THM TECHNISCHE HOCHSCHULE MITTELHESSEN | CAMPUS GIESSEN | MNI Mathematik, Naturwissenschaften und Informatik

Masterthesis

Überprüfung des No-Signaling Theorems an frei zugänglichen Quantencomputern

vorgelegt dem
Fachbereich Mathematik, Naturwissenschaften und Informatik
der Technischen Hochschule Mittelhessen
von

Cedric Schacht

10. Oktober 2025

Referent: Prof. Dr. Bettina Just
Co-Referent: Prof. Dr. Christoph Gallus

Acknowledgement

I gratefully acknowledge the Jülich Supercomputing Centre (https://www.fz-juelich.de/ias/jsc) for funding this project by providing computing time on the D-Wave Advantage™System JUPSI through the Jülich UNified Infrastructure for Quantum computing (JUNIQ).

Eidesstattliche Erklärung

Hiermit versichere ich, die vorliegende Arbeit selbstständig und unter ausschließlicher Verwendung der angegebenen Literatur und Hilfsmittel erstellt zu haben.

Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungsbehörde vorgelegt und auch nicht veröffentlicht.

Gießen, 10. Oktober 2025

Abstract

Mit Shor's Algorithmus ist gezeigt, dass für Schaltkreis Quantencomputer für bestimmte Probleme bessere Algorithmen existieren, als für klassische Computer.[1] Annealing Rechner lösen Optimierungsprobleme, viele klassische Berechnungsprobleme, wie die Faktorisierung von Zahlen, lassen sich als solche Optimierungsprobleme beschreiben.[2] Um in der Praxis Anwendung zu finden, ist es notwendig, die Limitationen der verschiedenen Rechnerarchitekturen zu berücksichtigen.

Diese Masterthesis untersucht das No-Signaling Theorem auf frei zugänglichen Quantencomputern. Um die No-Signaling Eigenschaft darzustellen, wird das Quantum Cube Model (erstmals beschrieben in Just 2020[3]) verwendet und auf die neuen Anforderungen erweitert.

Ziel ist es, Experimente durchzuführen, die einen Verstoß der No-Signaling Eigenschaft nachweisen. Dazu werden Experimente auf frei zugänglichen Schaltkreis Rechnern von IBM durchgeführt. Zusätzlich werden Experimente auf annealing Rechnern von D-Wave durchgeführt, um zu untersuchen, ob äquivalente Lösungen in Optimierungsproblemen gleichermaßen gefunden werden. Dazu werden aus booleschen Funktionen Optimierungsprobleme mit vielen gleichwertigen Lösungen erzeugt.

Diese Arbeit liefert drei sehr unterschiedliche Erkenntnisse:

1. Das Quantum Cube Model ist noch leistungsfähiger als bisher bekannt. So lässt sich die No-Signaling Eigenschaft von Quantensystemen graphisch mit dem Quantum Cube Model darstellen.

2. Zeitgleich mit dieser Arbeit zeigen Rybotycki u.a. 2025[4], dass Quanten Rechner die No-Signaling Eigenschaft in bestimmten Fällen brechen. In dieser Arbeit wird gezeigt, dass schon bei einfachen Schaltkreisen fehlerhafte Ergebnisse erzeugt werden, die nicht den Erwartungen entsprechen. Aussagen zu der No-Signaling Eigenschaft lassen sich so nicht zuverlässig treffen.

3. Diese Arbeit liefert erstmals ein Verfahren beliebige boolesche Funktionen als Optimierungsprobleme zu beschreiben. So lassen sich Optimierungsprobleme mit bestimmten Eigenschaften, wie einer großen Anzahl gleichwertiger Lösungen, erzeugen. Ein Optimierungsproblem auf Basis einer booleschen Funktion, die durch die Ungleichung des CHSH-Games inspiriert ist, zeigt in der Auswertung der Experimente ist eine Diskrepanz zwischen den Lösungen des annealing Rechners von D-Wave, und dem Dimod Simulator. So

[1] Peter W. Shor. „Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer". In: *SIAM Journal on Computing* 26.5 (Okt. 1997), S. 1484–1509. ISSN: 0097-5397, 1095-7111. DOI: 10.1137/S0097539795293172. arXiv: quant-ph/9508027. URL: http://arxiv.org/abs/quant-ph/9508027 (besucht am 15. 09. 2025).

[2] Shuxian Jiang u. a. „Quantum Annealing for Prime Factorization". In: *Scientific Reports* 8.1 (5. Dez. 2018), S. 17667. ISSN: 2045-2322. DOI: 10.1038/s41598-018-36058-z. URL: https://www.nature.com/articles/s41598-018-36058-z (besucht am 15. 09. 2025).

[3] Bettina Just. *Quantencomputing kompakt: Spukhafte Fernwirkung und Teleportation endlich verständlich*. IT kompakt. Berlin, Heidelberg: Springer, 2020. ISBN: 978-3-662-61888-2 978-3-662-61889-9. DOI: 10.1007/978-3-662-61889-9. URL: https://link.springer.com/10.1007/978-3-662-61889-9 (besucht am 05. 06. 2025).

[4] Tomasz Rybotycki u. a. „Violation of No-Signaling on a Public Quantum Computer". In: *Advanced Quantum Technologies* 8.6 (Juni 2025), S. 2400661. ISSN: 2511-9044, 2511-9044. DOI: 10.1002/qute.202400661. arXiv: 2409.11348 [quant-ph]. URL: http://arxiv.org/abs/2409.11348 (besucht am 07. 08. 2025).

ist der Anteil der optimalen Lösungen auf dem annealing Rechner deutlich höher, als in der Simulation. Die gemessenen Verteilungen weichen jedoch schon bei kleinen Optimierungsproblemen ausschließlich auf dem annealing Rechner von der erwarteten Verteilung ab, wenn Funktionen mit mehren gleichwertigen Lösungen berechnet werden.

Inhaltsverzeichnis

Abstract v

1 Einleitung 1

2 Grundlagen 5
2.1 Klassische Bits und Qubits . . . 5
2.2 Messung eines Qubits . . . 7
2.3 Schaltkreis Modell und Annealing . . . 9
2.4 Quantum Cube Model . . . 10
2.5 Matrix eines Schaltkreises . . . 13
2.6 Quadratic Unconstrained Binary Optimization . . . 15
2.7 No-Signaling Theorem . . . 18
2.8 Visualisierung des No-Signaling Theorems . . . 21
2.9 CHSH-Game . . . 27

3 Experimente 29
3.1 Schaltkreis Experimente . . . 30
3.1.1 Schaltkreis — $H|0\rangle$ . . . 30
3.1.2 Schaltkreis — $H \otimes H|00\rangle$ . . . 32
3.1.3 Schaltkreis — $CNOT_{0,1}(H \otimes \mathbf{I}_2)|00\rangle$ . . . 35
3.1.4 Schaltkreis — CHSH-Game . . . 38
3.2 Annealing Experimente . . . 41
3.2.1 QUBO 00 . . . 42
3.2.2 QUBO 10 . . . 44
3.2.3 QUBO Gleichverteilung über zwei Ergebnissen . . . 47
3.2.4 QUBO Gleichverteilung über vier Ergebnissen . . . 51
3.2.5 EPR als $f(a,b) : a \Leftrightarrow b$ . . . 54
3.2.6 Asymmetrische Funktion . . . 57
3.2.7 CHSH-Game . . . 62

4 Fazit 67

Literatur 69

A: Matrizen der Gatter 70

B: BQM Utils 71

Abbildungsverzeichnis

2.1 Quantum Cube Model für den Zustand $\sqrt{\frac{3}{4}}|0\rangle + \sqrt{\frac{1}{4}}e^{i\pi 1/3}|1\rangle$ . . 11
2.2 Wirkung von Hadamard auf den Zustand $|1\rangle$ 11
2.3 Quantum Cube Model für den Zustand $|q_1 q_2\rangle$ aus Gleichung 2.23 11
2.4 Wirkung von Hadamard auf den Zustand $|000\rangle$ 12
2.5 Messung des zweiten von zwei Qubits 12
2.6 Schaltkreis auf einem Qubit 13
2.7 Wirkung von Hadamard auf den Zustand $|0\rangle$ 14
2.8 Wirkung von PauliZ auf den Zustand $\frac{1}{\sqrt{2}}|0\rangle + |1\rangle$ 14
2.9 Wirkung von PauliX auf den Zustand $\frac{1}{\sqrt{2}}|0\rangle + |1\rangle$ 14
2.10 Schaltkreis mit drei Qubits 14
2.11 Erster Zeitschritt aus Abbildung 2.10 14
2.12 Aufteilung eines Systems mit drei Qubits auf Alice (Qubits 1 und 2) und Bob (Qubit 3) 21
2.13 Sicht von Alice auf das System aus drei Qubits 22
2.14 Sicht von Bob auf das System aus drei Qubits 23
2.15 Sicht von Alice auf unitäre Transformationen auf Bobs Qubit . . 23
2.16 Sicht von Bob auf unitäre Transformationen auf Alices Qubits . . 24
2.17 Sicht auf Alices Teilsystem vor und nach Messung von Bobs Qubit 24
2.18 Visualisierung des Skalarproduktes der oberen beiden Amplitudenvektoren von Alice 25
2.19 Visualisierung des Skalarproduktes der oberen beiden Amplitudenvektoren von Alice nach Messung von Bobs Qubit 25
2.20 Sicht auf Bobs Teilsystem nach Messung des ersten Qubits von Alice 26
2.21 Skalarprodukt des vorderen und hinteren Amplitudenvektors von Bob 26
2.22 Skalarprodukt des vorderen und hinteren Amplitudenvektors von Bob nach Messung des ersten Qubits von Alice 27
2.23 Darstellung eines Mixed States aus Sicht von Alice 27
2.24 Aufbau des CHSH-Games mit Kommunikationsbarriere 28

3.1 Schaltkreis Hadamard auf einem Qubit 30
3.2 Erwartete Ergebnisse des Schaltkreises aus Abbildung 3.1. 31
3.3 Wirkung von Hadamard auf $|0\rangle$ 31
3.4 Ergebnisse von 1000 Ausführungen des Schaltkreises aus Abbildung 3.1 (Single Hadamard) auf dem IBM Rechner „ibm_brisbane“. 32
3.5 Zwei Qubit Schaltkreis mit je einem Hadamard 33
3.6 Erwartete Ergebnisse des Schaltkreises aus Abbildung 3.5. 33
3.7 Wirkung von $H \otimes H$ auf $|00\rangle$ 34
3.8 Ergebnisse von 1000 Ausführungen des Schaltkreises aus Abbildung 3.5 (Double Hadamard) 34
3.9 Schaltkreis zur Erzeugung eines EPR-Paars 36
3.10 Erwartete Ergebnisse des Schaltkreises aus Abbildung 3.9. 36
3.11 Wirkung des Schaltkreises aus Abbildung 3.9 auf den Zustand $|00\rangle$ 37
3.12 Ergebnisse von 1000 Ausführungen des Schaltkreises aus Abbildung 3.9 (EPR-Schaltkreises) 37

3.13 Schaltkreis CHSH-Game mit Quantenorakel für Result Qubit. abbycross. *CHSH Game*. IBM Quantum Learning. Aug. 2025. URL: https://quantum.cloud.ibm.com/learning/en/courses/basics-of-quantum-information/entanglement-in-action/quantum.cloud.ibm.com/learning/en/courses/basics-of-quantum-information/entanglement-in-action/chsh-game (besucht am 27. 08. 2025) 38
3.14 Erwartete Ergebnisse des Schaltkreises aus Abbildung 3.13. . . . 40
3.15 Ergebnisse von 1000 Ausführungen des Schaltkreises aus Abbildung 3.13 (CHSH-Schaltkreises) 41
3.16 Erwartete Ergebnisse des QUBOs für Gleichung 3.23. 43
3.17 Ergebnisse der Simulation des QUBOs aus Gleichung 3.23 (QUBO 00). 43
3.18 Gemessene Ergebnisse des QUBOs für Gleichung 3.23 (QUBO 00). 44
3.19 Erwartete Ergebnisse des QUBOs aus für Gleichung 3.25. 45
3.20 Ergebnisse der Simulation des QUBOs aus für Gleichung 3.25 (QUBO 10). 46
3.21 Gemessene Ergebnisse des QUBOs aus für Gleichung 3.25 (QUBO 10). 47
3.22 Erwartete Ergebnisse des QUBOs aus Tabelle 3.2. 48
3.23 Ergebnisse der Simulation des QUBOs aus Tabelle 3.2 (Gleichverteilung über 2 Ergebnissen). 49
3.24 Ergebnisse des QUBOs aus Tabelle 3.2 (Gleichverteilung über 2 Ergebnissen). 50
3.25 Erwartete Ergebnisse des QUBOs aus Tabelle 3.3. 52
3.26 Ergebnisse der Simulation des QUBOs aus Tabelle 3.3 (Gleichverteilung über 4 Ergebnissen). 53
3.27 Erwartete Ergebnisse des QUBOs aus Tabelle 3.3 (Gleichverteilung über 4 Ergebnissen). 54
3.28 Erwartete Ergebnisse des QUBOs aus Tabelle 3.4. 55
3.29 Ergebnisse der Simulation des QUBOs aus Tabelle 3.4 (EPR als boolesche Funktion). 56
3.30 Erwartete Ergebnisse des QUBOs aus Tabelle 3.4 (EPR als boolesche Funktion). 57
3.31 Distanzen zwischen den optimalen Lösungen des asymmetrischen QUBOs aus Tabelle 3.5. 58
3.32 Ergebnisse der Simulation des QUBOs aus Tabelle 3.3. 59
3.33 Ergebnisse der Simulation des asymmetrischen QUBOs aus Tabelle 3.5. 60
3.34 Ergebnisse der Simulation des asymmetrischen QUBOs aus Tabelle 3.5. 61
3.35 Erwartete Ergebnisse des QUBOs aus Tabelle 3.6. 63
3.36 Ergebnisse der Simulation des QUBOs aus Tabelle 3.6 (CHSH-QUBO). 64
3.37 Erwartete Ergebnisse des QUBOs aus Tabelle 3.6 (CHSH-QUBO). 65

Listings

2.1 Übersetzen einer booleschen Funktion in ein BinaryQuadratic-Model in Python 17
3.1 Den Schaltkreis aus Abbildung 3.1 mit Qiskit anlegen 30
3.2 Den Schaltkreis aus Abbildung 3.5 mit Qiskit anlegen 33
3.3 EPR-Paar mit Qiskit anlegen 36
3.4 Python Code, um den Schaltkreis aus Abbildung 3.13 anzulegen. 39
3.5 Anlegen des Optimierungsproblems aus Gleichung 3.23 42
3.6 Anlegen des Optimierungsproblems aus Gleichung 3.25 45
3.7 Anlegen des Optimierungsproblems aus Tabelle 3.2 in Python . . 48
3.8 Anlegen des Optimierungsproblems aus Tabelle 3.3 in Python . . 51
3.9 Anlegen des Optimierungsproblems aus Tabelle 3.4 in Python . . 55
3.10 Anlegen des Optimierungsproblems aus Tabelle 3.5 in Python . . 59
3.11 Anlegen des Optimierungsproblems aus Tabelle 3.6 in Python . . 62
B.1 f_to_bqm 73
B.2 pretty_print_bqm 73

1 Einleitung

Quantencomputer können einige Probleme mit weniger Berechnungsschritten lösen als jeder (bislang bekannte) klassische Algorithmus. Bei Quantenalgorithmen ist meist das Schaltkreis Modell des Quantencomputing gemeint. So steht mit Shor's Algorithmus ein Algorithmus bereit, der unter anderem zum Faktorisieren von Zahlen verwendet werden kann.[1] Dieser Algorithmus ist besonders im Bereich der Kryptographie von Interesse.

Im Gegensatz zum Schaltkreis Modell existiert noch ein weiteres Modell, dass spezifisch Optimierungsprobleme berechnen kann.[2] Rechner dieses Modells werden annealing Rechner genannt. Die Möglichkeit Optimierungsprobleme effizient berechnen zu können, bietet einige Vorteile. Neben Problemen die klassisch als Optimierungsprobleme beschrieben werden, können einige klassische Berechnungsprobleme als Optimierungsprobleme formuliert werden. So ist auch die Berechnung von Primfaktoren, durch Umformung, als Optimierungsproblem auf annealing Rechnern möglich.[3]

Um in der Praxis Anwendung zu finden, ist es notwendig, die Limitationen der verschiedenen Rechnerarchitekturen zu berücksichtigen. Dies betrifft neben der Anzahl der Qubits, auch die Fehlertoleranz und Präzision des Systems.

Schaltkreis Rechner unterliegen einer gewissen Fehleranfälligkeit, so ist die No-Signaling Eigenschaft unter gewissen Bedingungen verletzt.[4] Im Rahmen dieser Arbeit werden Experimente auf Schaltkreis Rechnern durchgeführt. Die Resultate der Experimente zeigen, dass Schaltkreis Rechner schon vor einer Verletzung der No-Signaling Eigenschaft fehleranfälliger sind als erwartet.

Für annealing Rechner besteht die Limitation darin, wie schnell das System zur Berechnung abgekühlt werden kann. Diese Abkühlgeschwindigkeit hängt von der Differenz der beiden besten Lösungen ab, diese sind im Allgemeinen jedoch nicht bekannt (sonst ist das Problem ja schon gelöst).[5] In dieser Arbeit wird erstmalig untersucht, welche Verteilungen bei Ergebnissen auftreten, wenn es eine große Zahl an gleichwertigen optimalen Lösungen gibt.

Die No-Signaling Eigenschaft im Quantencomputing bedeutet, dass keine Operationen auf einem Teilsystem in einem anderen Teilsystem beobachtbar sind. So kann weder durch unitäre Transformationen, noch durch Messungen Information von einem Teilsystem zu einem anderen Teilsystem übertragen werden, ohne Kommunikation außerhalb des System.

---

[1] Peter W. Shor. „Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer". In: *SIAM Journal on Computing* 26.5 (Okt. 1997), S. 1484–1509. ISSN: 0097-5397, 1095-7111. DOI: 10.1137/S0097539795293172. arXiv: quant-ph/9508027. URL: http://arxiv.org/abs/quant-ph/9508027 (besucht am 15. 09. 2025).

[2] *The Advantage™ Quantum Computer | D-Wave*. URL: https://www.dwavequantum.com/solutions-and-products/systems/ (besucht am 16. 08. 2025).

[3] Fred Glover, Gary Kochenberger und Yu Du. „Quantum Bridge Analytics I: A Tutorial on Formulating and Using QUBO Models". In: (4. Nov. 2019). DOI: 10.48550/arXiv.1811.11538. URL: https://arxiv.org/abs/1811.11538 (besucht am 30. 08. 2025).

[4] Tomasz Rybotycki u. a. „Violation of No-Signaling on a Public Quantum Computer". In: *Advanced Quantum Technologies* 8.6 (Juni 2025), S. 2400661. ISSN: 2511-9044, 2511-9044. DOI: 10.1002/qute.202400661. arXiv: 2409.11348 [quant-ph]. URL: http://arxiv.org/abs/2409.11348 (besucht am 07. 08. 2025).

[5] Bruce Hajek. „Cooling Schedules for Optimal Annealing". In: *Math. Oper. Res.* 13.2 (1. Mai 1988), S. 311–329. ISSN: 0364-765X.

Um Änderungen im Systemzustand während eines Quantenalgorithmus darzustellen, kann das Quantum Cube Model verwendet werden. Dieses stellt die Amplituden (und Phasenwinkel) des Systemzustands an den Ecken eines Würfels dar.[6] Um auch die No-Signaling Eigenschaft visuell darzustellen, liefert diese Arbeit eine neue Erweiterung des Quantum Cube Models. Diese ermöglicht es, Operationen auf Teilsystemen aus Sicht eines anderen Systemteils darzustellen.

System	Experiment	Simulation: Anteil korrekter Lösungen	Simulation: Verteilung	Rechner: Anteil korrekter Lösungen	Rechner: Verteilung	Methode
IBM	3.1.1 Schaltkreis - $H\|0\rangle$	—		—	✓	χ^2-Test
	3.1.2 Schaltkreis - $(H \otimes H)\|00\rangle$	—		—	✗	χ^2-Test
	3.1.3 Schaltkreis - $CNOT_{0,1}(H \otimes \mathbf{I}_2)\|00\rangle$	—		98,2%	✗	χ^2-Test
	3.1.4 Schaltkreis - CHSH-Game	—		83,9%	✗	χ^2-Test
D-Wave	3.2.1 QUBO 00	32,2%	—	95,2%	—	—
	3.2.2 QUBO 10	29,8%	—	98%	—	—
	3.2.3 Gleichverteilung über zwei Ergebnissen	58,2%	✓	100%	✗	χ^2-Test
	3.2.4 Gleichverteilung über vier Ergebnissen	76,7%	✓	99,9%	✗	χ^2-Test
	3.2.5 EPR-Paar als $f(a,b): a \Leftrightarrow b$	51,5%	✓	99,1%	✗	χ^2-Test
	3.2.6 Asymmetrische Funktion	64,9%	✓	86,1%	✗	χ^2-Test
	3.2.7 CHSH-QUBO	69,2%	✓	43,5%	✗	χ^2-Test

— Nicht durchgeführt
✓ Erwartete Verteilung erfüllt
✗ Nicht aussagekräftig
✗ Erwartete Verteilung nicht erfüllt

Tabelle 1.1: Übersicht über die durchgeführten Experimente

Diese Arbeit ist wie folgt aufgebaut. In Kapitel 2 sind die mathematischen Grundlagen für die Experimente aus Kapitel 3 beschrieben. Dabei sind in Ab-

[6] Bettina Just. *Quantencomputing kompakt: Spukhafte Fernwirkung und Teleportation endlich verständlich*. IT kompakt. Berlin, Heidelberg: Springer, 2020. ISBN: 978-3-662-61888-2 978-3-662-61889-9. DOI: 10.1007/978-3-662-61889-9. URL: https://link.springer.com/10.1007/978-3-662-61889-9 (besucht am 05. 06. 2025).

schnitt 2.1 bis Abschnitt 2.3 die Gemeinsamkeiten und Unterschiede zwischen den beiden Modellen des Quantencomputing aufgeschlüsselt. Abschnitt 2.4 beschreibt das grundlegende Quantum Cube Model. Das spezifische Berechnungsmodell der Schaltkreis Rechner ist in Abschnitt 2.5 beschrieben. Äquivalent dazu folgt in Abschnitt 2.6 das Berechnungsmodell für annealing Rechner.

Quantencomputer unterliegen (in der Theorie) der No-Signaling Eigenschaft. Abschnitt 2.8 zeigt diese Eigenschaft für unitäre Transformationen und Messungen. Die zuvor erwähnte Darstellung der No-Signaling Eigenschaft ist in Abschnitt 2.8 beschrieben. Das CHSH-Game stellt einen gängigen Test der No-Signaling Eigenschaft dar. Die Überlegungen dieses Tests sind in Abschnitt 2.9 beschrieben.

Tabelle 1.1 zeigt die in Kapitel 3 durchgeführten Experimente, diese werden in zwei Metriken, soweit möglich, analysiert. Zum einen wird der Anteil korrekter Lösungen ermittelt. Zum anderen wird die Verteilung der korrekten Lösungen, soweit durchführbar, durch den χ^2-Test ermittelt.

Für Schaltkreis Rechner ist dies der Anteil an Lösungen mit Wahrscheinlichkeit größer 0, bei Experimenten auf dem annealing Rechner, ist dies der Anteil an Lösungen die optimal hinsichtlich des Problems sind. Bei Experimenten auf dem Schaltkreis Rechner, wird die gemessene Verteilung mit der erwarteten verglichen. Für Experimente auf dem annealing Rechner, wird eine Gleichverteilung über den optimalen Lösungen erwartet, diese wird mit den Ergebnissen der Simulation und des annealing Rechners verglichen.

Diese Arbeit liefert drei sehr unterschiedliche Erkenntnisse:

1. Das Quantum Cube Model ist noch leistungsfähiger als bisher bekannt. So lässt sich die No-Signaling Eigenschaft von Quantensystemen graphisch mit dem Quantum Cube Model darstellen.

2. Zeitgleich mit dieser Arbeit zeigen Rybotycki u.a. 2025[7], dass Quanten Rechner die No-Signaling Eigenschaft in bestimmten Fällen brechen. In dieser Arbeit wird gezeigt, dass schon bei einfachen Schaltkreisen fehlerhafte Ergebnisse erzeugt werden, die nicht den Erwartungen entsprechen. Aussagen zu der No-Signaling Eigenschaft lassen sich so nicht zuverlässig treffen.

3. Diese Arbeit liefert erstmals ein Verfahren beliebige boolesche Funktionen als Optimierungsprobleme zu beschreiben. So lassen sich Optimierungsprobleme mit bestimmten Eigenschaften, wie einer großen Anzahl gleichwertiger Lösungen, erzeugen. Ein Optimierungsproblem auf Basis einer booleschen Funktion, die durch die Ungleichung des CHSH-Games inspiriert ist, zeigt in der Auswertung der Experimente ist eine Diskrepanz zwischen den Lösungen des annealing Rechners von D-Wave, und dem Dimod Simulator. So ist der Anteil der optimalen Lösungen auf dem annealing Rechner deutlich höher, als in der Simulation. Die gemessenen Verteilungen weichen jedoch schon bei kleinen Optimierungsproblemen ausschließlich auf dem annealing Rechner von der erwarteten Verteilung ab, wenn Funktionen mit mehren gleichwertigen Lösungen berechnet werden.

[7] Tomasz Rybotycki u. a. „Violation of No-Signaling on a Public Quantum Computer“. In: *Advanced Quantum Technologies* 8.6 (Juni 2025), S. 2400661. ISSN: 2511-9044, 2511-9044. DOI: 10.1002/qute.202400661. arXiv: 2409.11348 [quant-ph]. URL: http://arxiv.org/abs/2409.11348 (besucht am 07. 08. 2025).

2 Grundlagen

In den folgenden Abschnitten werden die mathematischen Grundlagen für die Experimente in Kapitel 3 beschrieben.

Abschnitt 2.1 bis 2.3 beschreiben übergreifende Eigenschaften des Quantencomputing, die sowohl im Schaltkreis Modell, als auch im annealing Modell Anwendung finden. In Abschnitt 2.4 wird eine Darstellung vorgestellt, die Systemzustände und Zustandsänderungen im Schaltkreis Modell darstellt.

Die spezifischen mathematischen Grundlagen des Schaltkreis Modells sind in Abschnitt 2.5 aufgestellt. Parallel dazu ist in Abschnitt 2.6 das Berechnungsmodell für annealing Rechner beschrieben.

Abschnitt 2.7 und Abschnitt 2.9 beschreiben, dass Änderungen in einem System keine Änderungen in einem anderen System auslösen können, sowie einen Test, der die Nicht-Lokalität von Quantensystemen überprüft.

2.1 Klassische Bits und Qubits

Bits sind die kleinste Dateneinheit eines Computers. Ein Bit hat entweder den Wert 0 oder 1.[1] Klassische Bits haben zwei Eigenschaften, die in der klassischen Informatik so offensichtlich sind, dass diese häufig nicht weiter betrachtet werden. Zum einen hat ein Bit zu jedem Zeitpunkt genau einen Wert, der unabhängig davon gilt, ob dieser Wert auch bekannt ist. Auch ändert sich der Wert eines Bits nicht, (nur) weil dieser ausgelesen wird. Diese Eigenschaft wird auch Realismus genannt.[2]

Die zweite Eigenschaft ist die Lokalität. Wird ein Bit geändert, so hat dies keine Auswirkungen auf die anderen Bits im System. Die Änderung an diesem Bit kann sich im System höchstens mit Lichtgeschwindigkeit ausbreiten, und Änderungen an anderen Bits auslösen.

Für Qubits (Quanten-Bits) wird in dieser Arbeit zwischen Zustand und Wert unterschieden. Ein Qubit hat einen Zustand der durch Gleichung 2.1 beschrieben werden kann. Wobei α_0 und α_1 komplexe Zahlen sind (Gleichung 2.3), welche die Gleichung 2.2 erfüllen.

$$|q\rangle = \alpha_0 |0\rangle + \alpha_1 |1\rangle \tag{2.1}$$

$$|\alpha_0|^2 + |\alpha_1|^2 = 1 \tag{2.2}$$

$$\alpha_j \in \mathbb{C}, \qquad j \in \{0, 1\} \tag{2.3}$$

In Gleichung 2.4 bis Gleichung 2.7 werden häufig gebrauchte Zustände aufgeführt. $|0\rangle$ und $|1\rangle$ sind dabei die Standardbasiszustände, auch Computational Basis genannt. $|+\rangle$ und $|-\rangle$ bilden eine weitere Orthonormalbasis, bekannt als Hadamardbasis.

