

Cover Page



Universiteit Leiden



The following handle holds various files of this Leiden University dissertation:  
<http://hdl.handle.net/1887/80413>

**Author:** Steudtner, M.

**Title:** Methods to simulate fermions on quantum computers with hardware limitations

**Issue Date:** 2019-11-20

# Samenvatting

Dit proefschrift bevat een verzameling theoretische studies gericht op het aanpassen van quantumalgoritmen aan de hardware van quantumcomputers. Het overkoepelende onderwerp van dit onderzoek is om digitale quantumsimulatie mogelijk te maken, het proces van het benaderen van de grondtoestand van een willekeurig fysisch systeem met behulp van een quantumcomputer. De impact van een quantumcomputer kan potentieel heel groot zijn voor de simulatie van eigenschappen van moleculen en materialen. Er bestaan algoritmen voor de simulatie van elektronische eigenschappen, maar die zijn niet zonder meer toepasbaar op realistische quantumschakelingen. Het lijkt erop dat onze verwachtingen van een volwaardige quantumcomputer nog steeds onze mogelijkheden overtreffen om er zo één te bouwen. Om quantumsimulatie toch mogelijk te maken, proberen we in dit proefschrift quantumalgoritmen aan te passen aan drie verschillende soorten realistische beperkingen. In het bijzonder behandelen we twee van die beperkingen in de context van fermionische quantumsimulatie en bespreken we de derde in verband met quantumfoutcorrectie.

De eerste beperking is het tekort aan qubits. Omdat bestaande quantumcomputers hooguit enkele tientallen qubits bevatten, moeten we quantuminformatie op een efficiënte manier opslaan. Veel algoritmen zijn niet efficiënt, en dit is met name het geval bij digitale quantumsimulatie wanneer het fysieke systeem wordt gemodelleerd met behulp van de Jordan-Wigner-transformatie – een bekende manier om elektronische orbitalen te vervangen door een spinketen. Deze manier om informatie op te slaan verspilt qubits, omdat slechts een kleine hoeveelheid van de spinconfiguraties relevant zijn voor de simulatie van de elektronen. In hoofdstuk twee willen we daarom qubits sparen door ons te beperken tot alleen de relevante configuraties. De sleutel tot onze methode is dat

alle quantumgegevens die zijn opgeslagen in het qubit-geheugen een superpositie zijn van bitstrings die het gesimuleerde systeem beschrijven. Door het gebruik van binaire codes worden alleen fysisch relevante superposities toegewezen aan iets kortere bitstrings, en dus opgeslagen in minder qubits. Deze procedure gaat weliswaar ten koste van een langere duur van de simulatie, maar dat nadeel valt in het niet bij het voordeel dat de simulatie kan worden uitgevoerd met een kleiner aantal qubits.

De tweede beperking komt van de connectiviteit van qubits. Qubits hebben vaste posities en ons vermogen om bewerkingen uit te voeren is doorgaans beperkt tot nabijgelegen qubits. De meeste quantumcomputers worden op een tweedimensionaal vierkant rooster ontworpen, waarbij elke qubit gekoppeld kan worden aan de vier burens. De bestaande algoritmes van quantumsimulatie maken geen optimaal gebruik van deze connectiviteit, doordat de lokaliteit van interacties in de simulatie verloren gaat. Het is van essentieel belang dat lokale interacties tussen elektronen geïmplementeerd kunnen worden in een lokale groep van nabijgelegen qubits. Op deze manier kan het quantumalgoritme parallel worden geschakeld en blijft de looptijd kort. Gezien de limieten die zijn vastgesteld door decoherentie, kan de resulterende afname in algoritmische looptijd zelfs de beslissende factor zijn voor de haalbaarheid van een berekening. In hoofdstuk drie sleutelen we opnieuw aan de manier waarop fysische toestanden in het geheugen worden weergegeven. De Jordan-Wigner-transformatie, die de lokaliteit niet behoudt, wordt hier samengevoegd met een codelaag. Deze keer is de code echter quantum – een constructie met codewoorden die niet langer als bitstrings kunnen worden beschouwd. Om de lokaliteit te behouden betalen we een prijs, het aantal qubits moet verdubbeld worden. Maar het grote voordeel is dat het nu mogelijk wordt om tweedimensionale systemen in een constant aantal algoritmische stappen te simuleren.

Ten slotte beschouwen we de derde beperking, die vrij vaak onder het tapijt wordt geveegd. Voor apparaten met een groot aantal qubits zullen beperkingen voor parallele werking verschijnen vanwege een beperkte mogelijkheid om al de qubits tegelijkertijd aan te sturen. Een realistische lay-out voor een toekomstig quantumapparaat is een dicht opgepakte matrix van qubits, zo dicht zelfs dat ze worden gemanipuleerd door draden die contact maken met de periferie van de matrix. In hoofdstuk vier beschouwen we een dergelijk dwarsbalkontwerp voor spinqubits in quantumdots van silicium. De qubits worden aangestuurd door

spanningslijnen, en voor quantumbewerkingen moeten ten minste twee lijnen worden geactiveerd die alleen het deel van de chip beïnvloeden waarop ze elkaar snijden. Wanneer we echter proberen meerdere bewerkingen parallel uit te voeren, worden niet alleen de kruispunten die we wensen geactiveerd, maar ook andere kruispunten waar dit niet zou moeten. Bij die valse kruisingen worden ongewenste quantumoperaties geïnduceerd. Als uitweg gebruiken we een bijzondere eigenschap van de voorgestelde architectuur en verplaatsen we bijvoorbeeld elektronen (qubits) tussen aangrenzende quantumdots: wanneer de dots aan de onbedoelde kruisingen leeg zijn, kunnen ongewenste operaties het quantumgeheugen niet beschadigen. Door het aanbieden van ladingsverdelingen met deze eigenschappen, evenals instructies voor de werking van de lijnen, vinden we quantumfoutcorrectieprogramma's die compatibel zijn met de dwarsbalkarchitectuur.

