.15)$$

Beschouw nu figuur (3.12) waar we de magnetisatie uitzetten als functie van $\frac{\beta}{|J_1|}$ voor $J_2 = \frac{J_1}{2}$ en met $D = 8$ (dus accurater dan $D = 2$ voor oneven L zoals we hiervoor besproken hebben).



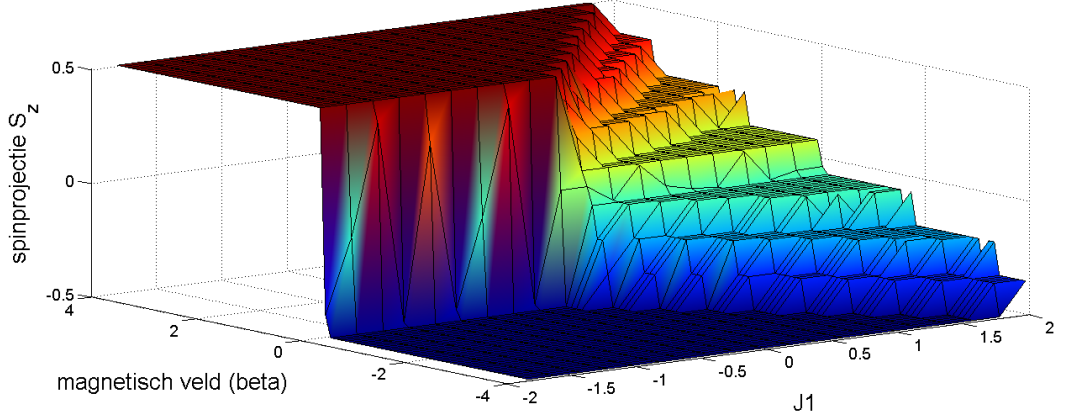
Figuur 3.11: Verstrelingsentropie S van een $J_1 - J_2$ keten als functie van de blokgrrootte, met totale lengte $L = 31$ en $J_1 = 0$, $J_2 = 4$.

Voor $J_1 > 0$ verandert de magnetisatie volgens een trapfunctie. De transitie naar een volledig gemagnetiseerde toestand gebeurt bij een magnetisch veld $|\beta| \approx 2.15 |J_1|$ (terwijl we bij $D = 2$ een transitie vonden bij $|\beta| \approx 2.175 |J_1|$). In het Heisenbergmodel vinden we een transitie bij $|\beta| = 2J_1$. Hier hebben we dus een groter veld nodig om alle spins met het magnetisch veld op te lijnen. Ter illustratie maken we ook nog een 3D plot voor een ketting met lengte $L = 7$



Figuur 3.12: Spinprojectie S_z als functie van het magnetisch veld $\frac{\beta}{|J_1|}$ met $J_2 = \frac{J_1}{2}$, $L = 7$, $\hbar = 1$, $D = 8$ voor $J_1 = [-2, -1, 1, 2]$.

en $D = 2$. Het resultaat is te vinden in figuur (3.13). Voor $J_1 < 0$ vinden we een analoog resultaat als bij het Heisenbergmodel; de spins zijn bij voorkeur reeds opgelijnd met elkaar en het magnetisch veld kan enkel de zin van de oplijning nog beïnvloeden. De precieze transities van het trapoppervlak bij $J_1 > 0$ zijn terug te vinden in figuur (3.12).



Figuur 3.13: Spinprojectie S_z als functie van het magnetisch veld β en J_1 met $J_2 = \frac{J_1}{2}$ en $L = 7$, $D = 2$, $\hbar = 1$.

3.3 Het bilinear biquadratic model

Het bilinear biquadratic (BB) model voor $S = 1$ ([20]) wordt beschreven door de volgende Hamiltoniaan

$$H_{BB} = J \sum_i \left(\cos \theta \left(\hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_{i+1} \right) + \sin \theta \left(\hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_{i+1} \right)^2 \right) - \beta \sum_i \hat{S}_i^z \quad (3.16)$$

Door het kwadraat uit te werken kan dit opnieuw in de standaardvorm (2.19) gebracht worden:

$$H_{BB} = -\beta \sum_i \hat{S}_i^z + J \sum_i \left\{ \frac{\cos \theta}{2} \left(\hat{S}_i^+ \hat{S}_{i+1}^- + \hat{S}_i^- \hat{S}_{i+1}^+ + 2\hat{S}_i^z \hat{S}_{i+1}^z \right) \right. \quad (3.17)$$

$$+ \frac{\sin \theta}{4} \left((S_i^+)^2 (S_{i+1}^-)^2 + (S_i^+ S_i^-) (S_{i+1}^- S_{i+1}^+) + (S_i^- S_i^+) (S_{i+1}^+ S_{i+1}^-) + (S_i^-)^2 (S_{i+1}^+)^2 + \right.$$

$$4(S_i^z)^2 (S_{i+1}^z)^2 + 2(S_i^+ S_i^z) (S_{i+1}^- S_{i+1}^z) + 2(S_i^- S_i^z) (S_{i+1}^+ S_{i+1}^z) \left. \right\}$$

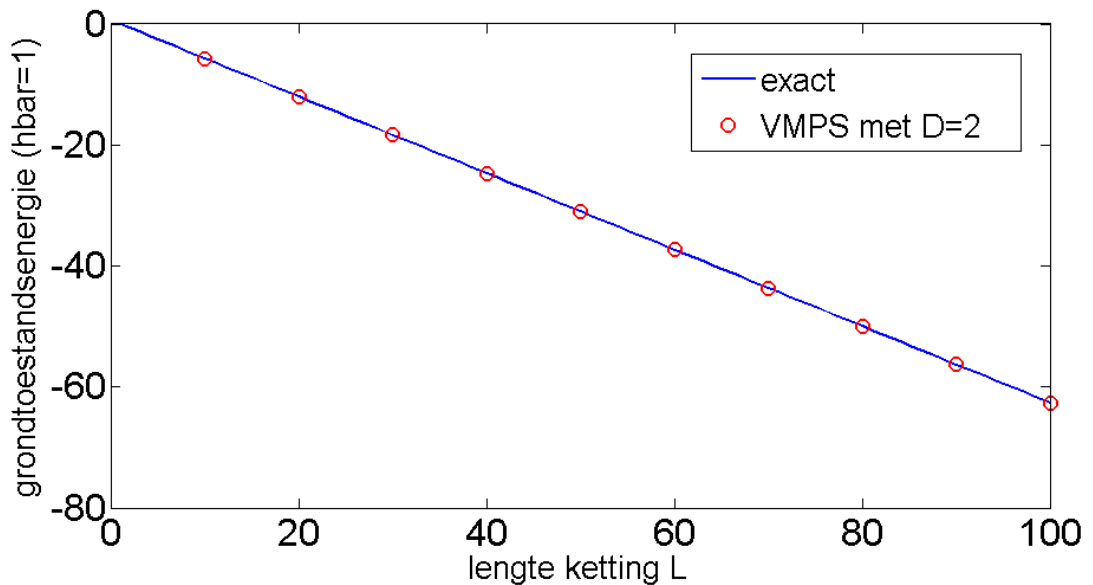
Door de Hamiltoniaan op deze manier te schrijven is het duidelijk dat we $\delta = 12$ interactie-termen hebben tussen site i en site $i + 1$, met interactie-range $\eta = 1$. Dit model is exact oplosbaar als $\theta = \arctan \frac{1}{3}$ (dan spreken we van het AKLT model, genoemd naar Affleck, Kennedy, Lieb en Tasaki), op voorwaarde dat de spin op site 1 en L gelijk is aan $\frac{1}{2}$ (zie hieronder voor iets meer uitleg). In dat geval kunnen we de bovenstaande Hamiltoniaan in verband brengen met een projectie-operator op een spin $S = 2$ ([5]) en men kan aantonen dat de exacte oplossing (voor open randvoorwaarden met spin $\frac{1}{2}$ op site 1 en L en spin 1 op de overige sites) dan gelijk is aan

$$E_0 = -\frac{2}{3} \cos \theta (L - 1). \quad (3.18)$$

Bij de implementatie van het AKLT-model hebben alle sites dus een spin $S = 1$ behalve de sites 1 en L die een spin $S = \frac{1}{2}$ hebben (hierdoor bekommen we een exact oplosbaar model dat we kunnen linken aan projectie-operatoren ([24]), net zoals we dat deden bij het exact oplosbaar $J_1 - J_2$ model; de geïnteresseerde lezer kan meer info vinden in ([25])). Dit kan geïmplementeerd worden door een zogenaamde edge-dimensie d_{edge} in te voeren. Alle sites 2 tot $L - 1$ hebben

in het AKLT-model een fysische dimensie 3 (overeenkomend met de 3 mogelijke spinprojecties), maar sites 1 en L hebben een fysische dimensie gelijk aan 2 (aangezien er slechts 2 mogelijke spinprojecties zijn), dus $d_{edge} = 2$. Bij het opstellen van de MPS en de verschillende X - en Y -blokken dienen we dus rekening te houden met deze kleine aanpassing. We vermelden voor de volledigheid dat alle matrices als (3×3) -matrices geïmplementeerd worden, maar op site 1 en L voegen we gewoon nullen toe aan de fysische (2×2) -matrices.

We controleren het exact oplosbare geval expliciet met het VMPS-programma bij $D = 2$. Het resultaat wordt weergegeven in figuur (3.14). Ook hier vinden we een perfecte overeenkomst. Dus de kwadratische term $(\hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_{i+1})^2$ kan goed beschreven worden met het programma. Dat we bij $D = 2$ opnieuw een exacte overeenkomst vinden heeft te maken met de specifieke structuur van de grondtoestand (net zoals bij het Majumdar-Ghosh model). Als we de Hamiltoniaan zouden uitrekenen zonder edge-dimensie dan blijkt dat de grondtoestand 4-voudig ontaard is als we open randvoorwaarden opleggen. Deze studie valt echter buiten het bestek van de thesis. We verwijzen hiervoor naar de literatuur ([26]). Voor de volledigheid vermelden we nog de referentie naar een meer theoretisch gericht artikel over het fasendiagram van het BB model ([20]). Daar wordt ook de verstrengelingsentropie berekend en vergeleken met analytische resultaten uit conforme veldentheorie (zoals we hierboven gedaan hebben in het geval van het isotroop Heisenbergmodel).



Figuur 3.14: Grondtoestandsenergie ($\hbar = 1$) als functie van L voor $\theta = \arctan \frac{1}{3}$.

Hoofdstuk 4

Fermionsystemen

Tot nu toe hebben we steeds spinsystemen behandeld terwijl de Hamiltoniaan (2.19) wel algemeen geldig is. Bij de implementatie zijn we echter wel uitgegaan van eigenschappen die specifiek gelden voor spinsystemen. We hebben reeds vermeld dat dit geen beperkingen oplegt aan de fysische systemen die we kunnen bestuderen aangezien via een geschikte transformatie een fermionsysteem kan afgebeeld worden op een spinsysteem. Dit is de zogenaamde Jordan-Wigner-transformatie (een beschrijving van deze transformatie voor het Hubbard model is te vinden in [27]). We kunnen echter ook het spin-algoritme recycleren om een werkend programma voor fermionische systemen te ontwikkelen. Het is deze laatste strategie die we in dit hoofdstuk zullen toepassen. De resultaten van het algoritme zullen getoetst worden aan het één-dimensionale Hubbard model. Uitbreidingen naar long-range fermionische kettingen worden ook besproken en we testen de methode met behulp van het (één-dimensionale) $t_1 - t_2$ model. We zullen echter eerst enkele belangrijke concepten invoeren uit de fundamentele veeldeeltjesfysica die van belang zijn in de verdere behandeling van fermionen.

4.1 Fundamentele aspecten van de veeldeeltjesfysica

Om een veeldeeltjessysteem te beschrijven is een elegant formalisme voorhanden dat ons toelaat veeldeeltjessystemen op een nette wijze te schrijven in termen van de één-deeltjestoestanden. In deze sectie zullen we een aantal fundamentele concepten uit de veeldeeltjesfysica invoeren om ze later te kunnen toepassen in de beschreven systemen. Een uitstekende behandeling van deze concepten is te vinden in ([28]). De volgende secties vormen een compacte samenvatting van hoofdstukken 1 en 2 uit ([28]). We verwijzen de lezer dan ook naar dit boek voor meer informatie en voor de afleiding van een aantal bewijzen waarvan hieronder slechts het eindresultaat wordt vermeld.

4.1.1 Inleiding

Aangezien de veeldeeltjesfysica de kwantummechanica van meerdere (N) deeltjes beschrijft, dienen we een aantal theoretische concepten in te voeren om dergelijke N -deeltjestoestanden op een compacte manier weer te geven (die we ontleen aan ([28])). Een N -deeltjestoestand (of productstate) noteren we in functie van *orthonormale* één-deeltjestoestanden $|\alpha_i\rangle$ als

$$|\alpha_1\alpha_2\dots\alpha_N\rangle = |\alpha_1\rangle|\alpha_2\rangle\dots|\alpha_N\rangle \quad (4.1)$$

met gepaste orthonormaliteitsrelatie

$$(\alpha_1 \alpha_2 \dots \alpha_N | \alpha'_1 \alpha'_2 \dots \alpha'_N) = \langle \alpha_1 | \alpha'_1 \rangle \langle \alpha_2 | \alpha'_2 \rangle \dots \langle \alpha_N | \alpha'_N \rangle = \delta_{\alpha_1, \alpha'_1} \delta_{\alpha_2, \alpha'_2} \dots \delta_{\alpha_N, \alpha'_N} \quad (4.2)$$

en bijhorende compleetheidsrelatie

$$\sum_{\alpha_1 \alpha_2 \dots \alpha_N} |\alpha_1 \alpha_2 \dots \alpha_N\rangle \langle \alpha_1 \alpha_2 \dots \alpha_N| = 1. \quad (4.3)$$

Deze productstates bevatten nog niet de correcte symmetrie die we aan respectievelijk fermionen en bosonen willen toekennen. Volgens het uitsluitingsprincipe van Pauli moet de golffunctie van een N -fermionsysteem compleet anti-symmetrisch zijn onder verwisseling van 2 willekeurige ononderscheidbare deeltjes. De compleet symmetrische golffunctie karakteriseert dan een N -bosonsysteem. Toestanden die reeds ge(anti)symmetriseerd zijn worden genoteerd als

$$|\alpha_1 \alpha_2 \dots \alpha_N\rangle. \quad (4.4)$$

Voor fermionen (die een anti-symmetrische golffunctie moeten hebben) geldt dan bijvoorbeeld

$$|\alpha_1 \alpha_2\rangle = -|\alpha_2 \alpha_1\rangle. \quad (4.5)$$

4.1.2 De Fock-ruimte

Om veeldeeltjestoestanden te beschrijven die voldoen aan bepaalde symmetrieën kan men met voordeel gebruik maken van de bezettingsgraadrepresentatie of tweede kwantisatie ([28]). We werken dit uit voor fermionen. In plaats van te werken met een vectorruimte met een vast aantal deeltjes werkt de tweede kwantisatie met een vectorruimte die een directe som is van de vacuümtoestand $|0\rangle$, de complete set van ééndeeltjestoestanden $\{|\alpha\rangle\}$, de complete set van geantisymmetriseerde tweedeeltjestoestanden $\{|\alpha_1 \alpha_2\rangle\}$ en zo verder tot de complete set van een oneindig aantal deeltjes de som afsluit. Dit is de zogenaamde Fock-ruimte en kan symbolisch uitgedrukt worden als

$$|0\rangle \oplus \{|\alpha\rangle\} \oplus \{|\alpha_1 \alpha_2\rangle\} \oplus \dots \oplus \{|\alpha_1 \dots \alpha_\infty\rangle\} = |0\rangle \left\{ \bigoplus_{n=1}^{\infty} \left| \prod_{i=1}^n \alpha_i \right\rangle \right\}. \quad (4.6)$$

4.1.3 Creatie- en annihilatie-operatoren

Om de Hamiltoniaan in tweede kwantisatie neer te schrijven, dienen we 2 operatoren in te voeren. Bij definitie ([28]) wordt de creatie-operator a_α^\dagger bepaald aan de hand van volgende relatie

$$a_\alpha^\dagger |\alpha_1 \alpha_2 \dots \alpha_N\rangle \equiv |\alpha \alpha_1 \alpha_2 \dots \alpha_N\rangle. \quad (4.7)$$

Men kan nu uitgaande van deze definitie ook de toegevoegde annihilatie-operator a_α bepalen. Men voert daartoe een gepaste compleetheidsrelatie in en maakt vervolgens gebruik van (4.7) om uiteindelijk

$$a_\alpha |\alpha_1 \alpha_2 \dots \alpha_N\rangle = (-1)^{i-1} |\alpha_1 \alpha_2 \dots \alpha_{i-1} \alpha_{i+1} \dots \alpha_N\rangle \delta_{\alpha, \alpha_i} \quad i = 1, \dots, N \quad (4.8)$$

te bekomen ([28]). Als $\alpha \neq \alpha_i$ voor $i = 1, \dots, N$ dan is dus $a_\alpha |\alpha_1 \alpha_2 \dots \alpha_N\rangle = 0$. We vermelden ook de anti-commutatierelaties waaraan deze operatoren moeten voldoen ([28]).

$$\{a_\alpha, a_\beta^\dagger\} = a_\alpha a_\beta^\dagger + a_\beta^\dagger a_\alpha = \delta_{\alpha, \beta} \quad (4.9)$$

$$\{a_\alpha, a_\beta\} = \{a_\alpha^\dagger, a_\beta^\dagger\} = 0 \quad (4.10)$$

Men kan inzien dat de naamgeving van de operatoren zinvol is door op te merken dat het inwerken van een creatie-operator a_α^\dagger op de vacuümtoestand $|0\rangle$ een ééndeeltjestoestand creëert. Past men op een dergelijke ééndeeltjestoestand opnieuw een creatie-operator a_β^\dagger toe dan bekomt men een geantisymmetriseerde tweedeeltjestoestand, enzovoort. Op dergelijke manier bekomt men alle complete sets die de Fock-ruimte opbouwen ([28]):

$$|\alpha_1 \alpha_2 \dots \alpha_N\rangle = a_{\alpha_1}^\dagger |\alpha_2 \dots \alpha_N\rangle = a_{\alpha_1}^\dagger a_{\alpha_2}^\dagger |\alpha_3 \dots \alpha_N\rangle = \dots = \prod_{i=1}^N a_{\alpha_i}^\dagger |0\rangle \quad (4.11)$$

Dat een dergelijke toestand anti-symmetrisch is (een primordiale vereiste voor fermionen) volgt onmiddellijk door het toepassen van (4.10). We merken nog op dat de bezettingsgraad n_{α_i} van een fermiontoestand α_i enkel 0 of 1 is (inachtgenomen dat de kwantumtoestand α_i ook informatie over de spin bevat). Zo schrijven we bijvoorbeeld ([28])

$$|n_{\alpha_1} = 1, n_{\alpha_2} = 1, n_{\alpha_3} = 1, 0, \dots, 0\rangle = |\alpha_1 \alpha_2 \alpha_3\rangle \quad (4.12)$$

4.1.4 Ééndeeltjesoperatoren in de Fock-ruimte

Beschouw een ééndeeltjesoperator F , dan wordt deze operator volledig bepaald door de matrixelementen $\langle \alpha | F | \beta \rangle$ wat men makkelijk kan inzien door opnieuw 2 compleetheidsrelaties in te voeren

$$F = \sum_{\alpha} \sum_{\beta} |\alpha\rangle \langle \alpha | F | \beta \rangle \langle \beta|. \quad (4.13)$$

Voor een ruimte met N deeltjes wordt de N -deeltjesoperator F_N gegeven door

$$F_N = \sum_{i=1}^N F(i) \quad (4.14)$$

waarbij $F(i)$ inwerkt op deeltje i . We wensen deze ééndeeltjesoperator (omdat elke component in F_N slechts inwerkt op een ééndeeltjestoestand) nu neer te schrijven in functie van de zopas ingevoerde creatie - en annihilatie-operatoren teneinde een operator te bekomen die gepast inwerkt op een Fock-ruimte. Men kan aantonen ([28]) dat een dergelijke Fock-operator bestaat en compleet equivalent is met (4.13). De corresponderende ééndeeltjes Fock-operator (genoteerd met \hat{F}) wordt gegeven door

$$\hat{F} = \sum_{\alpha\beta} \langle \alpha | F | \beta \rangle a_\alpha^\dagger a_\beta. \quad (4.15)$$

De kinetische energie-operator \hat{T} kan bijvoorbeeld in deze vorm uitgedrukt worden (met dan $\hat{F} = \hat{T}$).

4.1.5 Tweedeeltjesoperatoren in de Fock-ruimte

Een tweedeeltjesoperator werkt in op 2 toestanden en noteren we in het geval van een tweedeeltjesproductruimte als

$$V = \sum_{\alpha\beta} \sum_{\gamma\delta} |\alpha\beta\rangle \langle \alpha\beta | V | \gamma\delta\rangle \langle \gamma\delta|. \quad (4.16)$$

In een N -deeltjesruimte wordt de uitbreiding hiervan gegeven door

$$V_N = \sum_{i < j=1}^N V(i, j) = \frac{1}{2} \sum_{i \neq j}^N V(i, j) \quad (4.17)$$

waarbij elke $V(i, j)$ enkel werkt op de deeltjes i en j . Men kan aantonen ([28]) dat een dergelijke operator in tweede kwantisatie geschreven kan worden als

$$\hat{V} = \frac{1}{2} \sum_{\alpha\beta\gamma\delta} (\alpha\beta|V|\gamma\delta) a_\alpha^\dagger a_\beta^\dagger a_\delta a_\gamma. \quad (4.18)$$

Let hierbij vooral op de volgorde van γ en δ . Een voorbeeld van een dergelijke operator is de interactie-operator tussen 2 elektronen. Nu hebben we alle concepten in handen om de Hamiltoniaan van een veeldeeltjessysteem te beschrijven in tweede kwantisatie ([28]):

$$\hat{H} = \sum_{\alpha\beta} \langle \alpha|T|\beta \rangle a_\alpha^\dagger a_\beta + \frac{1}{2} \sum_{\alpha\beta\gamma\delta} (\alpha\beta|V|\gamma\delta) a_\alpha^\dagger a_\beta^\dagger a_\delta a_\gamma. \quad (4.19)$$

4.2 Van spin-roosters naar fermion-roosters

We zullen nu trachten fermionische systemen te bestuderen met behulp van ons algoritme. We ontwikkelen het programma in eerste instantie voor fermion Hamiltonianen van de vorm

$$\begin{aligned} H &= - \sum_{s \in (\uparrow, \downarrow)} \sum_{i < j} T_{ij} \left(a_{i,s}^\dagger a_{j,s} + a_{j,s}^\dagger a_{i,s} \right) + U \sum_{i=1}^L a_{i,\uparrow}^\dagger a_{i,\uparrow} a_{i,\downarrow}^\dagger a_{i,\downarrow} \\ &= - \sum_{s \in (\uparrow, \downarrow)} \sum_{i < j} T_{ij} \left(a_{i,s}^\dagger a_{j,s} + a_{j,s}^\dagger a_{i,s} \right) + U \sum_{i=1}^L n_{i,\uparrow} n_{i,\downarrow}. \end{aligned} \quad (4.20)$$

Grofweg komt het er dus op neer dat spin-operatoren vervangen zijn door creatie- en annihilatie-operatoren van \uparrow en \downarrow deeltjes (ook geassocieerd aan een bepaalde site). De getallen $T_{ij} \in \mathbb{R}$ spelen hier de rol van koppelconstante en $U \in \mathbb{R}$ kan gezien worden als de sterkte van de on-site repulsie ($U > 0$) of attractie ($U < 0$). Om een dergelijke Hamiltoniaan correct te implementeren in ons spin-algoritme zijn er 2 belangrijke aanpassingen nodig: enerzijds moeten we een gewenst deeltjesaantal opleggen en anderzijds zijn er additionele fasefactoren bij long-range systemen nodig. Voor we deze aanpassingen toelichten, moeten we natuurlijk eerst een lokale basis definiëren en geschikte matrixvoorstellungen voor de verschillende operatoren vinden.