[1] C. E. Shannon. „A Mathematical Theory of Communication“. In: *The Bell System Technical Journal* 27.3 (Juli 1948), S. 379–423. ISSN: 0005-8580. DOI: 10.1002/j.1538-7305.1948.tb01338.x. URL: https://ieeexplore.ieee.org/document/6773024/ (besucht am 16. 08. 2025).

[2] Bettina Just. *Quantencomputing kompakt: Spukhafte Fernwirkung und Teleportation endlich verständlich*. IT kompakt. Berlin, Heidelberg: Springer, 2020. ISBN: 978-3-662-61888-2 978-3-662-61889-9. DOI: 10.1007/978-3-662-61889-9. URL: https://link.springer.com/10.1007/978-3-662-61889-9 (besucht am 05. 06. 2025).

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \tag{2.4}$$

$$|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \tag{2.5}$$

$$|+\rangle = \frac{1}{\sqrt{2}} \cdot |0\rangle + \frac{1}{\sqrt{2}} \cdot |1\rangle = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \tag{2.6}$$

$$|-\rangle = \frac{1}{\sqrt{2}} \cdot |0\rangle - \frac{1}{\sqrt{2}} \cdot |1\rangle = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix} \tag{2.7}$$

In der Quantenmechanik wird die Bra-Ket-Notation (Gleichung 2.4 ff.) eingesetzt, um Zustände in unendlich-dimensionalen Hilberträumen zu modellieren. Im Rahmen dieser Arbeit genügt es, mit endlich dimensionalen Vektoren zu rechnen.

Die wichtigsten Operationen sind das Skalarprodukt, auch „inner product“ genannt. Die Ergebnisse sind eine Zahl, die oft als Wahrscheinlichkeit interpretiert wird. Und das Tensorprodukt, auch „outer product“ genannt, das Ergebnis hier ist ein skalierter Projektionsoperator (vergl. Abschnitt 2.2).

Wird ein Qubit objektiv gemessen, so ergibt sich ein Wert von $|0\rangle$ oder $|1\rangle$. Wobei das konkrete Ergebnis, je nach Zustand vor der Messung, mit unterschiedlichen Wahrscheinlichkeiten $|\alpha_0|^2$ für $|0\rangle$ und $|\alpha_1|^2$ für $|1\rangle$ auftritt. Bis zum Zeitpunkt der Messung ist nicht nur unbekannt, sondern auch unbestimmt, welcher Wert gemessen wird. Qubits verhalten sich also anders als klassische Bits, sie erfüllen die Realismus-Eigenschaft nicht.[3]

Nach der Messung in der Standardbasis befindet sich das Qubit nun nicht mehr im Ursprungszustand, sondern immer im Zustand $|0\rangle$ oder $|1\rangle$. Die Messung hat also eine Auswirkung auf den Zustand. In Abschnitt 2.2 ist genauer beschrieben, was eine Messung ist.

Betrachtet man ein System mit zwei Qubits, lässt sich der Zustand durch Gleichung 2.8 beschreiben. $|j\rangle$ beschreibt dabei den j-ten Einheitsvektor in 2^n, wobei n die Anzahl der Qubits im System ist.

Für ein System mit zwei Qubits, ergibt dies die Bedingung in Gleichung 2.9. Wird jetzt q_0 gemessen und liefert zum Beispiel einen Wert von $|0\rangle$, befindet sich das System danach im Zustand aus Gleichung 2.10. Wobei die Bedingung aus Gleichung 2.2 verallgemeinert gilt. Die Summe der Betragsquadrate der α_i muss immer gleich 1 sein.

$$|q_0 q_1\rangle = \alpha_0 |00\rangle + \alpha_1 |01\rangle + \alpha_2 |10\rangle + \alpha_3 |11\rangle \tag{2.8}$$

$$|\alpha_0|^2 + |\alpha_1|^2 + |\alpha_2|^2 + |\alpha_3|^2 = 1$$

$$\alpha_j \in \mathbb{C}, \quad j \in \{0,1\}^2 \tag{2.9}$$

$$|0 q_1\rangle = \alpha'_0 |00\rangle + \alpha'_1 |01\rangle \tag{2.10}$$

[3] Bettina Just. *Quantencomputing kompakt: Spukhafte Fernwirkung und Teleportation endlich verständlich*. IT kompakt. Berlin, Heidelberg: Springer, 2020. ISBN: 978-3-662-61888-2 978-3-662-61889-9. DOI: 10.1007/978-3-662-61889-9. URL: https://link.springer.com/10.1007/978-3-662-61889-9 (besucht am 05. 06. 2025).

Die Messung hat also Auswirkungen auf das gesamte System. Diese Änderung des Systems wirkt sich mit mehr als Lichtgeschwindigkeit aus,[4] und bricht die Lokalitätseigenschaft.

2.2 Messung eines Qubits

Die in diesem Abschnitt beschriebenen Formeln folgen, falls nicht abweichend gekennzeichnet, den Ausführungen aus Nielsen u.a. 2010.[5]

Befindet sich ein System mit einem Qubit im Zustand $|\varphi\rangle = \alpha|0\rangle + \beta|1\rangle$, so wird mit einer Wahrscheinlichkeit von $|\alpha|^2$ der Wert $|0\rangle$, und mit Wahrscheinlichkeit $|\beta|^2$ der Wert $|1\rangle$ gemessen.

Ein reiner Zustand (engl. Pure State) eines Systems ist ein Ket-Vektor $|\varphi\rangle$. Gemischte Zustände (engl. Mixed States) sind eine konvexe Kombination von reinen Zuständen (Gleichung 2.11).

$$p_1|\varphi_1\rangle + p_2|\varphi_2\rangle + \cdots + p_s|\varphi_s\rangle \tag{2.11}$$

Um zu bestimmen welchen Wert ein Qubit annimmt, wenn es gemessen wird, bietet sich die Betrachtung von Zuständen durch Dichtematrizen an.

Zu einem Zustand $|\varphi\rangle$ (Gleichung 2.12) ergibt sich die Dichtematrix ρ bezüglich der Standardbasis durch Gleichung 2.13. Wobei $\langle\varphi|$ der komplex-konjugierte Transponierte von $|\varphi\rangle$ ist.

$$|\varphi\rangle = \alpha|0\rangle + \beta|1\rangle \tag{2.12}$$

$$\rho = |\varphi\rangle\langle\varphi| = \begin{pmatrix}\alpha\\ \beta\end{pmatrix}\begin{pmatrix}\alpha^* & \beta^*\end{pmatrix} = \begin{pmatrix}|\alpha|^2 & \alpha\beta^*\\ \beta\alpha^* & |\beta|^2\end{pmatrix} \tag{2.13}$$

Dichtematrizen bieten den Vorteil, auch unbestimmte Zustände zu modellieren. So kann beschrieben werden, dass sich ein System entweder im Zustand $|0\rangle$ oder im Zustand $|+\rangle$ befindet. Zusätzlich lässt sich für alle möglichen Zustände festlegen, mit welchen klassischen Wahrscheinlichkeiten diese auftreten.

Gleichung 2.14 beschreibt den Zustand eines Qubit, dass sich mit Wahrscheinlichkeit $\frac{3}{4}$ im Zustand $|+\rangle$ und mit Wahrscheinlichkeit $\frac{1}{4}$ im Zustand $|0\rangle$ befindet.

$$\begin{aligned}\varrho &= \frac{3}{4}|+\rangle\langle+| + \frac{1}{4}|0\rangle\langle 0|\\ &= \frac{1}{8}\begin{pmatrix}5 & 3\\ 3 & 3\end{pmatrix}\end{aligned} \tag{2.14}$$

Ein solcher Zustand, der nur durch eine Dichtematrix beschrieben werden kann, nicht aber durch das Produkt eines Ket-Vektors mit seinem entsprechenden Bra-Vektor, heißt Mixed State.

[4] Juan Yin u. a. „Bounding the Speed of 'spooky Action at a Distance'". In: *Physical Review Letters* 110.26 (26. Juni 2013), S. 260407. ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.110.260407. arXiv: 1303.0614 [quant-ph]. URL: http://arxiv.org/abs/1303.0614 (besucht am 16. 08. 2025).

[5] Michael A. Nielsen und Isaac L. Chuang. *Quantum Computation and Quantum Information*. 10th anniversary ed. Cambridge ; New York: Cambridge University Press, 2010. 676 S. ISBN: 978-1-107-00217-3.

Um zu ermitteln mit welcher Wahrscheinlichkeit ein bestimmter Wert gemessen wird, wird eine Projektionsmatrix erstellt. Gleichung 2.15 zeigt die Projektionsmatrix für den Wert $|0\rangle$. Die Wahrscheinlichkeit den entsprechenden Zustand zu messen, ist durch das in Gleichung 2.16 beschriebene Produkt gegeben.

$$M_0 = |0\rangle \langle 0| \tag{2.15}$$

$$p_0 = \langle \varphi | M_0^\dagger M_0 | \varphi \rangle \tag{2.16}$$

$$|\varphi'\rangle = \frac{M_0 |\varphi\rangle}{\sqrt{p_0}} \tag{2.17}$$

Nach der Messung befindet sich das System im Zustand $|\varphi'\rangle$, der in Gleichung 2.17 beschrieben ist.

Äquivalent lässt sich der Zustand nach einer Messung durch Dichtematrizen beschreiben. Gleichung 2.18 zeigt, wie die Dichtematrix nach einer Messung berechnet wird.

$$\begin{aligned} \rho' &= \frac{M_i |\varphi\rangle \langle \varphi| M_i^\dagger}{p_0} \\ &= \frac{M_i \rho M_i^\dagger}{p_0} \end{aligned} \tag{2.18}$$

Dieses Konzept lässt sich auf Systeme mit zwei und mehr Qubits erweitern. Zur Beschreibung solcher Systeme wird die in Gleichung 2.19 festgelegte Notation für reine Zustände verwendet. Aus dieser Beschreibung ergibt sich eine Reihenfolge für die Qubits, die im Folgenden weiter verwendet wird.

$$|q_0 q_1 \dots q_n\rangle = |q_0\rangle \otimes |q_1\rangle \otimes \dots \otimes |q_n\rangle \tag{2.19}$$

Für ein System mit n Qubits ergeben sich 2^n mögliche Messergebnisse. Diese treten mit Wahrscheinlichkeit $|\alpha_i|^2$ auf, wobei α_i die Amplitude des jeweiligen Basisvektors ist.

Die Projektionsmatrizen haben dann die Form $2^n \times 2^n$. Eine Projektionsmatrix sieht dann wie in Gleichung 2.20 beschrieben aus. Für reine Zustände ist $p_{i_1 \dots i_n} = \alpha_{i_1 \dots i_n}$. Bei Mixed States ergibt sich diese aus der Linearkombination der entsprechenden reinen Zustände und der passenden Amplitude.

$$M_{i_1 \dots i_n} = |i_1 \dots i_n\rangle \langle i_1 \dots i_n| \tag{2.20}$$

Der Zustand nach einer Messung ist dann durch Gleichung 2.21 als Dichtematrix gegeben.

$$\rho' = \frac{M_{i_1 \dots i_n} \rho M_{i_1 \dots i_n}^\dagger}{p_{i_1 \dots i_n}} \tag{2.21}$$

In Systemen mit mehreren Qubits lassen sich auch einzelne Qubits messen, ohne alle Qubits messen zu müssen. Im mathematischen Modell ist das Vorgehen äquivalent zur Messung aller Qubits. Lediglich die Projektionsmatrix ist anders aufgebaut. Gleichung 2.22 zeigt, wie eine Projektionsmatrix aufgebaut ist, die

lediglich das erste Qubit auf den Wert $|0\rangle$ misst und die anderen Qubits im System nicht betrachtet.

$$M_{0_} = |0\rangle\langle 0| \otimes \mathbf{I}_{2^{n-1}} \tag{2.22}$$

Die Wahrscheinlichkeit das erste Qubit als $|0\rangle$ zu messen, ist dann durch die Summe der Beträge der Amplituden gegeben, an denen das erste Qubit $|0\rangle$ ist.

Allgemein wird für die Projektionsmatrix die Reihenfolge der Qubits im System betrachtet. Für jedes Qubit wird entweder die Dichtematrix eines Basiszustands ausgewählt, falls dieses Qubit gemessen werden soll, oder die Identitätsmatrix $\mathbf{I}_2$ ausgewählt, falls dieses Qubit nicht gemessen werden soll. Die gewählten Matrizen werden dann in der Reihenfolge der Qubits mit dem Tensorprodukt multipliziert. Die Wahrscheinlichkeit die ausgewählten Qubits in dem jeweiligen Zustand zu messen, ist wie schon bei einzelnen Qubits, durch das Skalarprodukt gegeben.

2.3 Schaltkreis Modell und Annealing

Im Quantencomputing wird zwischen zwei verschiedenen Modellen unterschieden, die sich nicht nur auf technologischer Ebene unterscheiden, sondern die sich auch darin unterscheiden, welche mathematischen Probleme damit berechnet werden können.

Auf der einen Seite steht das Schaltkreis Modell. Quantencomputer die auf diesem Modell aufbauen, haben einen Systemzustand der zu jedem Berechnungsschritt bekannt ist.

Es gibt verschiedene Hersteller solcher Quantencomputer, welche wiederum verschiedene Technologien verwenden. Zu den größten Anbietern dieser Quantencomputer gehört IBM. Es gibt verschiedene Technologien, die eingesetzt werden, um Quantencomputer zu konstruieren. Die Quantencomputer von IBM verwenden 1121 supraleitende Qubits.[6] Andere Quantencomputer verwenden einzelne Photonen oder eingefangene Ionen als Qubits.

In Abschnitt 2.4 wird ein Modell beschrieben, welches diese Systemzustände für Systeme mit wenigen Qubits visuell darstellt. Auf Schaltkreis Rechnern ist es zudem möglich, logische Gatter des klassischen Computing zu simulieren und mit Quantengattern nachzubauen.[7] Die dafür notwendigen Gatter sind in Abschnitt 2.5 beschrieben.

Das zweite Modell ist das „adiabatische quanten annealing“. Quantencomputer, die auf diesem Modell aufbauen, können wie eine Blackbox behandelt werden. Als Qubits werden hier supraleitende Drahtschleifen verwendet. Kühlt man diese nahe dem absoluten Nullpunkt ab, fließt ein Strom im oder gegen den Uhrzeigersinn. Die Richtung des Stromfluss wird erst bei der Messung determiniert, so wie auch bei den Qubits des Schaltkreismodells.[8]

[6] *IBM Quantum System Two: The Era of Quantum Utility Is Here | IBM Quantum Computing Blog*. URL: https://www.ibm.com/quantum/blog/quantum-roadmap-2033 (besucht am 16. 08. 2025).

[7] *Einführung in Das Quantencomputing - Teil 3*. openHPI. URL: https://open.hpi.de/courses/qc-intro-3-2023 (besucht am 16. 08. 2025).

[8] *The Advantage™ Quantum Computer | D-Wave*. URL: https://www.dwavequantum.com/solutions-and-products/systems/ (besucht am 16. 08. 2025).

Neben den Drahtschleifen gibt es noch Gewichte (engl. weights), die in Form von Magneten in die Nähe der Drahtschleifen gebracht werden können, um eine Bias für die Richtung festzulegen. Zusätzlich gibt es noch so genannte couplers, welche dafür sorgen, dass der Strom in zwei benachbarten Drahtschleifen nach Möglichkeit in die gleiche, bzw. entgegengesetzte Richtung fließt.

Bei einer Berechnung mit einem annealing Rechner. Wird das System durch weights und couplers zunächst vorbereitet und das System erhitzt. Zur Berechnung wird das System im zweiten Schritt auf eine Temperatur unter 20 mK abgekühlt und die Ströme in den Drahtschleifen gemessen.[9] Bei so geringen Temperaturen befindet sich das System in einem Zustand minimaler Energie. Durch geschicktes Anwenden der weights und couplers kann so ein Optimierungsproblem gelöst werden. In dieser Arbeit werden annealing Rechner von D-Wave verwendet, um solche Optimierungsproblem zu lösen.

Im Gegensatz zu Quantencomputern des Schaltkreismodells ist bei annealing Rechnern nicht bekannt, in welchen Zuständen sich das System befindet bis zum Zeitpunkt des Messens.

Annealing Rechner können lediglich Optimierungsprobleme lösen. Wie in Abschnitt 2.6 gezeigt wird, ist es jedoch möglich, viele Probleme als Optimierungsprobleme zu formulieren.

Die Art wie Probleme für die verschiedenen Modelle beschrieben werden können, ist in Abschnitt 2.5 für Schaltkreis Rechner und in Abschnitt 2.6 für annealing Rechner beschrieben.

2.4 Quantum Cube Model

Im Folgenden wird die Darstellung des Quantum Cube Models erläutert, die es ermöglicht Pure States zu visualisieren. Die Darstellung wurde erstmals in Just 2020[10] beschrieben.

Die Amplituden werden als Quadrate mit der Kantenlänge $|\alpha|$ in der unteren linken Ecke eines Einheitsquadrates gezeichnet. Zusätzlich wird das Einheitsquadrat um seinen Mittelpunkt entsprechend der Phase von α gedreht. So kann jeder Koeffizient α für sich dargestellt werden. Um den Gesamtzustand des Systems darzustellen, wird nun in den Ecken eines n-dimensionalen Würfels jeweils einer der Koeffizienten dargestellt, wobei n die Anzahl der Qubits im System ist. Für die Auswahl der korrekten Ecke für einen Koeffizienten gilt, jedem Qubit wird eine Raumdimension zugewiesen.

Abbildung 2.1 zeigt wie der Zustand $\sqrt{\frac{3}{4}}\,|0\rangle + \sqrt{\frac{1}{4}}e^{i\pi 1/3}\,|1\rangle$ dargestellt werden kann. Zusätzlich zu Zuständen bietet das Quantum Cube Model die Möglichkeit, auch Übergänge zwischen zwei Zuständen zu beschreiben. Abbildung 2.2 zeigt die Wirkung von Hadamard auf den Zustand $|1\rangle$.

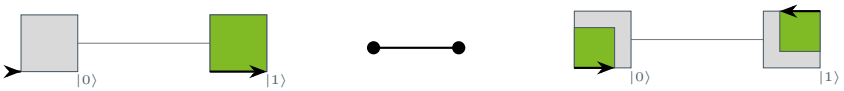
Für Systeme mit zwei Qubits ergeben sich vier Koeffizienten. Diese werden wie

[9] *How Cold Is It?* D-Wave Quantum Inc. 30. Apr. 2018. URL: https://support.dwavesys.com/hc/en-us/articles/360003681034-How-Cold-Is-It (besucht am 16. 08. 2025).

[10] Bettina Just. *Quantencomputing kompakt: Spukhafte Fernwirkung und Teleportation endlich verständlich*. IT kompakt. Berlin, Heidelberg: Springer, 2020. ISBN: 978-3-662-61888-2 978-3-662-61889-9. DOI: 10.1007/978-3-662-61889-9. URL: https://link.springer.com/10.1007/978-3-662-61889-9 (besucht am 05. 06. 2025).



Abbildung 2.1: Quantum Cube Model für den Zustand $\sqrt{\frac{3}{4}}\,|0\rangle + \sqrt{\frac{1}{4}}e^{i\pi 1/3}\,|1\rangle$



(a) Initialzustand $|1\rangle$ (b) Übergang (c) Endzustand $\sqrt{\frac{1}{2}}(|0\rangle - |1\rangle)$

Abbildung 2.2: Wirkung von Hadamard auf den Zustand $|1\rangle$

in Abbildung 2.3 dargestellt. Das most significant Qubit wird entlang der rechts-links-Achse, das andere Qubit entlang der oben-unten-Achse dargestellt.

$$|q_1 q_2\rangle = \sqrt{\frac{1}{2}}\,|00\rangle + \sqrt{\frac{1}{4}}e^{\pi/2}\,|01\rangle - \sqrt{\frac{1}{4}}\,|10\rangle \tag{2.23}$$

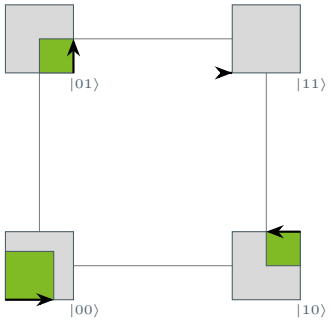


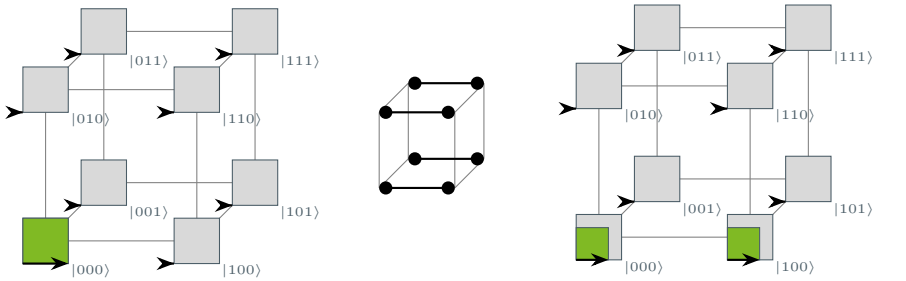
Abbildung 2.3: Quantum Cube Model für den Zustand $|q_1 q_2\rangle$ aus Gleichung 2.23

Abbildung 2.4 zeigt die Wirkung von Hadamard auf ein System mit drei Qubits. Am Übergang ist zu erkennen, dass entlang der Achse des ersten Qubits die Amplituden „verteilt“ werden.

Für die verschiedenen Gatter werden unterschiedliche Pfeile im Übergangsdiagramm verwendet. Diese werden im Abschnitt 2.5 genauer beschrieben.

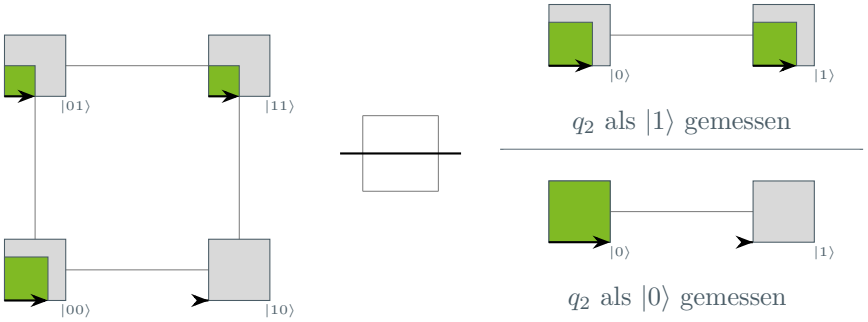
Messungen von Qubits können im Quantum Cube Model wie in Abbildung 2.5 dargestellt werden, wobei durch eine Messung im Allgemeinen zwei mögliche Zustände entstehen, solange es für beide Messergebnisse eine Wahrscheinlichkeit größer 0 gibt.

Wie in der Abbildung zu erkennen ist werden die Amplituden in den beiden möglichen Zuständen getrennt von einander normiert.



(a) Initialzustand $|000\rangle$ (b) Übergang (c) Endzustand $\sqrt{\frac{1}{2}}(|000\rangle + |100\rangle)$

Abbildung 2.4: Wirkung von Hadamard auf den Zustand $|000\rangle$



(a) Zustand vor der Messung (b) Messung (c) Mögliche Zustände nach der Messung

Abbildung 2.5: Messung des zweiten von zwei Qubits

Die Quantum Learning App ist eine Anwendung, die einige gängige Schaltkreise simuliert und die Ergebnisse mit dem Quantum Cube Model darstellt.[11]

[11] Cedric Schacht. *CedricSchacht/Quantum-Learing-App*. Version 1.0.0+1. 20. Juni 2024. URL: https://github.com/CedricSchacht/quantum-learing-app (besucht am 05. 06. 2025).

2.5 Matrix eines Schaltkreises

Die in diesem Abschnitt beschriebenen Formeln folgen, falls nicht abweichend gekennzeichnet, den Ausführungen aus Nielsen u.a. 2010.[12]

Schaltkreise bieten eine kompakte Notation, um Algorithmen für Quantencomputer zu beschreiben. Im Folgenden wird sowohl die Notation als Schaltkreisdiagramm, als auch die mathematische Grundlage für die Analyse dieser Schaltkreise beleuchtet.

Für Systeme mit einem Qubit wird eine horizontale Linie verwendet, um die Entwicklung des Zustands über die Zeit des Algorithmus zu beschreiben. Abbildung 2.6 zeigt, wie ein solcher Schaltkreis aussieht.



Abbildung 2.6: Schaltkreis auf einem Qubit

Jeder der Kästen auf der Linie entspricht einem Gatter, welches auf das Qubit angewendet wird. Dabei ist die Reihenfolge von links nach rechts relevant. Um den Systemzustand zu einem bestimmten Zeitschritt zu bestimmen, müssen die entsprechenden unitären Matrizen der Gatter von links an den vorhergehenden Zustand multipliziert werden.

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \tag{2.24}$$

$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \tag{2.25}$$

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \tag{2.26}$$

$$|\varphi'\rangle = ZH\,|\varphi\rangle \tag{2.27}$$

$$|\varphi''\rangle = X\,|\varphi'\rangle \tag{2.28}$$

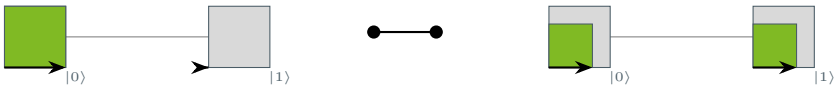
Gleichung 2.27 zeigt den Zustand des Systems nach den ersten zwei Transformationen, wobei in Gleichung 2.24 und 2.25 die Matrizen der Gatter beschrieben sind. Gleichung 2.28 zeigt den Zustand nach allen drei Transformationen. In Appendix A sind die Matrizen der gängigsten Gatter, sowie aller Gatter die für diese Arbeit relevant sind, aufgeführt.

In Abbildung 2.7 bis 2.9 sind die Zustände des Qubits im Verlauf des Schaltkreises dargestellt, ausgehend von dem Initialzustand $|\varphi\rangle = |0\rangle$.

Aufgrund des Assoziativgesetzes für die Multiplikation mit Matrizen lässt sich so eine Matrix aufstellen, welche die Wirkung des Schaltkreises auf einen Ausgangszustand beschreibt.

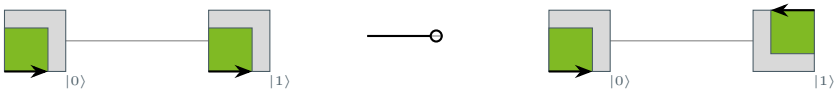
Ein Schaltkreis mit n Qubits hat einen Initialzustand mit 2^n Koeffizienten. Entsprechend muss die Matrix eines Zeitschritts auch die Form $2^n \times 2^n$ haben. Dies

[12] Michael A. Nielsen und Isaac L. Chuang. *Quantum Computation and Quantum Information*. 10th anniversary ed. Cambridge ; New York: Cambridge University Press, 2010. 676 S. ISBN: 978-1-107-00217-3.



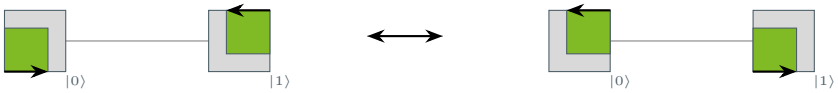
(a) Initialzustand $|0\rangle$ (b) Übergang (c) Zustand nach H

Abbildung 2.7: Wirkung von Hadamard auf den Zustand $|0\rangle$



(a) Zustand nach H (b) Übergang (c) Zustand nach Z

Abbildung 2.8: Wirkung von PauliZ auf den Zustand $\frac{1}{\sqrt{2}}|0\rangle + |1\rangle$



(a) Zustand nach Z (b) Übergang (c) Zustand nach X

Abbildung 2.9: Wirkung von PauliX auf den Zustand $\frac{1}{\sqrt{2}}|0\rangle + |1\rangle$

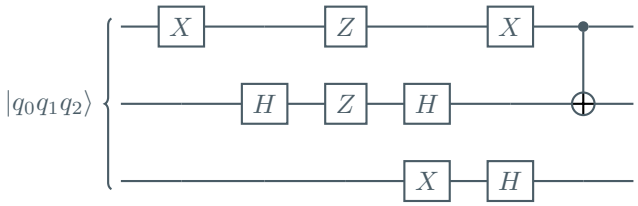


Abbildung 2.10: Schaltkreis mit drei Qubits

wird erreicht, indem die Matrizen der Gatter mit Identitätsmatrizen vertensort werden. In Abbildung 2.10 ist ein solcher Schaltkreis mit drei Qubits zu sehen. Um die Wirkung von PauliX auf dem ersten Qubit zu bestimmen, wird in Abbildung 2.11 der erste Zeitschritt betrachtet.

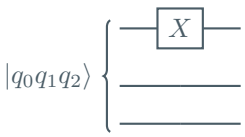


Abbildung 2.11: Erster Zeitschritt aus Abbildung 2.10

Wie in Gleichung 2.29 zu sehen ist, wird für alle Qubits, auf denen kein Gatter liegt, die Identitätsmatrix verwendet. Für alle Qubits, auf die ein Gatter ange-

wendet wird, wird die unitäre Matrix des Gatters verwendet. Die Reihenfolge der Matrizen ist entsprechend der Qubits im Schaltkreis von oben nach unten abzulesen.

$$U_0 = X \otimes \mathbf{I}_2 \otimes \mathbf{I}_2 \tag{2.29}$$
$$U_1 = \mathbf{I}_2 \otimes H \otimes \mathbf{I}_2 \tag{2.30}$$
$$U_2 = Z \otimes Z \otimes \mathbf{I}_2 \tag{2.31}$$
$$U_3 = \mathbf{I}_2 \otimes H \otimes X \tag{2.32}$$
$$U_4 = X \otimes \mathbf{I}_2 \otimes H \tag{2.33}$$
$$U_5 = CNOT_{0,1} \otimes \mathbf{I}_2 \tag{2.34}$$
$$U = U_5 U_4 U_3 U_2 U_1 U_0 \tag{2.35}$$

Gleichung 2.30 bis 2.34 zeigen die Matrizen für die anderen Zeitschritte, und Gleichung 2.35 zeigt die Matrix für den gesamten Schaltkreis. Der Endzustand nach Anwendung des Schaltkreises auf $|q_0 q_1 q_2\rangle$ ist gegeben durch $U |q_0 q_1 q_2\rangle$.

$$U |q_0 q_1 q_2\rangle \left(U |q_0 q_1 q_2\rangle\right)^\dagger = U |q_0 q_1 q_2\rangle \left(\langle q_0 q_1 q_2| U^\dagger\right) \tag{2.36}$$
$$= U(|q_0 q_1 q_2\rangle \langle q_0 q_1 q_2|)U^\dagger \tag{2.37}$$
$$= U \rho U^\dagger \tag{2.38}$$

Gleichung 2.36 bis 2.38 zeigen, dass die Matrix eines Schaltkreises auf einen Zustand angewendet werden kann, der als Dichtematrix (engl. Density Matrix) beschrieben ist.

2.6 Quadratic Unconstrained Binary Optimization

Berechnungsprobleme lassen sich in verschiedene Klassen unterteilen. Die Klasse der schwersten Berechnungsprobleme wird NP-schwer (engl. NP-hard) genannt. Für NP-schwere Berechnungsprobleme ist kein klassischer Algorithmus bekannt, der in Polynomialzeit eine Lösung für das Problem findet.[13]

Eines dieser NP-schweren Probleme ist das QUBO (Quadratic Unconstrained Binary Optimization). In dieser Arbeit werden nur Minimierungsprobleme als QUBO betrachtet. Eine Überführung eines Minimierungsproblems in ein Maximierungsproblem ist jedoch einfach möglich.

Gleichung 2.39 zeigt ein Optimierungsproblem, wobei x der Vektor der binären Entscheidungsvariablen und Q eine quadratische Matrix mit ausschließlich konstanten Einträgen ist.[14] Es ist üblich Q als symmetrische Matrix oder als Matrix in oberer Dreieckform aufzustellen. Die beiden üblichen Formen der Matrix lassen sich in die jeweils andere überführen. Das genaue Verfahren ist hier nicht weiter interessant.

[13] Michael Sipser. *Introduction to the Theory of Computation*. preliminary ed. The PWS Series in Computer Science. Boston, Mass. Bonn: PWS Publ, 1996. 239 S. ISBN: 978-0-534-95250-1, Seite 210 ff.

[14] vergl. Fred Glover, Gary Kochenberger und Yu Du. „Quantum Bridge Analytics I: A Tutorial on Formulating and Using QUBO Models“. In: (4. Nov. 2019). DOI: 10.48550/arXiv.1811.11538. URL: https://arxiv.org/abs/1811.11538 (besucht am 30. 08. 2025).

$$\text{QUBO: } \min y = x^\dagger Q x \tag{2.39}$$

$$b = b^2, \quad b \in \mathbb{B} \tag{2.40}$$

Die Matrix Q lässt sich als Polynom beschreiben. Wobei $b_i b_j c$ beschreibt, dass das Element $Q_{i,j} = c$ ist.

Aufgrund der Gleichung 2.40 werden die Terme, die nur von einer QUBO-Variablen abhängen, Linearterme genannt. Terme ohne Variable können ignoriert werden, diese ändern die optimale Belegung der Variablen nicht.

Im Folgenden wird gezeigt, wie eine boolesche Funktion f als QUBO beschrieben werden kann. Es handelt sich dabei jedoch nicht um eine Polynomialzeitreduktion, da der Suchraum für die optimale Lösung von 2^n auf 2^{2^n} erweitert wird. Eine Lösung für das Optimierungsproblem soll dabei die folgenden beiden Eigenschaften erfüllen:

1. Es soll genau eine Zeile der Wertetabelle der Funktion f ausgewählt werden.
2. Die ausgewählte Zeile soll den Funktionswert 1 (T) haben.

Gleichung 2.41 setzt die beiden Eigenschaften um. Für jede Zeile in der Wertetabelle wird eine Variable z_i eingeführt. Die erste Summe ist genau dann 0, wenn genau eine Zeile der Wertetabelle ausgewählt wird, also genau ein z_i gleich 1 ist. Der Faktor 2^{n-1} wird voran gestellt, um sicher zu stellen, dass die zweite Eigenschaft nur dann zur Lösung beiträgt, wenn die erste Eigenschaft erfüllt ist.

Da durch die erste Bedingung nur ein $z_i \neq 0$ ist, vereinfacht sich die zweite Summe zu $f(i)z_i$ für genau das i, an dem $z_i \neq 0$. Also -1 genau dann, wenn ein i ausgewählt wird, für das $f(i) = 1$ ist.

$$Q = 2^{n-1} \left(\sum_{i=0}^{2^n-1} z_i - 1 \right)^2 - \sum_{i=0}^{2^n-1} f(i) z_i \tag{2.41}$$

In Gleichung 2.42 ist die entsprechende QUBO-Matrix aufgestellt, die sich aus Gleichung 2.41 ergibt. Diese Matrix ist bereits in die obere Dreieckform überführt.

$$Q = \begin{pmatrix} -f(0) & 2^n & & \dots & 2^n \\ 0 & -f(1) & \ddots & & \vdots \\ \vdots & \ddots & \ddots & & \\ & & & & 2^n \\ 0 & \dots & & 0 & -f(2^n-1) \end{pmatrix} \tag{2.42}$$

Die Matrix in Gleichung 2.42 lässt sich in Python mit dem Code aus Listing 2.1 anlegen. `BinaryQuadraticModel` ist ein generalisierter Datentyp der `dimod` Bibliothek, von D-Wave und wird verwendet um das QUBO zu implementieren.

```python
from typing import Callable
import itertools
from dimod import BinaryQuadraticModel
from dimod import BINARY
def f_to_bqm(f: Callable[[int], bool], bits: int) ->
    BinaryQuadraticModel:

    zs = [i for i in range(2**bits)]
    linear = dict([
        (f"z_{z}", -1 if f(z) else 0) for z in zs
    ])
    quadratic = dict([
        ((f"z_{z_i}", f"z_{z_j}"), 2**bits)
        for z_i, z_j
        in itertools.product(zs, zs)
        if z_i > z_j
    ])
    bqm = BinaryQuadraticModel(
        linear,
        quadratic,
        0.0,
        BINARY
    )
    bqm.normalize()
    return bqm
```

Listing 2.1: Übersetzen einer booleschen Funktion in ein BinaryQuadraticModel in Python

2.7 No-Signaling Theorem

Die in diesem Abschnitt beschriebenen Formeln folgen, falls nicht abweichend gekennzeichnet, den Ausführungen aus Nielsen u.a. 2010.[15]

Ein reiner Zustand $|\varphi\rangle$ lässt sich als Linearkombination der Basiszustände der Teilsysteme A und B darstellen. Gleichung 2.43 verdeutlicht diese Zerlegung in Komponenten noch einmal, wobei die Indizes A und B angeben, dass es sich um einen Vektor der Größe des Teilsystems A bzw. B, bzw. eine Matrix der jeweils passenden Größe des Teilsystems handelt.

$$|\varphi\rangle = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \alpha_{ij} |i\rangle_A |j\rangle_B \tag{2.43}$$

Gleichung 2.44 zeigt, dass diese Zerlegung sich auch auf Mixed States übertragen lässt. Um die folgenden Formeln nicht unnötig zu überladen, werden die Ausführungen zum No-Signaling Theorem nun mit reinen Zuständen demonstriert.

$$\sum_n p_n \cdot |\varphi\rangle = \sum_n p_n \cdot \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \alpha_{ij} |i\rangle_A |j\rangle_B \tag{2.44}$$

Im Folgenden wird ein System im Zustand (gegeben durch seine Dichtematrix) ρ_{AB} betrachtet. Die partielle Spur (engl. partial trace) $\mathrm{Tr}_A(\rho_{AB})$ ist dann wie in Gleichung 2.45 definiert. Interpretiert werden kann die partielle Spur über A als eine Modellierung des Teilsystems B (Gleichung 2.46).

$$\begin{aligned} \mathrm{Tr}_A(\rho_{AB}) &= \sum_{ijl} \alpha_{ij}\alpha_{il}^* |j\rangle_B \langle l|_B \\ &= \sum_{jl} (\sum_i \alpha_{ij}\alpha_{il}^*) |j\rangle_B \langle l|_B \end{aligned} \tag{2.45}$$

$$\rho_B = \mathrm{Tr}_A(\rho_{AB}) \tag{2.46}$$

Das No-Signaling Theorem besagt, dass lokale Änderungen durch unitäre Transformationen oder Messungen in einem Teilsystem keine beobachtbaren Zustandsänderungen im restlichen System auslösen können.

Gleichung 2.47 zeigt, wie sich eine unitäre Transformation auf Teilsystem A in der Linearkombination auswirkt. Äquivalent zeigt Gleichung 2.48 den entsprechenden Bra-Vektor.

[15] Michael A. Nielsen und Isaac L. Chuang. *Quantum Computation and Quantum Information*. 10th anniversary ed. Cambridge ; New York: Cambridge University Press, 2010. 676 S. ISBN: 978-1-107-00217-3.

$$
\begin{aligned}
U_A \otimes \mathbf{I}_B \left|\varphi\right\rangle &= \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} \alpha_{ij} U_A \left|i\right\rangle_A \mathbf{I}_B \left|j\right\rangle_B \\
&= \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} \alpha_{ij} U_A \left|i\right\rangle_A \left|j\right\rangle_B \qquad (2.47) \\
\left\langle\varphi\right| U_A^\dagger \otimes \mathbf{I}_B &= \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} \alpha_{ij}^* \left\langle i\right|_A U_A^\dagger \left\langle j\right|_B \mathbf{I}_B \\
&= \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} \alpha_{ij}^* \left\langle i\right|_A U_A^\dagger \left\langle j\right|_B \qquad (2.48)
\end{aligned}
$$

Der Zustand des Teilsystems B ist dann, wie in Gleichung 2.46 beschrieben, durch die partielle Spur gegeben.

Zu zeigen ist, dass die partielle Spur über A den gleichen Zustand für das Teilsystem B liefert, unabhängig davon ob eine unitäre Transformation U_A auf dem Teilsystem A angewendet wird (Gleichung 2.49).

$$
\begin{aligned}
\rho_B &= \mathrm{Tr}_A(U_A \otimes \mathbf{I}_B \left|\varphi\right\rangle \left\langle\varphi\right| U_A^\dagger \otimes \mathbf{I}_B) \stackrel{?}{=} \mathrm{Tr}_A(\left|\varphi\right\rangle \left\langle\varphi\right|) \qquad (2.49) \\
\rho_B &= \mathrm{Tr}_A\left(\sum_{ijkl} \alpha_{ij} U_A \left|i\right\rangle_A \left|j\right\rangle_B \alpha_{kl}^* \left\langle k\right|_A U_A^\dagger \left\langle l\right|_B\right) \\
&= \sum_{ijkl} \mathrm{Tr}_A(\alpha_{ij} U_A \left|i\right\rangle_A \left|j\right\rangle_B \alpha_{kl}^* \left\langle k\right|_A U_A^\dagger \left\langle l\right|_B) \\
&= \sum_{ijkl} \alpha_{ij}\alpha_{kl}^* \,\mathrm{Tr}_A(U_A \left|i\right\rangle_A \left|j\right\rangle_B \left\langle k\right|_A U_A^\dagger \left\langle l\right|_B) \qquad (2.50)
\end{aligned}
$$

Aufgrund der Linearität können in Gleichung 2.50 die Summe und die Koeffizienten aus der Spur vorgezogen werden. Gleichung 2.51 zeigt, dass die unitäre Transformation für jede Kombination von Basiszuständen aus A und B wegfällt.

$$
\begin{aligned}
\mathrm{Tr}_A(U_A \left|i\right\rangle_A \left|j\right\rangle_B \left\langle k\right|_A U_A^\dagger \left\langle l\right|_B) &= \left|j\right\rangle_B \left\langle l\right|_B \left\langle i\right|_A U_A^\dagger U_A \left|k\right\rangle_A \\
&= \left|j\right\rangle_B \left\langle l\right|_B \left\langle i\right|_A \mathbf{I}_A \left|k\right\rangle_A \\
&= \left|j\right\rangle_B \left\langle l\right|_B \left\langle i\right|_A \left|k\right\rangle_A \\
&= \left|j\right\rangle_B \left\langle l\right|_B \delta_{ik} \qquad (2.51)
\end{aligned}
$$

Gleichung 2.51 kann in Gleichung 2.50 eingesetzt werden. Das Kronecker-Delta lässt sich kürzen, indem $i = k$ gesetzt wird (Gleichung 2.52). Mit Gleichung 2.53 ist somit gezeigt, dass unitäre Transformationen in Teilsystem A keine Auswirkungen auf das Teilsystem B haben.

$$
\begin{aligned}
\mathrm{Tr}_A(U_A \otimes \mathbf{I}_B \left|\varphi\right\rangle \left\langle\varphi\right| U_A^\dagger \otimes \mathbf{I}_B) &= \sum_{ijkl} \alpha_{ij}\alpha_{kl}^* \delta_{ik} \left|j\right\rangle_B \left\langle l\right|_B \\
&= \sum_{ijl} \alpha_{ij}\alpha_{il}^* \left|j\right\rangle_B \left\langle l\right|_B \qquad (2.52) \\
&= \sum_{jl} (\sum_{i} \alpha_{ij}\alpha_{il}^*) \left|j\right\rangle_B \left\langle l\right|_B \\
&= \mathrm{Tr}_A(\left|\varphi\right\rangle \left\langle\varphi\right|) \qquad (2.53)
\end{aligned}
$$

Zusätzlich wird im Folgenden gezeigt, dass auch eine Messung auf Teilsystem A keine Auswirkungen auf den Zustand des Teilsystems B haben kann.

M_i sei ein vollständiger Satz von Messoperatoren auf dem Teilsystem A (Gleichung 2.54). Aufgrund der Vollständigkeit ist die Summe aller Operatoren gleich der Identität auf dem gesamten System (Gleichung 2.56).

$$M_i = |i\rangle_A \langle i|_A \otimes \mathbf{I}_B \tag{2.54}$$

$$M_i^\dagger = |i\rangle_A \langle i|_A \otimes \mathbf{I}_B \tag{2.55}$$

$$\sum_i M_i^\dagger M_i = \mathbf{I} \tag{2.56}$$

Wie in Abschnitt 2.2 beschrieben, ist der Zustand nach einer Messung gegeben durch Gleichung 2.58, wobei p_i (Gleichung 2.57) die Wahrscheinlichkeit für das jeweilige Messergebnis beschreibt.

$$p_i = \langle\varphi| M_i^\dagger M_i |\varphi\rangle \tag{2.57}$$

$$|\varphi'\rangle = \frac{M_i |\varphi\rangle}{\sqrt{p_i}} \tag{2.58}$$

Wird jetzt ein Teilsystem gemessen und das Ergebnis ist nicht bekannt, lässt sich das System danach durch die Summe aller möglichen Zustände, gewichtet durch die Auftrittswahrscheinlichkeit, beschreiben. ρ' (Gleichung 2.59) beschreibt einen solchen Zustand.

$$\begin{aligned}\rho' &= \sum_i p_i \frac{M_i |\varphi\rangle}{\sqrt{p_i}} \frac{\langle\varphi| M_i^\dagger}{\sqrt{p_i}} \\ &= \sum_i M_i |\varphi\rangle \langle\varphi| M_i^\dagger\end{aligned} \tag{2.59}$$

Das Teilsystem B wird dann, wie in Gleichung 2.46 durch die partielle Spur, wie in Gleichung 2.60 beschrieben.

$$\begin{aligned}\rho'_B &= \mathrm{Tr}_A(\rho') \\ &= \mathrm{Tr}_A\left(\sum_i M_i |\varphi\rangle \langle\varphi| M_i^\dagger\right)\end{aligned} \tag{2.60}$$

In Gleichung 2.61 bis 2.62 wird die Linearität, sowie die zyklische Invarianz der Spur verwendet. Da der wie in Gleichung 2.56 gewählte Satz an Messoperatoren vollständig ist, lässt sich dieser wie in Gleichung 2.63 durch die Identität $\mathbf{I}$ ersetzen.

$$\mathrm{Tr}_A\left(\sum_i M_i |\varphi\rangle \langle\varphi| M_i^\dagger\right) = \sum_i \mathrm{Tr}_A(M_i |\varphi\rangle \langle\varphi| M_i^\dagger) \tag{2.61}$$

$$= \sum_i \mathrm{Tr}_A(M_i^\dagger M_i |\varphi\rangle \langle\varphi|)$$

$$= \mathrm{Tr}_A(\sum_i M_i^\dagger M_i |\varphi\rangle \langle\varphi|) \tag{2.62}$$

In Gleichung 2.63 und 2.64 wird deutlich, dass eine Messung in Teilsystem A keine Auswirkungen auf den beobachtbaren Zustand des Teilsystems B hat.

$$\mathrm{Tr}_A(\sum_i M_i^\dagger M_i \left|\varphi\right\rangle \left\langle\varphi\right|) = \mathrm{Tr}_A(\mathbf{I} \left|\varphi\right\rangle \left\langle\varphi\right|) \tag{2.63}$$

$$= \mathrm{Tr}_A(\left|\varphi\right\rangle \left\langle\varphi\right|)$$

$$= \rho_B \tag{2.64}$$

Signaling durch unitäre Transformation oder Messung eines Teilsystems sind damit innerhalb des mathematischen Modells nicht möglich.

2.8 Visualisierung des No-Signaling Theorems

Die Idee der partiellen Spur eines Systems wird im Folgenden anhand des Quantum Cube Models (Abschnitt 2.4) visualisiert. Betrachtet wird ein System aus drei Qubits. Wie in Abbildung 2.12 zu erkennen ist, liegen die ersten beiden Qubits (verantwortlich für links-rechts und unten-oben) in den Händen von Alice (blau). Das dritte Qubit (verantwortlich für vorne-hinten) liegt in den Händen von Bob (rot).

Die Amplituden werden nicht mit eingezeichnet, um die Darstellung nicht zu überladen.

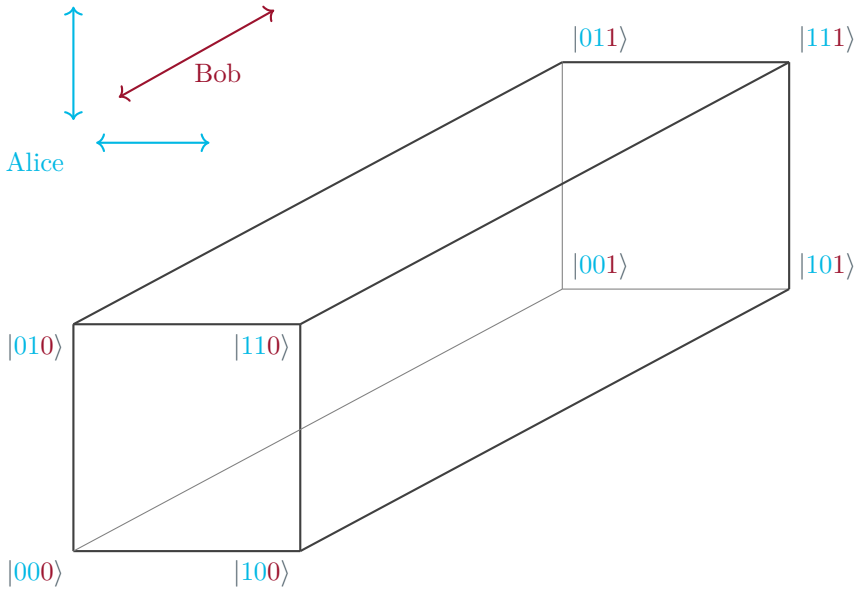


Abbildung 2.12: Aufteilung eines Systems mit drei Qubits auf Alice (Qubits 1 und 2) und Bob (Qubit 3)

Bei Kenntnis des Gesamtsystems gibt es acht Messergebnisse, deren Wahrscheinlichkeiten wie in Abschnitt 2.2 beschrieben durch ihre jeweiligen Amplituden definiert sind.

Aus Sicht von Alice gibt es lediglich vier Messergebnisse, $|00\rangle_A$, $|01\rangle_A$, $|10\rangle_A$ und $|11\rangle_A$, deren Wahrscheinlichkeiten jeweils durch einen Amplitudenvektor aus zwei Amplituden gegeben sind (Abbildung 2.13). Im Allgemeinen sind Alice die Koeffizienten dieser Vektoren unbekannt. Auch wenn Alice zu Beginn die Koeffizienten kannte, können sich diese durch Operationen auf Bobs Qubit verändern.

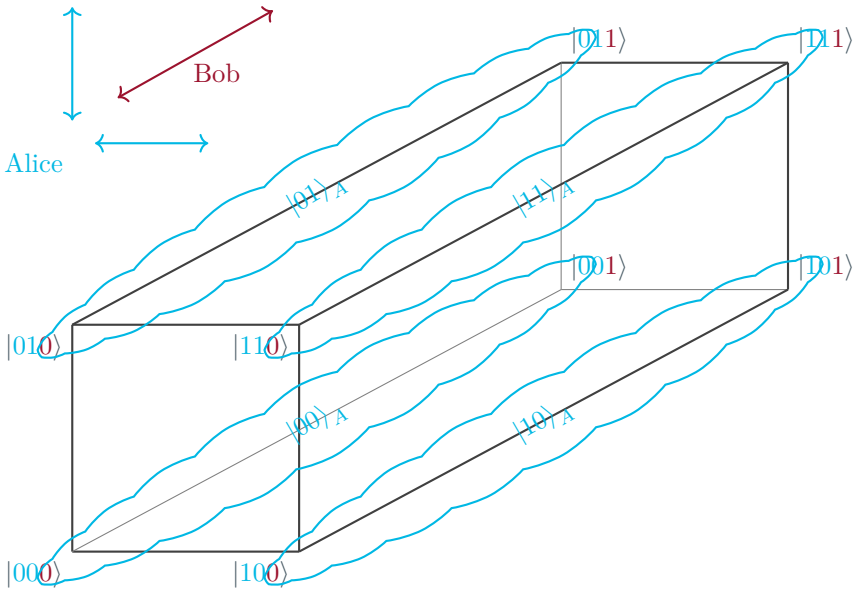


Abbildung 2.13: Sicht von Alice auf das System aus drei Qubits

Analog gibt es aus Sicht von Bob zwei Messergebnisse $|0\rangle$ und $|1\rangle$, deren Wahrscheinlichkeiten durch Amplitudenvektoren mit vier Koeffizienten gegeben sind (Abbildung 2.14). Auch Bob kennt die Koeffizienten (meist) nicht. Diese können sich verändern, indem Alice Operationen an ihren Qubits vornimmt.

Die Aussage des No-Signaling Theorems ist nun wie in Abschnitt 2.8 bereits beschrieben wie folgt: Durch unitäre Transformationen, die Bob an seinem Qubit vornimmt, ändern sich die paarweisen Skalarprodukte von Alices Amplitudenvektoren nicht. Auch ändern sich die Skalarprodukte nicht, durch eine Messung, die Bob an seinem Qubit vornimmt. Analoges gilt auch bei vertauschten Rollen für die paarweisen Skalarprodukte der Amplitudenvektoren von Bobs Qubit.

Es folgt die Darstellung des No-Signaling Theorems für unitäre Transformationen. In Abbildung 2.15 ist die Wirkung einer unitären Transformation dargestellt, die Bob auf sein Qubit anwendet. Unitäre Transformationen erhalten Skalarprodukte (Abschnitt 2.7). Damit folgt die Aussage des No-Signaling Theorems, denn es wird auf jeden Amplitudenvektor von Alice dieselbe unitäre Transformation angewendet. Analog dazu ist in Abbildung 2.16 dargestellt, dass Alice eine unitäre Transformation auf ihre Qubits anwendet und dass diese die Skalarprodukte von Bobs Amplitudenvektoren unverändert lassen.

Betrachtet wird nun die Sicht von Alice darauf, dass Bob sein Qubit misst. Illustriert wird dies anhand des Skalarproduktes der oberen beiden Amplitudenvektoren von Alice.

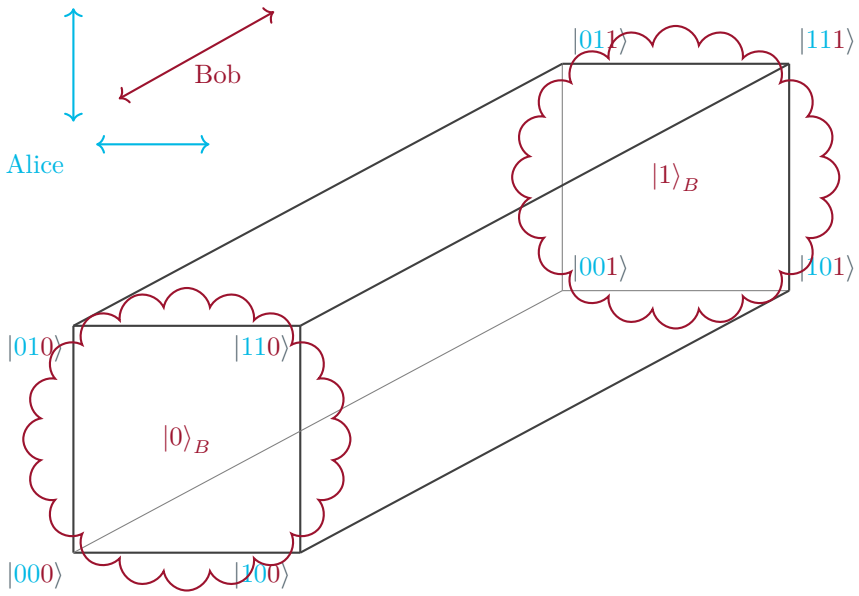


Abbildung 2.14: Sicht von Bob auf das System aus drei Qubits

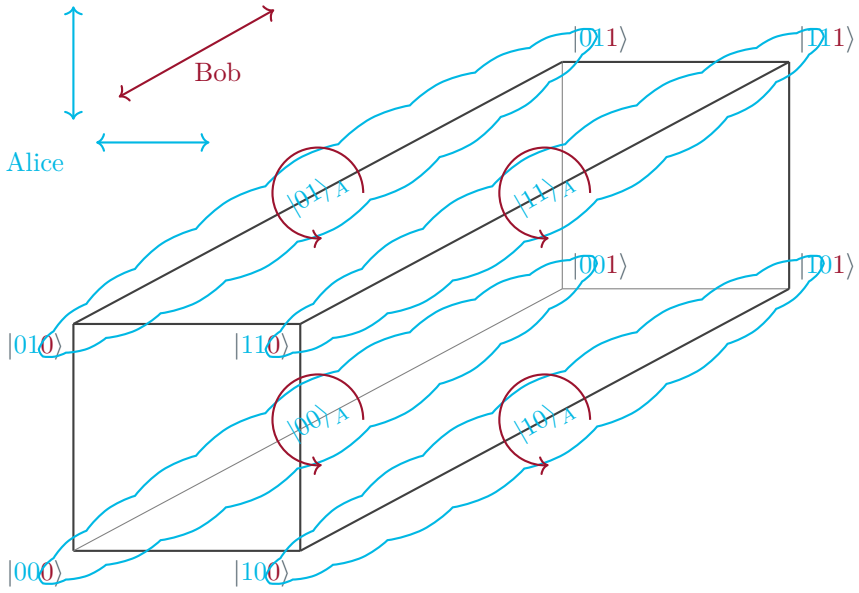


Abbildung 2.15: Sicht von Alice auf unitäre Transformationen auf Bobs Qubit

Bei der Messung von Bobs Qubit kommt mit einer Wahrscheinlichkeit p der Wert $|0\rangle_B$ heraus. Das System befindet sich dann im Zustand der Amplituden der vorderen Würfelseite, normiert mit $1/\sqrt{p}$. Analog kommt mit Wahrscheinlichkeit q (wobei $q = 1-p$, das ist hier aber nicht von Bedeutung) der Wert $|1\rangle_B$ heraus, das System befindet sich dann im Zustand der hinteren Würfelseite, normiert mit $1/\sqrt{q}$.