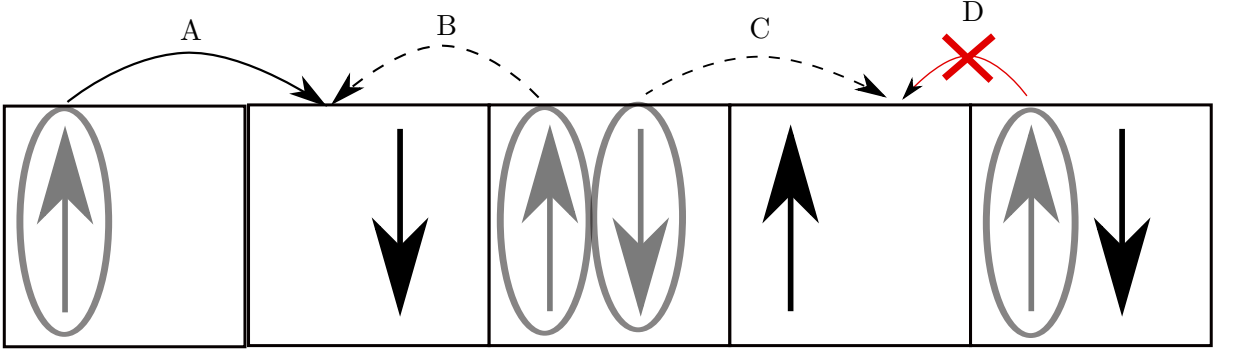
De fysische dimensie van een site is in dit geval gelijk aan 4 omdat er per site 4 mogelijke toestanden zijn: de site is onbezet wat overeenkomt met het vacuüm $|0\rangle$, de site bevat een spin up $|\uparrow\rangle$, de site bevat een spin down $|\downarrow\rangle$ en tot slot kan de site ook bezet zijn door 2 deeltjes met tegengestelde spin (Pauli) $|\uparrow\downarrow\rangle$. Als we als basis (**in die volgorde**) dus

$$\{|0\rangle, |\uparrow\rangle, |\downarrow\rangle, |\uparrow\downarrow\rangle\} \equiv \left\{ \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \right\}$$

vooropstellen, dan kunnen we de number-operators $\hat{n}_{i,s}$ uitdrukken als

$$\hat{n}_{i,\uparrow} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \hat{n}_{i,\downarrow} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Immers, voor $\hat{n}_{i,\uparrow}$ vormen de basisvectoren een set van eigenvectoren met eigenwaarde 0 of 1 al naar gelang site i geen of wel een spin up bevat (analoog voor $\hat{n}_{i,\downarrow}$). De conventie omtrent de volgorde van de basisvectoren moet men voor alles blijven aanhouden en is essentieel in de definitie van de matrices. Nu moeten we nog een matrixrepresentatie vinden voor de creatie- en annihilatie-operatoren. Daartoe starten we van een model dat enkel dichtste-nabuur interacties bevat (de uitbreiding naar long-range vergt de invoering van additionele fasefactoren zoals we verderop zullen zien). Beschouw voor de volgende behandeling figuur (4.1). De term $a_{i,s}^\dagger a_{j,s}$ met



Figuur 4.1: Fermionisch rooster met dichtste-nabuur interacties.

$i < j$ zal zich in dit geval reduceren tot $a_{i,s}^\dagger a_{i+1,s}$. Dat betekent dat deze operator een deeltje met spin s op site $i + 1$ annihileert en dat deeltje naar site i wil verplaatsen (want $a_{i,s}^\dagger$ creëert een deeltje met spin s op site i). Dit is enkel mogelijk als site i nog geen deeltje met spin s bevat (wegens $\{a_{i,s}^\dagger, a_{j,s'}^\dagger\} = 0, \forall(i, j, s, s')$) en als er effectief een deeltje met spin s aanwezig is op site $i + 1$ (wegens $a_{i+1,s} |0\rangle = 0$ en de anti-commutatierelatie $\{a_{i,s}, a_{j,s'}^\dagger\} = \delta_{i,j} \delta_{s,s'}$).

Zoals we hiervoor gezien hebben zullen creatie-operatoren aanleiding geven tot een bepaalde toestand van het rooster wanneer ze inwerken op het vacuüm. Op die manier kan men alle mogelijke toestanden $|\psi\rangle$ van het fermionisch rooster schrijven in termen van creatie-operatoren die op het vacuüm inwerken. Merk op dat een superpositie van dergelijke toestanden formeel overeenkomt met een MPS (hoewel er een aantal kleine aanpassingen nodig zijn qua fase, zie later). We vinden dus dat $\langle\psi| a_{i,s}^\dagger a_{i+1,s} |\psi\rangle = 0$ als er niet voldaan is aan de 2 bovengenoemde voorwaarden (wegens de anti-commutatierelaties). Bovendien moet men er ook nog op letten (als gevolg van de vastgelegde volgorde van de basisvectoren) dat er tekenwissels kunnen optreden als er deeltjes verspringen. Dit wordt geïllustreerd in figuur (4.1) waar in proces B een spin up geannihileerd wordt op site $i + 1$ en verplaatst wordt naar site i . Maar daarvoor moet die spin up wel eerst een spin down op site i passeren, zodat wegens $\{a_{i,s}, a_{j,s'}^\dagger\} = \delta_{i,j} \delta_{s,s'}$ een additioneel minteken zal optreden in de matrixvoorstelling. We gaan er immers van uit dat een site steeds een spin up bevat links van een spin down en niet omgekeerd (dit is louter conventie, maar eens

die vastgelegd is moeten we uiteraard consistent verder werken met deze keuze). Een analoog proces wordt voorgesteld door pijltje C. Proces D is verboden wegens Pauli (er kunnen nooit 2 gelijke spins op dezelfde site zitten, ofwel $\{a_{i,s}^\dagger, a_{j,s'}^\dagger\} = 0, \forall(i, j, s, s')$). In proces A gaat de spin up van site i naar de volgende site $i + 1$ zonder tekenwissel omdat er geen spin down meer aanwezig is op site i . Deze bevindingen laten nu toe om de creatie- en annihilatie-operatoren voor de spin up en spin down deeltjes te construeren in het geval we **enkel dichtste-nabuur** interacties beschouwen. Deze set van matrices hoort dus bij elkaar en is niet meer correct indien we de interactie van site i met site $i + n, n > 1$ beschouwen. Daarvoor zullen additionele fasefactoren nodig zijn (zie verder). De set matrices behorend bij dichtste-nabuur interacties is dus

$$\begin{aligned}
a_{i,\uparrow}^\dagger a_{i+1,\uparrow} &= \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} & a_{i,\uparrow} a_{i+1,\uparrow}^\dagger &= \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \\
a_{i,\downarrow}^\dagger a_{i+1,\downarrow} &= \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} & a_{i,\downarrow} a_{i+1,\downarrow}^\dagger &= \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{pmatrix}.
\end{aligned} \tag{4.21}$$

Deze matrices werken in op de 4-dimensionale ruimte $\{|0\rangle, |\uparrow\rangle, |\downarrow\rangle, |\uparrow\downarrow\rangle\}$ en geven de correcte interacties als men golf functies beschouwt die opgebouwd zijn uit creatie-operatoren die op het vacuüm inwerken (dit is makkelijk te controleren door gebruik te maken van de anti-commutatierelaties).

4.2.1 Opleggen van het deeltjesaantal: de kwadratische kostfunctie

Veronderstel dat we een fermionisch systeem beschouwen dat beschreven wordt door de Hamiltoniaan \hat{H} . Aangezien fermionen anti-commuteren kunnen er per site maximum twee deeltjes zitten (een spin up en een spin down deeltje). Dat betekent dat het maximaal aantal deeltjes op een rooster met lengte L gelijk is aan $N_{max} = 2L$. In wat volgt kijken we steeds naar systemen met een *gegeven en vast aantal* deeltjes $N \leq N_{max}$. Indien er op één site twee deeltjes aanwezig zijn en $U > 0$ wordt groter, dan zal de energie stijgen als gevolg van de repulsieve term. Immers, de operatoren $n_{i,\uparrow}$ en $n_{i,\downarrow}$ tellen of er respectievelijk een spin up of spin down deeltje aanwezig is op site i (zie ook in [28]). Van zodra $N > L$ zal er minstens één site zijn met twee deeltjes en naarmate N stijgt, zal het aantal positieve bijdragen tot de energie stijgen (als $U > 0$). Dit heeft tot gevolg dat het algoritme mogelijks deeltjes zal uitstoten om de energie terug te verlagen. Aangezien wij a priori wensen op te leggen hoeveel deeltjes er in het systeem zitten, moeten we er voor zorgen dat er geen deeltjes uitgestoten worden. Om te verzekeren dat het aantal deeltjes op het rooster gelijk blijft aan een bepaalde vooropgestelde waarde N voegen we nog een kwadratische kostfunctie $\hat{f}(N)$ toe aan de Hamiltoniaan:

$$\hat{f}(N) = \lambda(\hat{N} - N)^2. \tag{4.22}$$

We zullen dus niet \hat{H} gebruiken, maar wel

$$\hat{H}_N = \hat{H} + \lambda(\hat{N} - N)^2. \quad (4.23)$$

Hierbij is λ een parameter, met $\lambda \in \mathbb{R}_+$. De operator \hat{N} is de number-operator die telt hoeveel deeltjes er in het totale systeem aanwezig zijn:

$$\hat{N} = \sum_{s \in (\uparrow, \downarrow)} \sum_{i=1}^L a_{i,s}^\dagger a_{i,s} = \sum_{s \in (\uparrow, \downarrow)} \sum_{i=1}^L \hat{n}_{i,s} \quad (4.24)$$

met L het aantal sites en $\hat{n}_{i,s}$ de number-operator geassocieerd aan site i . De reden dat we $\lambda > 0$ moeten kiezen, is om ervoor te zorgen dat alle contributies waarvoor $\langle \psi | \hat{N} | \psi \rangle \neq N$ een positieve bijdrage zullen leveren met als gevolg dat de energie zal stijgen. Het meest gunstige geval wordt bekomen wanneer $\langle \psi | \hat{N} | \psi \rangle = N$ omdat de contributie van de kostfunctie dan wegvalt. Op die manier kunnen we het deeltjesaantal N opleggen.

Merk op dat $[\hat{H}, \hat{N}] = 0$. Dit impliceert dat beide operatoren gemeenschappelijk gediagonaliseerd kunnen worden en dat er deeltjesbehoud is. Dit betekent echter niet dat de waarde van N die wij opleggen (*wensen*) aanleiding zal geven tot de laagste energie voor een gegeven U . Zoals gezegd zal het energetisch voordeliger zijn om N te verkleinen als $U > 0$ te groot wordt. Precies omwille van dit feit is het nodig om een kostfunctie te gebruiken die een gewenste N kan opleggen. Nadat we met de Hamiltoniaan \hat{H}_N een geschikte golfunctie $|\psi_\lambda\rangle$ gevonden hebben, berekenen we met deze golfunctie $\langle \psi_\lambda | \hat{H} | \psi_\lambda \rangle = E$ om de energie behorend bij de oorspronkelijke Hamiltoniaan (met N deeltjes) te bepalen. Om er zeker van te zijn dat het deeltjesaantal overeenkomt met het gewenste aantal berekenen we na de optimalisatie ook nog $\langle \psi | \hat{N} | \psi \rangle$ en controleren of dit effectief gelijk is aan N .

We zullen nu een meer expliciete uitdrukking voor de kostfunctie afleiden in functie van de number-operatoren geassocieerd aan de spin up en spin down deeltjes op iedere site.

$$\begin{aligned} \frac{\hat{f}(N)}{\lambda} &= \hat{N}^2 - 2N\hat{N} + N^2\mathbb{I}_{4 \times 4} \\ &= \sum_{i,j} (\hat{n}_{i,\uparrow}\hat{n}_{j,\uparrow} + \hat{n}_{i,\uparrow}\hat{n}_{j,\downarrow} + \hat{n}_{i,\downarrow}\hat{n}_{j,\uparrow} + \hat{n}_{i,\downarrow}\hat{n}_{j,\downarrow}) - 2N \sum_i (\hat{n}_{i,\uparrow} + \hat{n}_{i,\downarrow}) + N^2\mathbb{I}_{4 \times 4} \\ &= 2 \sum_{i < j} (\hat{n}_{i,\uparrow}\hat{n}_{j,\uparrow} + \hat{n}_{i,\uparrow}\hat{n}_{j,\downarrow} + \hat{n}_{i,\downarrow}\hat{n}_{j,\uparrow} + \hat{n}_{i,\downarrow}\hat{n}_{j,\downarrow}) \\ &\quad + \sum_i (\hat{n}_{i,\uparrow}\hat{n}_{i,\uparrow} + \hat{n}_{i,\uparrow}\hat{n}_{i,\downarrow} + \hat{n}_{i,\downarrow}\hat{n}_{i,\uparrow} + \hat{n}_{i,\downarrow}\hat{n}_{i,\downarrow}) - 2N \sum_i (\hat{n}_{i,\uparrow} + \hat{n}_{i,\downarrow}) + \sum_i \frac{N^2}{L} \mathbb{I}_{4 \times 4} \end{aligned}$$

Dit is formeel dezelfde vorm als de oorspronkelijk ingevoerde fermion Hamiltoniaan. Qua implementatie moeten we er op letten dat deze Hamiltoniaan niet interfereert met de fysische Hamiltoniaan (er mogen dus geen interactie-termen komen tussen de 2 Hamiltonianen). Het programma wordt zo herschreven dat beide Hamiltonianen analoog, maar los van elkaar, behandeld worden. Bij het uitrekenen van de effectieve Hamiltoniaan worden de 2 contributies dan gewoonweg gesommeerd en de som wordt gebruikt in de eigenvaluesolver. Er worden dus ook verschillende X - en Y -blokken aangemaakt, corresponderend met de verschillende bijdragen.

Het algoritme verandert op zich echter niet: we beschouwen gewoon verschillende contributies die op analoge manier uitgerekend worden (we dienen wel in sommige gevallen nog fases in rekening te brengen, zie verderop).

Invloed van de kostparameter λ

Vermits de parameter U de grootte van de repulsie bepaalt, zullen we λ gunstig moeten kiezen in functie van de energieschaal van de Hamiltoniaan. Immers, als we λ kleiner kiezen als de meest relevante energieschaal van het systeem, dan zal de contributie $\lambda(\hat{N} - N)^2$ de energie niet voldoende kunnen doen stijgen in vergelijking met de daling in energie die je bekomt door deeltjes uit te stoten (precies omdat U in dat geval zal domineren op λ en beide geven een positieve bijdrage tot de energie, dus als λ te klein is, is het voordeliger om de contributie komende van U te wijzigen). Het is dus belangrijk om λ voldoende groot te kiezen (daarover later meer bij het Hubbard model).

4.2.2 Long-range fermionsystemen

We hebben reeds matrixvoorstellungen afgeleid in het geval van dichtste-nabuur interacties. Indien we long-range interacties beschouwen moeten we echter rekening houden met additionele tekenwissels als gevolg van de anti-commutatierelaties. Beschouw daartoe opnieuw figuur (4.1) en veronderstel dat we de spin up van de laatste site (site 5) willen verplaatsen naar de tweede site. Dit proces wordt voorgesteld door de operator $a_{2,\uparrow}^\dagger a_{5,\uparrow}$. Het systeem zelf bevindt zich in de toestand

$$|\psi\rangle = a_{1,\uparrow}^\dagger a_{2,\downarrow}^\dagger a_{3,\uparrow}^\dagger a_{3,\downarrow}^\dagger a_{4,\uparrow}^\dagger a_{5,\uparrow}^\dagger a_{5,\downarrow}^\dagger |0\rangle. \quad (4.25)$$

Lokaal (dus op de site zelf) zijn de matrixvoorstellungen (4.21) nog steeds in orde. Maar dit is in de veronderstelling dat de twee sites naast elkaar liggen. Als er meerdere sites tussenliggen dan zorgen de anti-commutatierelaties voor extra tekenwissels. Telkens een spin up of spin down gepasseerd wordt, moet men hierdoor een extra min-teken in rekening brengen. Concreet heeft de operator $a_{i,\uparrow}^\dagger a_{i+j,\uparrow}$, $j > 1$ dezelfde matrixvoorstelling als $a_{i,\uparrow}^\dagger a_{i+1,\uparrow}$ **maar** wel gecorrigeerd met een additionele fase-factor

$$(-1)^{\sum_{k=i+1}^{i+j-1} n_k} \quad (4.26)$$

met $n_k = n_{k,\uparrow} + n_{k,\downarrow}$ het aantal deeltjes op site k . Een even aantal deeltjes tussen i en $i + j$ zal dus geen invloed hebben, terwijl een oneven aantal deeltjes aanleiding geeft tot een extra fase -1 . In het voorbeeld van hierboven heeft $a_{2,\uparrow}^\dagger a_{5,\uparrow}$ dezelfde matrixvoorstelling als $a_{2,\uparrow}^\dagger a_{3,\uparrow}$ maar gecorrigeerd met een factor $(-1)^3$ omdat er drie spins tussen site 2 en site 5 aanwezig zijn. Door deze correctie wordt er (samen met de matrixvoorstellungen (4.21)) steeds voldaan aan de fundamentele anti-commutatierelaties. Dit kan geïmplementeerd worden door op te merken dat de MPS golf functie van een fermionisch systeem kan voorgesteld worden als

$$|\psi\rangle = \sum_{s_1 \dots s_L} M^{s_1} \dots M^{s_L} |s_1 \dots s_L\rangle \quad (4.27)$$

waarbij $s_i = \{|0\rangle, |\uparrow\rangle, |\downarrow\rangle, |\uparrow\downarrow\rangle\} \equiv \{1, 2, 3, 4\}$ en M^{s_i} opnieuw matrices zijn. Dat betekent dat wanneer s_i in de toestand 2 of 3 zit (dus als er maar 1 deeltje aanwezig is op site i) er voor site i een fasefactor -1 in rekening moet gebracht worden. In het algoritme volstaat het dus

om bij de constructie van de Y -blokken en bij het uitrekenen van de interactie tussen linker- en rechterblok in de effectieve Hamiltoniaan de corresponderende M^{s_i} te vermenigvuldigen met -1 indien $s_i = 2, 3$. Bij de constructie van de Y -blokken moet de fasefactor in rekening gebracht worden als de volgende site aan het vorige Y -blok gekleefd wordt. We hebben in dat geval contracties van de vorm

$$\sum_{k,m,\sigma_l} ((Y_{l-1}^L)_{\alpha+1}^n)_{k,m} M_{m,j}^{\sigma_l} M_{k,i}^{\sigma_l*} \quad (4.28)$$

die voor fermionen aangepast worden tot

$$\sum_{k,m,s_l} ((Y_{l-1}^L)_{\alpha+1}^n)_{k,m} M_{m,j}^{s_l} M_{k,i}^{s_l*} (-1)^{P(s_l)} \quad (4.29)$$

waarbij

$$P(s_l) = 2(\delta_{s_l,1} + \delta_{s_l,4}) + (\delta_{s_l,2} + \delta_{s_l,3}). \quad (4.30)$$

Bij de sommatie over s_l zal er op die manier automatisch rekening gehouden worden met de extra fasefactor. Een analoge aanpassing gebeurt bij het uitrekenen van de interactie-term tussen het linker- en rechterblok (waarbij de fasefactor in dit geval afkomstig is van de te optimaliseren site l) en bij de constructie van de Y^R -blokken. Deze aanpassingen zijn voldoende om alle extra fases in rekening te brengen. Men dient dit wel met enige voorzichtigheid aan te pakken, want niet alle operatoren geven aanleiding tot een fasefactor. Een number-operator $n_{i,s} = a_{i,s}^\dagger a_{i,s}$ bevat twee operatoren zodat het verplaatsen van een dergelijke operator over verschillende sites steeds twee fasefactoren per site zal geven omdat er telkens **twee** operatoren verschoven worden. Dit betekent dat er dus nooit een additionele fase in rekening gebracht moet worden (vermits $(-1)^{2\sum_{k=i+1}^{i+j-1} n_k} = 1$). Daarom is het nuttig om verschillende Hamiltonianen apart te behandelen (zie vorige sectie). De kostfunctie bevat bijvoorbeeld enkel number-operatoren zodat er nooit een additionele fase vereist is. De Hamiltoniaan met $a_{i,s}^\dagger a_{i+j,s}$ -termen heeft echter wel nood aan extra fasefactoren. Door beide apart te behandelen kunnen we meegeven aan het programma voor welke Hamiltonianen de fasefactoren in rekening moeten gebracht worden. Op die manier wordt alles netjes en correct behandeld.

4.3 Het Hubbard-model

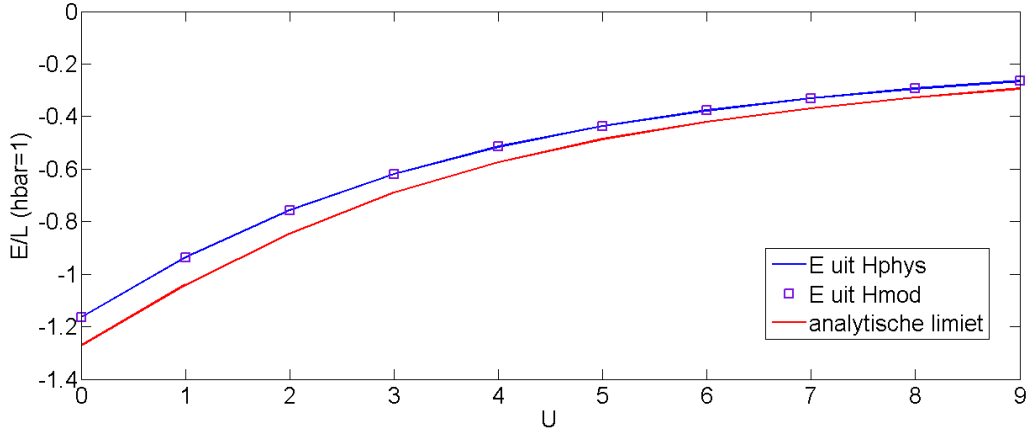
Om de accuraatheid van het programma te testen vertrekken we van een zeer befaamd model: het Hubbard model (in één dimensie). Dit model beschouwt dichtste-nabuur interacties (T) tussen fermionen en on-site repulsie (U). De Hubbard Hamiltoniaan ([29]) wordt gegeven door

$$H = -T \sum_{s \in (\uparrow, \downarrow)} \sum_{i=1}^{L-1} \left(a_{i,s}^\dagger a_{i+1,s} + a_{i+1,s}^\dagger a_{i,s} \right) + U \sum_{i=1}^L a_{i,\uparrow}^\dagger a_{i,\uparrow} a_{i,\downarrow}^\dagger a_{i,\downarrow}. \quad (4.31)$$

Additionele fasefactoren spelen hier nog geen rol aangezien we enkel dichtste-nabuur interacties beschouwen. In ([29]) wordt er aangetoond dat er (in de limiet $L \rightarrow \infty$) een exacte oplossing bestaat in het geval van half-filling, dat betekent dat het aantal deeltjes $N = \frac{N_{max}}{2} = L$. De exacte oplossing (voor $T = 1$ en $L \rightarrow \infty$) wordt gegeven door

$$\frac{E_0}{L} = -4 \int_0^\infty \frac{J_0(\omega) J_1(\omega)}{\omega \left(1 + \exp\left(\frac{U\omega}{2}\right) \right)} d\omega \quad (4.32)$$

waarbij $J_0(\omega)$ en $J_1(\omega)$ respectievelijk 0-de en 1-ste orde Bessel-functies zijn. Het integrandum valt vrij snel op nul en we plotten dit in MATLAB met behulp van een quadratuur-methode waarbij we de bovengrens van de integraal voldoende groot kiezen om het significante deel van het integrandum in rekening te kunnen brengen. Deze oplossing vergelijken we dan met het resultaat uit het VMPS-programma. De simulatie werd uitgevoerd met $L = N = 6$, $T = 1$ en $D = 16$, $\lambda = 10$ en het resultaat is te zien in figuur (4.2). Neem aan dat de Hamiltoniaan met de



Figuur 4.2: $\frac{E}{L}$ als functie van U voor een 1D-Hubbard model met $L = N = 6$, $T = 1$ en $D = 16$, $\lambda = 10$.

kostfunctie aanleiding geeft tot een golf functie $|\psi\rangle$. De blauwe curve geeft de verwachtingswaarde $\langle\psi|\hat{H}|\psi\rangle$ (steeds genoteerd als $\mathbb{H}\text{phys}$) terwijl de vierkantjes de verwachtingswaarde $\langle\psi|\hat{H}_N|\psi\rangle$ (steeds genoteerd als $\mathbb{H}\text{mod}$) geven. We zien dat er geen verschil is tussen beide oplossingen en dat het dus niet uitmaakt of we nu de fysische Hamiltoniaan of de gemodificeerde Hamiltoniaan gebruiken om uiteindelijk de energie te bepalen. We dienen echter wel op te merken dat dit enkel geldt als de kostparameter λ voldoende groot wordt gekozen zoals we zo dadelijk zullen zien. Daarnaast merken we ook nog op dat de energie boven de exacte energie ligt, wat niet onlogisch is aangezien de formule strikt gezien enkel geldt in de limiet $L \rightarrow \infty$. De vorm van de curve is echter wel hetzelfde zodat de asymptotische eigenschappen wel goed voorspeld worden. Ook het deeltjesaantal werd behouden voor elke waarde van U (iets wat we na elke simulatie controleerden via een extra functie die de verwachtingswaarde van de number-operator uitrekent). We berekenden daarnaast E/L ($\hbar = 1$) voor $U = 0$ en $D = 64$ voor grotere waarden van L en vergelijken dit met exacte resultaten (voor $U = 0$ heb je een ééndeeltjesprobleem en kan het exacte resultaat gevonden worden door diagonalisatie van een $L \times L$ matrix). In dit geval werd met $\lambda = 0$ ook steeds het deeltjesaantal behouden. Het resultaat is weergegeven in tabel (4.1). We zien een goede overeenkomst tussen de exacte waarden en de waarden bekomen met het VMPS programma. Voor $L = 6$ verwachten we een perfecte overeenkomst met het exacte resultaat omdat in dat geval $D = d^{\lfloor \frac{L}{2} \rfloor}$. Dit is inderdaad het geval (zie tabel (4.1)).