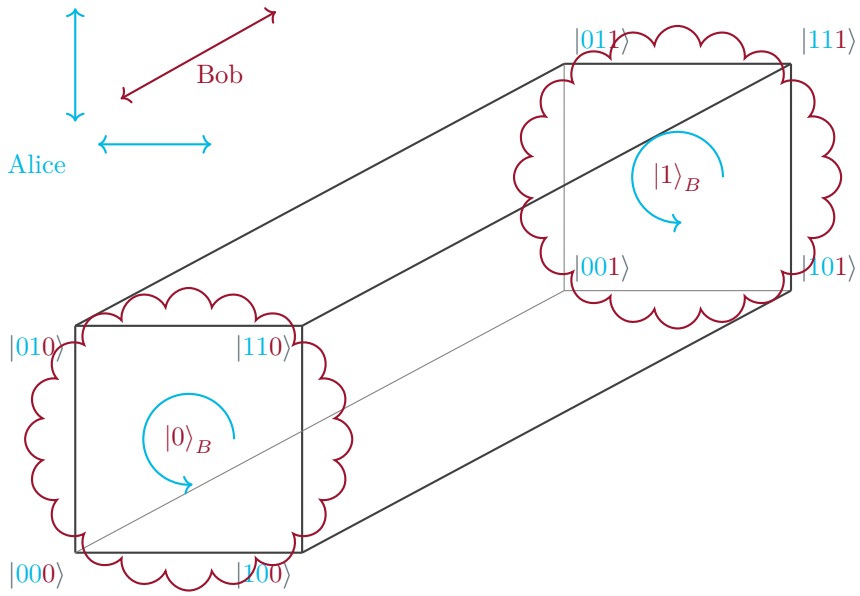


Abbildung 2.16: Sicht von Bob auf unitäre Transformationen auf Alices Qubits

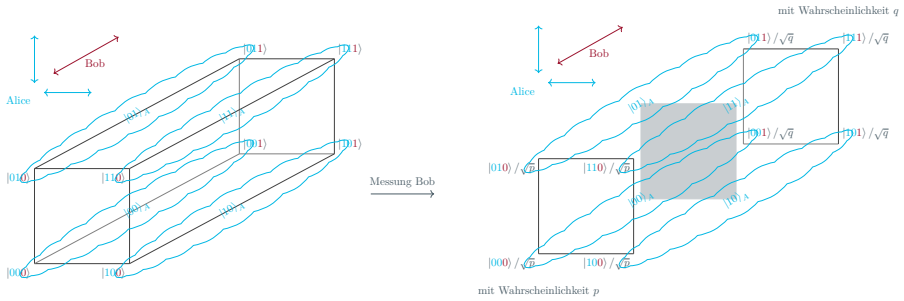


Abbildung 2.17: Sicht auf Alices Teilsystem vor und nach Messung von Bobs Qubit

Alice kennt das Messergebnis von Bob (ohne Kommunikation außerhalb des Systems) nicht. Abbildung 2.18 zeigt das Skalarprodukt der oberen beiden Amplituden von Alice vor der Messung von Bobs Qubit. Der Erwartungswert der Skalarprodukte ihrer Amplituden bleibt gleich, da sich die Wahrscheinlichkeiten mit der Normierung aufheben. Abbildung 2.19 zeigt das Skalarprodukt der oberen beiden Amplituden von Alice nach der Messung von Bobs Qubit. Es ist zu sehen, dass jeder Summand im Skalarprodukt gleich bleibt, weil $p/(\sqrt{p}*\sqrt{p}) = 1$ gilt (analog der Summand mit q).

Abbildung 2.20 illustriert die Situation aus Sicht von Bob, wenn Alice ihr erstes Qubit misst. Bob weiß nichts von der Messung und der Erwartungswert des Skalarproduktes seiner Amplitudenvektoren bleibt gleich.

Am Skalarprodukt des vorderen und hinteren Amplitudenvektors wird illustriert (Abbildung 2.21 und Abbildung 2.22), dass die Summanden des Skalarproduktes gleich bleiben. Der Faktor $p/(\sqrt{p} * \sqrt{p}) = 1$ gilt, wie schon zuvor.

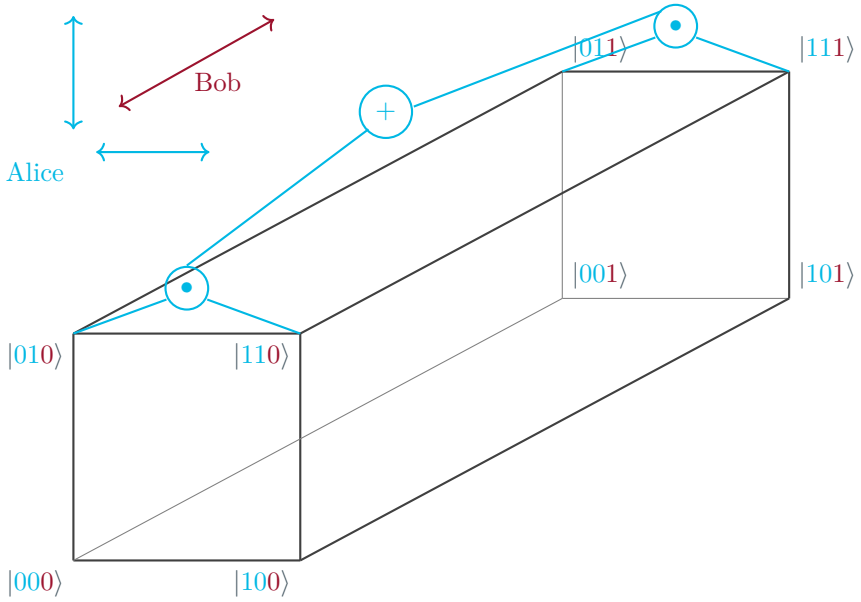


Abbildung 2.18: Visualisierung des Skalarproduktes der oberen beiden Amplitudenvektoren von Alice

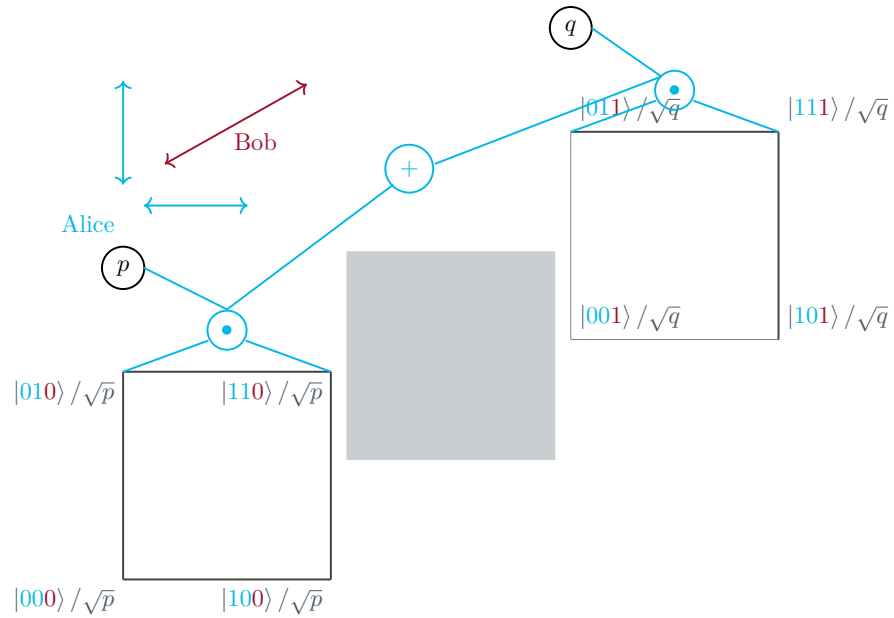


Abbildung 2.19: Visualisierung des Skalarproduktes der oberen beiden Amplitudenvektoren von Alice nach Messung von Bobs Qubit

Die reduzierte Dichtematrix von Alice enthält die paarweisen Skalarprodukte der Amplitudenvektoren von Alice, es ist eine 4×4-Matrix. In der Diagonalen stehen die Wahrscheinlichkeiten, dass Alice das entsprechende Messergebnis erhält.

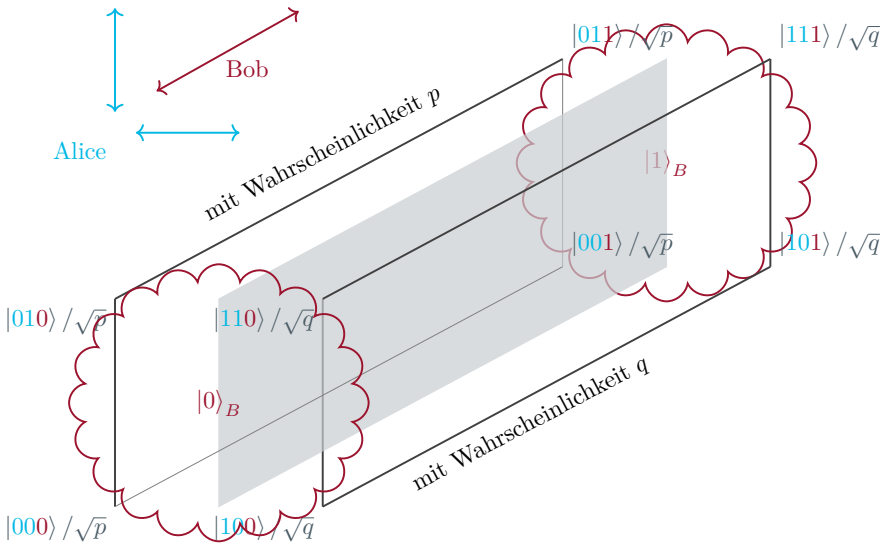


Abbildung 2.20: Sicht auf Bobs Teilsystem nach Messung des ersten Qubits von Alice

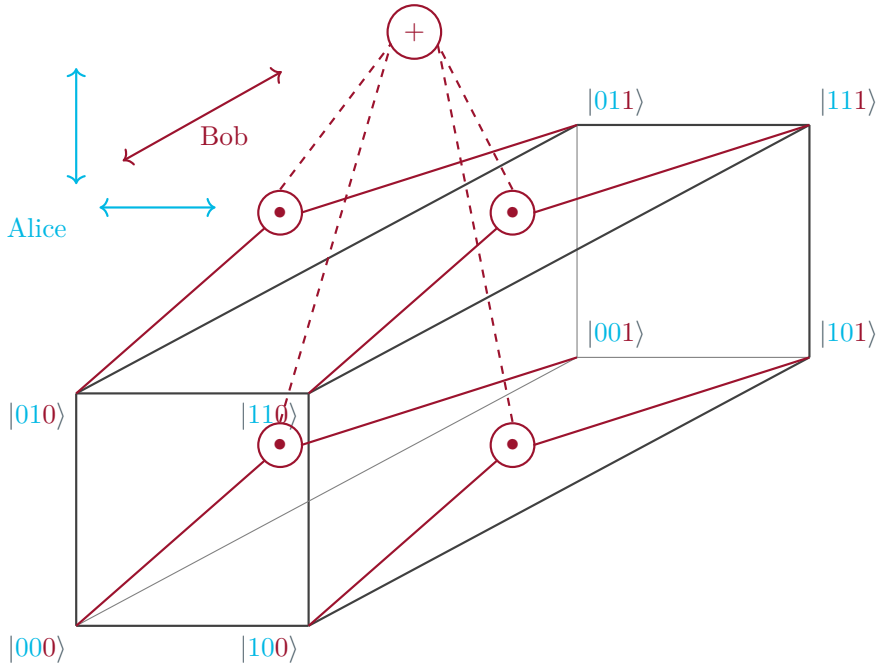


Abbildung 2.21: Skalarprodukt des vorderen und hinteren Amplitudenvektors von Bob

Die reduzierte Dichtematrix von Bob enthält entsprechend die paarweisen Skalarproduktes der Amplitudenvektoren von Bob, es ist eine 2×2-Matrix mit Wahrscheinlichkeiten für $|0\rangle_B$ und $|1\rangle_B$ in der Diagonalen.

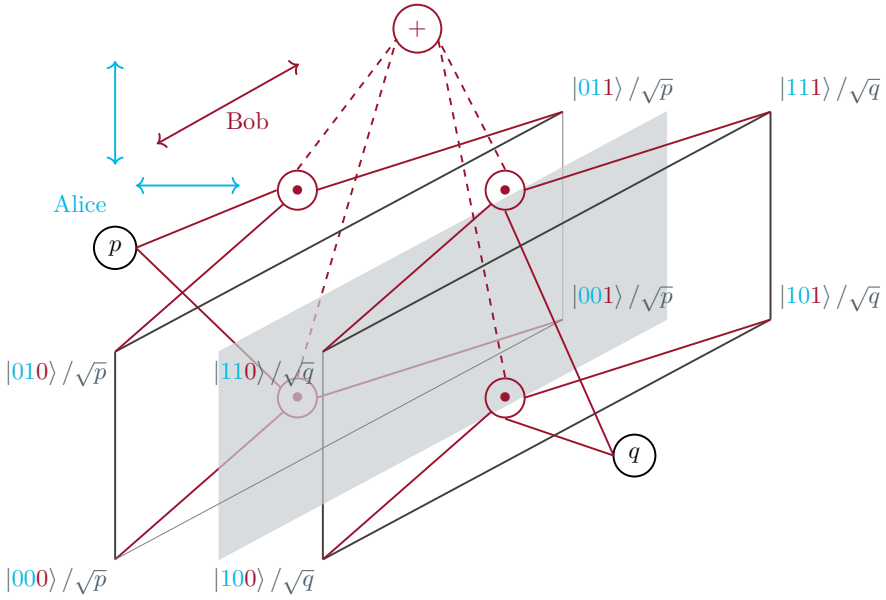


Abbildung 2.22: Skalarprodukt des vorderen und hinteren Amplitudenvektors von Bob nach Messung des ersten Qubits von Alice

Die Überlegungen können auf einen Mixed State übertragen werden. Bei einem Mixed state ist das System in einer konvexen Kombination aus Pure States. Abbildung 2.23 stellt dies aus Alices Sicht dar.

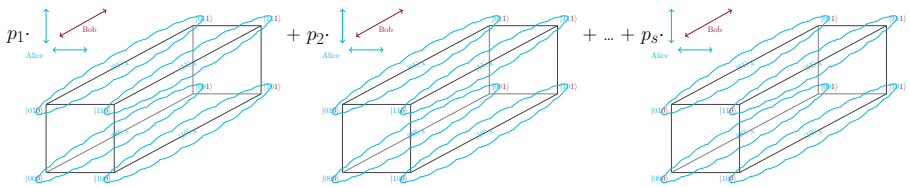


Abbildung 2.23: Darstellung eines Mixed States aus Sicht von Alice

Die reduzierte Dichtematrix von Alice ist dann die konvexe Kombination der reduzierten Dichtematrizen der beteiligten reinen Zustände.

2.9 CHSH-Game

Im Folgenden wird eine Ungleichung beschrieben, die genutzt wird, um die nicht Lokalität von Quantensystemen nachzuweisen.

In dieser Arbeit wird die Ungleichung als Grundlage einer booleschen Funktion verwendet. Die Funktion wird wie in Gleichung 2.65 definiert und in den folgenden Abschnitten dieser Arbeit als CHSH-Funktion bezeichnet.

$$\begin{aligned} chsh : \{0,1\}^4 &\to \{0,1\} \\ (x,y,a,b) &\mapsto x \wedge y \Leftrightarrow a \oplus b \end{aligned} \tag{2.65}$$

Die Funktion ist balanciert Es gibt also genau acht Belegungen die 0 ergeben und acht Belegungen die 1 ergeben.

Die Variablen der Funktion werden dabei in zwei Arten unterschieden: (x, y) werden im Folgenden als Challenge (Aufgabe) bezeichnet; (a, b) bilden die Response (Antwort).

Die Idee des CHSH-Games ist es, dass Alice (a) und Bob (b) jeweils eine Challenge zugewiesen bekommen. Dabei ist $x \in \{0, 1\}$ die Challenge für Alice und $y \in \{0, 1\}$ die Challenge für Bob. Alice und Bob können zu Beginn eine Strategie, festlegen. Danach werden Alice und Bob separiert, sodass keine weitere Kommunikation möglich ist. Der Aufbau und die Kommunikation zwischen Challenger und Alice bzw. Bob ist in Abbildung 2.24 dargestellt.

Die Aufgabe des CHSH-Games besteht darin, eine erfüllende Belegung für die Variablen (x, y, a, b) zu finden. Die Belegungen von $(x, y) \in \{0, 1\}^2$ werden dabei zufällig und mit gleicher Wahrscheinlichkeit ausgewählt. Alice erfährt dabei lediglich den Wert für x und Bob den Wert für y.

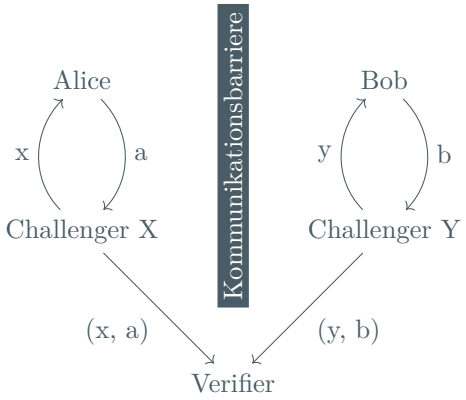


Abbildung 2.24: Aufbau des CHSH-Games mit Kommunikationsbarriere

Für den in Abbildung 2.24 beschriebenen Aufbau gibt es keine klassische Strategie, die eine höhere Gewinnchance als 75% liefert.[16] Teilen sich Alice und Bob hingegen ein verschränktes Qubit Paar (EPR-Paar), so ist eine Gewinnchance von etwa 85% möglich.[17]

Wie ein Schaltkreis aussehen kann, der eine Gewinnchance von mehr als 75% verspricht, ist in Unterabschnitt 3.1.4 beschrieben.

[16] Christoph Gallus, Pawel Blasiak und Emmanuel M. Pothos. „Winning a CHSH Game without Entangled Particles in a Finite Number of Biased Rounds: How Much Luck Is Needed?“ In: *Entropy* 25.5 (21. Mai 2023), S. 824. ISSN: 1099-4300. DOI: 10.3390/e25050824. URL: https://www.mdpi.com/1099-4300/25/5/824 (besucht am 11. 04. 2025).

[17] abbycross. *CHSH Game*. IBM Quantum Learning. Aug. 2025. URL: https://quantum.cloud.ibm.com/learning/en/courses/basics-of-quantum-information/entanglement-in-action/quantum.cloud.ibm.com/learning/en/courses/basics-of-quantum-information/entanglement-in-action/chsh-game (besucht am 27. 08. 2025).

3 Experimente

Die folgende Tabelle zeigt eine Übersicht der durchgeführten Experimente. Die Tabelle zeigt jeweils den Anteil korrekter Lösungen, für Schaltkreis Experimente ist dies der Anteil an Lösungen mit Wahrscheinlichkeit für das jeweilige Messergebnis $p > 0$. Für annealing Experimente ist eine korrekte Lösung die, die das Optimierungsproblem löst. Für die annealing Experimente wird zusätzlich eine Simulation durchgeführt. Zusätzlich zeigt die Tabelle ob die erwartete Verteilung erzielt wird. Für Schaltkreis Experimente kann diese wie in Abschnitt 2.5 berechnet werden. Für annealing Experimente wird eine Gleichverteilung über gleichwertigen Lösungen angenommen.

System	Experiment	Simulation: Anteil korrekter Lösungen	Simulation: Verteilung	Rechner: Anteil korrekter Lösungen	Rechner: Verteilung	Methode
IBM	3.1.1 Schaltkreis - $H\|0\rangle$	—		—	✓	χ^2-Test
	3.1.2 Schaltkreis - $(H \otimes H)\|00\rangle$	—		—	✗	χ^2-Test
	3.1.3 Schaltkreis - $CNOT_{0,1}(H \otimes \mathbf{I}_2)\|00\rangle$	—		98,2%	✗	χ^2-Test
	3.1.4 Schaltkreis - CHSH-Game	—		83,9%	✗	χ^2-Test
D-Wave	3.2.1 QUBO 00	32,2%	—	95,2%	—	—
	3.2.2 QUBO 10	29,8%	—	98%	—	—
	3.2.3 Gleichverteilung über zwei Ergebnissen	58,2%	✓	100%	✗	χ^2-Test
	3.2.4 Gleichverteilung über vier Ergebnissen	76,7%	✓	99,9%	✗	χ^2-Test
	3.2.5 EPR-Paar als $f(a,b) : a \Leftrightarrow b$	51,5%	✓	99,1%	✗	χ^2-Test
	3.2.6 Asymmetrische Funktion	64,9%	✓	86,1%	✗	χ^2-Test
	3.2.7 CHSH-QUBO 0	69,2%	✓	43,5%	✗	χ^2-Test

— Nicht durchgeführt
✓ Erwartete Verteilung erfüllt
✗ Nicht aussagekräftig
✗ Erwartete Verteilung nicht erfüllt

Tabelle 3.1: Übersicht der durchgeführten Experimente.

3.1 Schaltkreis Experimente

In den folgenden Abschnitten werden verschiedene Experimente auf Quanten Rechnern von IBM durchgeführt. Die Experimente bauen dabei auf den verschiedenen Eigenschaften des Quanten Rechners auf.

In Unterabschnitt 3.1.1 und Unterabschnitt 3.1.2 werden in zwei Experimente durchgeführt, die das Phänomen der Superposition untersuchen. In Unterabschnitt 3.1.3 wird ein Experiment zur Verschränkung von zwei Qubits untersucht. Unterabschnitt 3.1.4 führt die Phänomene der vorhergehenden Experimente zusammen.

3.1.1 Schaltkreis — $H\left|0\right\rangle$

Im Folgenden wird ein Experiment zur Gleichverteilung von Messergebnissen auf einem Qubit ausgeführt. Gesucht ist ein Schaltkreis, der ein Qubit so vorbereitet, dass ein Messen dieses Qubits mit Wahrscheinlichkeit $p(\left|0\right\rangle) = 0.5$ den Wert $\left|0\right\rangle$ und mit Wahrscheinlichkeit $p(\left|1\right\rangle) = 0.5$ den Wert $\left|1\right\rangle$ ergibt.

Abbildung 3.1 zeigt einen Schaltkreis, der die Eigenschaft erfüllt. Gleichung 3.1 zeigt, dass sich das System nach Hadamard im $\left|+\right\rangle$ Zustand befindet. Gleichung 3.3 und Gleichung 3.4 zeigen, dass die Ergebnisse $\left|0\right\rangle$ und $\left|1\right\rangle$ jeweils mit Wahrscheinlichkeit $\frac{1}{2}$ auftreten.

$$H\left|0\right\rangle = \sqrt{\frac{1}{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}\left|0\right\rangle = \sqrt{\frac{1}{2}}(\left|0\right\rangle + \left|1\right\rangle) = \left|+\right\rangle \tag{3.1}$$

$$M_j = \left|j\right\rangle\left\langle j\right| \tag{3.2}$$

$$\left\langle +\right| M_0^\dagger M_0 \left|+\right\rangle = \frac{1}{2} \tag{3.3}$$

$$\left\langle +\right| M_1^\dagger M_1 \left|+\right\rangle = \frac{1}{2} \tag{3.4}$$

Implementierung



Abbildung 3.1: Schaltkreis Hadamard auf einem Qubit

Der Schaltkreis kann mit dem Code aus Listing 3.1 mit Qiskit angelegt werden.

```
1 from qiskit import QuantumCircuit
2
3 qc = QuantumCircuit(1)
4 qc.h(0)
5 qc.measure_all()
6 qc.draw("mpl")
```

Listing 3.1: Den Schaltkreis aus Abbildung 3.1 mit Qiskit anlegen

Erwartete Verteilung

Abbildung 3.2 zeigt die erwartete Verteilung für die Messergebnisse $|0\rangle$ und $|1\rangle$.

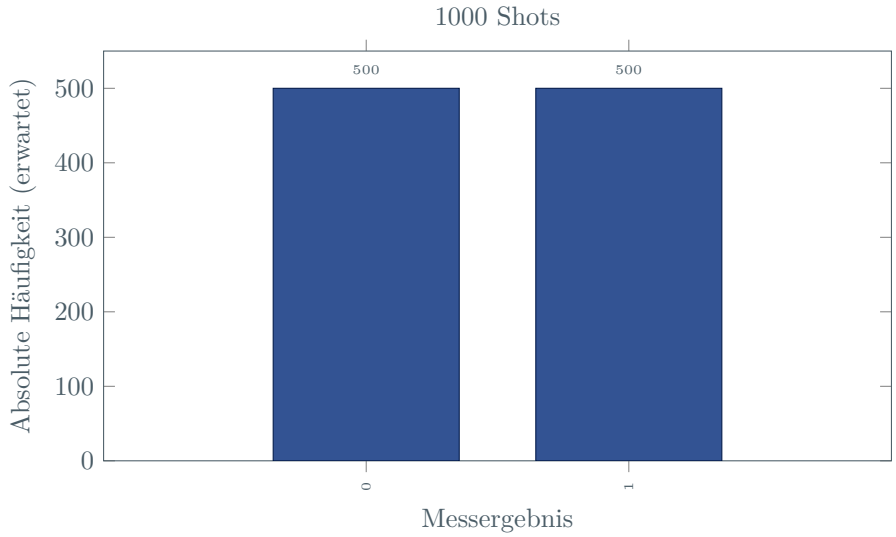


Abbildung 3.2: Erwartete Ergebnisse des Schaltkreises aus Abbildung 3.1.

Die Wirkung des Schaltkreises kann mit dem Quantum Cube Model, wie in Abbildung 3.3 dargestellt werden.



(a) Startzustand (b) Endzustand

Abbildung 3.3: Wirkung von Hadamard auf $|0\rangle$

Ausführungen auf IBM Brisbane

Abbildung 3.4 zeigt die tatsächliche Verteilung, gemessen auf dem „ibm_brisbane“ mit 1000 Ausführungen.

Mit dem χ^2-Test lässt sich ermitteln, ob die gemessene Verteilung zu der Erwartung passt.

H_0: Die Messergebnisse passen zur erwarteten Verteilung.

H_A: Die Messergebnisse weichen signifikant von der erwarteten Verteilung ab.

Der Wert für die χ^2-Verteilung ist für ein Signifikanzniveau $\alpha = 0,05$ bei einem Freiheitsgrad $\chi^2_{\text{kritisch}} = 3,841$.

Wie in Gleichung 3.5 zu erkennen, ist der empirische Wert $\chi^2_{\text{empirisch}} = 0,036$ deutlich kleiner als χ^2_{kritisch}. Somit wird die Nullhypothese angenommen.

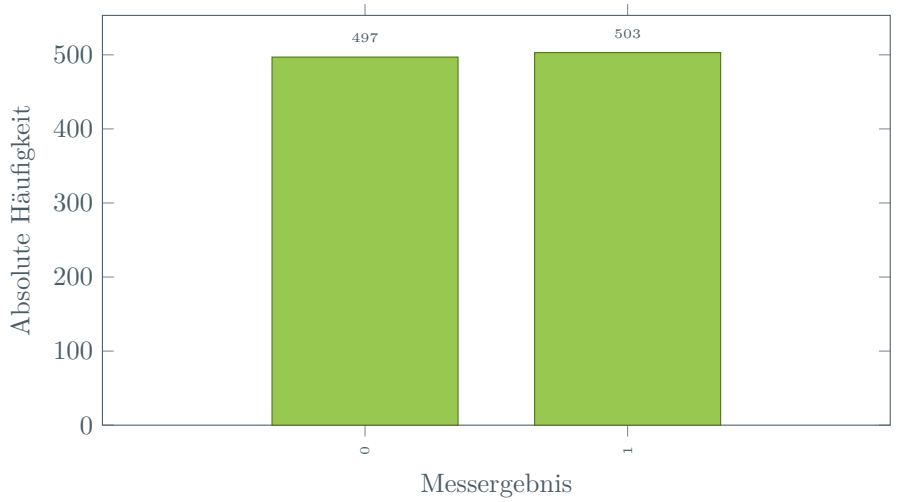


Abbildung 3.4: Ergebnisse von 1000 Ausführungen des Schaltkreises aus Abbildung 3.1 (Single Hadamard) auf dem IBM Rechner „ibm_brisbane“.

$$\begin{aligned}\chi^2_{\text{empirisch}} &= \frac{(497-500)^2}{500} + \frac{(503-500)^2}{500} \\ &= 0,036\end{aligned} \tag{3.5}$$

3.1.2 Schaltkreis — $H \otimes H \left|00\right\rangle$

Analog zu dem Experiment aus Unterabschnitt 3.1.1 wird ein Experiment auf zwei Qubits durchgeführt. Dabei soll eine Gleichverteilung aller vier Basiszustände geschaffen werden.

Wie in Gleichung 3.7 bis 3.10 zu erkennen ist, ergibt sich für jedes Messergebnis die gleiche Wahrscheinlichkeit. Dementsprechend ist eine Gleichverteilung, wie in Abbildung 3.6 zu sehen ist, zu erwarten.

$$M_{ij} = \left|ij\right\rangle \left\langle ij\right| \tag{3.6}$$

$$\left\langle \varphi \right| M_{00}^\dagger M_{00} \left| \varphi \right\rangle = \frac{1}{4} \tag{3.7}$$

$$\left\langle \varphi \right| M_{01}^\dagger M_{01} \left| \varphi \right\rangle = \frac{1}{4} \tag{3.8}$$

$$\left\langle \varphi \right| M_{10}^\dagger M_{10} \left| \varphi \right\rangle = \frac{1}{4} \tag{3.9}$$

$$\left\langle \varphi \right| M_{11}^\dagger M_{11} \left| \varphi \right\rangle = \frac{1}{4} \tag{3.10}$$

Implementierung

Abbildung 3.5 zeigt einen Schaltkreis der die Gleichverteilung über allen Basiszustände erzeugt. Dieser Schaltkreis kann mit dem Code aus Listing 3.2 in Qiskit angelegt werden.

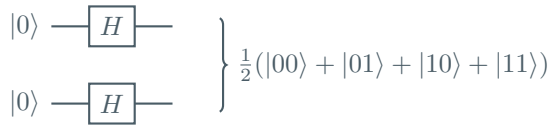


Abbildung 3.5: Zwei Qubit Schaltkreis mit je einem Hadamard

```
from qiskit import QuantumCircuit

qc = QuantumCircuit(2)
qc.h(0)
qc.h(1)
qc.draw("mpl")
```

Listing 3.2: Den Schaltkreis aus Abbildung 3.5 mit Qiskit anlegen

Erwartete Verteilung

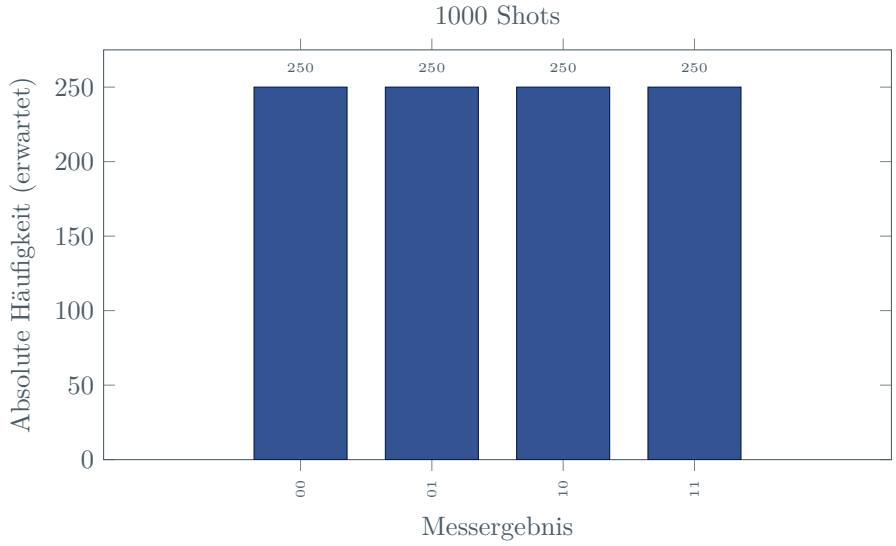
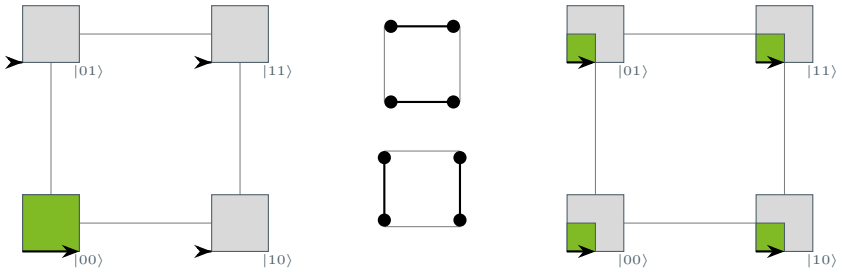


Abbildung 3.6: Erwartete Ergebnisse des Schaltkreises aus Abbildung 3.5.

Die Wirkung des Schaltkreises aus Abbildung 3.5 lässt sich auch mit dem Quantum Cube Model darstellen (Abbildung 3.7).



(a) Startzustand (b) Endzustand

Abbildung 3.7: Wirkung von $H \otimes H$ auf $|00\rangle$

Ausführungen auf IBM Brisbane

Betrachtet man die tatsächliche Verteilung aus Abbildung 3.8, ist zu erkennen, dass die Werte $|00\rangle$ und $|11\rangle$ häufiger vorkommen. Dies spricht für eine Verschränkung des Systems. Damit ist eine Verletzung der No-Signaling Eigenschaft nicht ausgeschlossen. Rybotycki u.a. 2025 zeigen dies auf einem deutlich Hardware näheren Ansatz.[1]

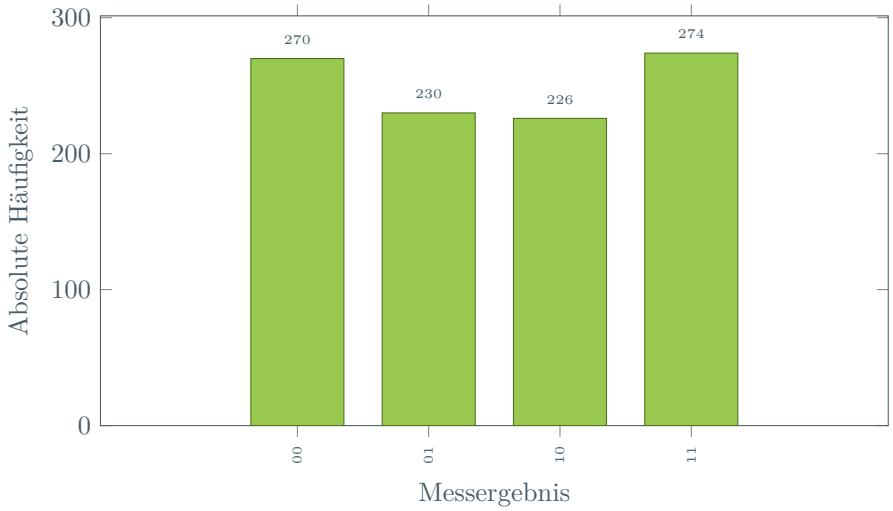


Abbildung 3.8: Ergebnisse von 1000 Ausführungen des Schaltkreises aus Abbildung 3.5 (Double Hadamard)

Wie in Unterabschnitt 3.1.1 lässt sich auch hier mit dem χ^2-Test überprüfen, ob die gemessenen Ergebnisse zu der erwarteten Verteilung passen.

H_0: Die Messergebnisse passen zur erwarteten Verteilung.

H_A: Die Messergebnisse weichen signifikant von der erwarteten Verteilung ab.

[1] Tomasz Rybotycki u. a. „Violation of No-Signaling on a Public Quantum Computer“. In: *Advanced Quantum Technologies* 8.6 (Juni 2025), S. 2400661. ISSN: 2511-9044, 2511-9044. DOI: 10.1002/qute.202400661. arXiv: 2409.11348 [quant-ph]. URL: http://arxiv.org/abs/2409.11348 (besucht am 07. 08. 2025).

Der Wert für die χ^2-Verteilung ist für ein Signifikanzniveau $\alpha = 0,05$ bei drei Freiheitsgraden $\chi^2_{\text{kritisch}} = 7,815$.

$\chi^2_{\text{empirisch}} = 7,808$ liegt genau an der Grenze. Die Nullhypothese kann also weder sicher angenommen, noch abgelehnt werden.

3.1.3 Schaltkreis — $CNOT_{0,1}(H \otimes \mathbf{I}_2)\,|00\rangle$

Im Folgenden wird ein Schaltkreis beschrieben, der eine Gleichverteilung über den beiden Messergebnissen $|00\rangle$ und $|11\rangle$ liefert. Der Schaltkreis ist so angelegt, dass bei Betrachtung eines einzelnen Qubits des Systems eine zufällige Verteilung über beiden Basiszuständen jeweils mit Wahrscheinlichkeit $1/2$ beobachtet wird.

Abbildung 3.9 zeigt einen Schaltkreis mit den gewünschten Eigenschaften. Der Endzustand wird im Folgenden mit $|\varphi\rangle$ (Gleichung 3.11) bezeichnet.

$$|\varphi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \tag{3.11}$$

Gleichung 3.12 und 3.13 zeigen, dass die Messung des ersten Qubits eine Gleichverteilung über den beiden möglichen Ergebnissen $|0\rangle$ und $|1\rangle$ liefert. Äquivalent dazu zeigen Gleichung 3.14 und 3.15 die Gleichverteilung für das zweite Qubit.

$$\langle\varphi|\,(|0\rangle\langle 0| \otimes \mathbf{I}_2)^\dagger(|0\rangle\langle 0| \otimes \mathbf{I}_2)\,|\varphi\rangle = \frac{1}{2} \tag{3.12}$$

$$\langle\varphi|\,(|1\rangle\langle 1| \otimes \mathbf{I}_2)^\dagger(|1\rangle\langle 1| \otimes \mathbf{I}_2)\,|\varphi\rangle = \frac{1}{2} \tag{3.13}$$

$$\langle\varphi|\,(\mathbf{I}_2 \otimes |0\rangle\langle 0|)^\dagger(\mathbf{I}_2 \otimes |0\rangle\langle 0|)\,|\varphi\rangle = \frac{1}{2} \tag{3.14}$$

$$\langle\varphi|\,(\mathbf{I}_2 \otimes |1\rangle\langle 1|)^\dagger(\mathbf{I}_2 \otimes |1\rangle\langle 1|)\,|\varphi\rangle = \frac{1}{2} \tag{3.15}$$

Diese Wahrscheinlichkeiten unterscheiden sich noch nicht von den Wahrscheinlichkeiten aus Unterabschnitt 3.1.2, bei denen auch eine Gleichverteilung der beiden möglichen Basiszustände jedes Qubits einzeln beobachtet wird. Der Unterschied in diesem Experiment wird deutlich, wenn die möglichen Messergebnisse des Gesamtsystems betrachtet werden. Gleichung 3.17 bis 3.20 zeigen, dass lediglich die Basiszustände $|00\rangle$ und $|11\rangle$ mit gleicher Wahrscheinlichkeit auftreten.

$$M_{ij} = |ij\rangle\langle ij| \tag{3.16}$$

$$\langle\varphi|\,M_{00}^\dagger M_{00}\,|\varphi\rangle = \frac{1}{2} \tag{3.17}$$

$$\langle\varphi|\,M_{11}^\dagger M_{11}\,|\varphi\rangle = \frac{1}{2} \tag{3.18}$$

$$\langle\varphi|\,M_{01}^\dagger M_{01}\,|\varphi\rangle = 0 \tag{3.19}$$

$$\langle\varphi|\,M_{10}^\dagger M_{10}\,|\varphi\rangle = 0 \tag{3.20}$$

Implementierung

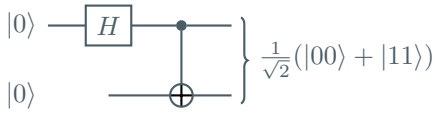


Abbildung 3.9: Schaltkreis zur Erzeugung eines EPR-Paars

Listing 3.3 zeigt wie der Schaltkreis aus Abbildung 3.9 mit Qiskit angelegt wird.

```
1 from qiskit import QuantumCircuit
2
3 qc = QuantumCircuit(2)
4 qc.h(0)
5 qc.cx(0, 1)
6 qc.draw("mpl")
```

Listing 3.3: EPR-Paar mit Qiskit anlegen

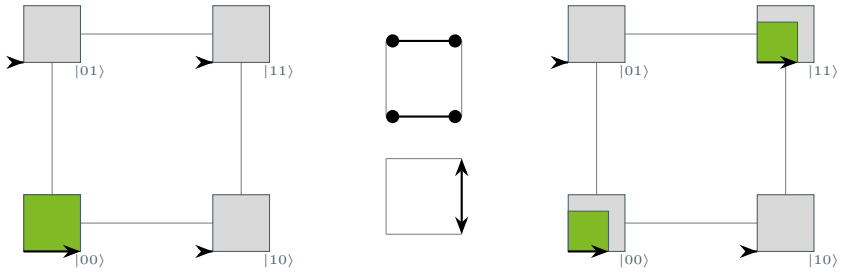
Erwartete Verteilung

Abbildung 3.10 zeigt die erwartete Verteilung der Messergebnisse bei 1000 Durchläufen des Experiments.



Abbildung 3.10: Erwartete Ergebnisse des Schaltkreises aus Abbildung 3.9.

Diese Verteilung lässt sich auch mit dem Quantum Cube Model wie in Abbildung 3.11 darstellen.



(a) Startzustand (b) Endzustand

Abbildung 3.11: Wirkung des Schaltkreises aus Abbildung 3.9 auf den Zustand $|00\rangle$

Ausführungen auf IBM Brisbane

In Abbildung 3.12 ist zu erkennen, dass 18 Messergebnisse aufgetreten sind, die nicht zur Verteilung passen. Somit ergibt sich ein Anteil korrekter Lösungen von $98,2\%$. Im Folgenden werden die falschen Messergebnisse nicht weiter berücksichtigt. Die erwartete Verteilung wird darauf angepasst.



Abbildung 3.12: Ergebnisse von 1000 Ausführungen des Schaltkreises aus Abbildung 3.9 (EPR-Schaltkreises)

Mit dem χ^2-Test lässt sich ermitteln, ob die gemessene Verteilung zu der Erwartung passt.

H_0: Die Messergebnisse passen zur erwarteten Verteilung.

H_A: Die Messergebnisse weichen signifikant von der erwarteten Verteilung ab.

Der Wert für die χ^2-Verteilung ist für ein Signifikanzniveau $\alpha = 0,05$ bei einem Freiheitsgrad $\chi^2_{\text{kritisch}} = 3,841$.

Wie in Gleichung 3.5 zu erkennen, ist der empirische Wert $\chi^2_{\text{empirisch}} = 4,709$ größer als χ^2_{kritisch}, somit wird die Nullhypothese abgelehnt.

$$\begin{aligned}\chi^2_{\text{empirisch}} &= \frac{(457-491)^2}{491} + \frac{(525-491)^2}{491} \\ &= 4,709\end{aligned} \tag{3.21}$$

3.1.4 Schaltkreis — CHSH-Game

Im Folgenden wird ein Experiment vorgestellt, dass die in Abschnitt 2.9 beschriebene Ungleichung als Schaltkreis so umsetzt, dass diese häufiger erfüllt ist als dies mit einer klassischen Strategie möglich ist. Abbildung 3.13 zeigt den Schaltkreis, der dies umsetzt.

Implementierung

Der Schaltkreis ist in vier Teilen aufgebaut unterteilt durch die roten Linien in Abbildung 3.13. Im ersten Schritt der Vorbereitung werden die Qubits a und b in einen EPR-Zustand gebracht (vergl. Unterabschnitt 3.1.3). Im zweiten Schritt werden die Qubits x und y in eine gleichmäßige Superposition über allen vier Basiszuständen (vergl. Unterabschnitt 3.1.2) überführt. In Schritt drei wird eine Drehung der Basis vorgenommen, um die Ungleichung durch die Ähnlichkeit der Basiswinkel zu erfüllen. Schritt vier führt ein Quantenorakel aus, welches das Qubit r in den Zustand $|chsh(a,b,x,y)\rangle$ überführt.[2]

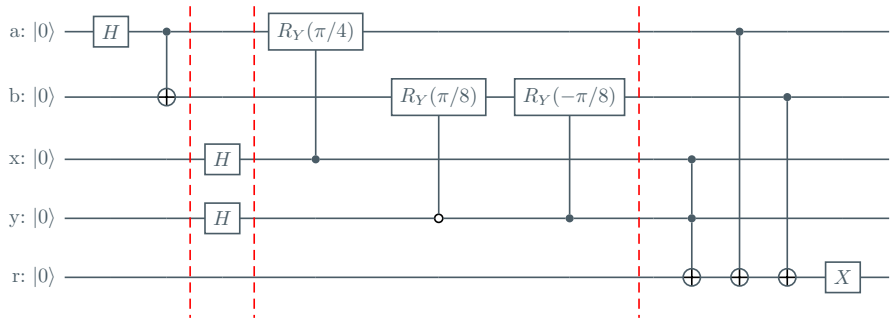


Abbildung 3.13: Schaltkreis CHSH-Game mit Quantenorakel für Result Qubit. abbycross. *CHSH Game*. IBM Quantum Learning. Aug. 2025. URL: https://quantum.cloud.ibm.com/learning/en/courses/basics-of-quantum-information/entanglement-in-action/quantum.cloud.ibm.com/learning/en/courses/basics-of-quantum-information/entanglement-in-action/chsh-game (besucht am 27. 08. 2025)

Der Schaltkreis lässt sich in Python mit Qiskit wie in Listing 3.4 zu sehen ist anlegen.

[2] Cedric Schacht. „Plattformübergreifende Anwendung Zur Darstellung von Quantenschaltkreisen Mit Proof of Concept Für Phase Kickback“. Technische Hochschule Mittelhessen, 30. Sep. 2023. URL: https://github.com/CedricSchacht/quantum-learing-app/blob/main/bachelor-thesis.pdf (besucht am 26. 08. 2025).

```python
from qiskit import QuantumCircuit, QuantumRegister, ClassicalRegister
from qiskit.circuit.library.standard_gates import RYGate
import numpy as np

alice = QuantumRegister(1, "alice")
bob = QuantumRegister(1, "bob")
qx = QuantumRegister(1, "x")
qy = QuantumRegister(1, "y")
r = QuantumRegister(1, "r")

chsh = QuantumCircuit(qx, qy, alice, bob, r)

# bell state
chsh.h(alice)
chsh.cx(alice, bob)

# challange
chsh.h(qx)
chsh.h(qy)

#alice preparation
u0 = RYGate(0).control(1, ctrl_state=0)
chsh.append(u0, [qx, alice])
u1 = RYGate(np.pi/4).control(1)
chsh.append(u1, [qx, alice])

#bob preparation
u2 = RYGate(np.pi/8).control(1, ctrl_state=0)
chsh.append(u2, [qy, bob])
u3 = RYGate(-np.pi/8).control(1)
chsh.append(u3, [qy, bob])

chsh.barrier()

chsh.ccx(qx, qy, r)
chsh.cx(alice, r)
chsh.cx(bob, r)
chsh.x(r)

chsh.measure_all()
```

Listing 3.4: Python Code, um den Schaltkreis aus Abbildung 3.13 anzulegen.

Erwartete Verteilung

In Abbildung 3.14 ist die erwartete Verteilung der Ergebnisse zu sehen. Die Bits sind dabei in der Reihenfolge *abxyr* äquivalent zur Reihenfolge im Schaltkreis Diagramm (Abbildung 3.13) gehalten. 16 der 32 Ergebnisse haben in der Verteilung kein Auftreten. Dies sind genau die Ergebnisse, in denen $chsh(abxy) \neq r$ die Erfüllung der CHSH Ungleichung nicht zum Wert von r passt.

Für die erwartete Verteilung ergibt sich eine Gewinnwahrscheinlichkeit von $p_{\text{winning}} \approx 0,798$. Diese ist damit größer als die optimale klassische Strategie, die, wie in Abschnitt 2.9 beschrieben, bei $0,75$ liegt.

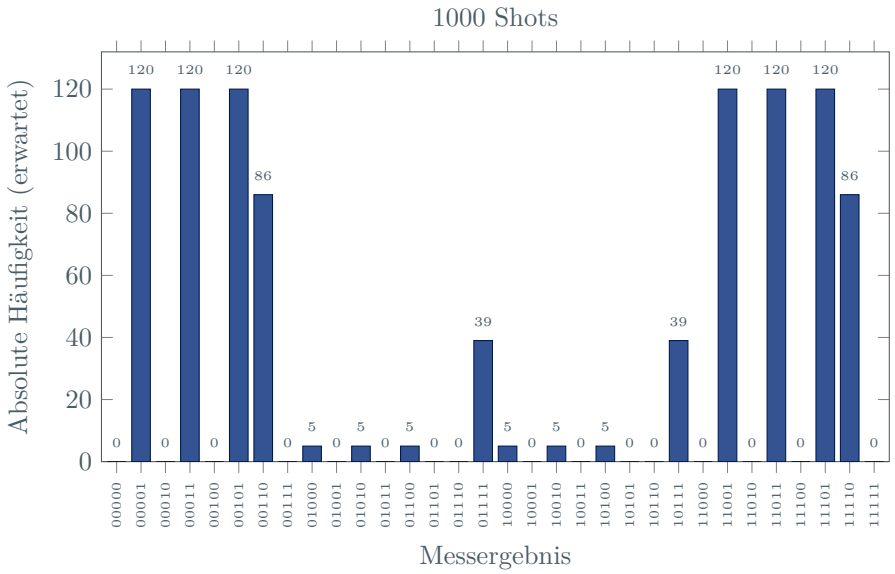


Abbildung 3.14: Erwartete Ergebnisse des Schaltkreises aus Abbildung 3.13.

Ausführungen auf IBM Brisbane

Abbildung 3.15 zeigt die Verteilung, die auf dem Quantencomputer von IBM gemessen ist. Der Anteil korrekter Lösungen liegt bei $83,9\%$. Im Folgenden wird die gemessene Verteilung bereinigt betrachtet. Die 161 Ergebnisse, die entsprechend der erwarteten Verteilung nicht auftreten können, werden nicht weiter betrachtet.

Der Anteil der Ergebnisse, die die Ungleichung erfüllen, ist mit $p'_{winning} = 0,708$ deutlich kleiner, und auch kleiner als die Gewinnwahrscheinlichkeit der klassischen Strategie.

Mit dem χ^2-Test lässt sich ermitteln, ob die gemessene Verteilung zu der Erwartung passt.

H_0: Die Messergebnisse passen zur erwarteten Verteilung.

H_A: Die Messergebnisse weichen signifikant von der erwarteten Verteilung ab.

Der Wert für die χ^2-Verteilung ist für ein Signifikanzniveau $\alpha = 0,05$ bei 15 Freiheitsgraden $\chi^2_{\text{kritisch}} = 24,996$.

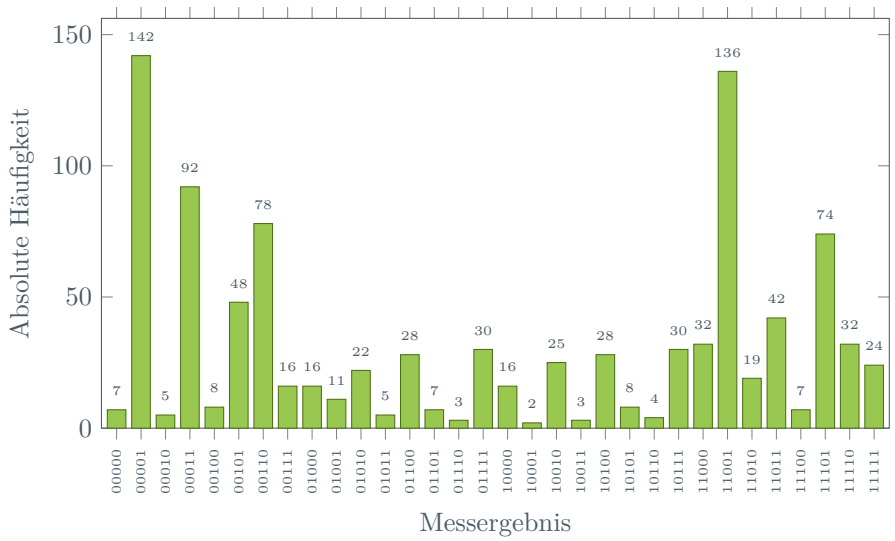


Abbildung 3.15: Ergebnisse von 1000 Ausführungen des Schaltkreises aus Abbildung 3.13 (CHSH-Schaltkreises)

Wie in Gleichung 3.22 zu erkennen ist, ist der empirische Wert $\chi^2_{\text{empirisch}} = 675,809$ deutlich größer als χ^2_{kritisch}. Somit wird die Nullhypothese klar abgelehnt.

$$\begin{aligned}\chi^2_{\text{empirisch}} = &\frac{(142-101)^2}{101} + \frac{(92-101)^2}{101} + \frac{(48-101)^2}{101} + \frac{(86-72)^2}{72} \\ &+ \frac{(16-4)^2}{4} + \frac{(22-4)^2}{4} + \frac{(28-4)^2}{4} + \frac{(30-33)^2}{33} \\ &+ \frac{(16-4)^2}{4} + \frac{(25-4)^2}{4} + \frac{(28-4)^2}{4} + \frac{(30-33)^2}{33} \\ &+ \frac{(136-101)^2}{101} + \frac{(42-101)^2}{101} + \frac{(74-101)^2}{101} + \frac{(32-72)^2}{72} \\ &= 675,809 \end{aligned} \tag{3.22}$$

3.2 Annealing Experimente

In den folgenden Abschnitten werden verschiedene Experimente auf einem annealing Rechner von D-Wave ausgeführt. Dabei liegt allen QUBO-Experimenten jeweils eine boolesche Funktion zugrunde, die als Optimierungsproblem betrachtet wird.

Wie die Funktionen als QUBO formuliert werden, ist in Abschnitt 2.6 beschrieben. Die Experimente in Unterabschnitt 3.2.1 bis 3.2.5 und Unterabschnitt 3.2.7 sind von den Experimenten aus Abschnitt 3.1 abgeleitet.

3.2.1 QUBO 00

Im Folgenden wird die boolesche Funktion $f(x_0, x_1)$ betrachtet, die nur an der Stelle $f(F, F) = T$ und sonst F ist (Gleichung 3.23).

$$f(x_0, x_1) = \begin{cases} T, & x_0 = F,\ x_1 = F \\ F, & \text{sonst} \end{cases} \tag{3.23}$$

Implementierung

Gleichung 3.24 zeigt, wie die Funktion als QUBO beschrieben wird.

$$Q = -0,25z_0 + z_0z_1 + z_0z_2 + z_0z_3 + z_1z_2 + z_1z_3 + z_2z_3 \tag{3.24}$$

```
1 from fToBqm import f_to_bqm
2 from prettyPrintBqm import pretty_print_bqm
3
4 def zero_zero(z_i: int) -> bool:
5     return z_i == 0
6
7 zero_zero_bqm = f_to_bqm(zero_zero, 2)
8 pretty_print_bqm(zero_zero_bqm)
```

Listing 3.5: Anlegen des Optimierungsproblems aus Gleichung 3.23

Erwartete Verteilung

Das Optimierungsproblem hat, wie in Abschnitt 2.6 bereits beschrieben, 2^N mögliche Ergebnisse. Wobei $N = 2^n$ die Anzahl der Zeilen in der Wertetabelle ist. Die möglichen Ergebnisse sind im Folgenden als „Bitmap *xyab*“ gekennzeichnet. Zusätzlich sind die Bitmaps, die genau einer Zeile der Wertetabelle entsprechen, als „Row r“ markiert.