4.3.1 Invloed van de kostparameter λ

We wensen nu ook even stil te staan bij de invloed van de kostparameter. We simuleerden het Hubbard model voor verschillende waarden van λ en bespreken hieronder kort de bevindingen.

L	$E_{exact}/L(*)$	E_{VMPS}/L
6	-1.16465306914	-1.16465306914
10	-1.20533483667	-1.20533346501
20	-1.23814899997	-1.23812247319
30	-1.24957737436	-1.24950078441
40	-1.25538985558	-1.25525451485
$\rightarrow \infty$	-1.27323764229 (**)	—

Tabel 4.1: $\frac{E}{L}$ als functie van L voor $U = 0$ ($D = 64, \lambda = 0$). (*) Waarden van S. Wouters. (**) Waarde uit formule (4.32).

Voor kleine waarden van λ ($\lambda < 10\epsilon$ met ϵ de relevante energieschaal van het fysisch systeem) valt het op dat het opdrijven van U ervoor zorgt dat $\langle \psi | \hat{H} | \psi \rangle$ en $\langle \psi | \hat{H}_N | \psi \rangle$ niet meer hetzelfde resultaat geven (het deeltjesaantal is in dat geval niet gelijk aan het gewenste aantal N). Daarnaast stijgt de energie als we λ opdrijven ($\lambda > 10\epsilon$) tot er een saturatie optreedt en het niet meer uitmaakt hoe groot we λ precies kiezen (λ mag wel niet oneindig groot worden). Dit valt te verklaren door het feit dat voor grotere U en te kleine λ het deeltjesaantal daalt. Het programma wil dus deeltjes uit het systeem werpen om op die manier de energie te doen dalen (want dan daalt ook de repulsie, die gekenmerkt wordt door de parameter U). Als λ niet groot genoeg is ten opzichte van U (of T omdat je de meest relevante energieschaal moet bekijken) dan zal de kostfunctie dus niet in staat zijn om het gewenste deeltjesaantal op te leggen. Daarom dat het belangrijk is om steeds een goede waarde voor λ te kiezen in functie van U en T . Een goede vuistregel is $\lambda \approx 10 \max(|T|, |U|), \forall T, U$ te nemen.

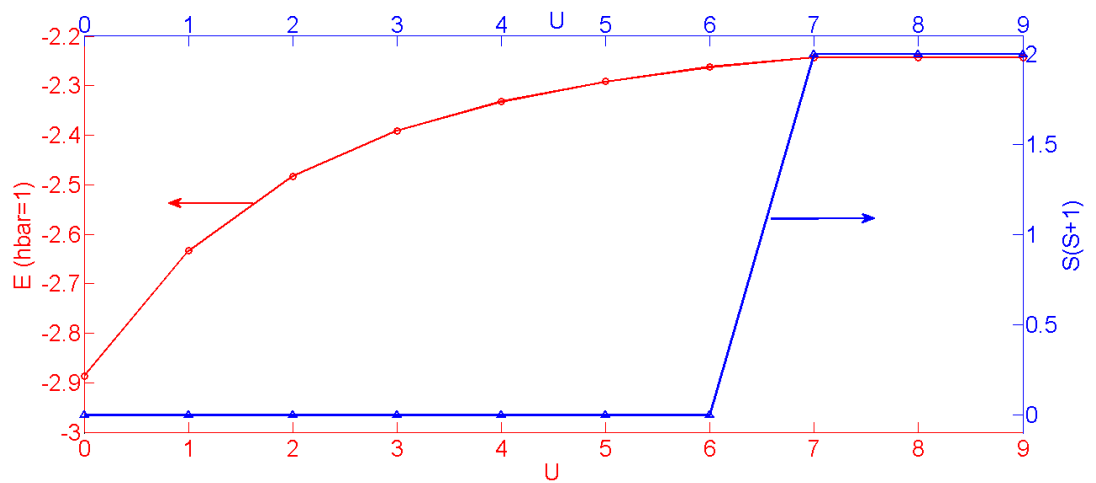
4.4 Het $t_1 - t_2$ model

Om te onderzoeken of de invloed van de additionele fasefactoren juist geïmplementeerd is, kijken we nu naar een meer geavanceerd model: het $t_1 - t_2$ model dat naast dichtste-nabuur ook tweede-orde interacties toelaat. De Hamiltoniaan van een $t_1 - t_2$ model ([30]) wordt gegeven door

$$H = - \sum_{s \in (\uparrow, \downarrow)} \sum_i \left(T_1 \left(a_{i,s}^\dagger a_{i+1,s} + a_{i+1,s}^\dagger a_{i,s} \right) + T_2 \left(a_{i,s}^\dagger a_{i+2,s} + a_{i+2,s}^\dagger a_{i,s} \right) \right) + U \sum_{i=1}^L a_{i,\uparrow}^\dagger a_{i,\uparrow} a_{i,\downarrow}^\dagger a_{i,\downarrow}. \quad (4.33)$$

In ([30]) vinden we een figuur met de energie en de grootte van de spin \hat{S}^2 als functie van U voor $L = 12$, $N = 6$ en $T_1 = 1$, $T_2 = -0.2$. Voor een bepaalde kritische waarde van U (U_c) wordt de energie onafhankelijk van U en vertoont $\langle \psi | \hat{S}^2 | \psi \rangle = S(S+1)$ een sprong van 0 naar $12 = 3(4)$. Het systeem gaat dan over naar de ferromagnetische toestand ([30]). De maximale spin wordt dan gegeven door $S = \frac{N}{2}$ (met N het aantal deeltjes). We pogen dit resultaat nu te reproduceren voor een meer bescheiden systeem met $L = 4$, $N = 2$ en $T_1 = 1$, $T_2 = -0.2$ ($D = 16, \lambda = 100$). We verwachten dat bij de transitie S gelijk wordt aan $S = 1$ (omdat we twee deeltjes in het systeem hebben) en dus $S(S+1) = 2$. Het resultaat wordt weergegeven in figuur (4.3). We vinden hetzelfde verband als in ([30]): een kritische waarde $U_c \approx 7$ die aangeeft wanneer de energie onafhankelijk wordt van U en bij dezelfde U_c een transitie van $S(S+1)$ van 0 naar 2 (wat we voorspeld hadden). We besluiten hieruit dat ook in het geval van long-range

fermionsystemen het programma naar behoren werkt.



Figuur 4.3: Energie (rode curve) en $S(S + 1)$ (blauwe curve) als functie van U voor een $t_1 - t_2$ model met $T_1 = 1$, $T_2 = -0.2$, $L = 4$, $N = 2$ en $\lambda = 100$.

Hoofdstuk 5

H_2 molecule

In het voorgaande hoofdstuk hebben we de VMPS-procedure uitgebreid tot fermionsystemen. De volgende logische stap is dan ook te kijken wat er gebeurt als we roosters van atomen bekijken. Het modelsysteem waarvan we vertrekken is de H_2 molecule. We associëren aan iedere site een $1s$ -orbitaal, zodat er op elke site slechts twee deeltjes kunnen zitten (net zoals in het Hubbard-model). Bij de studie van atomen op een rooster treden er echter belangrijke complicaties op en het doel van dit hoofdstuk is dan ook deze complicaties toe te lichten en een mogelijke oplossing voor te stellen. Hierbij zullen we wel trachten zoveel mogelijk technieken uit de vorige hoofdstukken te hergebruiken. We eindigen het hoofdstuk met een aantal algemene beschouwingen die toelaten om het algoritme uit te breiden tot grotere roosters ($L > 2$), de zogenaamde H -chains; de algemene gedachtengang die men moet volgen bij het bestuderen van moleculaire kettingen zal echter al duidelijk worden bij de studie van een H_2 molecule.

5.1 Hamiltoniaan van een moleculair systeem

Als we starten van de Born-Oppenheimer benadering dan wordt de (elektronische) Hamiltoniaan (in atomaire eenheden) van een rooster van N waterstofatomen gegeven door ([28]):

$$\hat{H}_e = -\frac{1}{2} \sum_i \nabla_i^2 + \sum_{i < j} \frac{1}{|\mathbf{r}_i - \mathbf{r}_j|} - \sum_{i=1}^N \sum_{A=1}^M \frac{1}{|\mathbf{r}_i - \mathbf{R}_A|} \quad (5.1)$$

waarbij \mathbf{r}_i de positiecoördinaat van het i -de elektron is en \mathbf{R}_A de positiecoördinaat van de A -de kern. De volledige Hamiltoniaan bevat daarnaast nog de repulsie tussen de kernen (die we echter vast beschouwen binnen het Born-Oppenheimer concept):

$$H_{kernen} = \sum_{A=1}^{M-1} \sum_{B=A+1}^M \frac{1}{|\mathbf{R}_A - \mathbf{R}_B|} \quad (5.2)$$

zodat de totale energie gegeven wordt door

$$E_e + \sum_{A=1}^{M-1} \sum_{B=A+1}^M \frac{1}{|\mathbf{R}_A - \mathbf{R}_B|} \quad (5.3)$$

met E_e de energie overeenkomend met \hat{H}_e . In het geval dat we hier beschouwen zijn er evenveel kernen als elektronen ($N = M = 2$). We herschrijven de Hamiltoniaan \hat{H}_e in tweede kwantisatie

(ingevoerd in vorig hoofdstuk):

$$\hat{H} = \sum_{\alpha\beta} \langle \alpha | O | \beta \rangle a_{\alpha}^{\dagger} a_{\beta} + \frac{1}{2} \sum_{\alpha\beta\gamma\delta} (\alpha\beta | V | \gamma\delta) a_{\alpha}^{\dagger} a_{\beta}^{\dagger} a_{\delta} a_{\gamma}. \quad (5.4)$$

waarbij $\{\alpha\} = \{\alpha_1, \dots, \alpha_N\}$ een *orthonormaal set* van eendeeltjestoestanden is ([28]). De kwantumgetallen per site α_i bevatten naast de spincoördinaat ook nog de radiale coördinaten. Naast de spin van het elektron moeten we immers ook rekening houden met het spatiale orbitaal waarin het elektron zich bevindt. In deze discussie beperken we ons zoals gezegd tot 1s-orbitalen. De operator \hat{O} bevat naast de kinetische energie van de elektronen ook de interactie-energie tussen de kernen en de elektronen en de operator \hat{V} bevat de interactie-energie tussen de elektronen. De matrixelementen worden dan gegeven door

$$\begin{aligned} \langle \alpha_i | O | \beta_j \rangle &= \delta_{m_{\alpha_i}, m_{\beta_j}} \int d\mathbf{r} \varphi_{\alpha_i}^*(\mathbf{r}) \hat{O} \varphi_{\beta_j}(\mathbf{r}) \\ (\alpha_i \beta_j | V | \gamma_k \delta_l) &= \delta_{m_{\alpha_i}, m_{\gamma_k}} \delta_{m_{\beta_j}, m_{\delta_l}} \int d\mathbf{r}_1 d\mathbf{r}_2 \varphi_{\alpha_i}^*(\mathbf{r}_1) \varphi_{\beta_j}^*(\mathbf{r}_2) \hat{V} \varphi_{\gamma_k}(\mathbf{r}_1) \varphi_{\delta_l}(\mathbf{r}_2). \end{aligned}$$

Voor de spatiale orbitalen φ gebruiken we Slater-orbitalen uit de STO-6G basis:

$$\varphi_{\alpha_k}(\mathbf{r}) = \sum_{i=1}^6 n_i c_i \exp\left(-\theta_i |\mathbf{r} - \mathbf{R}_{\alpha_k}|^2\right) \quad (5.5)$$

waarbij c_i en θ_i uit tabellen gevonden worden ([31]) en de n_i bepaald worden uit

$$n_i^2 \int d\mathbf{r} \exp(-2\theta_i r^2) = 1. \quad (5.6)$$

Deze relatie kan makkelijk uitgewerkt worden door gebruik te maken van

$$\int_0^{\infty} \sqrt{u} \exp(-u) du = \Gamma\left(\frac{3}{2}\right) = \frac{1}{2} \Gamma\left(\frac{1}{2}\right) = \frac{\sqrt{\pi}}{2}. \quad (5.7)$$

Op die manier bekommen we na enig rekenwerk

$$n_i = \left(\frac{2\theta_i}{\pi}\right)^{\frac{3}{4}} \quad (5.8)$$

zodat

$$\varphi_{\alpha_k}(\mathbf{r}) = \sum_{i=1}^6 \left(\frac{2\theta_i}{\pi}\right)^{\frac{3}{4}} c_i \exp(-\theta_i |\mathbf{r} - \mathbf{R}_{\alpha_k}|). \quad (5.9)$$

Let op dat we deze orbitalen nog moeten orthonormaliseren (zie verderop).

De vorm van de Hamiltoniaan (5.4) maakt meteen duidelijk waar de problemen schuilen. Nemen we als lokale basis immers opnieuw (zoals bij het Hubbard model)

$$\{|0\rangle, |\uparrow\rangle, |\downarrow\rangle, |\uparrow\downarrow\rangle\}$$

dan hebben we een product van vier operatoren en geen twee. Dit betekent een enorme toename in rekentijd en we wensen de rekentijd te beperken. We zullen dus een methode moeten ontwikkelen die zowel snel is als een voldoende accuraat resultaat geeft. Bovendien zijn de koppelconstanten uit de vorige hoofdstukken hier vervangen door ruimte-integralen. Daarnaast moeten we ook hier een gewenst deeltjesaantal opleggen (in ons geval $N = 2$ voor de H_2 molecule).

5.2 Bepaling van de ruimte-integralen

Om moleculaire kettingen te beschrijven dienen we als eerste aanpassing de gebruikte basisfuncties (dus de STO-6G basis in dit geval) te orthonormaliseren. De orbitalen worden geassocieerd aan een bepaald atoom (positie van de kern) met als gevolg dat de overlap tussen twee orbitalen gecentreerd op verschillende atomen niet nul zal zijn. We moeten de basis dus eerst transformeren naar een orthonormale basis. In ([32]) wordt aangetoond (via een kleinste kwadraten methode) dat de basisfuncties $\varphi_{\alpha_k}(\mathbf{r}) \equiv \varphi_k(\mathbf{r})$ (we kijken nu enkel naar de spatiale afhankelijkheid) kunnen geschreven worden in termen van een set orthonormale functies $\psi_j(\mathbf{r})$. We zullen de matrixelementen uitrekenen in de $|\psi_j\rangle$ basis en werken voor alle verdere berekeningen dus in de orthonormale basis (wat noodzakelijk is omdat er anders zaken binnen de tweede kwantisatie veranderen). Het verband tussen $\varphi_k(\mathbf{r})$ en $\psi_j(\mathbf{r})$ ([32]) wordt gegeven door

$$\begin{aligned}\psi_j(\mathbf{r}) &= \sum_k \varphi_k(\mathbf{r}) \left(\Delta^{-\frac{1}{2}} \right)_{k,j} \\ &= \sum_k \sum_{i=1}^6 n_i c_i \exp\left(-\theta_i |\mathbf{r} - \mathbf{R}_k|^2\right) \left(\Delta^{-\frac{1}{2}} \right)_{k,j}\end{aligned}\quad (5.10)$$

waarbij

$$\Delta_{k,j} = \langle \varphi_k | \varphi_j \rangle \quad (5.11)$$

de overlap-matrix is tussen de niet-orthogonale basisfuncties. De matrix $\Delta^{-\frac{1}{2}}$ wordt bekomen door de eigenwaarden (gecollecteerd in de diagonaalmatrix Σ) en de eigenvectoren (gecollecteerd in de matrix V) van Δ te berekenen en $\Delta^{-\frac{1}{2}}$ vervolgens te schrijven als

$$\Delta^{-\frac{1}{2}} = V \Sigma^{-\frac{1}{2}} V^{-1}. \quad (5.12)$$

De bovenstaande transformatie (5.10) staat ook gekend als de Lowdin transformatie. In het vervolg noteren we $\Delta^{-\frac{1}{2}}$ als Λ .

Een volgende stap bestaat erin de ruimte-integralen uit te rekenen. Voor basisfuncties van het type

$$\exp\left(-\zeta(\mathbf{r} - \mathbf{R})^2\right), \quad \zeta \in \mathbb{R} \quad (5.13)$$

met \mathbf{r} de elektron-coördinaat en \mathbf{R} de positie van de kern, worden alle nodige ruimte-integralen uitgewerkt in ([33]). We nemen de resultaten uit dit artikel dan ook over, maar dienen er op te letten dat we hier 6 verschillende functies per kern ($\exp(-\zeta_a(\mathbf{r} - \mathbf{R})^2)$, $a = 1 \dots 6$) beschouwen. Concreet betekent dit dat er voor elke basisfunctie in de ruimte-integraal een som moet ingevoerd worden over deze 6 termen. De kinetische energie-integraal (die uitgerekend wordt over 2 basisfuncties) zal dan bijvoorbeeld bestaan uit $36 = 6^2$ contributies waarbij elke contributie gegeven wordt door een uitdrukking uit ([33]). We vatten hieronder de resultaten samen en merken nogmaals op dat we de ruimte-integralen uitrekenen voor $1s$ -orbitalen.

5.2.1 Kinetische energie

Voor de kinetische energie zijn matrixelementen van het type $T_{ij} = \langle \psi_i | \hat{T} | \psi_j \rangle$ vereist. Deze elementen zijn diagonaal in de spin zodat $m_{s_i} = m_{s_j}$. De ψ_i zijn de orthonormale basisfuncties,

maar in ([33]) worden de ruimte-integralen uitgerekend in de $\varphi_k(\mathbf{r})$ -basis zodat er geldt

$$\begin{aligned} T_{ij} &= \sum_{k,l} \Lambda_{k,i}^* \Lambda_{l,j} \langle \varphi_k | \hat{T} | \varphi_l \rangle \\ &= \sum_{k,l} \sum_{a,b=1}^6 \Lambda_{k,i}^* \Lambda_{l,j} c_a c_b n_a n_b \Delta_{k,l}^{a,b} \zeta_a^b \left(3 - 2\zeta_a^b |\mathbf{R}_k - \mathbf{R}_l|^2 \right) \end{aligned} \quad (5.14)$$

met

$$\varphi_k(\mathbf{r}) = \sum_{i=1}^6 n_i c_i \exp\left(-\theta_i |\mathbf{r} - \mathbf{R}_k|^2\right), \quad \xi_a^b = \frac{\theta_a \theta_b}{\zeta_a^b}, \quad \zeta_a^b = \theta_a + \theta_b \quad (5.15)$$

en

$$\Delta_{k,l}^{a,b} = \left(\frac{\pi}{\zeta_a^b} \right)^{\frac{3}{2}} \exp\left(-\xi_a^b |\mathbf{R}_k - \mathbf{R}_l|^2\right). \quad (5.16)$$

We merken op dat $T_{ij} = T_{ji}$; dit is van belang om geen overbodige berekeningen te doen.

5.2.2 Attractieve elektron-kern interactie

Vervolgens dienen we de interactie van de M kernen met de elektronen te beschouwen. Daartoe moeten we het matrixelement $U_{ij} = \sum_{m=1}^M \langle \psi_i | \frac{1}{|\mathbf{r} - \mathbf{R}_m|} | \psi_j \rangle$ uitwerken. Ook hiervoor vinden we analytische uitdrukkingen in ([33])

$$U_{ij} = \sum_{m=1}^M \sum_{k,l} \sum_{a,b=1}^6 \Lambda_{k,i}^* \Lambda_{l,j} c_a c_b n_a n_b \Delta_{k,l}^{a,b} \left(2\sqrt{\frac{\zeta_a^b}{\pi}} F_0(U_{a,b}^{k,l,m}) \right) \quad (5.17)$$

met

$$U_{a,b}^{k,l,m} = \zeta_a^b \left| \mathbf{P}_{a,b}^{k,l} - \mathbf{R}_m \right|^2, \quad \mathbf{P}_{a,b}^{k,l} = \frac{\theta_a \mathbf{R}_k + \theta_b \mathbf{R}_l}{\theta_a + \theta_b} \quad (5.18)$$

en

$$F_0(U) = \int_0^1 dt \exp(-Ut^2). \quad (5.19)$$

De overige symbolen werden reeds verklaard bij de kinetische energie en zullen ook in wat volgt nog gebruikt worden. Ook hier geldt dat $U_{ij} = U_{ji}$.

5.2.3 Repulsieve elektron-elektron interactie

Tot slot dienen we nog de repulsie tussen de elektronen in rekening te brengen. Deze term hangt af van vier kern-posities i, j, k, l en wordt genoteerd als $(ij|V|kl) = V_{ijkl}$ (we werken dus niet met het gesymmetriseerd matrixelement, zie Hoofdstuk 2 van ([28])). Met behulp van ([33]) vinden we dan

$$\begin{aligned} V_{ijkl} &= \sum_{m,n,p,q} \sum_{a,b,c,d=1}^6 \Lambda_{m,i}^* \Lambda_{n,j}^* \Lambda_{p,k} \Lambda_{q,l} c_a c_b c_c c_d n_a n_b n_c n_d \\ &\left(\left(\zeta_a^c + \zeta_b^d \right)^{-\frac{1}{2}} K(\theta_a, \theta_c, \mathbf{R}_m, \mathbf{R}_p) K(\theta_b, \theta_d, \mathbf{R}_n, \mathbf{R}_q) F_0(T) \right) \end{aligned} \quad (5.20)$$

met

$$K(\theta_i, \theta_j, \mathbf{A}, \mathbf{B}) = \frac{\sqrt{2\pi}^{\frac{5}{4}}}{\zeta_i^j} \exp\left(-\frac{\theta_i \theta_j}{\zeta_i^j} |\mathbf{A} - \mathbf{B}|^2\right) \quad (5.21)$$

en

$$T = \frac{\zeta_a^c \zeta_b^d}{\zeta_a^c + \zeta_b^d} |\mathbf{R} - \mathbf{Q}|^2, \quad \mathbf{R} = \frac{\theta_a \mathbf{R}_m + \theta_c \mathbf{R}_p}{\zeta_a^c}, \quad \mathbf{Q} = \frac{\theta_b \mathbf{R}_n + \theta_d \mathbf{R}_q}{\zeta_b^d}. \quad (5.22)$$

Ook de V_{ijkl} bevat een aantal belangrijke symmetrieën wegens het reëel zijn van de golffuncties:

$$V_{ijkl} = V_{kjil} = V_{ilkj} = V_{jilk} = V_{klij} = V_{lijk} = V_{jkli} = V_{lkji}. \quad (5.23)$$

De repulsieve interactie zal de rekentijd enorm opdrijven als gevolg van de 4-center integralen. We kunnen echter een zinvolle vereenvoudiging doen zonder de fysica al te veel te veranderen. Van zodra de atomaire orbitalen voldoende ver van elkaar verwijderd zijn daalt de waarde van de interactie vrij snel. Het is dan ook overbodig om deze termen verder mee te nemen in de berekeningen. De fout die men hierdoor maakt blijft klein indien we de drempel zinvol kiezen. We kunnen bijvoorbeeld stellen dat orbitalen die L_{th} van elkaar verwijderd zijn bijna geen invloed meer hebben op elkaar wegens de screening van de tussenliggende orbitalen. Als gevolg daarvan zullen we alle interactietermen tussen sites die verder dan L_{th} van elkaar liggen niet meer berekenen. Op die manier kan de rekentijd beperkt worden en behouden we toch nog een accuraat resultaat (als L_{th} goed gekozen is; je mag de termen pas verwaarlozen als ze voldoende klein geworden zijn, bijvoorbeeld van de orde 10^{-6} , maar die keuze hangt natuurlijk af van de gewenste accuraatheid).

5.3 De tweedeeltjesinteractie \hat{V}

De eendeeltjes-operatoren \hat{T} en \hat{U} geven in tweede kwantisatie analoge Hamiltonianen als in het Hubbard model (waarbij de koppelsterktes nu bepaald worden door de ruimte-integralen). Voor deze termen kunnen we het fermionisch algoritme hergebruiken, maar voor de \hat{V} interactie zullen we een aangepast formalisme moeten ontwikkelen omdat we een product hebben van 4 operatoren ($a_\alpha^\dagger a_\beta^\dagger a_\delta a_\gamma$) en geen 2 zoals bij het Hubbard model. Om het geheel op een compacte en nette manier te implementeren zullen we

$$\frac{1}{2} \sum_{\alpha\beta\gamma\delta} (\alpha\beta|V|\gamma\delta) a_\alpha^\dagger a_\beta^\dagger a_\delta a_\gamma \quad (5.24)$$

herschikken in een *geordende* som (analoog aan de $\sum_{i<j}$ uit het Hubbard model) teneinde een iteratieve procedure mogelijk te maken waarbij het volgende blok uit het vorige geconstrueerd kan worden. We identificeren eerst de relevante kwantumgetallen $|\alpha\rangle \equiv |i, s\rangle$ met i de positie van de i -de kern en s de spin van een elektron in het orbitaal geassocieerd aan de i -de site. Aangezien de V -interactie spin-onafhankelijk is geldt er meteen

$$(\alpha\beta|V|\gamma\delta) \equiv ((i, s)(j, s')|V|(k, s_3)(l, s_4)) = \delta_{s, s_3} \delta_{s', s_4} (ij|V|kl). \quad (5.25)$$

We zullen de interactie nu opdelen in een stuk bestaande uit operatoren die slechts op één site inwerken, een stuk met operatoren die op twee sites inwerken, een stuk dat op drie sites inwerkt en tot slot een stuk dat op vier sites inwerkt. We zullen steeds meer en meer geordende producten invoeren en een gepaste herlabeling toepassen om zo tot een geordende som te komen. In het onderstaande noteren we $a_{i,s}^\dagger a_{j,s'}^\dagger a_{k,s} a_{l,s'}$ als $-a_{i,s}^\dagger a_{j,s'}^\dagger a_{l,s'} a_{k,s}$ als f_{ijkl} (de (s, s') indices worden weggelaten maar we letten er wel op dat ze in onderstaande afleiding steeds in de volgorde

$(s - s' - s - s')$ blijven staan in f_{ijkl} ; de eerste en derde operator hebben dus steeds een spin s en de andere twee een spin s'). De eerste 2 benedenindices in f_{ijkl} hebben betrekking op de creatieoperatoren en de laatste 2 op de annihilatieoperatoren. Zo is bijvoorbeeld $a_{k,s}^\dagger a_{j,s'}^\dagger a_{l,s} a_{i,s'}$ gelijk aan f_{kjli} . Op die manier kan men de tweedeeltjesinteractie als volgt herschrijven (de som over (s, s') en de factor $-\frac{1}{2}$ worden vanaf de tweede regel weggelaten en pas in de laatste regel terug ingevoerd)

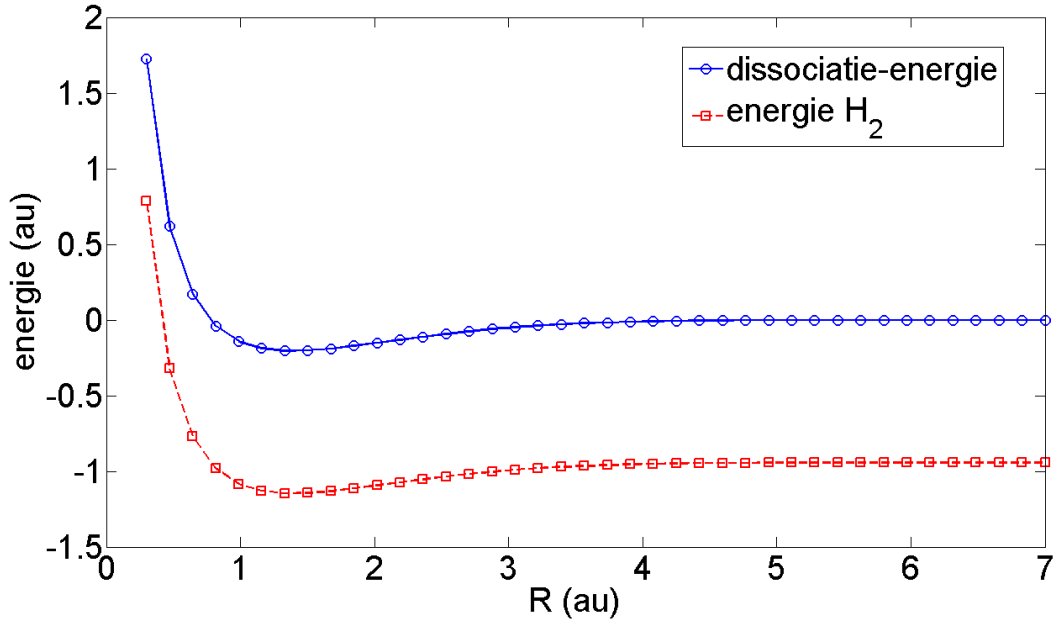
$$\begin{aligned}
\hat{V} &= -\frac{1}{2} \sum_{i,j,k,l} \sum_{s,s'} V_{ijkl} f_{ijkl} \\
&= \sum_{i,j} \left(\sum_{k<l} + \sum_{k>l} + \sum_{k=l} \right) V_{ijkl} f_{ijkl} \\
&= \sum_{i,j,k} V_{ijkk} f_{ijkk} + \sum_{i,j} \sum_{k<l} (V_{ijkl} f_{ijkl} + V_{ijlk} f_{ijlk}) \\
&= \sum_{i,j} V_{ijjj} f_{ijjj} + \sum_i \sum_{j<k} (V_{ijkk} f_{ijkk} + V_{ikjj} f_{ikjj}) + \\
&\quad \sum_i \left(\sum_{j<k<l} + \sum_{(j=k)<l} + \sum_{k<j<l} + \sum_{k<(j=l)} + \sum_{k<l<j} \right) (V_{ijkl} f_{ijkl} + V_{ijlk} f_{ijlk}) \\
&= \dots \\
&= -\frac{1}{2} \sum_{s,s'} \left(\sum_i V_{iiii} f_{iiii} + \sum_{i<j} V_{ij}^2 + \sum_{i<j<k} V_{ijk}^3 + \sum_{i<j<k<l} V_{ijkl}^4 \right) \tag{5.26}
\end{aligned}$$

Als we steeds ordening blijven invoeren dan bekommen we uiteindelijk vier bijdragen voor de tweedeeltjesinteractie. De eerste bijdrage werkt lokaal op site i (met sterkte V_{iiii}) terwijl de tweede bijdrage V_{ij}^2 alle interactietermen met twee sites (i, j) bevat. Deze bijdragen zijn dus volledig analoog als in het Hubbard model en men kan dan ook zonder meer het fermionprogramma hergebruiken. De bijdragen V_{ijk}^3 en V_{ijkl}^4 bevatten echter de interactie van drie respectievelijk vier sites en zijn volledig nieuw in het opzicht dat er geen X en Y blokken zijn in het bestaande programma die in staat zijn om deze termen te implementeren. We vermelden hier ook dat V^3 maar liefst $\delta_3 = 60$ verschillende interactietermen bevat en V^4 heeft er $\delta_4 = 48$ (ter vergelijking: het isotroop Heisenbergmodel heeft er maar $\delta = 3$). Voor het geval dat wij bestuderen, de H_2 molecule, hebben we echter enkel de V_{ij}^2 bijdrage nodig (die volledig uitgewerkt staat in Appendix E). We zullen op het einde wel aangeven hoe deze V^3 en V^4 bijdragen kunnen geïmplementeerd worden.

5.4 Energie en dissociatiecurve van een H_2 molecule

In dit onderdeel trachten we de accuraatheid van het programma te testen door de resultaten te vergelijken met resultaten afkomstig van Full Configuration Interaction (FCI) berekeningen. De overeenkomst met experimentele waarden zal over het algemeen niet goed zijn omdat we in een STO-6G basis werken (een minimale basisset). We zullen daarom alles vergelijken met FCI berekeningen in diezelfde STO-6G basis (omdat we met $D = 4$ in dit geval de volledige Hilbertruimte beschouwen).

We berekenen de energie en de dissociatiecurve van een H_2 molecule. Het resultaat is te vinden in figuur (5.1) en met R noteren we de interatomaire afstand (alle resultaten staan in atomaire eenheden). We zien dat de asymptotische waarden van de dissociatiecurve goed



Figuur 5.1: Energie en dissociatiecurve van een H_2 molecule in atomaire eenheden, $D = 4$.

overeenkomen met de exacte curve (Hoofdstuk 10 van ([28])): sterke repulsie voor korte interatomaire afstanden en convergentie naar nul voor grote interatomaire afstanden. De energie (rode vierkanten) van de H_2 molecule vertoont een minimum bij $E_{min} = -1.145977$ Hartree (of dus $E_{min} = -31.183180$ eV) bij een intermoleculaire afstand van 0.73503102 \AA . De dissociatie-energie is gelijk aan $E_{de} = -5.527$ eV wat ongeveer 1 eV afwijkt ten opzichte van de experimentele waarde (-4.462604 eV ([28])). Wegens onze opmerking van hierboven is dit dus niet onlogisch. Als we het minimum in de energie E_{min} vergelijken met de waarde bekomen uit exacte diagonalisatie¹ (-1.14597774737 Hartree, deze waarde werd bekomen met ThING ([34])) dan vinden we een treffende overeenkomst. Ook voor andere interatomaire afstanden werd een uitstekende overeenkomst gevonden met FCI berekeningen.

5.5 Theoretische beschouwingen voor long-range systemen

In dit onderdeel bespreken we de basis theoretische concepten die nodig zijn om het algoritme uit te breiden tot long-range systemen. De belangrijkste aandachtspunten kwamen reeds aan bod bij de behandeling van een H_2 molecule, maar toch zijn er een aantal niet-triviale aanpassingen nodig om het geheel ook te laten werken voor moleculaire roosters met $L > 2$.

¹Via FCI berekening.

5.5.1 X/Y -blokken voor V^3 en V^4

Om het geheel op een overzichtelijke wijze te implementeren zullen we verschillende Y_L en Y_R blokken moeten invoeren, afhankelijk van het aantal operatoren dat erin vervat zit. De X -blokken bevatten nog steeds volledig gecontraheerde operatoren (en hebben dus geen externe operator-index meer). Voor V^3 zullen we dus Y -blokken hebben met één of twee operatoren en voor V^4 hebben we Y -blokken met één, twee of drie operatoren. De X -blokken voor V^3 bevatten dan drie operatoren (X^3) en die voor V^4 bevatten er vier (X^4). Vermits er een zeer groot aantal interactietermen is, zullen we een bovengrens η (de interactie-range) instellen om te vermijden dat er teveel contracties moeten uitgerekend worden. De interactie-range wordt als volgt bepaald: een X - of Y -blok kan nooit twee operatoren bevatten die verder uit elkaar liggen dan η . Neem bijvoorbeeld een X^3 -blok met zeven sites en $\eta = 3$, dan is het onmogelijk dat dit blok een contractie bevat tussen de operatoren (1,5 en j) met j een willekeurige andere site groter dan 5 omdat site 1 en 5 verder dan drie sites uit elkaar liggen. Zo kan er ook nooit een interactie zijn tussen site 2 en 6 of site 3 en 7. Door een geschikte definitie voor de Y -blokken kunnen we dit op een compacte manier implementeren.

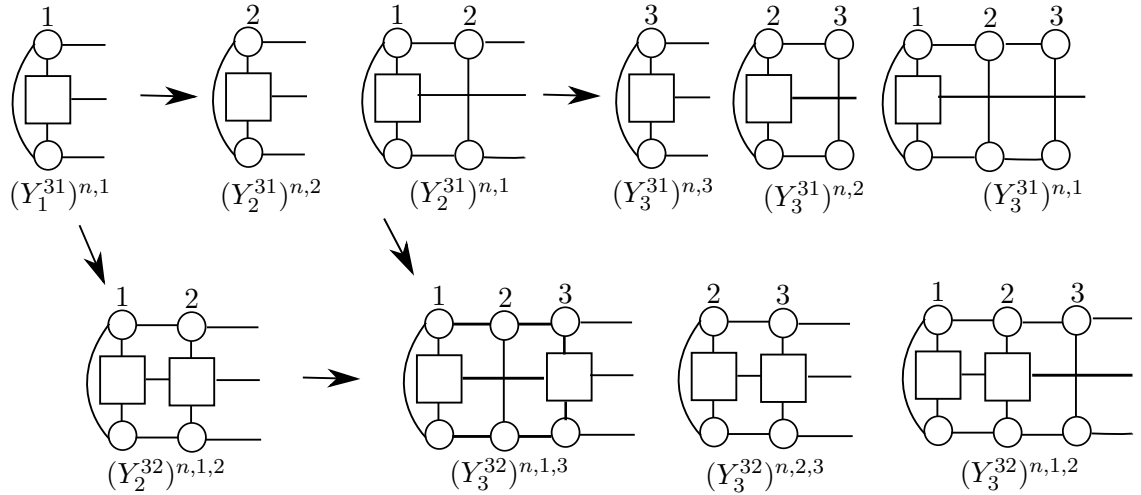
De Y -blokken

We dienen eerst de Y -blokken te construeren en merken op dat deze verschillend zijn voor V^3 en V^4 omdat V^3 en V^4 andere interactietermen bevatten. We voeren volgende notaties in voor Y -blokken die sites 1 tot en met l bevatten:

- $\left(Y_{L,l}^{31}\right)_{i,j}^{n,\alpha}$ is een linker Y -blok voor de V^3 interactie en het bevat slechts 1 site α waarop een operator aanwezig is. De index n kan waarden aannemen van 1 tot en met $\delta_3 = 60$ en komt dus overeen met één van de 60 interactietermen. De indices (i, j) zijn matrix-indices horend bij elke interactieterm ($n = 1 \dots \delta_3$).
- $\left(Y_{L,l}^{32}\right)_{i,j}^{n,\alpha,\beta}$ is een linker Y -blok voor de V^3 interactie en het bevat 2 sites α en β (bij constructie zal steeds $\alpha < \beta$) waarop een operator aanwezig is. De andere indices zijn reeds verklaard bij $\left(Y_{L,l}^{31}\right)_{i,j}^{n,\alpha}$. Bemerkt dat wegens de ingevoerde interactierange η er steeds geldt dat $\beta - \alpha \leq \eta$.
- $\left(Y_{L,l}^{41}\right)_{i,j}^{n,\alpha,\beta}$ is een linker Y -blok voor de V^4 interactie en het bevat 1 site α waarop een operator aanwezig is. De andere indices zijn reeds verklaard bij $\left(Y_{L,l}^{31}\right)_{i,j}^{n,\alpha}$, maar in dit geval loopt n van 1 tot $\delta_4 = 48$.
- $\left(Y_{L,l}^{42}\right)_{i,j}^{n,\alpha,\beta}$ is het linker Y -blok voor de V^4 interactie (analoge discussie als bij $\left(Y_{L,l}^{32}\right)_{i,j}^{n,\alpha,\beta}$)
- $\left(Y_{L,l}^{43}\right)_{i,j}^{n,\alpha,\beta,\gamma}$ is een linker Y -blok voor de V^4 interactie en het bevat 3 sites α , β en γ (bij constructie zal steeds $\alpha < \beta < \gamma$) waarop een operator aanwezig is. Er geldt dat $\beta - \alpha \leq \eta$ en $\gamma - \beta \leq \eta$ zodat $\gamma - \alpha \leq 2\eta$.

Voor elk Y -blok wordt een lijst bijgehouden die de sites bevat waarop een operator aanwezig is. Immers, in de constructie van de Y -blokken wordt de interactiesterkte $V_{P(ijkl)}$ nog niet in rekening gebracht zodat we moeten weten welke $V_{P(ijkl)}$ met welke term in een Y -blok moet

vermenigvuldigd worden. Hierbij is $P(ijkl)$ een permutatie van de indices $(ijkl)$ omdat voor een gegeven n de indices $(ijkl)$ in een andere volgorde kunnen staan wegens de herordening (de permutatie is dus een functie van n). Stel dat we bijvoorbeeld $(Y_{L,3}^{32})^{n,\alpha,\beta}$ bekijken, dan bevat dit blok alle termen die twee operatoren bevatten en vervat zijn in een blok met sites 1 tot en met 3. Dat betekent dat het koppel (α,β) de waarden $(1,2), (1,3), (2,3)$ kan aannemen (op voorwaarde dat $\eta \geq 2$). Het $(Y_{L,3}^{32})^{n,\alpha,\beta}$ -blok bevat in dit geval dus drie termen (niet gesommeerd, maar gewoon 3 aparte termen die elk nog indices (n,i,j) hebben) en de corresponderende site-lijst is in dat geval $(1,2), (1,3), (2,3)$. Een voorbeeld van de iteratieve constructie voor de Y^3 -blokken wordt gegeven in figuur (5.2). Het is dus duidelijk dat de termen met één operator



Figuur 5.2: Iteratieve constructie van de Y^3 -blokken.

iteratief opgebouwd worden uit de vorige termen met één operator samen met een nieuwe term en dat termen met twee operatoren opgebouwd worden uit de vorige termen met één en twee operatoren (ofwel contractie met een operator ofwel contractie met een site). De constructie voor de Y^4 -blokken is volkomen analoog met als extra stap de constructie van een blok met drie operatoren (die geconstrueerd wordt uit één van de vorige blokken met twee en drie operatoren). De constructie voor de rechterblokken verloopt volledig analoog aan de constructie van de linkerblokken (waarbij ook hier een lijst aangemaakt wordt die voor elke term opslaat welke sites er een operator bevatten).

Fases

Net zoals in het long-range fermion model moeten we fases in rekening brengen als we een site l (zonder operator) contraheren met een Y -blok. Voor de linkerblokken moeten we een fase -1 in rekening brengen als voor de gegeven $n = 1 \dots \delta_{3/4}$ het aantal operatoren links van l oneven is. Voor de rechterblokken wordt er een fase -1 in rekening gebracht als het aantal operatoren rechts van l oneven is. Het feit dat de fase afhankelijk wordt van n is een gevolg van de herordeningsprocedure. Voor V^3 zijn er bijvoorbeeld $\delta_3 = 60$ interactie-termen waarbij de operatoren in een geordende som staan. Algemeen geldt dat

$$V^3 = \sum_{i < j < k} V_{P(ijk)}^3 O_i O_j O_k \quad (5.27)$$

waarbij $O_{i,j,k}$ operatoren zijn op sites (i, j, k) . Soms bevat O_i twee operatoren (vb. een $a_{i,\uparrow}^\dagger a_{i,\uparrow}$) en moet er geen fase in rekening gebracht worden voor alle sites groter dan i en kleiner dan j . Vanaf j moet er dan wel een fase -1 in rekening gebracht worden omdat er in totaal vier operatoren zijn en er al twee op site i zitten, zodat de overige twee automatisch op j en k zitten (omdat je drie sites moet hebben waarop een operator zit). Van zodra site k bereikt is, moet er opnieuw geen fase meer in rekening gebracht worden omdat er nu vier operatoren links van site l zitten (in het geval we de constructie van linkerblokken beschouwen). Analoge redeneringen gaan op voor het geval dat er twee operatoren op site j of site k zitten. Voor de V^4 -interactie is er alternerend een fase $-1/1/-1/1$ voor de linkerblokken omdat elke site nu slechts één operator bevat.

De X -blokken

Voor de constructie van de X^3 -blokken zullen we gebruik maken van de vorige X^3 -blokken (die we dan contraheren met de betreffende site zoals vroeger) en de Y^{32} -blokken (die we contraheren met de derde operator, net zoals we vroeger een contractie beschouwden van de tweede operator met een Y -blok). Voor de X^4 -blokken zullen we gebruik maken van de vorige X^4 -blokken en de Y^{43} -blokken. We moeten er nu wel voor zorgen dat de juiste interactiesterktes $V_{P^n(ijkl)}$ in rekening gebracht worden (in de notatie hebben we reeds de n -afhankelijkheid van $P^n(ijkl)$ in rekening gebracht). Concreet kunnen we de iteratieve constructie van de **linker** X -blokken geassocieerd aan site l in formulevorm schrijven als (de onderindex L voor het linkerblok is weggelaten voor de duidelijkheid)

$$(X_l^3)_{i,j} = \sum_{k,m,s_l} (X_{l-1}^3)_{k,m} M_{m,j}^{s_l} M_{k,i}^{s_l^*} + \sum_{\substack{k,m,n \\ s_l, s_l', \alpha, \beta}} V_{P^n(\alpha\beta l)} (Y_{l-1}^{32})_{k,m}^{n,\alpha,\beta} M_{m,j}^{s_l} M_{k,i}^{s_l'^*} (O_l^3)_{s_l, s_l'}^n \quad (5.28)$$

$$(X_l^4)_{i,j} = \sum_{k,m,s_l} (X_{l-1}^4)_{k,m} M_{m,j}^{s_l} M_{k,i}^{s_l^*} + \sum_{\substack{k,m,n \\ s_l, s_l', \alpha, \beta, \gamma}} V_{P^n(\alpha\beta\gamma l)} (Y_{l-1}^{43})_{k,m}^{n,\alpha,\beta,\gamma} M_{m,j}^{s_l} M_{k,i}^{s_l'^*} (O_l^4)_{s_l, s_l'}^n. \quad (5.29)$$

Hierbij zijn $(O_l^3)_{s_l, s_l'}^n$ en $(O_l^4)_{s_l, s_l'}^n$ respectievelijk de derde en vierde operator uit de geordende som en hebben we in $V_{P^n(\alpha\beta l)}$ rekening gehouden met het feit dat er nu maar 3 sites zijn met een operator op. De interactiesterkte is dan functie van de site-index l en de indices (α, β) (of (α, β, γ) voor de V^4 interactie) van het Y -blok. Net als vroeger definiëren we gepaste randvoorwaarden voor de blokken zodat er geen extra contributies ontstaan (zo zijn de (X_l^3) -blokken bijvoorbeeld nul zolang we geen drie sites in een blok hebben). Een volledig analoge procedure kan gebruikt worden om de rechter X -blokken te construeren.

5.5.2 Constructie van de lokale matrices

Om het algoritme te kunnen implementeren moeten we beschikken over een matrixrepresentatie van de operatoren geassocieerd aan een bepaalde site. We zullen dit hieronder in enig detail uitwerken voor de V^3 interactie. Zoals reeds gezegd bevat deze interactie $\delta_3 = 60$ verschillende interactietermen. We lijsten er hieronder een aantal op en geven daarna aan hoe men voor deze

operatoren een geschikte matrixvoorstelling kan vinden.

$$a_{i,\uparrow}^\dagger a_{k,\downarrow}^\dagger a_{l,\uparrow} a_{l,\downarrow} \quad (5.30)$$

$$a_{i,\downarrow}^\dagger a_{k,\downarrow} a_{k,\uparrow} a_{l,\uparrow}^\dagger \quad (5.31)$$

$$a_{i,\uparrow} a_{i,\downarrow} a_{k,\uparrow}^\dagger a_{l,\downarrow}^\dagger \quad (5.32)$$

Zonder aan algemeenheid in te boeten stellen we $i = 1$, $k = 2$ en $l = 3$ (de index j wordt hier dus niet gebruikt, want we beschouwen de interactie tussen drie sites zodat één site inderdaad twee operatoren met dezelfde index (vb. $k = j$) bevat). Wanneer er sites tussen i, k, l liggen (en die sites bevatten uiteraard geen operator) dan is de enige aanpassing het in rekening brengen van een eventuele fase (zoals we reeds hiervoor besproken hebben). Om de matrixelementen op te stellen zullen we gebruik maken van de anti-commutatierelaties en de conventie voor de lokale basis. Als basis nemen we

$$\left\{ |0\rangle, a_{\uparrow}^\dagger |0\rangle = |\uparrow\rangle, a_{\downarrow}^\dagger |0\rangle = |\downarrow\rangle, a_{\uparrow}^\dagger a_{\downarrow}^\dagger |0\rangle = |\uparrow\downarrow\rangle \right\} \equiv \{1, 2, 3, 4\}. \quad (5.33)$$

Dit betekent dat als een site twee deeltjes bevat, we bij conventie aannemen dat links de spin up zit en rechts de spin down. Een algemene golf functie op een rooster van L sites (met $s_i = 1, 2, 3, 4$) wordt dan

$$|\psi\rangle = \sum_s M^{s_1} M^{s_2} \dots M^{s_L} |s_1 \dots s_L\rangle. \quad (5.34)$$