Wie in Abbildung 3.16 zu erkennen ist, ist ausschließlich das Ergebnis „(Row 0) Bitmap 0001“ zu erwarten.

Simulation

Abbildung 3.17 zeigt die Ergebnisse der Simulation. Wie zu erkennen ist, liegt der Anteil der korrekten Lösungen bei $32,2\%$.

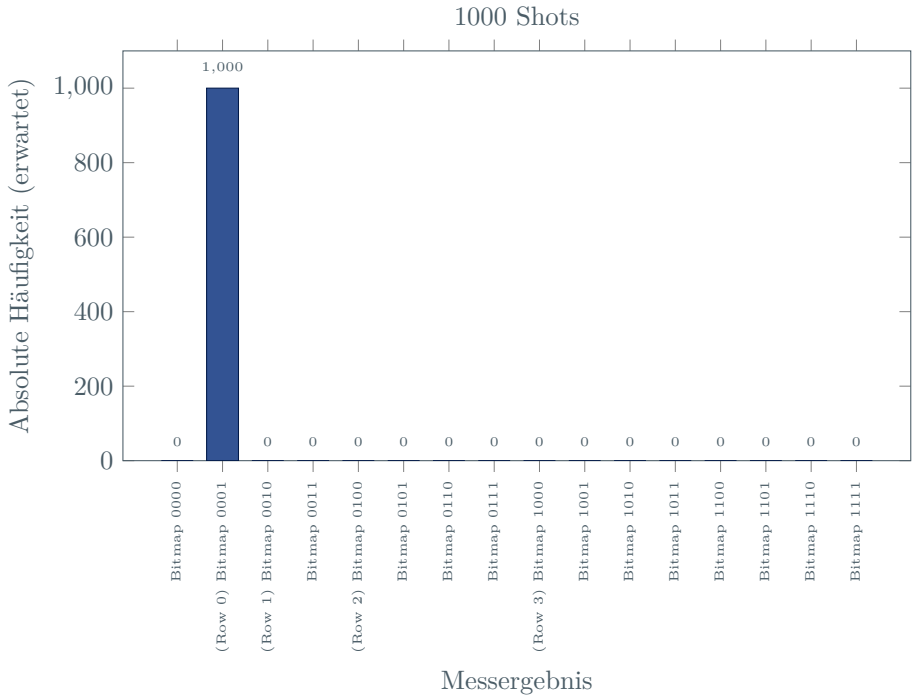


Abbildung 3.16: Erwartete Ergebnisse des QUBOs für Gleichung 3.23.

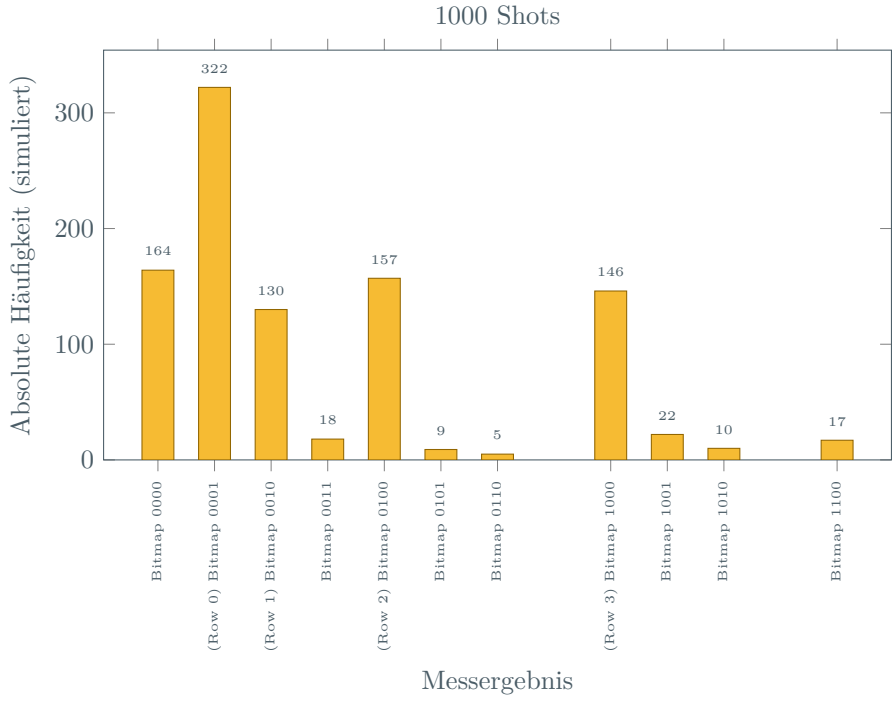


Abbildung 3.17: Ergebnisse der Simulation des QUBOs aus Gleichung 3.23 (QUBO 00).

Ausführungen auf Jupsi

Abbildung 3.18 zeigt die Ergebnisse der Berechnung auf einem annealing Rechner. Auffällig dabei ist, dass der Simulator deutlich weniger oft die optimale Lösung findet, als dies auf dem annealing Rechner der Fall ist.

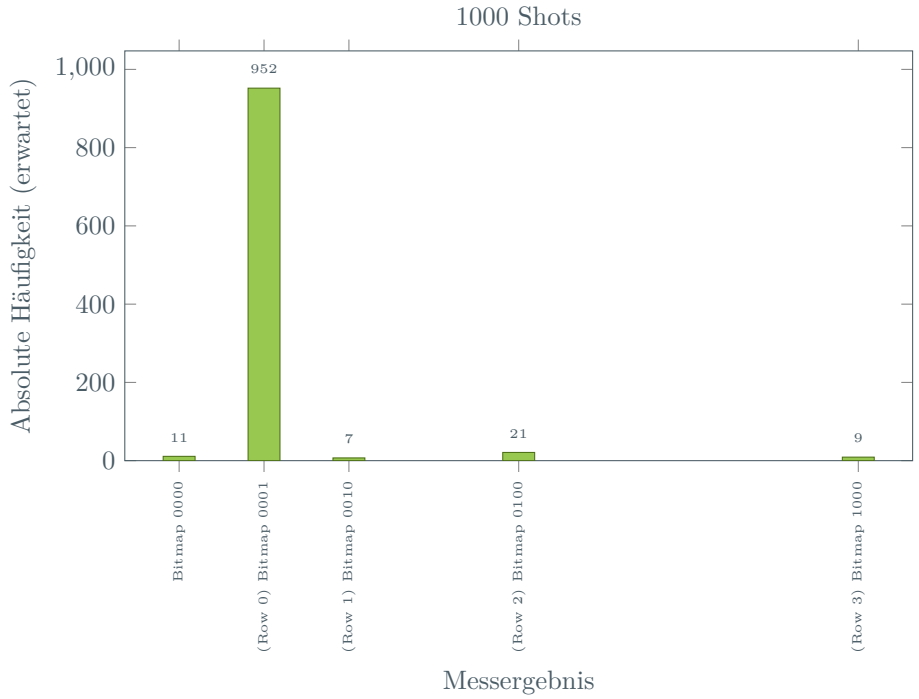


Abbildung 3.18: Gemessene Ergebnisse des QUBOs für Gleichung 3.23 (QUBO 00).

In Abbildung 3.18 ist außerdem zu erkennen, dass lediglich „Bitmap 0000“ mit 11 Auftritten keiner Zeile der Wertetabelle entspricht. Dies passt auch zu den Ergebnissen des Simulators, welcher „Bitmap 0000“ deutlich öfter als die anderen Lösung auswählt, die keiner eindeutigen Zeile entsprechen.

3.2.2 QUBO 10

Äquivalent zu der Funktion aus Unterabschnitt 3.2.1 lässt sich eine Funktion $f(x_0, x_1)$ aufstellen, die nur in der dritten Zeile der Wertetabelle T ist (Gleichung 3.25).

$$f(x_0, x_1) = \begin{cases} T, & x_0 = T,\ x_1 = F \\ F, & \text{sonst} \end{cases} \tag{3.25}$$

Implementierung

Gleichung 3.26 zeigt die Übersetzung der booleschen Funktion als QUBO, wie dies in Abschnitt 2.6 beschrieben ist.

$$Q = -0,25z_2 + z_0z_1 + z_0z_2 + z_0z_3 + z_1z_2 + z_1z_3 + z_2z_3 \tag{3.26}$$

```
1 from fToBqm import f_to_bqm
2 from prettyPrintBqm import pretty_print_bqm
3
4 def one_zero(z_i: int) -> bool:
5     return z_i == 0
6
7 one_zero_bqm = f_to_bqm(one_zero, 2)
8 pretty_print_bqm(one_zero_bqm)
```

Listing 3.6: Anlegen des Optimierungsproblems aus Gleichung 3.25

Erwartete Verteilung

Abbildung 3.19 zeigt die erwartete Verteilung. Lediglich das Ergebnis „(Row 2) Bitmap 0100“ stellt die optimale Lösung dar.

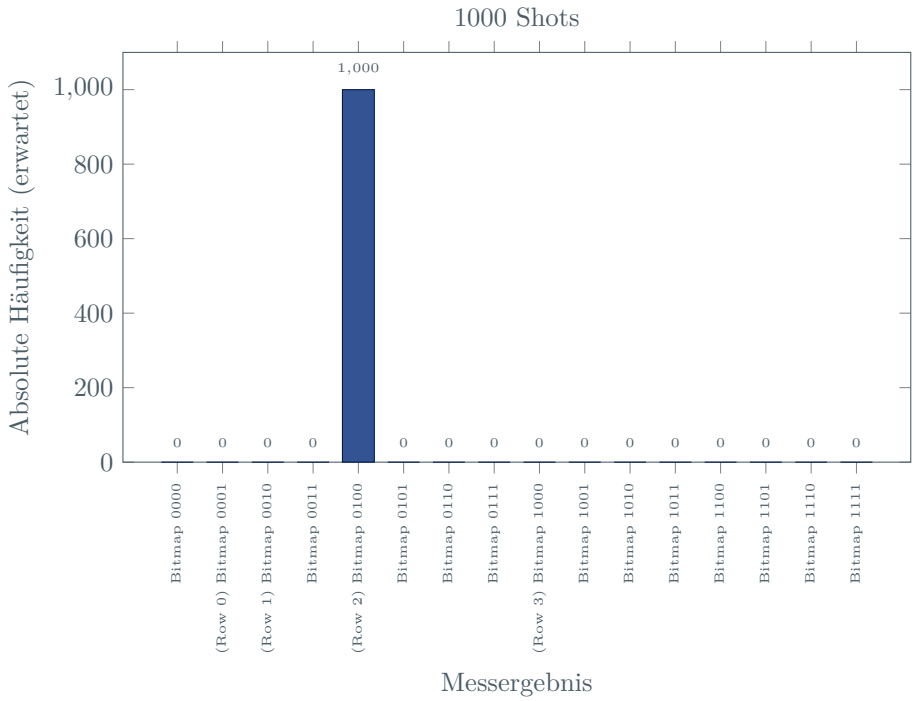


Abbildung 3.19: Erwartete Ergebnisse des QUBOs aus für Gleichung 3.25.

Simulation

Betrachtet man die Ergebnisse des Simulators (Abbildung 3.20), ist wie schon in Unterabschnitt 3.2.1 die optimale Lösung am häufigsten vertreten. Damit liegt der Anteil an korrekten Lösungen bei 29, 8%. Es treten zusätzlich signifikant viele Ergebnisse auf, die anderen Zeilen der Wertetabelle entsprechen. Abgesehen von einigen Ausreißern tritt „Bitmap 0000“ als unerwartetes Ergebnis außerhalb der eindeutigen Zeilen auf.

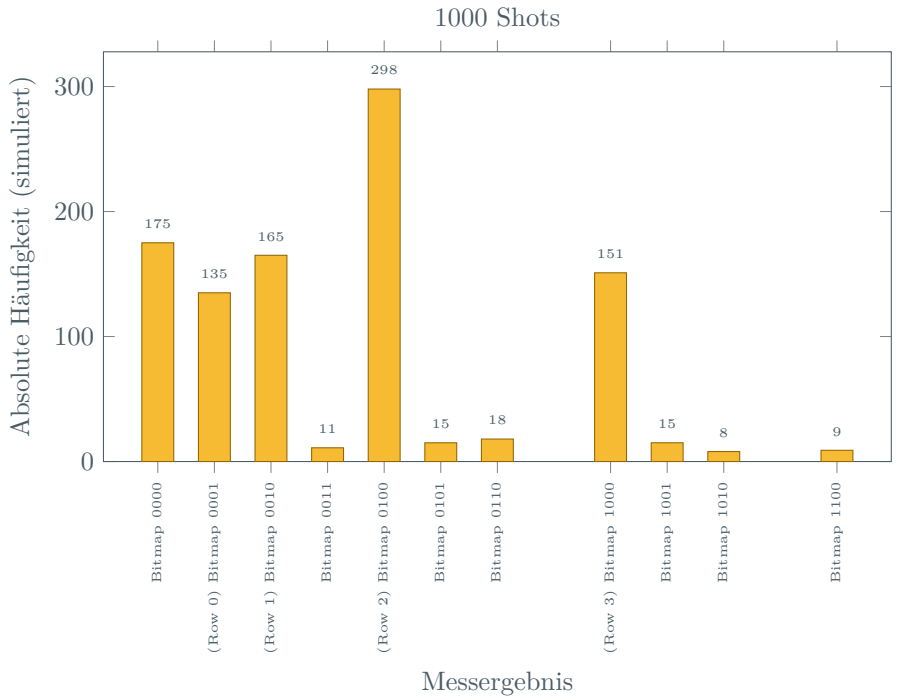


Abbildung 3.20: Ergebnisse der Simulation des QUBOs aus für Gleichung 3.25 (QUBO 10).

Ausführungen auf Jupsi

Im Vergleich dazu stehen die Ergebnisse des annealing Rechners (Abbildung 3.21), die wie schon im vorhergehenden Experiment deutlich häufiger die optimale Lösung abbilden. Der Anteil korrekter Lösungen liegt hier bei 98, 0%. Auffällig ist außerdem, dass wie schon zuvor lediglich das Ergebnis „Bitmap 0000“ unter den Lösungen liegt, die keiner eindeutigen Zeile zugeordnet werden können.

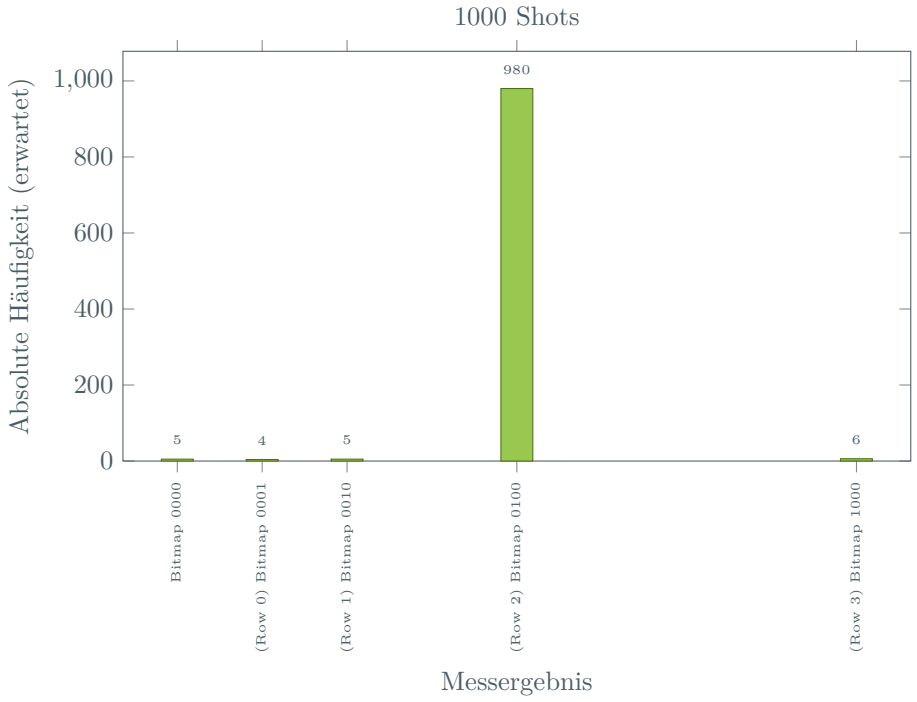


Abbildung 3.21: Gemessene Ergebnisse des QUBOs aus für Gleichung 3.25 (QUBO 10).

3.2.3 QUBO Gleichverteilung über zwei Ergebnissen

Das folgende Experiment erzeugt eine Verteilung über den beiden Ergebnissen, die genau einer Zeile zugeordnet werden können. Da die Funktion symmetrisch ist, ist eine Gleichverteilung über den beiden Ergebnissen zu erwarten.

Das Optimierungsproblem wird aus der Funktion aus Tabelle 3.2 erzeugt. Die Wertetabelle hat in beiden Zeilen das Ergebnis T.

z_i	x	$f(x)$
z_0	F	T
z_1	T	T

Tabelle 3.2: Wertetabelle der konstanten Funktion $f(x) = True$

Implementierung

Mit dem Code aus Listing 3.7 lässt sich die boolesche Funktion in die QUBO Form bringen, das resultierende Optimierungsproblem ist in Gleichung 3.27 dargestellt.

$$Q = -0,5z_0 - 0,5z_1 + z_0z_1 \tag{3.27}$$

```python
from fToBqm import f_to_bqm
from prettyPrintBqm import pretty_print_bqm

def uniform_distribution(z_i: int) -> bool:
    return True

bqm1 = f_to_bqm(uniform_distribution, 1)
pretty_print_bqm(bqm1)
```

Listing 3.7: Anlegen des Optimierungsproblems aus Tabelle 3.2 in Python

Erwartete Verteilung

Die erwartete Verteilung der Ergebnisse ist in Abbildung 3.24 dargestellt. Lediglich die beiden Ergebnisse die mit „Row“ gekennzeichnet sind, sollten auftreten.

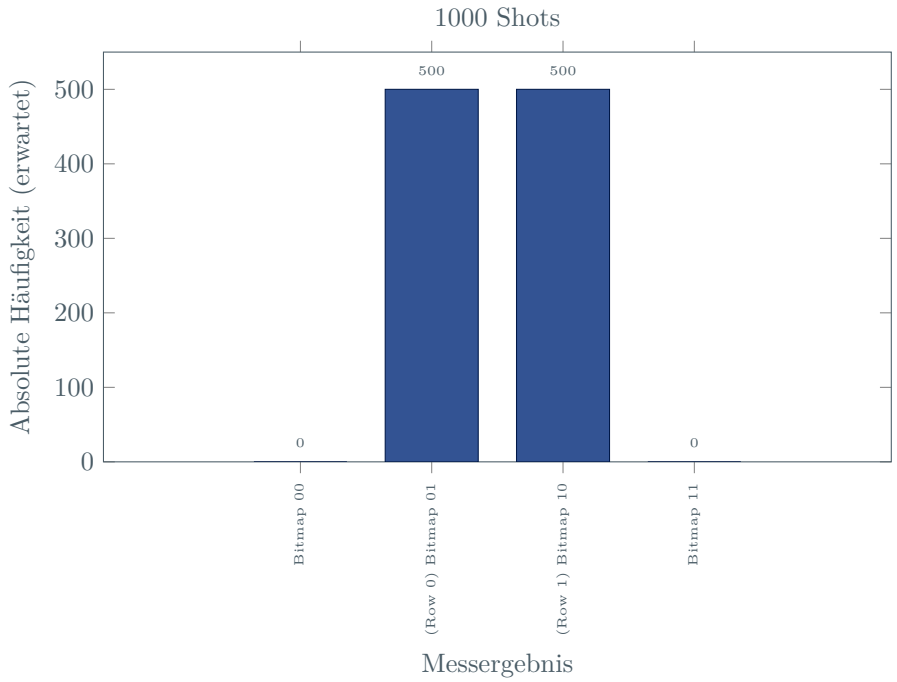


Abbildung 3.22: Erwartete Ergebnisse des QUBOs aus Tabelle 3.2.

Simulation

Abbildung 3.23 zeigt die Ergebnisse der Simulation. Der Anteil korrekter Lösungen liegt bei $58,2\%$. Im Folgenden werden die falschen Ergebnisse nicht weiter betrachtet.

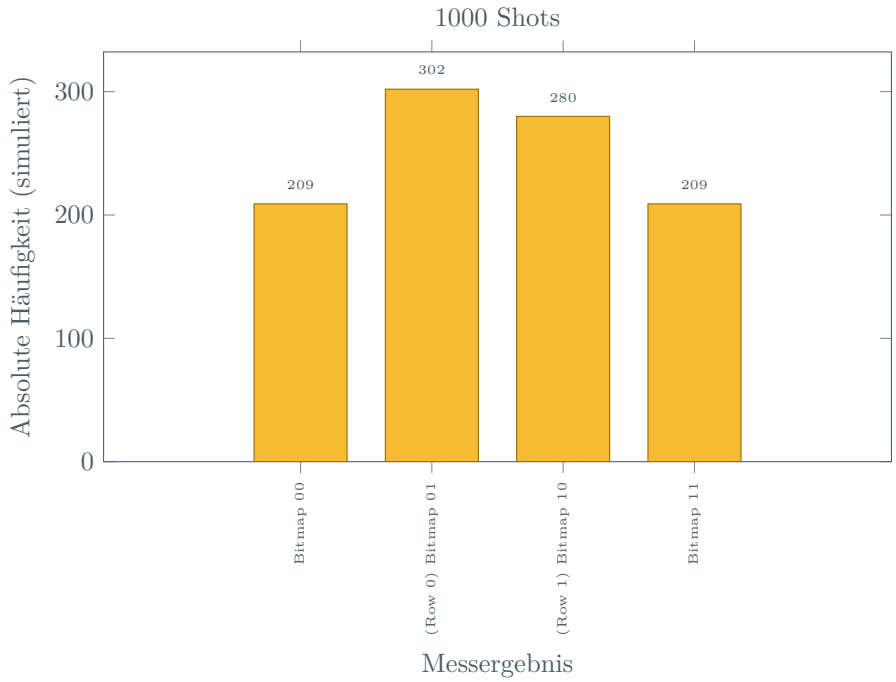


Abbildung 3.23: Ergebnisse der Simulation des QUBOs aus Tabelle 3.2 (Gleichverteilung über 2 Ergebnissen).

Mit dem χ^2-Test lässt sich ermitteln, ob die gemessene Verteilung der erwarteten Gleichverteilung entspricht.

H_0: Die Messergebnisse passen zur erwarteten Verteilung.

H_A: Die Messergebnisse weichen signifikant von der erwarteten Verteilung ab.

Der Wert für die χ^2-Verteilung ist für ein Signifikanzniveau $\alpha = 0,05$ bei einem Freiheitsgrad $\chi^2_{\text{kritisch}} = 3,841$.

Wie in Gleichung 3.28 zu erkennen ist, ist der empirische Wert $\chi^2_{\text{empirisch}} = 0,832$ deutlich kleiner als χ^2_{kritisch}. Somit wird die Nullhypothese angenommen.

$$\begin{aligned} \chi^2_{\text{empirisch}} &= \frac{(302-291)^2}{291} + \frac{(280-291)^2}{291} \\ &= 0,832 \end{aligned} \tag{3.28}$$

Ausführung auf Jupsi

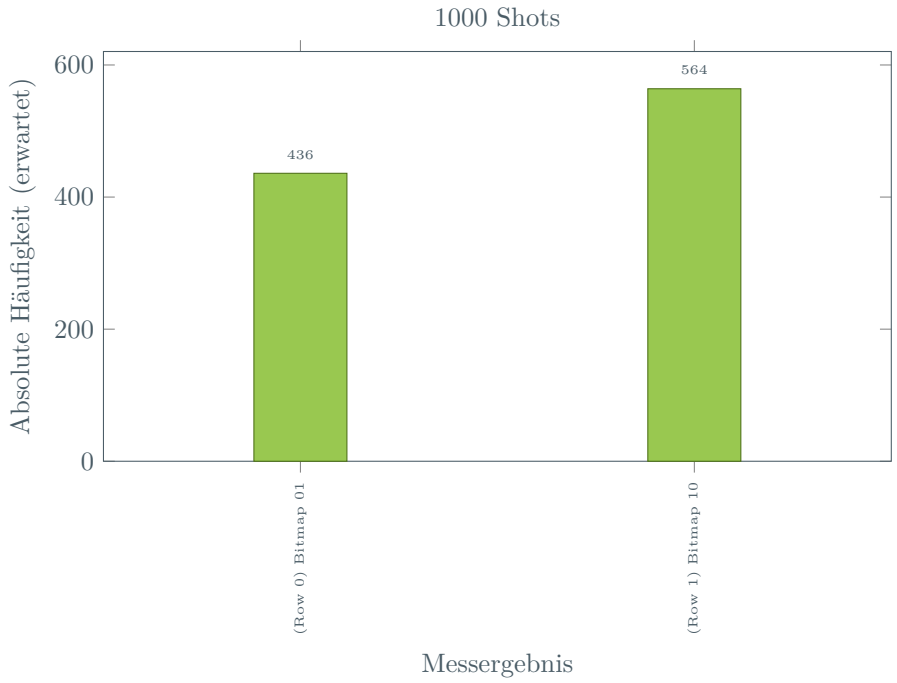


Abbildung 3.24: Ergebnisse des QUBOs aus Tabelle 3.2 (Gleichverteilung über 2 Ergebnissen).

Die Ergebnisse der Ausführung auf dem annealing Rechner von D-Wave (Abbildung 3.24) zeigen, dass ausschließlich optimale Lösungen gefunden wurden. Mit dem χ^2-Test lässt sich ermitteln, ob die gemessene Verteilung, der erwarteten Gleichverteilung entspricht.

H_0: Die Messergebnisse passen zur erwarteten Verteilung.

H_A: Die Messergebnisse weichen signifikant von der erwarteten Verteilung ab.

Der Wert für die χ^2-Verteilung ist für ein Signifikanzniveau $\alpha = 0,05$ bei einem Freiheitsgrad $\chi^2_{\text{kritisch}} = 3,841$.

Wie in Gleichung 3.29 zu erkennen, ist der empirische Wert $\chi^2_{\text{empirisch}} = 10,784$ größer als χ^2_{kritisch}. Somit wird die Nullhypothese abgelehnt. Eine Gleichverteilung ist also nicht zu erkennen.

$$\begin{aligned} \chi^2_{\text{empirisch}} &= \frac{(436-500)^2}{500} + \frac{(564-500)^2}{500} \\ &= 10,784 \end{aligned} \tag{3.29}$$

3.2.4 QUBO Gleichverteilung über vier Ergebnissen

Die konstante boolesche Funktion $f(x_1, x_2) = True$ (Tabelle 3.3) stellt die Grundlage für das folgende Optimierungsproblem dar. Erwartet wird, wie schon in Unterabschnitt 3.2.3, eine Gleichverteilung über den Ergebnissen die genau einer Zeile in der Wertetabelle zugeordnet werden können.

z_i	x_1	x_2	$f(x_1, x_2)$
z_0	F	F	T
z_1	F	T	T
z_2	T	F	T
z_3	T	T	T

Tabelle 3.3: Wertetabelle der konstanten Funktion $f(x_1, x_2) = True$

Die Begründung für die Gleichverteilung ist, wie auch im letzten Experiment, die Symmetrie der Funktion. x_1 und x_2 sind in der Funktion unabhängig. Für jedes Ergebnis kann die Wahrscheinlichkeit mit $0,25$ abgeschätzt werden.

Implementierung

Listing 3.8 zeigt, wie die boolesche Funktion in ein Optimierungsproblem überführt werden kann. Das daraus resultierende Optimierungsproblem ist in Gleichung 3.30 aufgestellt.

$$Q = -0,25(z_0 + z_1 + z_2 + z_3) + z_0z_1 + z_0z_2 + z_0z_3 + z_1z_2 + z_1z_3 + z_2z_3 \tag{3.30}$$

```
1 from fToBqm import f_to_bqm
2 from prettyPrintBqm import pretty_print_bqm
3
4 def uniform_distribution(z_i: int) -> bool:
5     return True
6
7 bqm2 = f_to_bqm(uniform_distribution, 2)
8 pretty_print_bqm(bqm2)
```

Listing 3.8: Anlegen des Optimierungsproblems aus Tabelle 3.3 in Python

Erwartete Verteilung

In Abbildung 3.25 ist die erwartete Verteilung dargestellt.