We wensen nu de matrixelementen $\langle\psi| \hat{H} |\psi\rangle$ te bepalen. We kijken eerst naar de term (5.30) en schrijven de relevante operatoren van $|\psi\rangle$ uit (de coëfficiënten M worden weggelaten want die zijn niet van belang in de bepaling van het matrixelement). We herschrijven de operatoren met gebruik van de anti-commutatierelaties en evalueren de matrixelementen uiteindelijk ook met behulp van deze relaties. Als gevolg van deze relaties zien we eerst en vooral dat er slechts een beperkt aantal kets zijn waarop (5.30) kan inwerken zonder nul te genereren. De operator $a_{3,\uparrow} a_{3,\downarrow}$ is enkel verschillend van nul als site 3 in de toestand $|\uparrow\downarrow\rangle$ zit. Daarnaast kan $a_{2,\downarrow}^\dagger$ enkel inwerken op site 2 als die site nog geen spin down deeltjes bevat en tot slot kan $a_{1,\uparrow}^\dagger$ enkel inwerken op site 1 als die site nog geen spin up bevat (Pauli). In totaal zijn er dus vier mogelijke basiskets die een niet-triviaal resultaat opleveren:

$$|0\rangle_1 |0\rangle_2 a_{3,\uparrow}^\dagger a_{3,\downarrow}^\dagger |0\rangle_3 \quad (5.35)$$

$$|0\rangle_1 a_{2,\uparrow}^\dagger |0\rangle_2 a_{3,\uparrow}^\dagger a_{3,\downarrow}^\dagger |0\rangle_3 \quad (5.36)$$

$$a_{1,\downarrow}^\dagger |0\rangle_1 |0\rangle_2 a_{3,\uparrow}^\dagger a_{3,\downarrow}^\dagger |0\rangle_3 \quad (5.37)$$

$$a_{1,\downarrow}^\dagger |0\rangle_1 a_{2,\uparrow}^\dagger |0\rangle_2 a_{3,\uparrow}^\dagger a_{3,\downarrow}^\dagger |0\rangle_3 \quad (5.38)$$

wat eenvoudig te controleren is via bovenstaande discussie. We onderzoeken nu in welke toestand de kets ((5.35) tot en met (5.38)) terechtkomen wanneer de operator (5.30) erop inwerkt en zullen daarbij vooral moeten letten op de fase van het matrixelement. Daartoe merken we op dat er geen fases geïntroduceerd worden wanneer we een vacuümtoestand $|0\rangle_{i=1,2,3}$ door de verschillende creatie- en annihilatie-operatoren trekken terwijl het permuteren van operatoren wel aanleiding geeft tot fasefactoren als gevolg van de anti-commutatierelaties. We werken dit nu in meer detail

uit voor (5.35):

$$\begin{aligned}
& a_{1,\uparrow}^\dagger a_{2,\downarrow}^\dagger a_{3,\uparrow} a_{3,\downarrow} \left(|0\rangle_1 |0\rangle_2 a_{3,\uparrow}^\dagger a_{3,\downarrow}^\dagger |0\rangle_3 \right) \\
&= \left(a_{1,\uparrow}^\dagger |0\rangle_1 \right) \left(a_{2,\downarrow}^\dagger |0\rangle_2 \right) \left(a_{3,\uparrow} a_{3,\downarrow} a_{3,\uparrow}^\dagger a_{3,\downarrow}^\dagger |0\rangle_3 \right) \\
&= |\uparrow\rangle_1 |\downarrow\rangle_2 \left(-a_{3,\uparrow} a_{3,\uparrow}^\dagger a_{3,\downarrow} a_{3,\downarrow}^\dagger |0\rangle_3 \right) \\
&= |\uparrow\rangle_1 |\downarrow\rangle_2 \left(- \left(1 - a_{3,\uparrow}^\dagger a_{3,\uparrow} \right) \left(1 - a_{3,\downarrow}^\dagger a_{3,\downarrow} \right) |0\rangle_3 \right) \\
&= |\uparrow\rangle_1 |\downarrow\rangle_2 (-|0\rangle_3). \tag{5.39}
\end{aligned}$$

Wanneer (5.30) inwerkt op (5.35) kom je dus terecht in de toestand $|\uparrow\rangle_1 |\downarrow\rangle_2 |0\rangle_3$. Dit resultaat geeft aanleiding tot de volgende matrixelementen (voor de respectieve lokale operatoren op sites 1, 2, 3):

$$\begin{aligned}
\langle \uparrow | a_{1,\uparrow}^\dagger | 0 \rangle &= 1 \\
\langle \downarrow | a_{2,\downarrow}^\dagger | 0 \rangle &= 1 \\
\langle 0 | a_{3,\uparrow} a_{3,\downarrow} | \uparrow \downarrow \rangle &= -1. \tag{5.40}
\end{aligned}$$

Het volstaat nu om de matrixelementen geassocieerd aan de inwerking van (5.30) op (5.38) te bepalen om alle mogelijke niet-triviale matrixelementen te kennen. Immers, de termen (5.36) en (5.37) bevatten steeds een combinatie van een vacuümtoestand (zoals hiervoor) gecombineerd met een spin up of down (waarvan we zo dadelijk de matrixelementen zullen afleiden).

$$\begin{aligned}
& a_{1,\uparrow}^\dagger a_{2,\downarrow}^\dagger a_{3,\uparrow} a_{3,\downarrow} \left(a_{1,\downarrow}^\dagger |0\rangle_1 a_{2,\uparrow}^\dagger |0\rangle_2 a_{3,\uparrow}^\dagger a_{3,\downarrow}^\dagger |0\rangle_3 \right) \\
&\stackrel{(*)}{=} \left((-1)^3 a_{1,\uparrow}^\dagger a_{1,\downarrow}^\dagger |0\rangle_1 \right) \left((-1)^2 a_{2,\downarrow}^\dagger a_{2,\uparrow}^\dagger |0\rangle_2 \right) (-|0\rangle_3) \\
&= (-|\uparrow\downarrow\rangle) |\downarrow\uparrow\rangle_2 (-|0\rangle_3) \\
&\stackrel{(**)}{=} (-|\uparrow\downarrow\rangle_1) (-|\uparrow\downarrow\rangle_2) (-|0\rangle_3) \tag{5.41}
\end{aligned}$$

Hierbij hebben we in (*) gebruik gemaakt van $\{O_i, O_j\} = 0$ als $i \neq j$ voor $O_{i,j} = a_{i,j}^\dagger$ of $O_{i,j} = a_{i,j}$ zodat het *passeren* van p operatoren in dat geval aanleiding geeft tot een fasefactor $(-1)^p$. Dit heeft ook voor gevolg dat als O_j bestaat uit twee operatoren (bijvoorbeeld een $a^\dagger a$ -combinatie) er geen fasefactoren geïntroduceerd worden omdat elk van de twee operatoren aanleiding geeft tot een fase (-1) zodat je dus een factor $((-1)^2)^n = 1, \forall n$ krijgt. Daarnaast hebben we in (**) de definitie van onze basis gehanteerd: de vierde basisvector is immers $|\uparrow\downarrow\rangle = -|\downarrow\uparrow\rangle$. Nu zijn we in staat om de matrices geassocieerd aan de operator (5.30) in de lokale basis uit te schrijven (let op de volgorde van de basisvectoren om de juiste rijen en kolommen te vullen)

$$a_{1,\uparrow}^\dagger = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \end{pmatrix}, a_{2,\downarrow}^\dagger = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{pmatrix}, a_{3,\uparrow} a_{3,\downarrow} = \begin{pmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}. \tag{5.42}$$

Let wel op dat deze matrices enkel gelden voor de specifieke combinatie (5.30); voor een andere combinatie zoals bijvoorbeeld (5.31) zullen we andere matrices krijgen. Bij de constructie van alle matrices dienen we dus vooral te letten op fasefactoren geïntroduceerd door het omwisselen

van operatoren enerzijds en fasefactoren geïntroduceerd door de keuze van de basis anderzijds. Met behulp van voorgaande discussie kan men nu controleren dat de matrixvoorstelling van (5.31) gegeven wordt door

$$a_{1,\downarrow}^\dagger = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}, a_{2,\downarrow} a_{2,\uparrow} = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, a_{3,\uparrow}^\dagger = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad (5.43)$$

en deze van (5.32) door

$$a_{1,\uparrow} a_{1,\downarrow} = \begin{pmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, a_{2,\uparrow}^\dagger = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \end{pmatrix}, a_{3,\downarrow}^\dagger = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{pmatrix}. \quad (5.44)$$

5.5.3 Energie van een algemene H-chain

Nu we besproken hebben hoe de verschillende blokken geconstrueerd worden, kunnen we ook de contributie van deze blokken tot de energie uitwerken. De V^3 en V^4 interacties zullen aanleiding geven tot een aanzienlijk aantal nieuwe termen (die allemaal moeten opgenomen worden in de iteratieve eigenvaluesolver). We lijsten ze hieronder voor de volledigheid allemaal op:

- $X_{L,l-1}^3$ en $X_{L,l-1}^4$ met site l
- $X_{R,l+1}^3$ en $X_{R,l+1}^4$ met site l
- $(Y_{L,l-1}^{32})$ met site l
- $(Y_{R,l+1}^{32})$ met site l
- $(Y_{L,l-1}^{43})$ met site l
- $(Y_{R,l+1}^{43})$ met site l
- $(Y_{L,l-1}^{3A})$ met $(Y_{R,l+1}^{3B})$, waarbij $A + B = 3$
- $(Y_{L,l-1}^{31})$ met site l en $(Y_{R,l+1}^{31})$
- $(Y_{L,l-1}^{4A})$ met $(Y_{R,l+1}^{4B})$, waarbij $A + B = 4$
- $(Y_{L,l-1}^{4A})$ met site l en $(Y_{R,l+1}^{4B})$, waarbij $A + B = 3$

Bij de contractie van dergelijke blokken moeten we opnieuw (waar nodig) rekening houden met additionele fasefactoren zoals we reeds vroeger besproken hebben. Op dit moment beschikken we over alle objecten om het algoritme uit te breiden tot long-range moleculaire systemen en de implementatie kan zeer analoog verlopen als deze bij de H_2 molecule.

Hoofdstuk 6

Conclusies

In dit hoofdstuk zullen we de verschillende resultaten kort en bondig proberen samenvatten.

In Hoofdstuk 2 hebben we aangetoond dat Matrix Product States (MPSs) uitermate geschikt zijn voor de beschrijving van ééndimensionale systemen op een rooster met lengte L . Een belangrijk concept daarbij is de truncatiedimensie D die zegt hoe groot we de Hilbertruimte laten worden. Er werd bewezen dat in het geval $D = d^{\lfloor \frac{L}{2} \rfloor}$ de methode exact is (met d de fysische dimensie). Met behulp van de MPSs hebben we de Schrödingervergelijking geassocieerd aan een sterk gecorreleerd spinsysteem herschreven tot een eigenwaardeprobleem (dit is ook mogelijk voor fermionische systemen). Door de energie te schrijven in een X/Y -blokvorm werd het bovendien duidelijk dat we een iteratieve procedure konden ontwikkelen om de blokken uit elkaar op te bouwen. Dit is uiteraard voordelig om de rekentijd te beperken. Op het einde van het hoofdstuk werd de convergentie bestudeerd en werd het concept verstrengelingsentropie S_E (een maat voor de kwantumcorrelatie tussen 2 systemen) ingevoerd. We voerden een kwantitatieve maat in voor de snelheid van de convergentie via de richtingscoëfficiënt κ van een $\log(\Delta E)$ versus $-\log(D)^2$ curve. Hoe groter κ , hoe sneller er convergentie optreedt. Voor systemen met een kleine S_E kan men volstaan met een lage D om het systeem efficiënt te simuleren (de parameters van het beschouwde model spelen hierin een belangrijke rol omdat ze de grootte van S_E sterk kunnen beïnvloeden).

In Hoofdstuk 3 werden de numerieke resultaten van een aantal spinsystemen vergeleken met analytische resultaten. Voor een isotroop Heisenbergmodel met lengte $L = 100$ bekwamen we een zeer goede overeenkomst (met de theoretische Bethe Ansatz) voor een truncatiedimensie van slechts $D = 8$ (terwijl exacte diagonalisatie een Hilbertruimte met dimensie $D = 2^{50} \approx 10^{15}$ vereist). In het geval van een exact oplosbaar $J_1 - J_2$ model (met even L) werd aangetoond dat een truncatiedimensie van $D = 2$ volstaat. Dit is een gevolg van een dimerisatie van de grondtoestand. Het nul worden van de verstrengelingsentropie voor subblokken met een even lengte ($S(2n) = 0, \forall n \in \mathbb{N}$) is hiermee consistent. Daarmee werd het verband tussen truncatiedimensie en verstrengelingsentropie verduidelijkt. Tot slot werden de numerieke resultaten van het exact oplosbaar bilinear biquadratic model (dat hogere-orde machten in de spin bevat, maar nog steeds een interactierange $\eta = 1$ heeft) vergeleken met de theoretische waarde en ook hier bleek de overeenkomst treffend.

In Hoofdstuk 4 maakten we de overgang naar fermionsystemen en via een handige numerieke ingreep konden we het faseprobleem (gerelateerd aan de anti-commutatierelaties van fermionen) oplossen. Het invoeren van een kwadratische kostfunctie stelde ons bovendien in staat om een deeltjesaantal op te leggen. Men dient de kostparameter gunstig te kiezen in functie van de repulsieve interactie. Als de kostparameter te klein is, dan zal de kostfunctie niet in staat zijn om het gewenste deeltjesaantal op te leggen bij sterke repulsie. Voor het Hubbard model bij half-filling (waarvoor een exact resultaat gekend is in de limiet $L \rightarrow \infty$) berekenden we de energie in functie van de repulsieve interactie en vonden daarbij voor kleine L reeds hetzelfde functionele verband als het exacte resultaat en voor grotere L konden we numeriek aantonen dat de energie voor $U = 0$ steeds beter de theoretische limiet (bij $L \rightarrow \infty$) benadert. Daarnaast bestudeerden we ook het $t1 - t2$ model (waar we rekening moeten houden met additionele fases). We toonden aan dat dit model voor bepaalde waarden van de parameters ferromagnetisch gedrag begint te vertonen vanaf een bepaalde U_c . Dit bleek mooi in overeenstemming met de literatuur zodat we konden besluiten dat het algoritme ook efficiënte resultaten oplevert voor algemene long-range 1D fermionsystemen.

In Hoofdstuk 5 hebben we het algoritme verder uitgebreid om ook moleculaire systemen te kunnen beschrijven. We zijn daarbij uitgegaan van de H_2 molecule (met elektronen in een $1s$ -orbitaal) om de nodige aanpassingen te bespreken. Voor de H_2 molecule vonden we dezelfde resultaten als FCI berekeningen (die we als benchmark gebruiken aangezien we werken met een minimale basisset en het vergelijken met experimentele resultaten dus niet zinvol is). We eindigden het hoofdstuk met een aantal algemene theoretische beschouwingen om het algoritme verder uit te breiden naar long-range systemen. Binnen dit topic zijn nog heel wat aanpassingen mogelijk. Een belangrijk punt is het ontwikkelen van een uitbreiding om ook p, d, f, \dots -orbitalen te kunnen beschouwen. Daar schuilt echter een groot probleem omdat het ganse algoritme uitgaat van een lineaire ketting. Voor waterstofatomen op een ketting met enkel $1s$ -orbitalen is een lineaire ketting een natuurlijke ordening, maar als je andere atomen op een ketting wil plaatsen dan stelt zich onmiddellijk het probleem wat je met de hogere-orde orbitalen moet doen. Daartoe moeten projectie-methodes worden ontwikkeld die in staat zijn de relevante fysica van 3D ruimtelijke orbitalen efficiënt te projecteren op een 1D-rooster.

Als uiteindelijke conclusie kunnen we wel stellen dat de 3 doelen van de thesis bereikt zijn: de implementatie van een algoritme voor spinsystemen (en bijhorend vat krijgen op de theorie van MPSs), de uitbreiding hiervan naar fermionische roosters en tot slot het berekenen van de energie van een H_2 molecule (wat opnieuw een aantal aanpassingen vergde om het algoritme uit te breiden tot moleculaire systemen).

Bijlage A

Numerieke implementatie van de VMPS methode

A.1 Implementatie

De volledige code is in MATLAB geschreven en bestaat uit een aantal subblokken die allen gebruikt worden in het hoofdprogramma `main.m`. In het hoofdstuk over de VMPS methode hebben we uitvoerig de verschillende stappen besproken om te komen tot een handelbaar eigenwaardeprobleem. Het is de bedoeling om in deze Appendix de theoretische behandeling te vertalen naar een snel en accuraat algoritme om de grondtoestandsenergie te berekenen. Alvorens de verschillende subblokken in detail te bespreken, is het handig om de algemene flowchart van het volledige algoritme even te bekijken (zie figuur (A.1)).

Gereduceerde QR-decompositie

Aangezien het onze uiteindelijke bedoeling is om een snel algoritme te bekomen, kunnen we een computationele vereenvoudiging doorvoeren door gebruik te maken van een gereduceerde QR-decompositie ([4]). Immers, als $m > n$ dan zullen de laatste $m - n$ rijen van R nul zijn. De matrix M kan dan geschreven worden als

$$M = Q_1 R_1 \tag{A.1}$$

waarbij Q_1 nu een $(m \times n)$ matrix is die echter in het meest algemene geval niet meer unitair is, maar waarvoor wel nog steeds geldt dat $Q_1^\dagger Q_1 = I$. De matrix R_1 is dan logischerwijze een $(n \times n)$ matrix. Voor het geval $m < n$ hebben we iets analoog zodat je finaal een decompositie van de vorm

$$M = Q_1 R_1 \tag{A.2}$$

krijgt met Q_1 een $(m \times \min(m, n))$ matrix en R_1 een $(\min(m, n) \times n)$ matrix.

A.1.1 Het sweep algoritme

We zullen steeds een ketting met lengte L beschouwen en opdelen in een linkerblok met sites 1 tot $l - 1$, een rechterblok met sites $l + 1$ tot L en een site l . We optimaliseren elke site l apart en houden daarbij de andere matrices op de overige sites vast. Op die manier vinden we voor

elke site l een gewoon eigenwaardeprobleem (zie hoofdstuk over de VMPS methode). Eens we nieuwe matrices voor site l gevonden hebben bekijken we site $l + 1$ (of $l - 1$ afhankelijk van de richting waarin men de ketting doorloopt, zie hieronder) en herhalen dezelfde procedure. De linker- en rechterblokken veranderen dus elke iteratiestap en afhankelijk van de richting waarin we lopen vergroten of verkleinen deze blokken. We verduidelijken dit door te veronderstellen dat het algoritme start op site L (wat in ons algoritme het geval is). Nadat we site L geoptimaliseerd hebben gaan we naar site $L-1$ (cyclus naar links). Het is duidelijk dat in dit geval het rechterblok groter wordt en het linkerblok kleiner. Wanneer we uiteindelijk bij site 1 komen, zullen we in de volgende stap naar site 2 gaan (cyclus naar rechts), zodat nu het linkerblok groter wordt en het rechterblok kleiner. We voeren steeds deze twee cycli uit. Het algoritme zal dus nooit stoppen ná een cyclus naar links, maar zal steeds nog in de cyclus naar rechts gaan. Indien op het einde van die cyclus de energie geconvergeerd¹ is binnen een bepaalde accuraatheid, dan stopt het algoritme. Indien dit niet het geval is start er een nieuwe cyclus naar links gevolgd door een cyclus naar rechts. Op het einde wordt opnieuw gekeken naar de energie. Dit cyclisch overlopen van alle sites noemt men het *sweep-algoritme*. We bespreken in wat volgt de verschillende stappen afzonderlijk en geven daarbij aan welke subblokken gebruikt worden.

A.1.2 Generatie en normalisatie van een MPS

Het programma heeft een aantal inputs nodig voor de constructie van een initiële MPS. Dit zijn de lengte van de ketting L , de fysische dimensie van de operatoren d (in het geval van een spin- $\frac{1}{2}$ -operator is dat dus bijvoorbeeld $d = 2$) en de truncatie-dimensie D . Voor elke site wordt een random MPS $M_{a_{l-1}, a_l}^{\sigma_l}$ gegenereerd met als beperking dat de matrixdimensie nooit groter kan worden dan $(D \times D)$. Alvorens we met deze MPS kunnen werken, moeten we zorgen dat de matrices een correcte normalisatie hebben (om een gewoon eigenwaardeprobleem te bekomen). Aangezien wij starten met de sweep $L \rightarrow 1$ wensen we dus dat alle matrices links-genormaliseerd zijn (men kan de volledige redenering ook omdraaien en starten met een cyclus naar rechts, maar dan moeten we uiteraard rechts-genormaliseerde matrices gebruiken als initiële MPS). Het normaliseren van de matrices gebeurt via het subblok `normalize.m`.

De functie `normalize.m` heeft 2 argumenten: enerzijds de matrices op site l die men wil normaliseren en anderzijds de sweep-richting (naar links wordt aangeduid met 'rl' en naar rechts wordt aangeduid met 'lr'). Het programma heeft ook 2 outputs: de genormaliseerde matrices en de matrix L of R uit de LQ of QR decompositie. Deze laatste worden dan vermenigvuldigd met respectievelijk de matrices op site $l - 1$ (deze worden vermenigvuldigd met matrix L) of de matrices op site $l + 1$ (deze worden met matrix R vermenigvuldigd). Aangezien wij links-genormaliseerde matrices wensen, is het dus logisch dat we als argument 'lr' meegeven en als output krijgen we de genormaliseerde matrices en de matrix R zoals geïllustreerd wordt in onderstaand code-fragment.

```
for l=1:L-1
    [mps{l},R]=normalize(mps{l},'lr');
```

¹We stellen bij definitie dat het systeem geconvergeerd is als $|E_{n+1} - E_n| < E_{acc}$ waarbij E_n de energie is in iteratiestap n en E_{acc} een vooraf ingestelde accuraatheid.

```

mps{1+1}=contract(R,2,2,mps{1+1},3,1);
end

```

Alle mogelijke vermenigvuldigingen worden uitgevoerd door de functie `contract.m` die we iets uitvoeriger behandelen in Appendix B. We geven alleen mee dat de argumenten van het programma `contract.m` eerst en vooral de twee objecten zijn die moeten vermenigvuldigd worden, het aantal indices dat elk object heeft (een matrix zal dus twee indices hebben, een tensor zal er drie hebben en verderop zullen we ook objecten tegenkomen met vier indices) én ook de indices van beide objecten waarover gesommeerd wordt. De precieze werking van de `contract` functie is op dit moment echter niet essentieel voor het begrijpen van het algoritme en is enkel van belang om een meer volledig begrip van de code te hebben (hoe alle vermenigvuldigingen dus precies uitgevoerd worden). De normalisatie van de laatste set matrices (op site L of site 1) wordt zodanig gekozen dat de totale norm gelijk aan 1 is.

A.1.3 Implementatie van de Hamiltoniaan

Om de initiële blokken te construeren hebben we uiteraard nood aan de Hamiltoniaan die de fysica van het systeem beschrijft. Het ingeven van deze Hamiltoniaan is niet triviaal en daarom wensen we kort even aandacht te besteden aan hoe de input-file van een Hamiltoniaan er moet uitzien. Het programma is zo geschreven dat het input-files van de volgende vorm kan inlezen

```
[H,d,d_edge,delta,int_range]=Hamiltoniaan(arg_1,arg_2,...,arg_n)
```

waarbij `Hamiltoniaan(arg_1,arg_2,...,arg_n)` de input-file is die de gebruiker zelf moet schrijven. Voor elke Hamiltoniaan moet dus een aparte input-file gemaakt worden die de volledige fysica van het systeem bevat, met nog een aantal parameters die door de gebruiker kunnen worden bepaald (deze parameters worden als argumenten meegegeven aan de functie `Hamiltoniaan(arg_1,arg_2,...,arg_n)`, waarbij de lengte L een verplicht argument is). Als output geeft `Hamiltoniaan(arg_1,arg_2,...,arg_n)` dan de Hamiltoniaan H , de fysische dimensie d , de eventuele edge dimensie `d_edge`, het aantal interactietermen `delta` ($n = 1 \dots \delta$) en de interactierange η gegeven door `int_range` (deze laatste 3 parameters moeten ook door de gebruiker ingegeven worden wanneer een bepaald fysisch model wordt ingegeven). De input-file moet steeds zo geschreven worden dat men als Hamiltoniaan H een array met dimensies $(d \times d \times (1 + \delta(1 + \eta)) \times L)$ krijgt. Concreet komt het er dus op neer dat aan elke site ($1 \dots L$) een set van $(1 + \delta(1 + \eta))$ $(d \times d)$ -matrices wordt geassocieerd.

De MATLAB-notatie voor de m -de $(d \times d)$ -matrix op site l is $H(:, :, m, l)$, waarbij $(:, :)$ equivalent is met $(1:d, 1:d)$ (het eerste argument geeft de rij-indices en het tweede de kolom-indices). Hieronder vatten we de regels samen die men moet naleven bij het ingeven van een Hamiltoniaan.

- De eerste matrix is steeds de lokale operator op site l , dus $H(:, :, 1, l) = \hat{O}_l$
- De volgende δ matrices op site l komen overeen met de matrices C_l^m (die gedefinieerd werden in het hoofdstuk over de VMPS methode), dus $H(:, :, 1 + n, l) = C_l^m$ (waarbij $n = 1 \dots \delta$)

- De daaropvolgende δ matrices op site l komen overeen met de matrices $(D_l^n)_{\alpha=1}$, dus $H(:, :, 1 + \delta + n, l) = (D_l^n)_{\alpha=1}$ (waarbij $n = 1 \dots \delta$)
- ...
- Op die manier vinden we dus als algemene regel: $H(:, :, 1 + \alpha\delta + n, l) = (D_l^n)_\alpha$ (waarbij $n = 1 \dots \delta$ en $\alpha = 1 \dots \eta$)

Het enige waar men dan nog moet op letten is dat de juiste $C_l^n / (D_l^n)_\alpha$ met de juiste $C_m^n / (D_m^n)_\alpha$ gecombineerd worden. Om te verduidelijken wat we hiermee bedoelen verwijzen we naar Appendix C waar een aantal input-files van gesimuleerde Hamiltonianen zijn opgenomen, voorzien met de nodige commentaar omtrent de schikking van alle matrices.