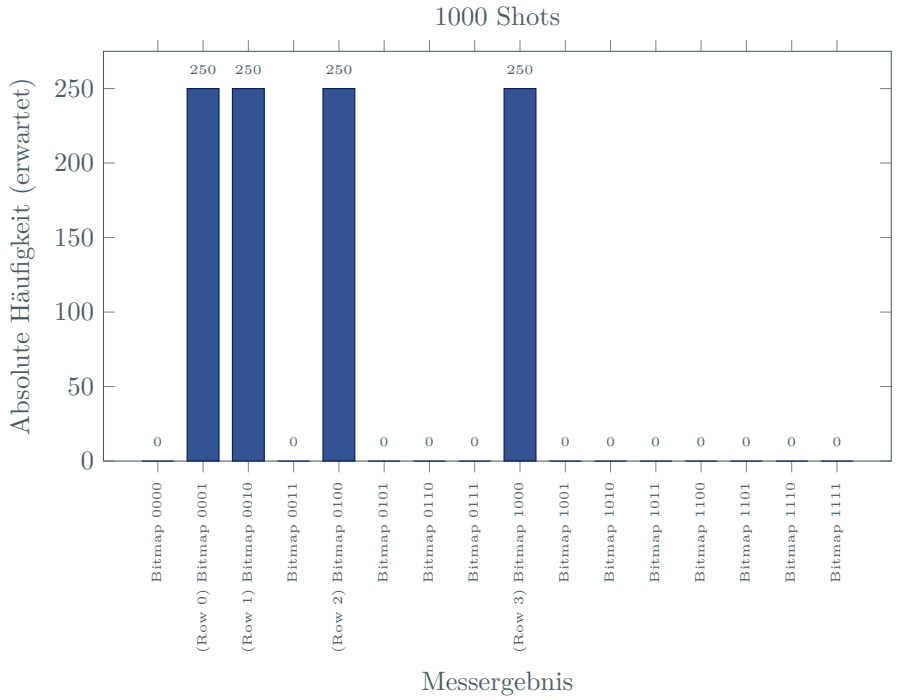


Abbildung 3.25: Erwartete Ergebnisse des QUBOs aus Tabelle 3.3.

Simulation

Die Ergebnisse der Simulation des Optimierungsproblems sind in Abbildung 3.26 dargestellt. Der Anteil korrekter Lösungen liegt bei $76,7\%$.

Der χ^2-Test für die bereinigten simulierten Ergebnisse (Gleichung 3.31), liegt mit $\chi_{\text{empirisch}} = 1,891$ deutlich unter dem kritischen Wert. $\chi_{\text{kritisch}} = 7,808$ bei einem Signifikanzniveau von $\alpha = 0,05$ und drei Freiheitsgraden, somit ist für die simulierten Ergebnisse eine Gleichverteilung gegeben.

H_0: Die Messergebnisse passen zur erwarteten Verteilung.

H_A: Die Messergebnisse weichen signifikant von der erwarteten Verteilung ab.

$$\begin{aligned}\chi^2_{\text{empirisch}} &= \frac{(177-192)^2}{192} + \frac{(196-192)^2}{192} + \frac{(203-192)^2}{192} + \frac{(191-192)^2}{192} \\ &= 1,891 \end{aligned} \tag{3.31}$$

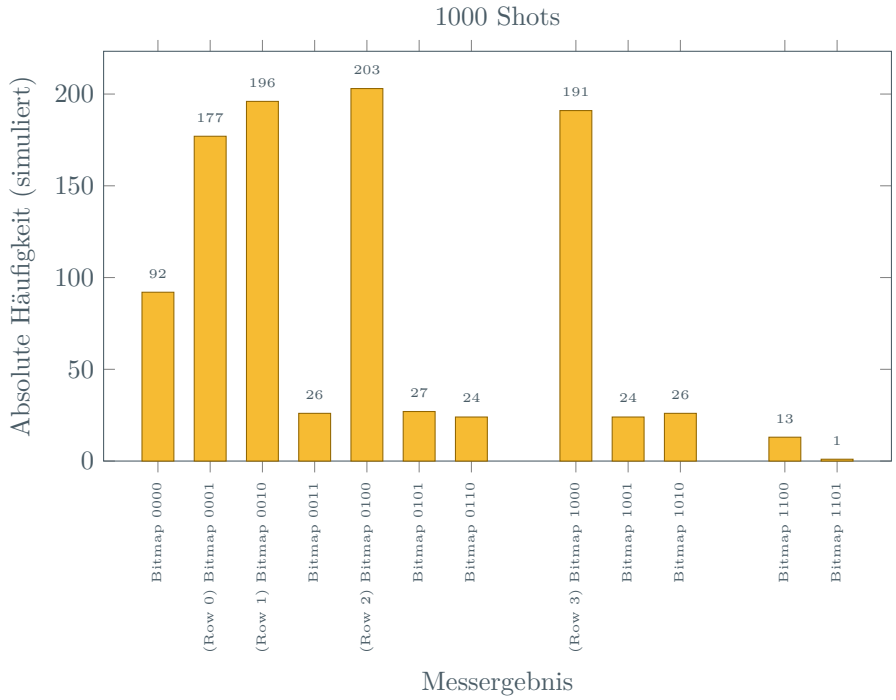


Abbildung 3.26: Ergebnisse der Simulation des QUBOs aus Tabelle 3.3 (Gleichverteilung über 4 Ergebnissen).

Ausführungen auf Jupsi

Im Gegensatz zu den Ergebnissen der Simulation ist unter den Ergebnissen des annealing Rechners nur ein Ergebnis, das nicht genau einer Zeile zugeordnet werden kann. Der Anteil korrekter Lösungen liegt hier bei $99,9\%$. Der annealing Rechner löst das Optimierungsproblem demnach deutlich öfter.

Der χ^2-Test für die bereinigten Ergebnisse (Gleichung 3.32) liegt mit $\chi_{\text{empirisch}} = 66,620$ deutlich über dem kritischen Wert, $\chi_{\text{kritisch}} = 7,808$ bei einem Signifikanzniveau von $\alpha = 0,05$ und drei Freiheitsgraden. Somit ist für die Ergebnisse keine Gleichverteilung gegeben.

H_0: Die Messergebnisse passen zur erwarteten Verteilung.

H_A: Die Messergebnisse weichen signifikant von der erwarteten Verteilung ab.

$$\begin{aligned}\chi^2_{\text{empirisch}} &= \frac{(235-250)^2}{250} + \frac{(175-250)^2}{250} + \frac{(353-250)^2}{250} + \frac{(236-250)^2}{250} \\ &= 66,620 \end{aligned} \tag{3.32}$$

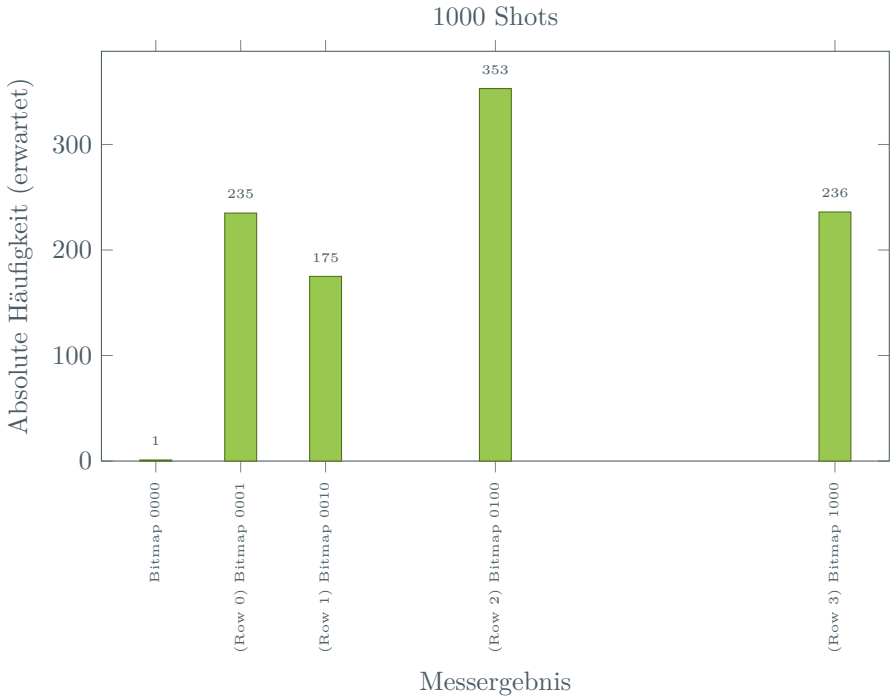


Abbildung 3.27: Erwartete Ergebnisse des QUBOs aus Tabelle 3.3 (Gleichverteilung über 4 Ergebnissen).

3.2.5 EPR als $f(a, b) : a \Leftrightarrow b$

Aus der erwarteten Verteilung des EPR-Experiments (Unterabschnitt 3.1.3) lässt sich eine boolesche Funktion ableiten. Die Wertetabelle für $f(x_1, x_2) = a \Leftrightarrow b$ ist in Tabelle 3.4 gegeben.

z_i	x_1	x_2	$f(x_1, x_2)$
z_0	F	F	T
z_1	F	T	F
z_2	T	F	F
z_3	T	T	T

Tabelle 3.4: Wertetabelle der Funktion $f(x_1, x_2) = a \Leftrightarrow b$

Implementierung

Listing 3.9 zeigt, wie die boolesche Funktion als Optimierungsproblem in Python implementiert werden kann. Das daraus resultierend Optimierungsproblem ist in Gleichung 3.33 aufgestellt.

$$Q = -0,25(z_0 + z_3) + z_0z_1 + z_0z_2 + z_0z_3 + z_1z_2 + z_1z_3 + z_2z_3 \tag{3.33}$$

```python
from fToBqm import f_to_bqm
from prettyPrintBqm import pretty_print_bqm

def epr(z_i: int) -> bool:
    a = z_i & 0b10 != 0
    b = z_i & 0b01 != 0
    return a == b

epr_bqm = f_to_bqm(epr, 2)
pretty_print_bqm(epr_bqm)
```

Listing 3.9: Anlegen des Optimierungsproblems aus Tabelle 3.4 in Python

Erwartete Verteilung

Die erwartete Verteilung der Ergebnisse ist in Abbildung 3.28 dargestellt. Eine Gleichverteilung über den beiden Ergebnissen „(Row 0) Bitmap 0001“ und „(Row 3) Bitmap 1000“ wird aufgrund der Symmetrie der boolesche Funktion erwartet.

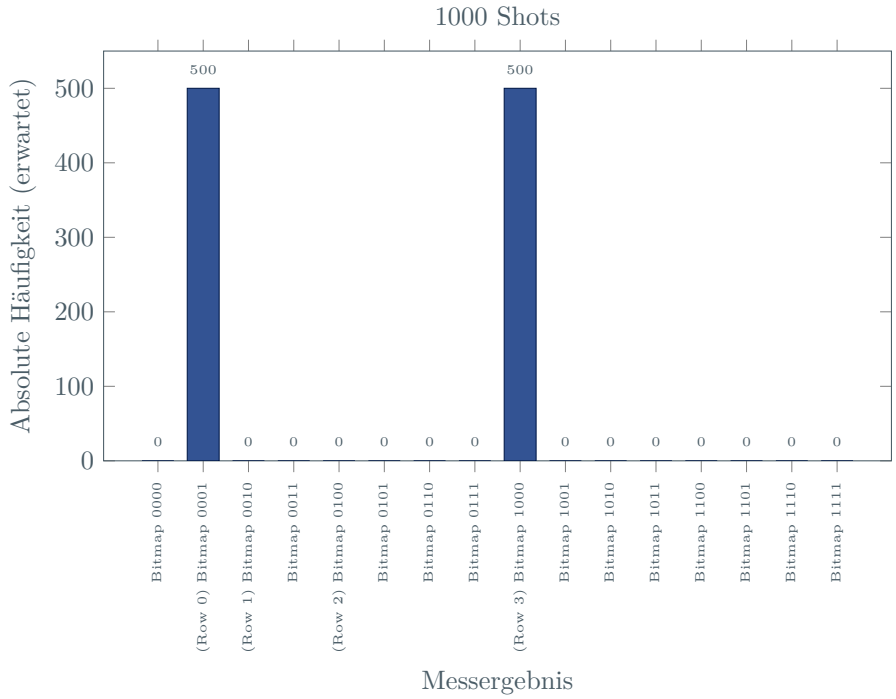


Abbildung 3.28: Erwartete Ergebnisse des QUBOs aus Tabelle 3.4.

Simulation

Abbildung 3.29 zeigt die Ergebnisse der Simulation. Der Anteil korrekter Lösungen liegt bei $51,5\%$. Wie bereits die Simulation in Unterabschnitt 3.2.4 liefert diese Simulation mit $48,5\%$ einen nennenswerten Anteil an Ergebnissen, die nicht in der Nähe einer optimalen Lösungen liegen. Auch liegt der Anteil der Null-Lösungen („Bitmap 0000“) im Vergleich zu den anderen Lösungen ohne eindeutige Zeile der Wertetabelle, wie schon in den vorherigen Experimenten, sehr hoch.

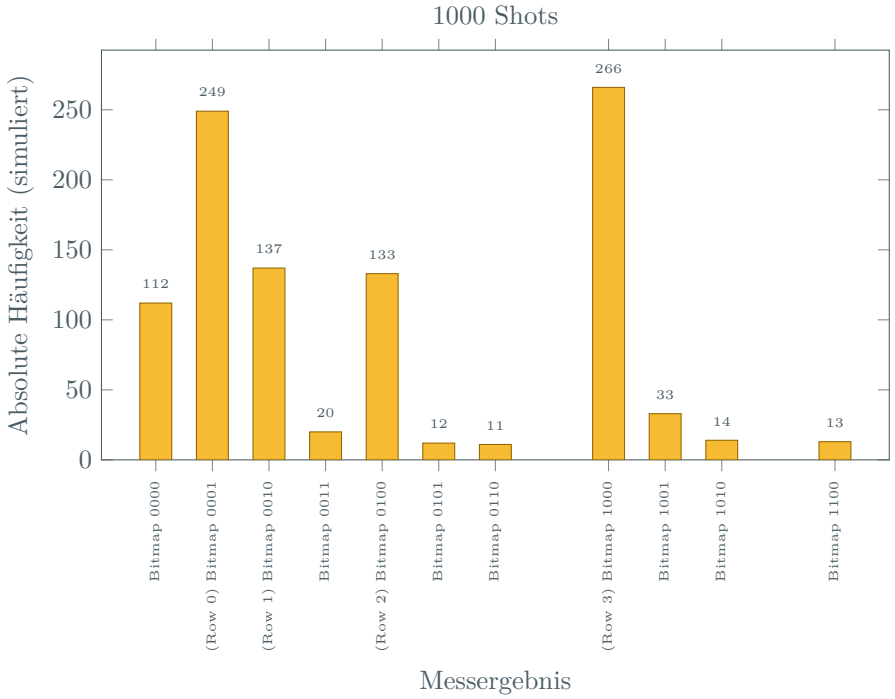


Abbildung 3.29: Ergebnisse der Simulation des QUBOs aus Tabelle 3.4 (EPR als boolesche Funktion).

Der χ^2-Test für die bereinigten simulierten Ergebnisse (Gleichung 3.31) liegt mit $\chi_{\text{empirisch}} = 0,562$ deutlich unter dem kritischen Wert $\chi_{\text{kritisch}} = 3,841$ bei einem Signifikanzniveau von $\alpha = 0,05$ und einem Freiheitsgrad, somit ist für die simulierten Ergebnisse eine Gleichverteilung gegeben.

H_0: Die Messergebnisse passen zur erwarteten Verteilung.

H_A: Die Messergebnisse weichen signifikant von der erwarteten Verteilung ab.

$$\begin{aligned} \chi^2_{\text{empirisch}} &= \frac{(249-258)^2}{258} + \frac{(266-258)^2}{258} \\ &= 0,562 \end{aligned} \tag{3.34}$$

Ausführungen auf Jupsi

Die Ergebnisse der Berechnung auf dem annealing Rechner (Abbildung 3.30) zeigen, dass deutlich häufiger eine optimale Lösung gefunden wird. Der Anteil korrekter Lösungen liegt hier bei $99,1\%$.

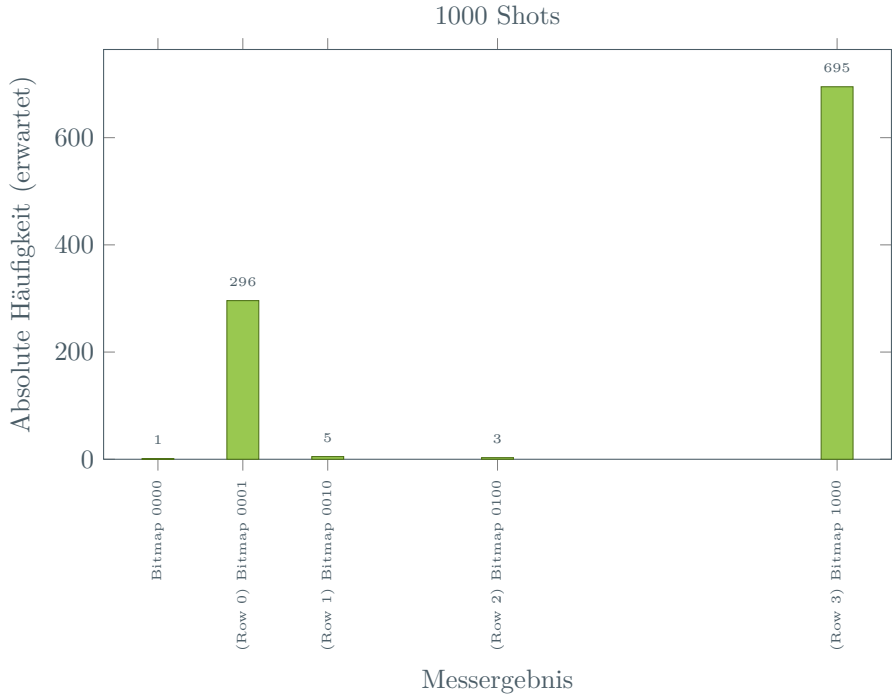


Abbildung 3.30: Erwartete Ergebnisse des QUBOs aus Tabelle 3.4 (EPR als boolesche Funktion).

Der χ^2-Test für die bereinigten Ergebnisse (Gleichung 3.32) liegt mit $\chi_{\text{empirisch}} = 1344,486$ deutlich über dem kritischen Wert $\chi_{\text{kritisch}} = 3,841$ bei einem Signifikanzniveau von $\alpha = 0,05$ und einem Freiheitsgrad. Somit ist für die Ergebnisse keine Gleichverteilung gegeben.

H_0: Die Messergebnisse passen zur erwarteten Verteilung.

H_A: Die Messergebnisse weichen signifikant von der erwarteten Verteilung ab.

$$\begin{aligned} \chi^2_{\text{empirisch}} &= \frac{(296-496)^2}{496} + \frac{(695-496)^2}{496} \\ &= 1344,486 \end{aligned} \tag{3.35}$$

3.2.6 Asymmetrische Funktion

Die Experimente in Unterabschnitt 3.2.1 bis 3.2.5 beruhen auf booleschen Funktionen, die eine Symmetrie aufweisen. Die optimalen Ergebnisse dieser Experimente haben zu den anderen optimalen Ergebnissen jeweils die gleiche Distanz. Das folgende Experiment ist so entworfen, dass diese Symmetrieeigenschaft nicht gegeben ist.

z_i	x_1	x_2	x_3	$f(x_1, x_2, x_3)$
z_0	F	F	F	T
z_1	F	F	T	T
z_2	F	T	F	F
z_3	F	T	T	T
z_4	T	F	F	F
z_5	T	F	T	F
z_6	T	T	F	T
z_7	T	T	T	F

Tabelle 3.5: Wertetabelle der asymmetrischen Funktion $f(x_1, x_2, x_3)$

Tabelle 3.5 zeigt die Wertetabelle der booleschen Funktion. Die unterschiedlichen Distanzen sind in Abbildung 3.31 als gewichteter Graph dargestellt. Drei der optimalen Lösungen liegen mit einer mittleren Distanz von $1,\overline{3}$ und einer maximalen Distanz zwischen zwei Lösungen von 2 näher, als die vierte Lösung, die eine mittleren Distanz von $2,\overline{3}$ und einer maximalen Distanz zwischen zwei Lösungen von 3 weiter entfernt liegt. Da die boolesche Funktion in vier der acht Zeilen T ist, ist diese trotz der Asymmetrie balanciert.

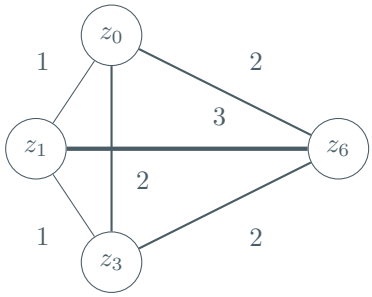


Abbildung 3.31: Distanzen zwischen den optimalen Lösungen des asymmetrischen QUBOs aus Tabelle 3.5.

Implementierung

Die boolesche Funktion kann mit dem Code aus Listing 3.10 in Python angelegt werden. Die resultierende QUBO-Gleichung ist in Gleichung 3.36 aufgestellt.

$$Q = -0,125(z_0 + z_1 + z_3 + z_6) + \sum_{k,l} \frac{\delta_{kl}}{2} z_k z_l \tag{3.36}$$

```python
from fToBqm import f_to_bqm
from prettyPrintBqm import pretty_print_bqm

def asymetric(z_i):
    return (z_i == 0b000 or
            z_i == 0b001 or
            z_i == 0b011 or
            z_i == 0b110)

asymetric_bqm = f_to_bqm(asymetric, 3)
pretty_print_bqm(asymetric_bqm)
```

Listing 3.10: Anlegen des Optimierungsproblems aus Tabelle 3.5 in Python

Erwartete Verteilung

In den folgenden Diagrammen sind nur die Messergebnisse dargestellt, die in der Simulation oder der Ausführung auf dem annealing Rechner mindestens einmal aufgetreten sind. In Abbildung 3.32 ist eine Gleichverteilung über den vier optimalen Lösungen dargestellt. Eine Gleichverteilung wird erwartet, da kein besseres Modell zur Verfügung steht.

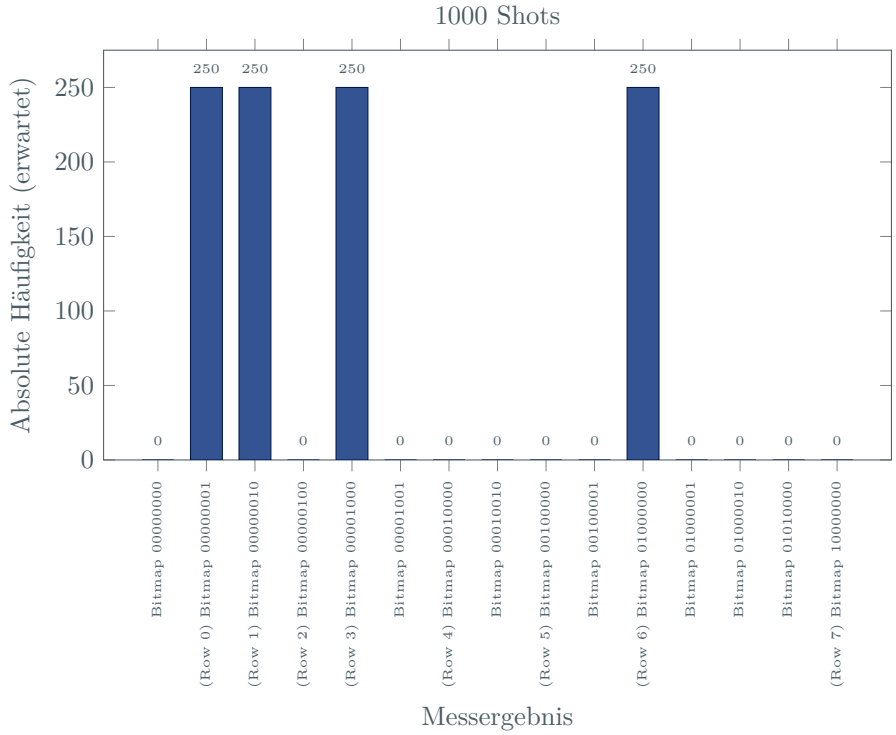


Abbildung 3.32: Ergebnisse der Simulation des QUBOs aus Tabelle 3.3.

Simulation

Die Ergebnisse der Simulation (Abbildung 3.33) zeigen, dass die Annahme einer Gleichverteilung passt. Der Simulator liefert, wie schon in den letzten Experimenten, einige Ergebnisse die nicht zu den optimalen Ergebnissen gehören. Der Anteil korrekter Lösungen liegt bei $64,9\%$.

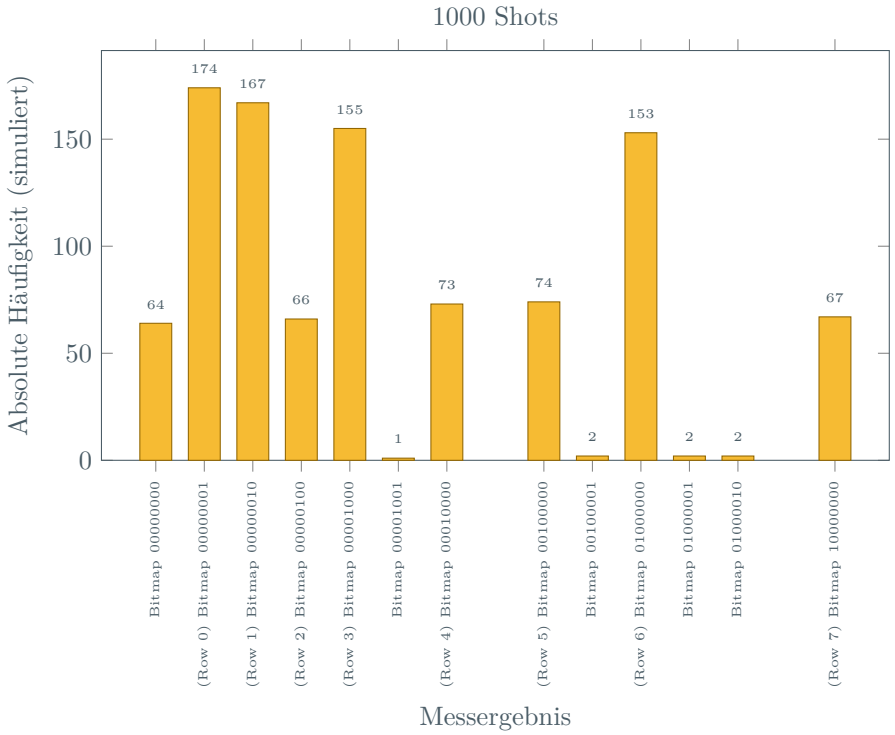


Abbildung 3.33: Ergebnisse der Simulation des asymmetrischen QUBOs aus Tabelle 3.5.

Der χ^2-Test für die bereinigten simulierten Ergebnisse (Gleichung 3.37) liegt mit $\chi_{\text{empirisch}} = 1,846$ deutlich unter dem kritischen Wert $\chi_{\text{kritisch}} = 7,808$ bei einem Signifikanzniveau von $\alpha = 0,05$ und drei Freiheitsgraden. Somit ist für die simulierten Ergebnisse eine Gleichverteilung gegeben.

H_0: Die Messergebnisse passen zur erwarteten Verteilung.

H_A: Die Messergebnisse weichen signifikant von der erwarteten Verteilung ab.

$$\begin{aligned}\chi^2_{\text{empirisch}} &= \frac{(174-162)^2}{162} + \frac{(167-162)^2}{162} + \frac{(155-162)^2}{162} + \frac{(153-162)^2}{162} \\ &= 1,846\end{aligned} \tag{3.37}$$

Ausführungen auf Jupsi

Die Ergebnisse des annealing Rechners sind in Abbildung 3.34 dargestellt. Wie schon in den vorherigen Experimenten liefert der annealing Rechner häufiger die optimalen Lösungen. Der Anteil korrekter Lösungen liegt bei $86,1\%$.