A.1.4 Initialisatie van de blokken

Als we vertrekken met een cyclus naar links ($L \rightarrow 1$) dan moeten we beschikken over de linkerblokken voor we het sweep-algoritme starten. Daartoe zullen we de linkerblokken geassocieerd aan sites 1 tot en met $L - 1$ construeren en stockeren in het geheugen. Voor site L moeten we uiteraard geen linkerblok construeren omdat er geen site $L + 1$ meer is. We gebruiken daartoe de iteratieve constructie voorgesteld in figuur (2.7)(b). Onderstaande code geeft de precieze implementatie en we zullen hieronder een aantal stappen trachten te verduidelijken.

```

for i=1:(L-1)
    mps_i=mps{i};
    [row,col,dm]=size(mps_i);

    if((d_edge~=0) && (i==1))
        %modify the Hamiltonian in case d_edge~=0
        H_i=H(1:d_edge,1:d_edge,:,i);
    else
        H_i=H(:, :, :, i);
    end

    YL{i}=zeros(col,col,delta,int_range);
    XL{i}=zeros(col,col);

    for m=1:int_range
        if(i==1)
            YLi=zeros(1,1,delta);
            [YL{i}(:, :, :, m), YRf]=construct_Yblocks(YLi(:, :, :), 0, mps_i, N, H_i, m, int_range, 'lr');
        else
            if(m==int_range)
                YLold=zeros(row,row,delta);
            else
                YLold=YL{i-1}(:, :, :, m+1);
            end
            [YL{i}(:, :, :, m), YRf]=construct_Yblocks(YLold, 0, mps_i, N, H_i, m, int_range, 'lr');
        end

        %construct X block

```

```

end
if(i==1)
    XLold=zeros(1,1,1);
    YLold=zeros(1,1,delta);
else
    XLold=XL{i-1};
    YLold=YL{i-1}(:, :, :, 1);
end
[XL{i},XRf]=construct_Xblocks(XLold,0,YLold,0,mps_i,N,H_i,1,'lr');
end

```

Binnen de `for`-lus die de sites afloopt is er nog een extra lus die loopt van 1 tot η . De blokken zelf worden geconstrueerd in de programma's

```

construct_Yblocks(YL,YR,mps_l,N,H_l,m,int_range,'direction')
construct_Xblocks(XL,XR,YL,YR,mps_l,N,H_l,1,'direction');

```

Het programma `construct_Yblocks(YL,YR,mps_l,N,H_l,m,int_range,'direction')` heeft als argumenten het `YL`-blok van de vorige site ($l - 1$) (bij een sweep naar rechts) of het `YR`-blok van de vorige site ($l + 1$) (bij een sweep naar links) en de matrices van site l (`mps_l`). Daarnaast moet men het getal $N = 1 + \delta$ en η (`=int_range`) ingeven en ook alle operatoren geassocieerd aan site l die vervat zitten in `H_l`. Verder is er nog een controle-parameter `m`, die we zo dadelijk bespreken, en tot slot geeft men ook nog de sweeprichting '`direction`'. De controle-parameter `m` (die in bovenstaande code samenvalt met de waarde van de interne lus, die loopt van 1 tot `int_range`) zorgt ervoor dat als `m=int_range` we rekening houden met de randvoorwaarde $(Y_{l-1}^L)_{\alpha+1}^n = 0$ als $\alpha = \eta$. Daarnaast moeten we binnen de interne lus over `m` ook nog een extra `if`-conditie toevoegen als we op site $l = 1$ zitten om de randvoorwaarde $(Y_{l-1}^L)_{\alpha+1}^n = 0$ als $l = 1$ te kunnen opleggen. Dan geven we aan het programma een `YL` mee die allemaal nullen bevat, zodat de contractie gewoon een nulblok zal opleveren en dus geen invloed zal hebben (zoals het hoort volgens de randvoorwaarde). De extra `if`-conditie (`if(m=int_range)`) voor het geval $l \neq 1$ dient gewoon om ervoor te zorgen dat de $m + 1$ niet buiten zijn grenzen gaat (als $m = \eta$ zou dit immers het geval zijn). Voor de constructie van de `X`-blokken maken we gebruik van zowel het `X`- als het `Y`-blok op de vorige site (de argumenten van de `construct_Xblocks.m` functie zijn volledig analoog aan deze van de `construct_Yblocks.m`, behalve dat hier ook nog de `Y`-blokken worden meegegeven en er geen nood is aan de kennis van de interactierange). Als $l = 1$ moeten we ook hier allemaal nullen meegeven om te voldoen aan de randvoorwaarde $X_{l-1}^L = 0$ en $(Y_{l-1}^L)_1 = 0$ als $l = 1$ (wat opnieuw geïmplementeerd is met een `if`-conditie). De controle-parameter wordt hier gewoon op 1 gezet (elke waarde mag, er moet gewoon een argument meegegeven worden om geen syntax-fouten te krijgen), maar heeft in de sweep naar links geen invloed bij de constructie van de `X`-blokken.

A.1.5 De sweep-cycli

Na de initialisatie van de blokken kan de sweep-cyclus gestart worden (in ons geval starten we met de cyclus naar links: $L \rightarrow 1$). Een `for`-lus loopt alle sites L tot en met 1 af. Bij elke stap in de lus moet het eigenwaardeprobleem (2.60) opgelost worden en daarvoor moeten we beschikken over de blokken links en rechts van een bepaalde site l . Bij de start zitten we op

site L en hebben we dus enkel nood aan een linker-blok, dat we uit het geheugen halen (dit blok werd immers in de initialisatie-fase geconstrueerd). De rechterblokken worden allemaal op nul gesteld, zodat het geen kwaad kan om deze blokken mee te geven aan de eigenvaluesolver; aangezien ze nul zijn zullen ze ook geen bijdrage leveren aan \hat{H}_{eff} . De eigenvaluesolver is de kern van het sweep-algoritme en we zullen deze hieronder dan ook in detail bespreken.

Iteratieve eigenvaluesolver

MATLAB beschikt over een iteratieve eigenvaluesolver `eigs` die toelaat om voor bepaalde types matrices het eigenwaardeprobleem snel op te lossen. De iteratieve solver gokt gewoon een oplossing $|\nu_{guess}\rangle$ en berekent dan het matrix-vectorproduct $\hat{H}_{eff}|\nu_{guess}\rangle$ om een nieuwe gok te genereren. Dit gaat verder tot er convergentie optreedt. De standaard-syntax van de `eigs` functie (nuttig voor ons probleem, namelijk het vinden van de kleinste eigenwaarde) wordt hieronder gegeven

```
opts.v0=mps
eigs(H,n,1,'sr',opts)
```

waarbij H de matrix is waarvoor je het probleem $Hx = \lambda x$ wil oplossen, n de dimensie van de vector x , 1 geeft aan dat we maar 1 eigenwaarde willen, `'sr'` geeft aan dat het de eigenwaarde met het kleinste reëel deel moet zijn (dus dit is goed, aangezien wij op zoek gaan naar de grondtoestandsenergie) en `opts` is een optie om aan te geven dat we de vector `opts.v0=mps` als initiële gok wensen te gebruiken. Concreet komt het er dus op neer dat we de matrices op een site herschikken tot een vector (zoals aangegeven in hoofdstuk 2) en dat we als initiële gok voor de eigenvector steeds de vector (\approx matrices) nemen die in de vorige stap berekend werd voor site l . Men kan dit algoritme echter enorm versnellen door de matrix H niet expliciet te construeren, maar onmiddellijk het matrix-vectorproduct te berekenen. Dit gebeurt via een zogenaamde `function handle`. Daartoe gebruiken we de syntax

```
opts.v0=mps
eigs(@(x)Heff_nu(arg_1,arg_2,...,x,...,arg_q),n,1,'sr',opts)
```

waarbij x de vector is die `eigs` uiteindelijk zal teruggeven als resultaat. Hier gebruiken we deze vector echter als argument van een externe MATLAB functie `Heff_nu.m` die we zelf geschreven hebben. Deze functie gebruikt de vector x naast de X - en Y -blokken (en nog een aantal andere argumenten die de dimensie van de matrices op site l , de interactierange en de waarde $N = 1 + \delta$ bevatten) om het matrix-vectorproduct $\hat{H}_{eff}|\nu\rangle$ uit te rekenen. Aangezien `eigs` een vector x moet genereren, maar `Heff_nu.m` gebruik maakt van matrices (zie formules (2.52) tot en met (2.57)) moeten we de vector x binnen de functie `Heff_nu.m` eerst terug omzetten naar een set van d matrices en op het einde de bekomen set van matrices terug omzetten naar een vector die dan in de volgende iteratiestap van het `eigs`-algoritme opnieuw gebruikt wordt als argument van de functie `Heff_nu.m`. Dit proces gaat verder tot de methode convergeert² en de functie `eigs.m` uiteindelijk een vector `EVec` (de eigenvector) teruggeeft met de corresponderende eigenwaarde `EVal`. De bekomen eigenvector moet nu nog omgezet worden naar een set matrices om alles terug in de vorm van een MPS te krijgen. Deze set matrices wordt vervolgens genormaliseerd (analoog aan de normalisatie van de initiële MPS) en daarna gebruikt om het **rechter** blok behorend bij site l (in de eerste stap dus $l = L$) te construeren. De bovenstaande discussie vertaalt zich in code als het hieronder gegeven fragment.

²Convergentie van het `eigs`-algoritme betekent dat er een stabiele eigenvector gevonden is.

```

mpsv=mps{1};
YRi=zeros(1,1,delta,int_range);
XRi=zeros(1,1,1);
YLi=YL{1-1};
XLi=XL{1-1};

%reshape mps to a vector
n=dM*rowM*colM;
mpsv=reshape(mpsv,[n,1]);
%initial guess
opts.v0=mpsv;
[EVec,EVal]=eigs(@(x)Heff_nu(XLi,XRi,YLi,YRi,H_1,x,rowM,...
...colM,dM,int_range,N),n,1,'sr',opts);
%reshape the vector back to mps
E=EVal;
mps{1}=reshape(EVec,[rowM,colM,dM]);

[mps{1},Left]=normalize(mps{1},'r1');
mps{1-1}=contract(Left,2,1,mps{1-1},3,2);
% construct Y block
[YLf,YR{1}(:,:,,1)]=construct_Yblocks(0,YRi(:,:,,1),...
...mps{1},N,H_1,1,int_range,'r1');
for m=2:int_range
    YR{1}(:,:,,m)=zeros(rowM,rowM,delta);
end

```

Voor de constructie van het Y^R blok geassocieerd aan site $l = L$ merken we op dat enkel de term voor $\alpha = 1$ overblijft wegens de randvoorwaarde $(Y_{l+1}^R)_{\alpha-1} = 0$ als $l = L$. De andere termen voor $\alpha = 2 \dots \eta$ worden op nul gesteld. De controle-parameter in de sweep-richting 'r1' zorgt ervoor dat je bij $\alpha = 1$ de eerste term uit de iteratieve constructie van figuur (2.9)(b) berekent en bij $\alpha \neq 1$ de tweede term. Bij $\alpha = 1$ is de tweede term inderdaad altijd nul wegens de randvoorwaarde $(Y_{l+1}^R)_{\alpha-1} = 0$ als $\alpha = 1$. Voor de volledigheid geven we ook nog de code van `Heff_nu.m`.

```

%reshape mps to array of d matrices
mps=reshape(x,[rowM,colM,d]);
delta=N-1;

%term 1:
term1=contract(XL,2,2,mps,3,1);

%term 2:
term2=contract(mps,3,3,Hloc(:,:,,1),2,1);

%term 3:
term3=contract(mps,3,2,XR,2,2);

%term 4:
term4=contract(mps,3,3,Hloc(:,:,,2:N),3,1);
term4=contract(YL(:,:,,1),3,2,term4,4,1);

%term 5
term5=contract(mps,3,3,Hloc(:,:,,(N+1):(N+delta)),3,1);
term5=contract(term5,4,2,YR(:,:,,1),3,1);

```

```

%term 6:
term6=zeros(rowM,colM,d);

for m=2:1:int_range

    B_alpha=Hloc(:,:(N+delta*(m-1)+1):(N+delta*m));
    YR_alpha=YR(:, :, :, m);

    term6int1=contract(mps,3,3,B_alpha,3,1);
    term6int1=contract(term6int1,4,2,YR_alpha,3,1);

    YR_transpose=permute(YR(:, :, :, m-1), [2,1,3]);
    term6int2=contract(mps,3,2,YR_transpose,3,1);
    term6int2=contract(YL(:, :, :, m),3,2,term6int2,4,1);

    term6=term6+term6int1+term6int2;
end

sum=term1+term2+term3+term4+term5+term6;
%sum=term1+term2+term3+term4+term5;

%reshape output array of matrices back to a vector
nu_new=reshape(sum, [rowM*colM*d,1]);

```

Zoals reeds gezegd berekent deze functie dus de 6 termen ((2.52) tot en met (2.57)) met de vector x die eerst wordt omgezet naar een set van matrices via de `reshape` functie. Op het einde wordt de bekomen set matrices terug omgezet in een vector.

Vervolg van sweep-algoritme

Eens de geoptimaliseerde matrices op site L bepaald zijn (we starten immers op site L) gaan we naar site $L - 1$. Voor alle sites $l \neq 1, L$ wordt volgend onderdeel van de code gebruikt.

```

mpsv=mps{l};
%reshape mps to a vector
n=dM*rowM*colM;
mpsv=reshape(mpsv, [n,1]);
%initial guess
opts.v0=mpsv;
[EVec,EVal]=eigs(@(x)Heff_nu(XL{l-1},XR{l+1}),...
...YL{l-1},YR{l+1},H_l,x,rowM,colM,dM,int_range,N),n,1,'sr',opts);
%reshape the vector back to mps
E=EVal;
mps{l}=reshape(EVec, [rowM,colM,dM]);
[mps{l},Left]=normalize(mps{l}, 'rl');
mps{l-1}=contract(Left,2,1,mps{l-1},3,2);
%construct Y block
for m=1:1:int_range
    if(m==1)
        control1=1;
        YRold=0;
    else

```

```

        control1=0;
        YRold=YR{1+1}(:,:,,m-1);
    end
    [YLf,YR{1}(:,:,,m)]=construct_Yblocks(0,YRold,...
        ...mps{1},N,H_1,control1,int_range,'r1');
end

```

Het grootste verschil is dat we nu zowel blokken uit het geheugen gebruiken ($XL\{1-1\}$, $YL\{1-1\}$) en blokken die in de vorige stap zijn geconstrueerd ($XR\{1+1\}$, $YR\{1+1\}$; merk nogmaals op dat de vorige site inderdaad één hoger ligt omdat we naar links gaan). Daarnaast verloopt de constructie van de Y^R -blokken analoog aan die van de Y^L -blokken. Als $m=1$ (equivalent met $\alpha = 1$) dan mogen we enkel de eerste term in rekening brengen en anders mogen we enkel de tweede term uit de iteratieve constructie in rekening brengen. Eens we op site $l = 1$ komen moeten we geen nieuw Y^R -blok meer maken want er is geen site links van $l = 1$. Voor elke l hebben we nu wel al de Y^R -blokken, maar we moeten ook nog de X^R -blokken construeren. Dit gebeurt via onderstaande code:

```

%construction of the X block
XR{1}=zeros(rowM,rowM);
if(l>1)
    for m=1:1:int_range
        if(l==L)
            XRold=zeros(1,1);
            YRold=zeros(1,1,delta);
        else
            XRold=XR{1+1};
            YRold=YR{1+1}(:,:,,m);
        end
        %the first and second component of XR should only be
        %constructed once, thus construct them only when m=1=control1
        control1=m;
        [XLf,XR_int]=construct_Xblocks(0,XRold,0,YRold,mps{1},...
            ...N,H_1,control1,'r1');
        XR{1}=XR{1}+XR_int;
    end
end
end

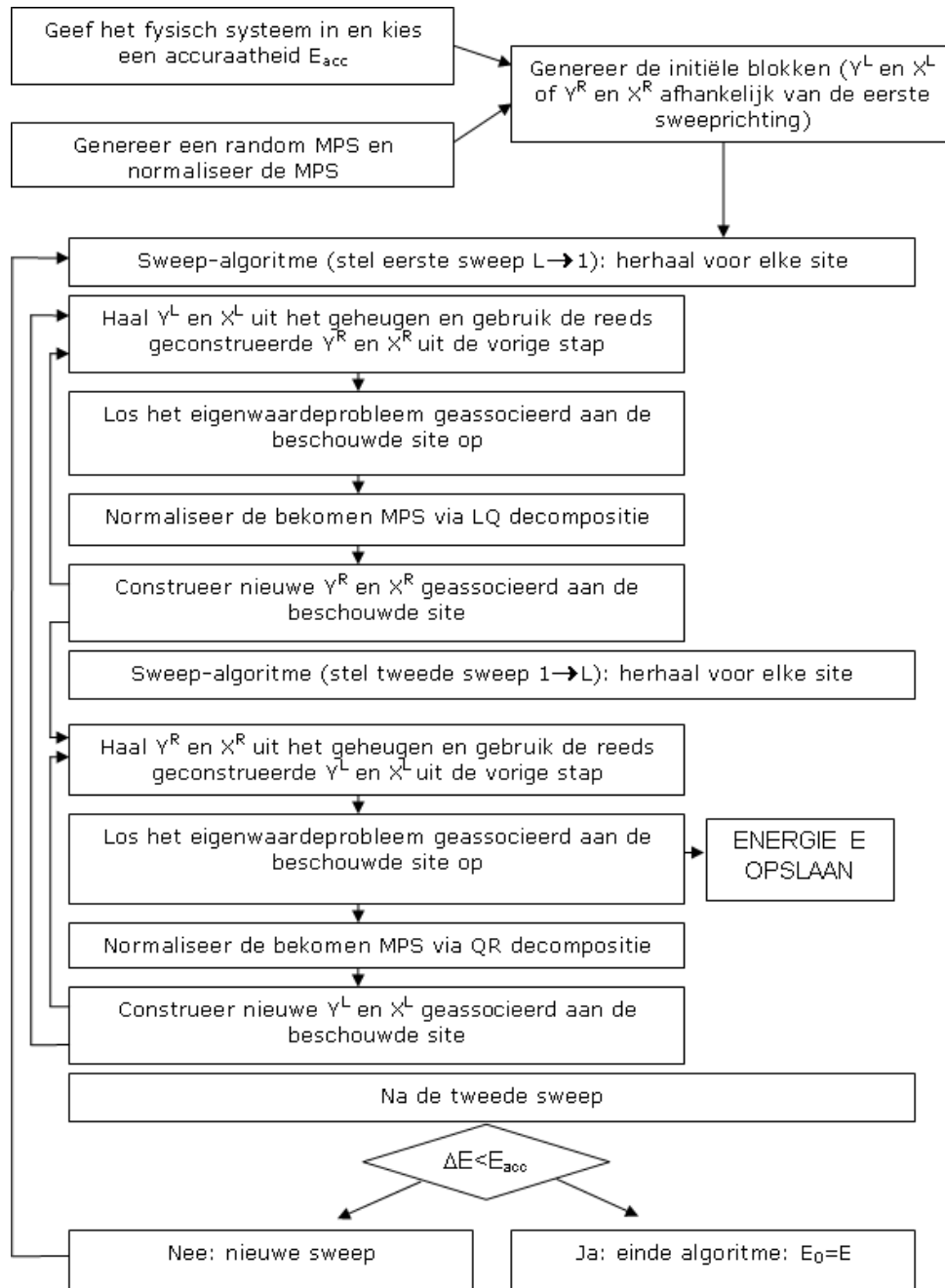
```

De `if`-conditie binnen de `for`-lus wordt gebruikt om de randvoorwaarde $X_{l+1}^R = 0$ en $(Y_{l+1}^R)_1 = 0$ als $l = L$ op te leggen. Verder is er voor het X^R -blok een `for`-lus van 1 tot η omdat de formule in figuur (2.10)(b) immers aangeeft dat men een som moet nemen over $\alpha = 1 \dots \eta$. De eerste 2 termen heeft men echter maar 1 keer nodig, vandaar dat we ook hier een controle-parameter gebruiken. We stellen via deze parameter in dat de eerste 2 termen geconstrueerd worden als $m=1$ (deze worden dan opgeteld bij de derde term). Als echter $m>1$ mag je enkel de laatste term in rekening brengen, omdat je anders de eerste 2 termen er $(\eta - 1)$ keer teveel zou bij tellen als je de volledige `for`-lus doorloopt.

Andere sweeprichting

We hebben het algoritme besproken in het geval we starten van rechts (bij site L) en naar links gaan. De andere richting wordt sowieso ook eenmaal doorlopen (het algoritme zal nooit stoppen na een cyclus naar links, maar altijd na een cyclus naar rechts). Wat betreft de redenering is alles volledig analoog met dat verschil dat je nu **linker**blokken zal construeren tijdens de sweep

en **rechter**blokken uit het geheugen zal halen terwijl het in de cyclus naar links net andersom is. Ook de **if**-condities zullen lichtjes aangepast worden om rekening te houden met de iteratieve formules zoals weergegeven in figuren (2.7)(b) en (2.8)(b). We merken wel op dat we reeds vroeger in het algoritme linkerblokken hebben geconstrueerd, namelijk bij de initialisatie. De constructie van de blokken binnen de sweeplus is volledig hetzelfde, zodat de bespreking gegeven bij de initialisatie ook geldt in de sweeplus.



Figuur A.1: Flowchart van het volledige VMPS-algoritme

Bijlage B

Additionele uitleg bij de MATLAB-code

B.1 De functie `contract.m`

Deze functie is geïnspireerd op een MATLAB-code uit ([5]). Vooral de gebruikte MATLAB-commando's `reshape` en `permute` worden veelvuldig gebruikt. Voor informatie over deze commando's verwijzen we naar de MATLAB Help. We geven hieronder enkel wat algemene info over de `contract.m` functie; voor meer specifieke details verwijzen we naar de volgende secties waar een aantal voorbeelden concreet uitgewerkt worden.

In de volledige VMPS-code moet men meermaals vermenigvuldigingen uitvoeren tussen matrices, tussen matrices en sets van matrices, enzovoort. Daarom werd een aparte functie `contract.m` geschreven die al deze bewerkingen uitvoert. De input/output karakteristiek van deze functie wordt hieronder gegeven

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

waarbij `T` de output is en zowel een matrix als een set (of array) van matrices kan zijn. De argumenten moeten als volgt geïnterpreteerd worden: `X` en `Y` zijn de objecten die we willen vermenigvuldigen; `numindX` en `numindY` geven aan hoeveel indices object `X` respectievelijk `Y` bevat en `indX` en `indY` geeft aan welke index van `X` gecontraheerd wordt met welke index van `Y` (op uitzonderingen na, zie verder). Als `X` en `Y` beide matrices zijn is dus `numindX=2` en `numindY=2` en als je de vermenigvuldiging `X*Y` wil verwezenlijken geef je `indX=2` en `indY=1` op, conform de definitie van een matrixvermenigvuldiging ($(XY)_{i,j} = \sum_k X_{ik}Y_{kj}$). De `contract.m` functie bevat een hele reeks `if`-condities om de verschillende combinaties `numindX, indX, numindY, indY` te onderscheiden en voor elke combinatie de juiste contractie terug te geven. In de volgende secties geven we een aantal voorbeelden van het expliciete gebruik van de `contract.m` functie. We merken wel op dat de `contract.m` functie soms meervoudige sommen uitrekent, zodat je dan niet eenduidig kan weergeven over welke indices er wordt gesommeerd (precies omdat de sommen lopen over meerdere indices). In dat geval kozen we een combinatie die nog niet gebruikt werd (en probeerden we alleszins 1 set indices waarover gesommeerd wordt, weer te geven in de `contract.m` functie).

B.2 De functie construct Yblocks

B.2.1 Constructie van de Y^L -blokken

Voor de constructie van de Y^L blokken gaan we uit van de algemene formule (waarbij de complexe toevoeging is weggelaten omdat we steeds met reële matrices werken)

$$((Y_l^L)_\alpha^n)_{i,j} = \sum_{k,\sigma_l,\sigma'_l} M_{k,j}^{\sigma'_l} M_{k,i}^{\sigma_l} ((D_l^n)_\alpha)_{\sigma_l,\sigma'_l} + \sum_{k,m,\sigma_l} ((Y_{l-1}^L)_{\alpha+1}^n)_{k,m} M_{m,j}^{\sigma_l} M_{k,i}^{\sigma_l} \quad (\text{B.1})$$

met $l = 1, 2, \dots, L - 1$ en $\alpha = 1, 2, \dots, \eta$ en waarbij $(Y_{l-1}^L)_{\alpha+1}^n = 0$ als $l = 1$ of $\alpha = \eta$. Hierbij is M^{σ_l} een set van d matrices geassocieerd aan site l , $(D_l^n)_\alpha$ een set van $\delta\eta$ ($d \times d$)-matrices en $(Y_{l-1}^L)_{\alpha+1}^n$ het Y^L -blok geassocieerd aan de vorige site, met gepaste randvoorwaarden. De constructie hieronder is voor vaste α , dat betekent dat voor elke α de beschreven contracties moeten herhaald worden (dit is logisch aangezien we η Y^L -blokken per site willen construeren). We laten daarom de index α weg. We bekijken eerst de eerste term en doen de berekening in 2 tussenstappen:

$$\begin{aligned} \sum_{k,\sigma_l,\sigma'_l} M_{k,j}^{\sigma'_l} M_{k,i}^{\sigma_l} (D_l^n)_{\sigma_l,\sigma'_l} &= \sum_{k,\sigma_l} M_{k,i}^{\sigma_l} \left(\sum_{\sigma'_l} M_{k,j}^{\sigma'_l} (D_l^n)_{\sigma_l,\sigma'_l} \right) \\ &= \sum_{k,\sigma_l} M_{k,i}^{\sigma_l} T_{k,j}^{\sigma_l,n} \\ &= (S_1^n)_{i,j} \end{aligned} \quad (\text{B.2})$$

We bekijken de verschillende bewerkingen nu wat meer in detail. De bewerking

$$T_{k,j}^{\sigma_l,n} = \sum_{\sigma'_l} M_{k,j}^{\sigma'_l} (D_l^n)_{\sigma_l,\sigma'_l} \quad (\text{B.3})$$

komt erop neer dat we elk matrixelement $M_{k,j}^{\sigma'_l}$ vermenigvuldigen met het element $(D_l^n)_{\sigma_l,\sigma'_l}$ en dit voor elke waarde van n . Om dit nu makkelijk via een gewone matrixvermenigvuldiging te kunnen schrijven maken we van D_l^n een blokmatrix met d^2 blokken waarbij elk blok een diagonaalmatrix is met de elementen $(D_l^n)_{\sigma_l,\sigma'_l}$. De dimensies van de blokken worden zo gekozen dat ze dimensioneel matchen op de matrices $M^{\sigma'_l}$. Daarna permuteren we de blokmatrix omdat we voor iedere rij-index van D_l^n de matrices $M^{\sigma'_l}$ moeten vermenigvuldigen met de kolom-elementen van D_l^n . Een dergelijke vermenigvuldiging met een diagonaalmatrix zorgt er inderdaad voor dat we elk matrixelement $M_{k,j}^{\sigma'_l}$ vermenigvuldigen met het element $(D_l^n)_{\sigma_l,\sigma'_l}$. Als resultaat bekomen we een array van matrices. Bij een fysische dimensie $d = 2$ en 3 interactietermen $\delta = 3$ is dat bijvoorbeeld een array met 6 matrices, waarbij de array 2 rijen heeft en 3 kolommen. De tweede bewerking wordt dan

$$(S_1^n)_{i,j} = \sum_{k,\sigma_l} M_{k,i}^{\sigma_l} T_{k,j}^{\sigma_l,n} = \sum_{k,\sigma_l} M_{i,k}^{\sigma_l T} T_{k,j}^{\sigma_l,n} \quad (\text{B.4})$$

waar we de getransponeerde van de matrices M^{σ_l} hebben ingevoerd. Om dit alles met één matrixvermenigvuldiging te kunnen verwezenlijken organiseren we de array in een grote matrix bestaande uit de blokken geconstrueerd in de vorige bewerking en permuteren we M^{σ_l} (die we

ook rangschikken in blokken). Op die manier worden alle blokken juist vermenigvuldigd. Tot slot breken we de bekomen matrix terug op in δ matrices omdat we immers nog een derde index hebben. We illustreren deze behandeling in figuur (B.1) waarbij de bovenste helft de eerste bewerking illustreert en de onderste helft de tweede. Deze bewerkingen worden uitgevoerd in de `contract.m` functie. De eerste bewerking gebeurt via

`contract(M,3,3,D,3,1)`

met voor de hand liggende notaties van de objecten die vermenigvuldigd worden en de tweede bewerking bekomt men via

`contract(M,3,1,T,4,1)`.

Om de tweede term te berekenen gaan we volledig analoog te werk (dus door een gepast herschikken, permuteren en omvormen van (sets van) matrices). We geven hier enkel de tussenstappen in formulevorm en geven ook aan welke syntax moet gebruikt worden om deze bewerkingen uit te voeren met de `contract.m` functie. De tweede term (waarbij we de index $\alpha + 1$ opnieuw laten vallen) wordt

$$\begin{aligned} \sum_{k,m,\sigma_l} (Y_{l-1}^L)_{k,m}^n M_{m,j}^{\sigma_l} M_{k,i}^{\sigma_l} &= \sum_{k,\sigma_l} M_{k,i}^{\sigma_l} \sum_m (Y_{l-1}^L)_{k,m}^n M_{m,j}^{\sigma_l} \\ &= \sum_{k,\sigma_l} M_{i,k}^{\sigma_l T} T_{k,j}^{n,\sigma_l} \\ &= (S_2^n)_{i,j}. \end{aligned} \quad (\text{B.5})$$

Om $T_{k,j}^{n,\sigma_l}$ te bekomen, gebruiken we

`contract(Y,3,2,M,3,1)`

en voor $(S_2^n)_{i,j}$ gebruiken we

`contract(M,3,1,T,4,2)`.

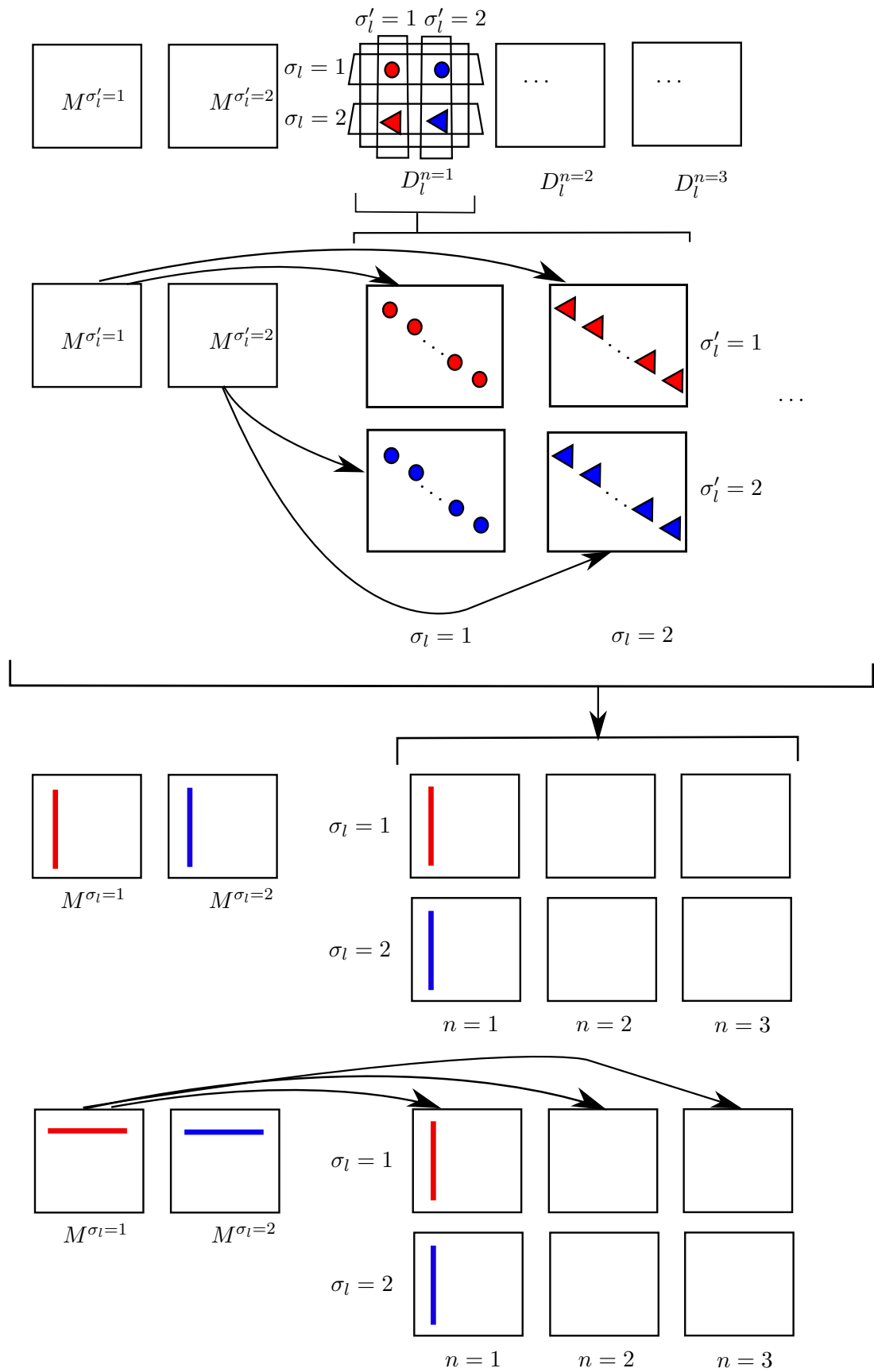
B.2.2 Constructie van de Y^R -blokken

Voor de constructie van de Y^R -blokken gaan we volledig analoog te werk. Ook hier zullen we enkel in formulevorm de tussenbewerkingen vermelden met de bijhorende syntax om deze effectief uit te voeren. De algemene formule voor Y^R is

$$((Y_l^R)^\alpha)_{i,j} = \sum_{k,\sigma_l,\sigma'_l} M_{j,k}^{\sigma'_l} M_{i,k}^{\sigma_l} (C_l^n)_{\sigma_l,\sigma'_l} \delta_{\alpha,1} + \sum_{k,m,\sigma_l} ((Y_{l+1}^R)^\alpha)_{k,m} M_{j,m}^{\sigma_l} M_{i,k}^{\sigma_l} \quad (\text{B.6})$$

met $l = L, L - 1, \dots, 2$ en $\alpha = 1, 2, \dots, \eta$ waarbij $(Y_{l+1}^R)_{\alpha-1} = 0$ als $l = L$ of $\alpha = 1$. Uitwerken van de eerste term geeft (ook hier laten we de index α weg):

$$\begin{aligned} \sum_{k,\sigma_l,\sigma'_l} M_{j,k}^{\sigma'_l} M_{i,k}^{\sigma_l} (C_l^n)_{\sigma_l,\sigma'_l} &= \sum_{k,\sigma_l} M_{i,k}^{\sigma_l} \sum_{\sigma'_l} M_{j,k}^{\sigma'_l} (C_l^n)_{\sigma_l,\sigma'_l} \\ &= \sum_{k,\sigma_l} M_{i,k}^{\sigma_l} T_{j,k}^{\sigma_l,n} \\ &= \sum_{k,\sigma_l} M_{i,k}^{\sigma_l} T_{k,j}^{(\sigma_l,n)T} \\ &= (S_1^n)_{i,j} \end{aligned}$$



Figuur B.1: Voorbeeld van een contractie.

waarbij we $T_{j,k}^{\sigma_l,n}$ berekenen met

`contract(M,3,3,C,3,1)`.

en $(S_1^n)_{i,j}$ met

`contract(M,3,3,T,4,3)`.

Voor de tweede term krijgen we

$$\begin{aligned}
\sum_{k,m,\sigma_l} (Y_{l+1}^R)_{k,m}^n M_{j,m}^{\sigma_l} M_{i,k}^{\sigma_l} &= \sum_{m,\sigma_l} M_{j,m}^{\sigma_l} \sum_k M_{i,k}^{\sigma_l} (Y_{l+1}^R)_{k,m}^n \\
&= \sum_{m,\sigma_l} M_{j,m}^{\sigma_l} T_{i,m}^{\sigma_l,n} \\
&= \sum_{m,\sigma_l} T_{i,m}^{\sigma_l,n} M_{m,j}^{\sigma_l} \\
&= (S_2^n)_{i,j}
\end{aligned}$$

waarbij we $T_{i,m}^{\sigma_l,n}$ berekenen met

`contract(M,3,2,Y,3,1)`.

en $(S_2^n)_{i,j}$ met

`contract(T,4,3,M,3,3)`.

B.3 De functie construct Xblocks

B.3.1 Constructie van de X^L -blokken

De algemene formule voor een X^L -blok wordt gegeven door

$$\begin{aligned}
(X_l^L)_{i,j} &= \sum_{k,\sigma_l,\sigma'_l} M_{k,j}^{\sigma'_l} M_{k,i}^{\sigma_l} (\hat{O}_l)_{\sigma_l,\sigma'_l} + \sum_{k,m,\sigma_l} (X_{l-1}^L)_{k,m} M_{m,j}^{\sigma_l} M_{k,i}^{\sigma_l} + \\
&\quad \sum_{k,m,n,\sigma_l,\sigma'_l} ((Y_{l-1}^L)_1^n)_{k,m} M_{m,j}^{\sigma_l} M_{k,i}^{\sigma'_l} (C_l^n)_{\sigma_l,\sigma'_l}
\end{aligned} \tag{B.7}$$

met $l = 1, 2, \dots, L-1$ en waarbij $X_{l-1}^L = 0$ en $(Y_{l-1}^L)_1 = 0$ als $l = 1$. Voor wat betreft de contracties kunnen we dankbaar gebruik maken van de contracties die we vonden voor de Y^L -blokken. De eerste term is immers dezelfde als de eerste term van een Y^L -blok waarbij de index n nu echter slechts 1 waarde aanneemt (en waarbij ook geen index α meer voorkomt). Het komt er dus op neer $((D_l^n)_\alpha)_{\sigma_l,\sigma'_l}$ te vervangen door $(\hat{O}_l)_{\sigma_l,\sigma'_l}$ in de syntax van de `contract.m` functie. Voor de tweede term is dit ook het geval: n kan opnieuw maar 1 waarde aannemen en Y_{l-1}^L wordt vervangen door X_{l-1}^L . De syntax voor de eerste twee termen (of dus 4 bewerkingen, 2 voor elke term) wordt dan respectievelijk

`contract(M,3,3,0,3,1)`

`contract(M,3,1,T,4,1)`

voor de eerste en

contract(X,3,2,M,3,1)

contract(M,3,1,T,4,2)

voor de tweede. De derde term is echter nieuw en we breken de berekening op in 3 stukken, wat we hieronder verduidelijken met behulp van de formule:

$$\begin{aligned}
\sum_{k,m,n,\sigma_l,\sigma'_l} ((Y_{l-1}^L)_1^n)_{k,m} M_{m,j}^{\sigma_l} M_{k,i}^{\sigma'_l} (C_l^n)_{\sigma_l,\sigma'_l} &= \sum_{k,n,\sigma_l,\sigma'_l} M_{k,i}^{\sigma'_l} (C_l^n)_{\sigma_l,\sigma'_l} \sum_m ((Y_{l-1}^L)_1^n)_{k,m} M_{m,j}^{\sigma_l} \\
&= \sum_{k,\sigma'_l} M_{k,i}^{\sigma'_l} \sum_{n,\sigma_l} T_{k,j}^{n,\sigma_l} (C_l^n)_{\sigma_l,\sigma'_l} \\
&= \sum_{k,\sigma'_l} M_{i,k}^{\sigma'_l T} S_{k,j}^{\sigma'_l} \\
&= U_{i,j}
\end{aligned}$$

De 3 contracties (voor de constructie van $T_{k,j}^{n,\sigma_l}$, $S_{k,j}^{\sigma'_l}$ en $U_{i,j}$) zijn respectievelijk

contract(Y,3,2,M,3,1)

contract(T,4,3,C,3,2)

contract(M,3,1,S,3,1).

B.3.2 Constructie van de X^R -blokken

De algemene formule voor een X^R -blok wordt gegeven door

$$\begin{aligned}
(X_l^R)_{i,j} &= \sum_{k,\sigma_l,\sigma'_l} M_{j,k}^{\sigma'_l} M_{i,k}^{\sigma_l} (\hat{O}_l)_{\sigma_l,\sigma'_l} + \sum_{k,m,\sigma_l} (X_{l+1}^R)_{k,m} M_{j,m}^{\sigma_l} M_{i,k}^{\sigma_l} + \\
&\quad \sum_{\alpha=1}^{\eta} \sum_{k,m,n,\sigma_l,\sigma'_l} ((Y_{l+1}^R)_\alpha^n)_{k,m} M_{j,m}^{\sigma_l} M_{i,k}^{\sigma'_l} ((D_l^n)_\alpha)_{\sigma_l,\sigma'_l} \tag{B.8}
\end{aligned}$$

met $l = L, L-1, \dots, 2$ en waarbij $X_{l+1}^R = 0$ en $(Y_{l+1}^R)_1 = 0$ als $l = L$. Voor de eerste 2 termen kunnen we dezelfde opmerking maken als bij de constructie van de X^L -blokken, namelijk dat ze met dezelfde contracties kunnen behandeld worden als hun overeenkomstige termen voor het Y^R -blok zodat de 4 contracties hier gegeven worden door (2 voor elke term):

contract(M,3,3,0,3,1)

contract(M,3,3,T,4,3)

contract(M,3,2,X,3,1)

contract(T,4,3,M,3,3).

Voor de derde term zijn er (zoals bij X^L) 3 bewerkingen nodig. We verduidelijken dit opnieuw aan de hand van de formule waarbij we de index α weglaten, en we er dus rekening mee moeten houden dat deze berekening η keer moet gedaan worden om vervolgens alle matrices (behorend

bij een specifieke α) te sommeren:

$$\begin{aligned}
& \sum_{k,m,n,\sigma_l,\sigma'_l} (Y_{l+1}^R)_{k,m}^n M_{j,m}^{\sigma_l} M_{i,k}^{\sigma'_l} (D_l^n)_{\sigma_l,\sigma'_l} \\
&= \sum_{k,m,n,\sigma'_l} (Y_{l+1}^R)_{k,m}^n M_{i,k}^{\sigma'_l} \sum_{\sigma_l} M_{j,m}^{\sigma_l} (D_l^n)_{\sigma_l,\sigma'_l} \\
&= \sum_{k,\sigma'_l} M_{i,k}^{\sigma'_l} \sum_{m,n} (Y_{l+1}^R)_{k,m}^n T_{j,m}^{\sigma'_l,n} \\
&= \sum_{k,\sigma'_l} M_{i,k}^{\sigma'_l} \sum_{m,n} (Y_{l+1}^R)_{k,m}^n T_{m,j}^{(\sigma'_l,n)T} \\
&= \sum_{k,\sigma'_l} M_{i,k}^{\sigma'_l} S_{k,j}^{\sigma'_l} \\
&= U_{i,j}
\end{aligned}$$

De 3 contracties (voor de constructie van $T_{j,m}^{\sigma'_l,n}$, $S_{k,j}^{\sigma'_l}$ en $U_{i,j}$) zijn respectievelijk

`contract (M,3,3,D,3,1)`

`contract (Y,3,2,T,4,2)`

`contract (M,3,2,S,3,3)`.

Het enige dat nu nog rest zijn de contracties voor de energie, maar die verlopen volledig analoog aan de contracties hierboven beschreven.

Bijlage C

Input-files van gebruikte Hamiltonianen

In deze Appendix geven we 2 voorbeelden van hoe we de input-file van een Hamiltoniaan moeten opstellen.

C.1 Het Heisenbergmodel

Het Heisenbergmodel wordt gegeven door de volgende Hamiltoniaan (zie hoofdstuk 3):

$$\hat{H} = \frac{J}{2} \sum_{i=1}^{L-1} \left(\hat{S}_i^+ \hat{S}_{i+1}^- + \hat{S}_i^- \hat{S}_{i+1}^+ + 2\hat{S}_i^z \hat{S}_{i+1}^z \right) - \beta \sum_{i=1}^L \hat{S}_i^z. \quad (\text{C.1})$$

Dit is een model met enkel dichtste-nabuur interacties en 3 ($=\delta$) interactie-termen per site. Per site zal de Hamiltoniaan dus $(1+3(1+1))=7$ termen bevatten. De input-file ziet er dan als volgt uit voor fysische dimensie $d = 2$ (spin- $\frac{1}{2}$): (indien de dimensie op site 1 en L dezelfde is als de dimensie op de overige sites, dan stellen we `d_edge=0`)

```
function [H_isoHeis,d,d_edge,delta,int_range]=isoHeisenberg(L,J,beta)

%model of the isotropic Heisenberg Hamiltonian in a magnetic field beta
%beta = 0: no magnetic field
%beta \= 0 and J = 0: Hamiltonian for a magnetic field only

Jz=(J)*eye(2,2);

Splus=[0 1;0 0];
Smin=[0 0;1 0];
Sz=[0.5 0;0 -0.5];
BSz=[-beta/2 0;0 beta/2];

%physical dimension
d=2;
d_edge=0;
%number of interaction terms
delta=3;
int_range=1;
```

```

H_isoHeis=randn(d,d,1+delta*(1+int_range),L);

for j=1:1:L
    H_isoHeis(:,:,1,j)=BSz;
    H_isoHeis(:,:,2,j)=Splus;
    H_isoHeis(:,:,3,j)=Smin;
    H_isoHeis(:,:,4,j)=Sz;
    H_isoHeis(:,:,5,j)=(J/2).*Smin;
    H_isoHeis(:,:,6,j)=(J/2).*Splus;
    H_isoHeis(:,:,7,j)=Jz*Sz;
end
end

```

Voor sites l wordt $C_l^1 = S^+$, $C_l^2 = S^-$, $C_l^3 = S^z$ zodat de geassocieerde $(D_l^n)_\alpha$ gegeven worden door $(D_l^1)_\alpha = (J/2) S^-$, $(D_l^2)_\alpha = (J/2) S^+$, $(D_l^3)_\alpha = J_z S^z$ (en $\alpha = 1$). Men dient dus te letten op de kruiscombinatie van de C - en D -operatoren (een S^+ met een S^- en vice versa). Het is nu makkelijk te controleren met de schema's van figuren (2.7),(2.8),(2.9) en (2.10) dat de gegeven schikking inderdaad leidt tot de correcte interacties tussen alle sites. We merken ook nog op dat de lokale operator inderdaad op plaats 1 staat in de Hamiltoniaan.

C.2 Het $J_1 - J_2$ model

Het $J_1 - J_2$ model wordt beschreven door de volgende Hamiltoniaan (zie hoofdstuk 3):

$$\hat{H} = \frac{J_1}{2} \sum_{i=1}^{L-1} \left(\hat{S}_i^+ \hat{S}_{i+1}^- + \hat{S}_i^- \hat{S}_{i+1}^+ + 2\hat{S}_i^z \hat{S}_{i+1}^z \right) + \frac{J_2}{2} \sum_{i=1}^{L-2} \left(\hat{S}_i^+ \hat{S}_{i+2}^- + \hat{S}_i^- \hat{S}_{i+2}^+ + 2\hat{S}_i^z \hat{S}_{i+2}^z \right). \quad (\text{C.2})$$

Naast dichtste-nabuur interacties laat dit model ook tweede-orde interacties toe (d.w.z dat een spintoestand op site l kan interageren met zowel site $l + 1$ als met site $l + 2$). Per site zal de Hamiltoniaan dus $(1+3(1+2))=10$ termen bevatten. De input-file ziet er dan als volgt uit voor fysische dimensie $d = 2$ (spin- $\frac{1}{2}$):

```

function [H_MajGho,d,d_edge,delta,int_range]=MajGho(L,J1,J2,beta)

Splus=[0 1;0 0];
Smin=[0 0;1 0];
Sz=[0.5 0;0 -0.5];
BSz=[-beta/2 0;0 beta/2];

%physical dimension
d=2;
d_edge=0;
%number of interaction terms
delta=3;
int_range=2;

H_MajGho=randn(d,d,1+delta*(1+int_range),L);

for j=1:1:L
    H_MajGho(:,:,1,j)=BSz;
    H_MajGho(:,:,2,j)=Smin;

```

```

H_MajGho(:, :, 3, j)=Splus;
H_MajGho(:, :, 4, j)=Sz;
H_MajGho(:, :, 5, j)=(J1/2).*Splus;
H_MajGho(:, :, 6, j)=(J1/2).*Smin;
H_MajGho(:, :, 7, j)=J1.*Sz;
H_MajGho(:, :, 8, j)=(J2/2).*Splus;
H_MajGho(:, :, 9, j)=(J2/2).*Smin;
H_MajGho(:, :, 10, j)=J2.*Sz;
end
end

```

Net als bij het Heisenbergmodel verloopt de constructie via een kruiscombinatie van S^+ - en S^- -operatoren. Er is nu echter een tweede-orde interactie zodat de Hamiltoniaan 3 termen meer bevat voor elke site. De schikking van de operatoren blijft echter hetzelfde, enkel de koppelsterkte verandert.

Bijlage D

Exact oplosbaar $J_1 - J_2$ model

We zullen de grondtoestandsenergie van een exact oplosbaar $J_1 - J_2$ model ($J_2 = \frac{J_1}{2}$) berekenen door expliciet de matrixvoorstelling van de projectie-operator uit te werken (waarom we dit doen, kan men vinden in hoofdstuk 3). Daartoe maken we gebruik van angulaire momentum-algebra, namelijk het koppelen van spins. Het koppelen van spins gebeurt door een lineaire combinatie te maken van mogelijke spinprojecties waarbij elke term gewogen wordt met een Clebsch-Gordan coëfficiënt ([35]). De mogelijke waarden van de gekoppelde spin s (bekomen door de koppeling van s_1 en s_2) worden gevonden aan de hand van de driehoeksrelatie

$$|s_1 - s_2| \leq s \leq |s_1 + s_2|. \quad (\text{D.1})$$

Voor het koppelen van 2 spin- $\frac{1}{2}$ vinden we makkelijk (met $s_1 = s_2 = \frac{1}{2}$ en dus $s = 0$ of $s = 1$)

$$\begin{aligned} |11\rangle &= |\uparrow\rangle |\uparrow\rangle \triangleq |\uparrow\uparrow\rangle \\ |10\rangle &= \frac{1}{\sqrt{2}} (|\uparrow\downarrow\rangle + |\downarrow\uparrow\rangle) \\ |1-1\rangle &= |\downarrow\downarrow\rangle \\ |00\rangle &= \frac{1}{\sqrt{2}} (|\uparrow\downarrow\rangle - |\downarrow\uparrow\rangle) \end{aligned}$$

waarbij we de gekoppelde spins genoteerd hebben met $|sm_s\rangle$ en de ongekoppelde spins met $|\frac{1}{2}m_s\rangle \triangleq |m_s\rangle = |\uparrow\rangle$ of $|\downarrow\rangle$ (bemerkt ook de definitie van de notatie in de eerste regel; deze is niet hetzelfde als de definitie van $|\uparrow\uparrow\rangle$ in hoofdstuk 4). Nu moeten we deze gekoppelde spin nogmaals koppelen met een spin- $\frac{1}{2}$ om een koppeling van 3 spins te bekomen. We noteren deze totale gekoppelde spin als S (bekomen door de koppeling van $s = 0$ of $s = 1$ met $s_3 = \frac{1}{2}$). Om de mogelijke $|Sm_S\rangle$ te vinden, maken we gebruik van de algemene spinkoppelingsformule ([35])

$$|Sm_S\rangle = \sum_{m_s} \sum_{m_{s_3}} \langle 1m_s \frac{1}{2}m_{s_3} | Sm_S \rangle |1m_s\rangle | \frac{1}{2} m_{s_3} \rangle. \quad (\text{D.2})$$

In bovenstaande formule zijn $\langle 1m_s \frac{1}{2}m_{s_3} | Sm_S \rangle$ de Clebsch-Gordan coëfficiënten die verschillend van nul zijn als $m_s + m_{s_3} = m_S$. We willen nu de projectie-operator $P_{\frac{3}{2}}$ op een spin- $\frac{3}{2}$ subruimte construeren:

$$P_{\frac{3}{2}} = \left| \begin{smallmatrix} 3 & 3 \\ 2 & 2 \end{smallmatrix} \right\rangle \left\langle \begin{smallmatrix} 3 & 3 \\ 2 & 2 \end{smallmatrix} \right| + \left| \begin{smallmatrix} 3 & 1 \\ 2 & 2 \end{smallmatrix} \right\rangle \left\langle \begin{smallmatrix} 3 & 1 \\ 2 & 2 \end{smallmatrix} \right| + \left| \begin{smallmatrix} 3 & -1 \\ 2 & -2 \end{smallmatrix} \right\rangle \left\langle \begin{smallmatrix} 3 & -1 \\ 2 & -2 \end{smallmatrix} \right| + \left| \begin{smallmatrix} 3 & -3 \\ 2 & -2 \end{smallmatrix} \right\rangle \left\langle \begin{smallmatrix} 3 & -3 \\ 2 & -2 \end{smallmatrix} \right| \quad (\text{D.3})$$

waarbij we alle $|\frac{3}{2}m_S\rangle$ moeten bepalen via (D.2). We vinden achtereenvolgens:

$$\begin{aligned} \left| \frac{3}{2} \frac{3}{2} \right\rangle &= |11\rangle \left| \frac{1}{2} \frac{1}{2} \right\rangle = |\uparrow\uparrow\rangle |\uparrow\rangle \triangleq |\uparrow\uparrow\uparrow\rangle \\ \left| \frac{3}{2} \frac{1}{2} \right\rangle &= \frac{1}{\sqrt{3}} (|\uparrow\uparrow\downarrow\rangle + |\uparrow\downarrow\uparrow\rangle + |\downarrow\uparrow\uparrow\rangle) \\ \left| \frac{3}{2} - \frac{1}{2} \right\rangle &= \frac{1}{\sqrt{3}} (|\downarrow\downarrow\uparrow\rangle + |\uparrow\downarrow\downarrow\rangle + |\downarrow\uparrow\downarrow\rangle) \\ \left| \frac{3}{2} - \frac{3}{2} \right\rangle &= |\downarrow\downarrow\downarrow\rangle \end{aligned}$$

Aangezien de 3 spins elk 2 mogelijke spin-projecties (up of down) hebben, wordt de projectie-operator in de ongekoppelde representatie voorgesteld door een (8×8) matrix in de volgende basis

$$\{|\downarrow\downarrow\downarrow\rangle, |\downarrow\downarrow\uparrow\rangle, |\downarrow\uparrow\downarrow\rangle, |\uparrow\downarrow\downarrow\rangle, |\downarrow\uparrow\uparrow\rangle, |\uparrow\downarrow\uparrow\rangle, |\uparrow\uparrow\downarrow\rangle, |\uparrow\uparrow\uparrow\rangle\}. \quad (\text{D.4})$$

Dit geeft uiteindelijk de volgende matrixvoorstelling:

$$P_{\frac{3}{2}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/3 & 1/3 & 1/3 & 0 & 0 & 0 & 0 \\ 0 & 1/3 & 1/3 & 1/3 & 0 & 0 & 0 & 0 \\ 0 & 1/3 & 1/3 & 1/3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/3 & 1/3 & 1/3 & 0 \\ 0 & 0 & 0 & 0 & 1/3 & 1/3 & 1/3 & 0 \\ 0 & 0 & 0 & 0 & 1/3 & 1/3 & 1/3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}. \quad (\text{D.5})$$

De volgende stap in het bewijs steunt op het aantonen van de volgende identiteit:

$$P_{\frac{3}{2}}(\hat{\mathbf{S}}_i, \hat{\mathbf{S}}_{i+1}, \hat{\mathbf{S}}_{i+2}) \equiv \frac{2}{3} \left(\hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_{i+1} + \hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_{i+2} + \hat{\mathbf{S}}_{i+1} \circ \hat{\mathbf{S}}_{i+2} \right) + \frac{1}{2}. \quad (\text{D.6})$$

Om deze identiteit te bewijzen maken we gebruik van de volgende matrixelementen:

$$\begin{aligned} \langle m'_i m'_{i+1} m'_{i+2} | S_i^+ S_{i+1}^- | m_i m_{i+1} m_{i+2} \rangle &= \delta_{m_i, -1/2} \delta_{m'_i, 1/2} \delta_{m_{i+1}, 1/2} \delta_{m'_{i+1}, -1/2} \delta_{m_{i+2}, m'_{i+2}} \\ \langle m'_i m'_{i+1} m'_{i+2} | S_i^- S_{i+1}^+ | m_i m_{i+1} m_{i+2} \rangle &= \delta_{m_i, 1/2} \delta_{m'_i, -1/2} \delta_{m_{i+1}, -1/2} \delta_{m'_{i+1}, 1/2} \delta_{m_{i+2}, m'_{i+2}} \\ \langle m'_i m'_{i+1} m'_{i+2} | S_i^z S_{i+1}^z | m_i m_{i+1} m_{i+2} \rangle &= m_i m_{i+1} \prod_{k=i}^{i+2} \delta_{m_k, m'_k} \\ \langle m'_i m'_{i+1} m'_{i+2} | S_i^+ S_{i+2}^- | m_i m_{i+1} m_{i+2} \rangle &= \delta_{m_i, -1/2} \delta_{m'_i, 1/2} \delta_{m_{i+1}, m'_{i+1}} \delta_{m_{i+2}, 1/2} \delta_{m'_{i+2}, -1/2} \\ \langle m'_i m'_{i+1} m'_{i+2} | S_i^- S_{i+2}^+ | m_i m_{i+1} m_{i+2} \rangle &= \delta_{m_i, 1/2} \delta_{m'_i, -1/2} \delta_{m_{i+1}, m'_{i+1}} \delta_{m_{i+2}, -1/2} \delta_{m'_{i+2}, 1/2} \\ \langle m'_i m'_{i+1} m'_{i+2} | S_i^z S_{i+2}^z | m_i m_{i+1} m_{i+2} \rangle &= m_i m_{i+2} \prod_{k=i}^{i+2} \delta_{m_k, m'_k} \end{aligned}$$

Als we ook nog rekening houden met

$$\hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_j = \frac{1}{2} \left(S_i^+ S_j^- + S_i^- S_j^+ \right) + S_i^z S_j^z \quad (\text{D.7})$$

dan kan men na enig rekenwerk aantonen dat het rechterlid van (D.6) dezelfde matrixvoorstelling heeft als (D.5). Nu moeten we nog een verband leggen met de Majumdar-Ghosh Hamiltoniaan. Beschouw daartoe

$$\begin{aligned}
\sum_{i=1}^{\infty} P_{\frac{3}{2}}(\hat{\mathbf{S}}_i, \hat{\mathbf{S}}_{i+1}, \hat{\mathbf{S}}_{i+2}) &= \sum_{i=1}^{\infty} \left(\frac{2}{3} \left(\hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_{i+1} + \hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_{i+2} + \hat{\mathbf{S}}_{i+1} \circ \hat{\mathbf{S}}_{i+2} \right) + \frac{1}{2} \right) \\
&= \sum_{i=1}^{\infty} \left(\frac{2}{3} \left(2\hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_{i+1} + \hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_{i+2} \right) + \frac{1}{2} \right) \\
&= \sum_{i=1}^{\infty} \frac{4}{3} \left(\left(\hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_{i+1} + \frac{1}{2} \hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_{i+2} \right) + \frac{3}{8} \right) \tag{D.8}
\end{aligned}$$

De meest algemene $J_1 - J_2$ Hamiltoniaan wordt gegeven door

$$\hat{H}_{MG} = J_1 \sum_i \left(\hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_{i+1} + \frac{J_2}{J_1} \hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_{i+2} \right) \tag{D.9}$$

Stel dat we een oneindig lange ketting $L \rightarrow \infty$ beschouwen in het speciale geval $J_2 = \frac{J_1}{2}$ (het Majumdar-Ghosh punt) dan is het duidelijk dat

$$\hat{H}_{MG} = \frac{3J_1}{4} \sum_{i=1}^L P_{\frac{3}{2}}(\hat{\mathbf{S}}_i, \hat{\mathbf{S}}_{i+1}, \hat{\mathbf{S}}_{i+2}) - \frac{3}{8} J_1 L. \tag{D.10}$$

Om de energie minimaal te maken in de grondtoestand $|\Theta\rangle$ zal er moeten gelden dat¹

$$\sum_i P_{\frac{3}{2}} |\Theta\rangle = 0 \tag{D.11}$$

zodat de energie in de grondtoestand E_0 van een Majumdar-Ghosh ketting ($J_2 = \frac{J_1}{2}$) in de limiet gegeven wordt door

$$E_0 = -\frac{3}{8} J_1 L. \tag{D.12}$$

Voor dit bijzondere geval kunnen we de resultaten bekomen met het VMPS-algoritme vergelijken met het analytische resultaat. Een alternatief en sneller bewijs² voor (D.8) bestaat uit het kwadrateren van

$$\hat{\mathbf{S}} = \hat{\mathbf{S}}_1 + \hat{\mathbf{S}}_2 + \hat{\mathbf{S}}_3 \tag{D.13}$$

en te stellen dat $\hat{\mathbf{S}}_2 \hat{\mathbf{S}}_3 = \hat{\mathbf{S}}_1 \hat{\mathbf{S}}_2$. Als men dan als algemene gedaante van $P_{\frac{3}{2}}$ de volgende uitdrukking

$$P_{\frac{3}{2}} = a \hat{\mathbf{S}}^2 + b$$

vooropstelt en eist dat

$$\begin{aligned}
P_{\frac{3}{2}} |S = \frac{1}{2}\rangle &= 0 \\
P_{\frac{3}{2}} |S = \frac{3}{2}\rangle &= 1
\end{aligned}$$

¹Een projectie-operator P is idempotent, $P^2 = P$, zodat de eigenwaarden 0 of 1 zijn.

²Met dank aan S. Wouters.

dan vinden we de volgende uitdrukkingen

$$\begin{aligned}\frac{3}{4}a + b &= 0 \\ \frac{15}{4}a + b &= 1\end{aligned}$$

waaruit $a = \frac{1}{3}$ en $b = -\frac{1}{4}$. Na enig rekenwerk bekomt men dan

$$P_{\frac{3}{2}} = \frac{4}{3} \left(\left(\hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_{i+1} + \frac{1}{2} \hat{\mathbf{S}}_i \circ \hat{\mathbf{S}}_{i+2} \right) + \frac{3}{8} \right)$$

waarna de verdere stappen volledig analoog verlopen als deze in het eerste bewijs.

Bijlage E

Tweedeeltjesinteractie voor de H_2 molecule

De tweedeeltjesinteractie V kan opgesplitst worden (zoals aangetoond in hoofdstuk 5) in een som van volgende termen:

- een stuk dat de on-site interactie bevat, dus $V(i, i, i, i)$
- termen die een combinatie van operatoren op twee verschillende sites bevatten: V^2
- termen die een combinatie van operatoren op drie verschillende sites bevatten: V^3
- termen die een combinatie van operatoren op vier verschillende sites bevatten: V^4

We zullen hieronder alle termen met operatoren op twee verschillende sites en hun corresponderende matrixvoorstellingen oplijsten (dit zijn dus de elementen die we nodig hebben voor de H_2 molecule). Let op dat we daarbij werken met het directe matrixelement (zie Hoofdstuk 2 van ([28])) en dus niet met het gesymmetriseerd matrixelement. De V^2 -interactie wordt gegeven door

$$\begin{aligned}
 V^2 = & \sum_{i < j} V_{ijjj} \left(a_{i,\uparrow}^\dagger a_{j,\downarrow}^\dagger a_{j,\downarrow} a_{j,\uparrow} + a_{i,\downarrow}^\dagger a_{j,\uparrow}^\dagger a_{j,\uparrow} a_{j,\downarrow} + a_{i,\uparrow} a_{j,\downarrow}^\dagger a_{j,\uparrow}^\dagger a_{j,\downarrow} + a_{i,\downarrow} a_{j,\uparrow}^\dagger a_{j,\downarrow}^\dagger a_{j,\uparrow} \right) + \\
 & V_{jiii} \left(a_{i,\uparrow}^\dagger a_{i,\downarrow} a_{i,\uparrow} a_{j,\downarrow}^\dagger + a_{i,\downarrow}^\dagger a_{i,\uparrow} a_{i,\downarrow} a_{j,\uparrow}^\dagger + a_{i,\downarrow}^\dagger a_{i,\uparrow}^\dagger a_{i,\uparrow} a_{j,\downarrow} + a_{i,\uparrow}^\dagger a_{i,\downarrow}^\dagger a_{i,\downarrow} a_{j,\uparrow} \right) + \\
 & V_{iiij} \left(a_{i,\uparrow}^\dagger a_{i,\downarrow} \left(-a_{j,\downarrow}^\dagger a_{j,\uparrow} \right) + a_{i,\downarrow}^\dagger a_{i,\uparrow} \left(-a_{j,\uparrow}^\dagger a_{j,\downarrow} \right) + a_{i,\downarrow}^\dagger a_{i,\uparrow}^\dagger a_{j,\uparrow} a_{j,\downarrow} + a_{i,\downarrow} a_{i,\uparrow} a_{j,\uparrow}^\dagger a_{j,\downarrow}^\dagger \right) + \\
 & V_{ijij} (n_{i,\uparrow} n_{j,\downarrow} + n_{i,\downarrow} n_{j,\uparrow}) + (V_{ijij} - V_{iiij}) n_{i,\uparrow} n_{j,\uparrow} + (V_{ijij} - V_{iiij}) n_{i,\downarrow} n_{j,\downarrow}. \quad (\text{E.1})
 \end{aligned}$$

Nu moeten we voor elk van deze combinaties de matrixelementen in de lokale basis uitwerken en we gaan daarbij te werk zoals ook beschreven is in hoofdstuk 5, maar nu construeren we de matrices geassocieerd aan twee verschillende sites terwijl we in hoofdstuk 5 de behandeling gedaan hebben voor drie verschillende sites. De redeneringen zijn echter volledig gelijklopend en we kiezen opnieuw $j = i + 1$ en houden in het algoritme rekening met eventuele fasefactoren als er zich op site i (en dus ook op site j) een oneven aantal operatoren bevinden. De

matrixvoorstellungen worden hieronder weergegeven.

$$a_{i,\uparrow}^\dagger a_{j,\downarrow}^\dagger a_{j,\downarrow} a_{j,\uparrow} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad (\text{E.2})$$

$$a_{i,\downarrow}^\dagger a_{j,\uparrow}^\dagger a_{j,\uparrow} a_{j,\downarrow} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad (\text{E.3})$$

$$a_{i,\uparrow} a_{j,\downarrow}^\dagger a_{j,\uparrow}^\dagger a_{j,\downarrow} = \begin{pmatrix} 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \end{pmatrix} \quad (\text{E.4})$$

$$a_{i,\downarrow} a_{j,\uparrow}^\dagger a_{j,\downarrow}^\dagger a_{j,\uparrow} = \begin{pmatrix} 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \quad (\text{E.5})$$

$$a_{i,\uparrow}^\dagger a_{i,\downarrow} a_{i,\uparrow} a_{j,\downarrow}^\dagger = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{pmatrix} \quad (\text{E.6})$$

$$a_{i,\downarrow}^\dagger a_{i,\uparrow} a_{i,\downarrow} a_{j,\uparrow}^\dagger = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad (\text{E.7})$$

$$a_{i,\downarrow}^\dagger a_{i,\uparrow}^\dagger a_{i,\uparrow} a_{j,\downarrow} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad (\text{E.8})$$

$$a_{i,\uparrow}^\dagger a_{i,\downarrow}^\dagger a_{i,\downarrow} a_{j,\uparrow} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad (\text{E.9})$$

$$a_{i,\uparrow}^\dagger a_{i,\downarrow} \left(-a_{j,\downarrow}^\dagger a_{j,\uparrow} \right) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad (\text{E.10})$$

$$a_{i,\downarrow}^\dagger a_{i,\uparrow} \left(-a_{j,\uparrow}^\dagger a_{j,\downarrow} \right) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad (\text{E.11})$$

$$a_{i,\downarrow}^\dagger a_{i,\uparrow}^\dagger a_{j,\uparrow} a_{j,\downarrow} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad (\text{E.12})$$

$$a_{i,\downarrow} a_{i,\uparrow} a_{j,\uparrow}^\dagger a_{j,\downarrow}^\dagger = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \quad (\text{E.13})$$

$$n_{i,\uparrow} n_{j,\downarrow} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (\text{E.14})$$

$$n_{i,\downarrow} n_{j,\uparrow} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (\text{E.15})$$

$$n_{i,\uparrow} n_{j,\uparrow} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (\text{E.16})$$

$$n_{i,\downarrow} n_{j,\downarrow} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (\text{E.17})$$

Voor V^3 en V^4 hebben we respectievelijk $\delta_3 = 60$ en $\delta_3 = 48$ verschillende termen, zodat we in totaal $\delta_3 + \delta_4 = 108$ verschillende matrixcombinaties hebben. Dit is een groot aantal en we hebben dan ook een programma ontwikkeld dat een logica bevat om dergelijke matrixvoorstellingen te genereren door enerzijds in te geven hoe de creatie- en annihilatie-operatoren ten opzichte van elkaar staan en anderzijds te vermelden welke operatoren zich op dezelfde site bevinden. Dit kan men dan gebruiken indien men het programma wil uitbreiden naar long-range moleculaire kettingen.

Bibliografie

- [1] V. Van Speybroeck, “Simulations and modeling for the nanoscale,” 2010. Cursusnota’s UGent.
- [2] J. Paldus, J. Cizek, M. Saute, and A. Laforgue, “Correlation problems in atomic and molecular systems. vi. coupled cluster approach to open-shell systems,” *Physical Review A*, vol. 17, Maart 1978.
- [3] A. Halkier, H. Larsen, J. Olsen, and P. Jorgensen, “Full configuration interaction benchmark calculations of first-order one-electron properties of bh and hf,” *Journal of Chemical Physics*, vol. 110, Januari 1999.
- [4] U. Schollwock, “The density-matrix renormalization group in the age of matrix product states,” 2010. arXiv:1008.3477v1 [cond-mat.str-el].
- [5] F. Verstraete, V. Murg, and J. Cirac, “Matrix product states, projected entangled pair states, and variational renormalization group methods for quantum spin systems,” *Advances in Physics*, vol. 57, Maart-April 2008.
- [6] K. Wilson, “The renormalization group and critical phenomena,” 1982. Nobel Lecture.
- [7] M. Fisher, “Renormalization group theory: Its basis and formulation in statistical physics,” *Reviews of Modern Physics*, vol. 70, April 1998.
- [8] K. Wilson, “Renormalization group and critical phenomena,” *Physical Review B*, vol. 4, November 1971.
- [9] D. Shirkov, “Evolution of the bogoliubov renormalization group.” arXiv:hep-th/9909024v1.
- [10] B. Delamotte, “A hint of renormalization.” arXiv:hep-th/0212049v3.
- [11] U. Schollwock, “The density-matrix renormalization group,” 2004. arXiv:cond-mat/0409292v1 [cond-mat.str-el].
- [12] S. White, “Density-matrix algorithms for quantum renormalization groups,” *Physical Review B*, vol. 48, Oktober 1993.
- [13] M. Karbach and G. Muller, “Introduction to the bethe ansatz i,” 1997. arXiv:cond-mat/9809162 v1.
- [14] J. Preskill, “Quantum computation,” 2011. Cursus Caltech.
- [15] P. Calabrese and J. Cardy, “Entanglement entropy and quantum field theory,” *Journal of Statistical Mechanics: Theory and Experiment*, Juni 2004.
- [16] L. Amico, R. Fazio, A. Osterloh, and V. Vedral, “Entanglement in many-body systems,” 2008. arXiv:quant-ph/0703044v3.
- [17] G. Vidal, J. Latorre, E. Rico, and A. Kitaev, “Entanglement in quantum critical phenomena,” *Physical Review Letters*, vol. 90, Juni 2003.
- [18] N. Schuch, M. Wolf, F. Verstraete, and J. Cirac, “Entropy scaling and simulability by matrix product sates,” 2008. arXiv:0705.0292v2 [quant-ph].

- [19] J. Hachmann, W. Cardoen, and G. Chan, “Multireference correlation in long molecules with the quadratic scaling density matrix renormalization group,” *J. Chem. Phys.*, vol. 125, Oktober 2006.
- [20] S. Manmana, A. Lauchli, F. Essler, and M. F., “Phase diagram and continuous pair-unbinding transition of the bilinear-biquadratic $s=1$ heisenberg chain in a magnetic field.” arXiv:1012.4518v1 [cond-mat.str-el].
- [21] R. Chhajlany, P. Tomczak, A. Wojcik, and J. Richter, “Entanglement in the majumdar-ghosh model,” *Physical Review*, Maart 2007.
- [22] M. Karbach, K. Hu, and G. Muller, “Introduction to the bethe ansatz ii,” 1998. arXiv:cond-mat/9809163 v1.
- [23] M. Karbach, K. Hu, and G. Muller, “Introduction to the bethe ansatz iii,” 2000. arXiv:cond-mat/0008018 v1.
- [24] H. Fan, V. Korepin, and V. Roychowdhury, “Entanglement in a valence-bond-solid state.” arXiv:quant-ph/0406067v3.
- [25] I. Affleck, T. Kennedy, E. Lieb, and H. Tasaki, “Rigorous results on valence-bond ground states in antiferromagnets,” *Physical Review Letters*, vol. 59, Augustus 1987.
- [26] S. Geraedts and E. Sorensen, “Exact results for the bipartite entanglement entropy of the aklt spin-1 chain.” arXiv:0912.4733v2 [cond-mat.stat-mech].
- [27] S. Barnes and S. Maekawa, “A jordan-wigner transformation for the t-j and hubbard models with holes,” *Journal of Physics: Condensed Matter*, December 2001.
- [28] W. Dickhoff and D. Van Neck, *Many-Body Theory Exposed! Propagator Description of Quantum Mechanics in Many-Body Systems, 2nd Edition*. World Scientific Publishing, 2008.
- [29] E. Lieb and F. Wu, “The one-dimensional hubbard model: A reminiscence,” 2002. arXiv:cond-mat/0207529v2 [cond-mat.str-el].
- [30] S. Daul and R. Noack, “Ferromagnetic transition and phase diagram of the one-dimensional hubbard model with next-nearest-neighbor hopping,” 1997. arXiv:cond-mat/9712310v1 [cond-mat.str-el].
- [31] EMSL Basis Set Exchange, <https://bse.pnl.gov/bse/portal>.
- [32] B. Carlson and J. Keller, “Orthogonalization procedures and the localization of wannier functions,” *Physical Review*, vol. 105, Januari 1957.
- [33] S. Obara and A. Saika, “Efficient recursive computation of molecular integrals over cartesian gaussian functions,” *J. Chem. Phys.*, vol. 84, April 1986.
- [34] S. Wouters. ThING, <https://github.com/SebWouters/ThING>.
- [35] B. Bransden and C. Joachain, *Physics of Atoms and Molecules*. Pearson Education Limited, 2003.