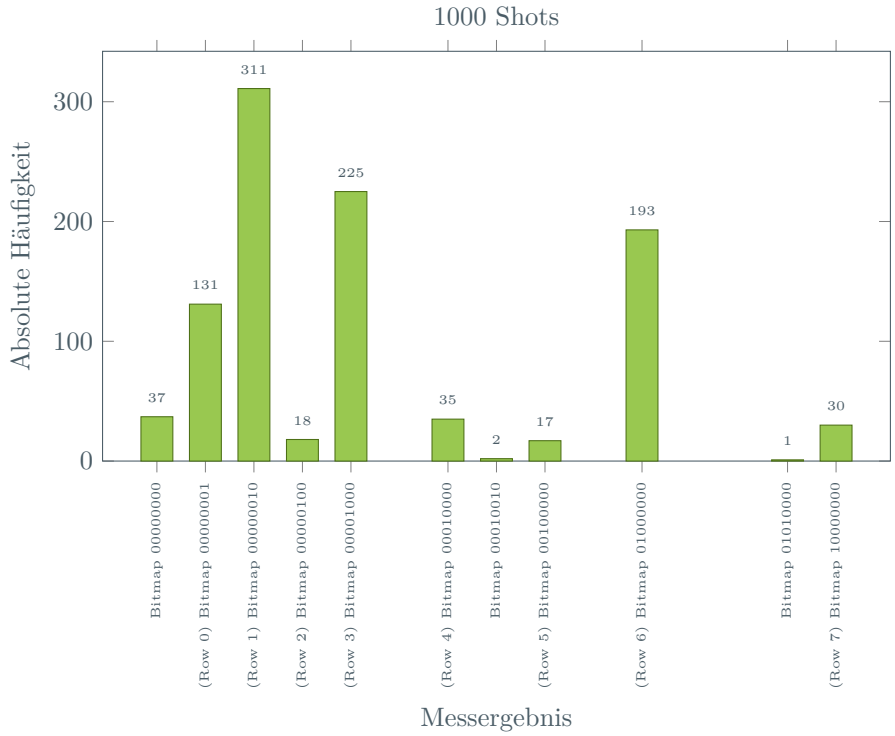


Abbildung 3.34: Ergebnisse der Simulation des asymmetrischen QUBOs aus Tabelle 3.5.

Der χ^2-Test für die bereinigten Ergebnisse (Gleichung 3.38) liegt mit $\chi_{\text{empirisch}} = 90,251$ deutlich über dem kritischen Wert $\chi_{\text{kritisch}} = 7,808$ bei einem Signifikanzniveau von $\alpha = 0,05$ und drei Freiheitsgraden. Somit ist für die Ergebnisse keine Gleichverteilung gegeben.

H_0: Die Messergebnisse passen zur erwarteten Verteilung.

H_A: Die Messergebnisse weichen signifikant von der erwarteten Verteilung ab.

$$\begin{aligned}\chi^2_{\text{empirisch}} &= \frac{(131-215)^2}{215} + \frac{(311-215)^2}{215} + \frac{(225-215)^2}{215} + \frac{(193-215)^2}{215} \\ &= 90,251\end{aligned} \tag{3.38}$$

3.2.7 CHSH-Game

Analog zu den Experimenten der letzten Abschnitte lässt sich aus der Ungleichung des CHSH-Games (Abschnitt 2.9) eine boolesche Funktion aufstellen. In Tabelle 3.6 ist die Wertetabelle der CHSH-Funktion dargestellt. Wie zu erkennen ist, lässt sich die Wertetabelle in vier Bereiche unterteilen. Drei dieser Bereiche sind ähnlich, der vierte Bereich ist invertiert.

z_i	a	b	x	y	$chsh(a,b,x,y)$	z_i	a	b	x	y	$chsh(a,b,x,y)$
z_0	F	F	F	F	T	z_8	T	F	F	F	T
z_1	F	F	F	T	F	z_9	T	F	F	T	F
z_2	F	F	T	F	F	z_{10}	T	F	T	F	F
z_3	F	F	T	T	T	z_{11}	T	F	T	T	T
z_4	F	T	F	F	T	z_{12}	T	T	F	F	F
z_5	F	T	F	T	F	z_{13}	T	T	F	T	T
z_6	F	T	T	F	F	z_{14}	T	T	T	F	T
z_7	F	T	T	T	T	z_{15}	T	T	T	T	F

Tabelle 3.6: Wertetabelle der CHSH-Funktion

Implementierung

Mit dem Code aus Listing 3.11 lässt sich die Funktion als Optimierungsproblem in Python implementieren. Das resultierende Optimierungsproblem ist in Gleichung 3.39 aufgestellt.

$$Q = -0,0625(z_0 + z_3 + z_4 + z_7 + z_8 + z_{11} + z_{13} + z_{14}) + \sum_{k,l} \frac{\delta_{kl}}{2} z_k z_l \quad (3.39)$$

```
from fToBqm import f_to_bqm
from prettyPrintBqm import pretty_print_bqm

def chsh(z_i):
    a = z_i & 0b1000 != 0
    b = z_i & 0b0100 != 0
    x = z_i & 0b0010 != 0
    y = z_i & 0b0001 != 0
    return (a and b) == (x != y)

chsh_bqm = f_to_bqm(chsh, 3)
pretty_print_bqm(chsh_bqm)
```

Listing 3.11: Anlegen des Optimierungsproblems aus Tabelle 3.6 in Python

Erwartete Verteilung

Die erwartete Verteilung ist in Abbildung 3.35 dargestellt. Wie schon im vorherigen Experiment sind nur die Ergebnisse aufgetragen, die mindestens einmal in der erwarteten, simulierten oder gemessenen Verteilung aufgetreten sind.

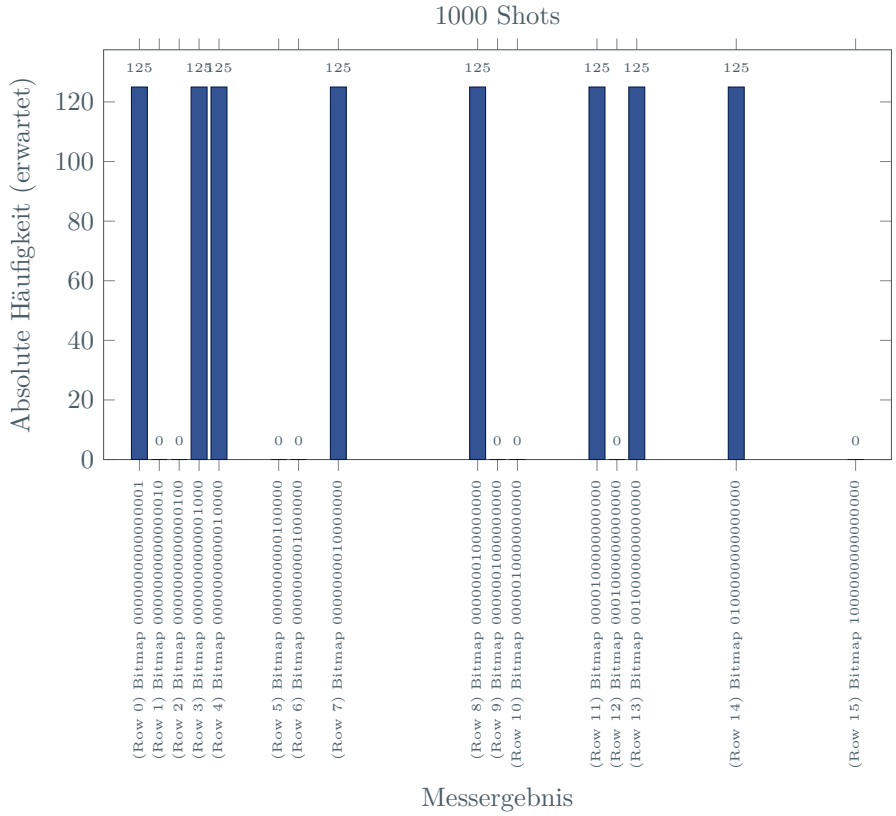


Abbildung 3.35: Erwartete Ergebnisse des QUBOs aus Tabelle 3.6.

Simulation

Abbildung 3.36 zeigt die Verteilung der Simulation. Der Anteil korrekter Ergebnisse liegt bei $69,2\%$.

Der kritische Wert für der χ^2-Verteilung mit sieben Freiheitsgraden liegt bei $\chi_{\text{kritisch}} = 14,067$. Der χ^2-Test für die bereinigten simulierten Ergebnisse (Gleichung 3.40) liegt mit $\chi_{\text{empirisch}} = 13,701$ also im Akzeptanzbereich. Wie schon in den vorherigen Experimenten liefert der Simulator jedoch mit etwa 30% einen hohen Anteil an Lösungen, die nicht zu den optimalen Lösungen gehören.

H_0: Die Messergebnisse passen zur erwarteten Verteilung.

H_A: Die Messergebnisse weichen signifikant von der erwarteten Verteilung ab.

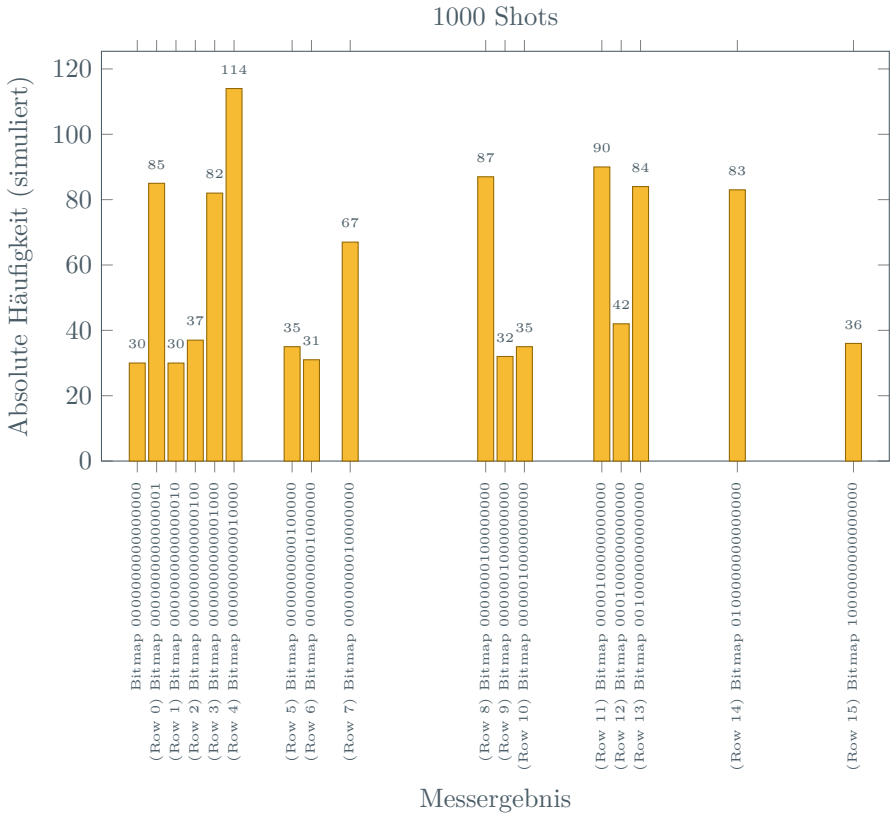


Abbildung 3.36: Ergebnisse der Simulation des QUBOs aus Tabelle 3.6 (CHSH-QUBO).

$$\begin{aligned}\chi^2_{\text{empirisch}} &= \frac{(85-87)^2}{87} + \frac{(82-87)^2}{87} + \frac{(114-87)^2}{87} + \frac{(67-87)^2}{87} \\ &\quad + \frac{(87-87)^2}{87} + \frac{(90-87)^2}{87} + \frac{(84-87)^2}{87} + \frac{(83-87)^2}{87} \\ &= 13,701 \end{aligned} \tag{3.40}$$

Ausführungen auf Jupsi

Im Gegensatz zu den vorherigen Experimenten liefert der annealing Rechner für dieses Optimierungsproblem, mit nur $43,5\%$, erstmals weniger oft einer optimale Lösung (Abbildung 3.37) als der Simulator.

Der χ^2-Test für die bereinigten Ergebnisse (Gleichung 3.41) liegt mit $\chi_{\text{empirisch}} = 769,241$ also deutlich außerhalb des Akzeptanzbereichs.

H_0: Die Messergebnisse passen zur erwarteten Verteilung.

H_A: Die Messergebnisse weichen signifikant von der erwarteten Verteilung ab.

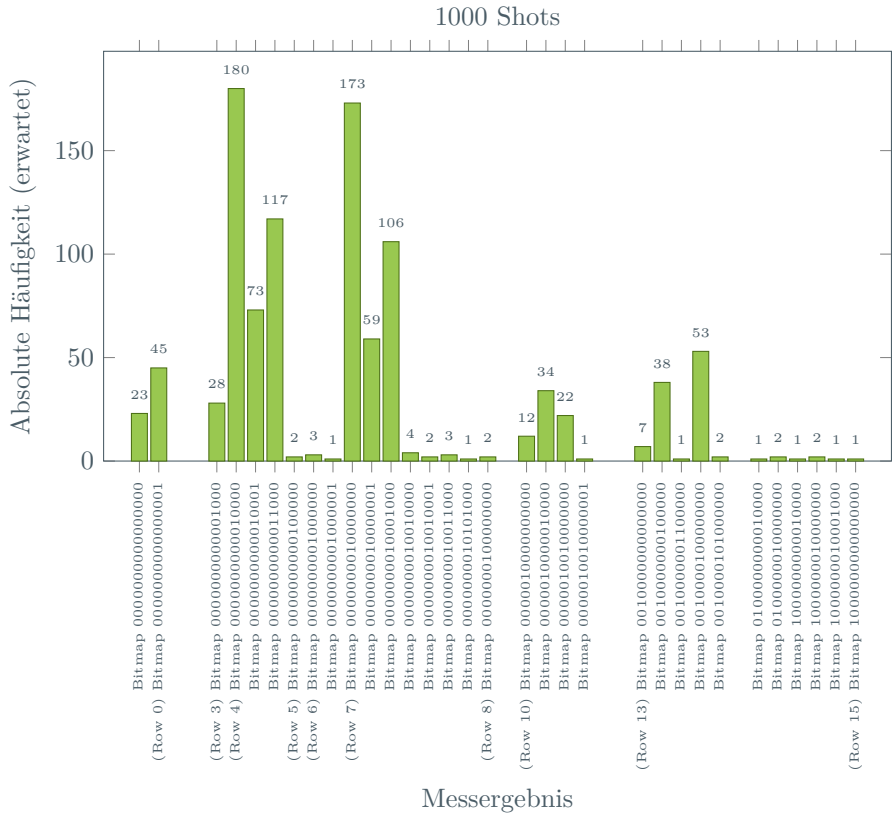


Abbildung 3.37: Erwartete Ergebnisse des QUBOs aus Tabelle 3.6 (CHSH-QUBO).

$$\begin{aligned}\chi^2_{\text{empirisch}} &= \frac{(45-54)^2}{54} + \frac{(28-54)^2}{54} + \frac{(180-54)^2}{54} + \frac{(173-54)^2}{54} \\ &\quad + \frac{(2-54)^2}{54} + \frac{(0-54)^2}{54} + \frac{(7-54)^2}{54} + \frac{(0-54)^2}{54} \\ &= 769,241 \end{aligned} \tag{3.41}$$

4 Fazit

Diese Arbeit untersucht das No-Signaling Theorem auf freizugänglichen Quanten Computern. Im Rahmen dieser Arbeit konnte gezeigt werden, dass das Quantum Cube Model erweitert werden kann, um die Sicht auf Teilsystemen darzustellen. Diese Darstellung kann dazu verwendet werden, das No-Signaling Theorem zu visualisieren.

Zur Untersuchung des No-Signaling Theorems wurden Experimente auf IBM Quanten Rechnern durchgeführt. Die Ergebnisse dieser Experimente zeigen, dass Quanten Rechner schon deutlich vor einer möglichen Verletzung des No-Signaling Theorems fehlerhafte Ergebnisse auftreten. So konnte gezeigt werden, dass bereits bei Schaltkreisen mit zwei Qubits und zwei Gattern, nicht die erwarteten Verteilungen erzielt werden.

Die Ergebnisse des CHSH-Game Schaltkreises zeigen, dass die Fehleranfälligkeit des IBM Quanten Rechners so groß ist, dass die Ungleichung seltener erfüllt ist, als dies bei einer optimalen klassischen Strategie der Fall ist.

Darüber hinaus ist eine neue Methode vorgestellt worden, boolesche Funktionen als Optimierungsprobleme zu beschreiben. Diese Methode kann verwendet werden, um die Performance von annealing Rechnern zu messen. Im Rahmen dieser Arbeit wurden boolesche Funktionen mit bis zu vier Variablen als QUBO formuliert.

Die Ergebnisse zeigen, dass für die durchgeführten Experimente der Simulator die erwartete Gleichverteilung über den optimalen Lösungen liefert. Überraschend ist, dass bei Ausführung der ersten Experimente auf Jupsi der Anteil optimaler Lösungen höher ist, als bei dem Simulator, diese jedoch nicht der erwarteten Gleichverteilung folgen. Für das letzte Experiment (CHSH-QUBO) ändert sich diese Beobachtung, Der Simulator erreicht in diesem Fall häufiger korrekte Ergebnisse, als der annealing Rechner.

In den Ergebnissen der annealing Experimente ist ein weiterer Trend erkennbar. Unter den nicht optimalen Lösungen tritt das Ergebnis „Bitmap 0…0“ häufiger auf, als andere Lösungen mit gleichem Wert. Dieser Trend ist über alle untersuchten Experimente hinweg, sowohl in den Ergebnissen der Simulationen, als auch in den Ergebnissen des annealing Rechners, beobachtbar.

Diese Arbeit liefert drei sehr unterschiedliche Erkenntnisse:

1. Das Quantum Cube Model ist noch leistungsfähiger als bisher bekannt, so lässt sich die No-Signaling Eigenschaft von Quantensystemen graphisch mit dem Quantum Cube Model darstellen.

2. Zeitgleich mit dieser Arbeit zeigen Rybotycki u.a. 2025[1], dass Quanten Rechner die No-Signaling Eigenschaft in bestimmten Fällen brechen. In dieser Arbeit wird gezeigt, dass schon bei einfachen Schaltkreisen fehlerhafte Ergebnisse erzeugt werden, die nicht den Erwartungen entsprechen. Aussagen zu der No-Signaling Eigenschaft lassen sich so nicht zuverlässig treffen.

[1] Tomasz Rybotycki u. a. „Violation of No-Signaling on a Public Quantum Computer“. In: *Advanced Quantum Technologies* 8.6 (Juni 2025), S. 2400661. ISSN: 2511-9044, 2511-9044. DOI: 10.1002/qute.202400661. arXiv: 2409.11348 [quant-ph]. URL: http://arxiv.org/abs/2409.11348 (besucht am 07. 08. 2025).

3. Diese Arbeit liefert erstmals ein Verfahren beliebige boolesche Funktionen als Optimierungsprobleme zu beschreiben. So lassen sich Optimierungsprobleme mit bestimmten Eigenschaften, wie einer großen Anzahl gleichwertiger Lösungen, erzeugen. Ein Optimierungsproblem auf Basis einer booleschen Funktion, die durch die Ungleichung des CHSH-Games inspiriert ist, zeigt in der Auswertung der Experimente ist eine Diskrepanz zwischen den Lösungen des annealing Rechners von D-Wave, und dem Dimod Simulator. So ist der Anteil der optimalen Lösungen auf dem annealing Rechner deutlich höher, als in der Simulation. Die gemessenen Verteilungen weichen jedoch schon bei kleinen Optimierungsproblemen ausschließlich auf dem annealing Rechner von der erwarteten Verteilung ab, wenn Funktionen mit mehren gleichwertigen Lösungen berechnet werden.

In den folgenden Semestern wir die neue Darstellung des No-Signaling Theorems in Vorlesungen zum Quanten Computing an der THM eingesetzt. Dabei wird das Verständnis der Teilnehmer, zu den Aussagen des No-Signaling Theorems, evaluiert.

Im letzten Jahr ist der freie Zugang zu Quanten Computern zunehmend eingeschränkt worden. So war noch vor einem Jahr die Registrierung in der IBM Cloud ohne Zahlungsmethoden möglich, und bis zu zehn Minuten Rechenzeit kostenlos verfügbar. Dieser Zugang ist zum aktuellen Zeitpunkt nicht möglich. Gleiches gilt für die Cloud von D-Wave. Auch diese stellt zum aktuellen Zeitpunkt keine freie Rechenzeit zur Verfügung.

Mit dieser Entwicklung ist es notwendig, dass die verfügbaren Simulatoren möglichst genau die Hardware abbilden, auf die der Zugang nicht mehr möglich ist. Die Ergebnisse der annealing Experimente zeigen jedoch eine deutliche Diskrepanz zwischen Simulator und realer Hardware.

Für zukünftige Arbeiten ergeben sich zwei maßgebliche Fragestellungen, die im Rahmen dieser Arbeit nicht beantwortet werden konnten. Die Optimierungsprobleme, die durch das neu beschriebene Verfahren aus booleschen Funktionen aufgebaut werden, lassen sich mit QAOA-Algorithmen auf Schaltkreis Rechnern lösen. Die Qualität der QAOA Ergebnisse kann dann mit denen des annealing Rechners verglichen werden. Weiterhin ist die Frage, wie alle gleichwertigen optimalen Lösungen auf annealing Rechnern zuverlässig gefunden werden können, von besonderem Interesse.

Literatur

abbycross. *CHSH Game*. IBM Quantum Learning. Aug. 2025. URL: https://quantum.cloud.ibm.com/learning/en/courses/basics-of-quantum-information/entanglement-in-action/quantum.cloud.ibm.com/learning/en/courses/basics-of-quantum-information/entanglement-in-action/chsh-game (besucht am 27. 08. 2025).

Einführung in Das Quantencomputing - Teil 3. openHPI. URL: https://open.hpi.de/courses/qc-intro-3-2023 (besucht am 16. 08. 2025).

Gallus, Christoph, Pawel Blasiak und Emmanuel M. Pothos. „Winning a CHSH Game without Entangled Particles in a Finite Number of Biased Rounds: How Much Luck Is Needed?“ In: *Entropy* 25.5 (21. Mai 2023), S. 824. ISSN: 1099-4300. DOI: 10.3390/e25050824. URL: https://www.mdpi.com/1099-4300/25/5/824 (besucht am 11. 04. 2025).

Glover, Fred, Gary Kochenberger und Yu Du. „Quantum Bridge Analytics I: A Tutorial on Formulating and Using QUBO Models“. In: (4. Nov. 2019). DOI: 10.48550/arXiv.1811.11538. URL: https://arxiv.org/abs/1811.11538 (besucht am 30. 08. 2025).

Hajek, Bruce. „Cooling Schedules for Optimal Annealing“. In: *Math. Oper. Res.* 13.2 (1. Mai 1988), S. 311–329. ISSN: 0364-765X.

How Cold Is It? D-Wave Quantum Inc. 30. Apr. 2018. URL: https://support.dwavesys.com/hc/en-us/articles/360003681034-How-Cold-Is-It (besucht am 16. 08. 2025).

IBM Quantum System Two: The Era of Quantum Utility Is Here | IBM Quantum Computing Blog. URL: https://www.ibm.com/quantum/blog/quantum-roadmap-2033 (besucht am 16. 08. 2025).

Jiang, Shuxian u. a. „Quantum Annealing for Prime Factorization“. In: *Scientific Reports* 8.1 (5. Dez. 2018), S. 17667. ISSN: 2045-2322. DOI: 10.1038/s41598-018-36058-z. URL: https://www.nature.com/articles/s41598-018-36058-z (besucht am 15. 09. 2025).

Just, Bettina. *Quantencomputing kompakt: Spukhafte Fernwirkung und Teleportation endlich verständlich*. IT kompakt. Berlin, Heidelberg: Springer, 2020. ISBN: 978-3-662-61888-2 978-3-662-61889-9. DOI: 10.1007/978-3-662-61889-9. URL: https://link.springer.com/10.1007/978-3-662-61889-9 (besucht am 05. 06. 2025).

Nielsen, Michael A. und Isaac L. Chuang. *Quantum Computation and Quantum Information*. 10th anniversary ed. Cambridge ; New York: Cambridge University Press, 2010. 676 S. ISBN: 978-1-107-00217-3.

Rybotycki, Tomasz u. a. „Violation of No-Signaling on a Public Quantum Computer“. In: *Advanced Quantum Technologies* 8.6 (Juni 2025), S. 2400661. ISSN: 2511-9044, 2511-9044. DOI: 10.1002/qute.202400661. arXiv: 2409.11348 [quant-ph]. URL: http://arxiv.org/abs/2409.11348 (besucht am 07. 08. 2025).

Schacht, Cedric. *CedricSchacht/Quantum-Learing-App*. Version 1.0.0+1. 20. Juni 2024. URL: https://github.com/CedricSchacht/quantum-learing-app (besucht am 05. 06. 2025).

– „Plattformübergreifende Anwendung Zur Darstellung von Quantenschaltkreisen Mit Proof of Concept Für Phase Kickback“. Technische Hochschule Mittelhessen, 30. Sep. 2023. URL: https://github.com/CedricSchacht/quantum-learing-app/blob/main/bachelor-thesis.pdf (besucht am 26. 08. 2025).

Shannon, C. E. „A Mathematical Theory of Communication“. In: *The Bell System Technical Journal* 27.3 (Juli 1948), S. 379–423. ISSN: 0005-8580. DOI: 10.1002/j.1538-7305.1948.tb01338.x. URL: https://ieeexplore.ieee.org/document/6773024/ (besucht am 16. 08. 2025).

Shor, Peter W. „Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer“. In: *SIAM Journal on Computing* 26.5 (Okt. 1997), S. 1484–1509. ISSN: 0097-5397, 1095-7111. DOI: 10.1137/S0097539795293172. arXiv: quant-ph/9508027. URL: http://arxiv.org/abs/quant-ph/9508027 (besucht am 15. 09. 2025).

Sipser, Michael. *Introduction to the Theory of Computation*. preliminary ed. The PWS Series in Computer Science. Boston, Mass. Bonn: PWS Publ, 1996. 239 S. ISBN: 978-0-534-95250-1.

The Advantage™ Quantum Computer | D-Wave. URL: https://www.dwavequantum.com/solutions-and-products/systems/ (besucht am 16. 08. 2025).

Yin, Juan u. a. „Bounding the Speed of ‘spooky Action at a Distance’“. In: *Physical Review Letters* 110.26 (26. Juni 2013), S. 260407. ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.110.260407. arXiv: 1303.0614 [quant-ph]. URL: http://arxiv.org/abs/1303.0614 (besucht am 16. 08. 2025).

A: Matrizen der Gatter

$$H = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix} \qquad X = \begin{bmatrix}0 & 1\\ 1 & 0\end{bmatrix} \qquad R_x(\theta) = \begin{bmatrix}\cos\left(\frac{\theta}{2}\right) & -i\sin\left(\frac{\theta}{2}\right)\\ -i\sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right)\end{bmatrix}$$

$$S = \begin{bmatrix}1 & 0\\ 0 & i\end{bmatrix} \qquad Y = \begin{bmatrix}0 & -i\\ i & 0\end{bmatrix} \qquad R_y(\theta) = \begin{bmatrix}\cos\left(\frac{\theta}{2}\right) & -\sin\left(\frac{\theta}{2}\right)\\ \sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right)\end{bmatrix}$$

$$T = \begin{bmatrix}1 & 0\\ 0 & e^{i\pi/4}\end{bmatrix} \qquad Z = \begin{bmatrix}1 & 0\\ 0 & -1\end{bmatrix} \qquad R_z(\theta) = \begin{bmatrix}e^{-i\theta/2} & 0\\ 0 & e^{i\theta/2}\end{bmatrix}$$

$$\text{CNOT} = \begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1\\ 0 & 0 & 1 & 0\end{bmatrix} \qquad \text{Controlled-U (CU)} = \begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & u_{00} & u_{01}\\ 0 & 0 & u_{10} & u_{11}\end{bmatrix}$$

$$\text{Toffoli (CCNOT)} = \begin{bmatrix}1 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1\\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0\end{bmatrix}$$

B: BQM Utils

```python
from typing import Callable
import itertools
from dimod import BinaryQuadraticModel
from dimod import BINARY
def f_to_bqm(f: Callable[[int], bool], bits: int) ->
    BinaryQuadraticModel:

    zs = [i for i in range(2**bits)]
    linear = dict([
        (f"z_{z}", -1 if f(z) else 0) for z in zs
    ])
    quadratic = dict([
        ((f"z_{z_i}", f"z_{z_j}"), 2**bits)
        for z_i, z_j
        in itertools.product(zs, zs)
        if z_i > z_j
    ])
    bqm = BinaryQuadraticModel(
        linear,
        quadratic,
        0.0,
        BINARY
    )
    bqm.normalize()
    return bqm
```

Listing B.1: f_to_bqm

```python
from dimod import BinaryQuadraticModel
def pretty_print_bqm(bqm: BinaryQuadraticModel):
    print("Binary Quadratic Model:")
    print("-" * 30)

    print(f"Variables: (Type: {bqm.vartype.name})")
    for v in bqm.variables:
        print(f"  {v}")

    print("\nLinear biases:")
    for v, bias in bqm.linear.items():
        print(f"  {v}: {bias}")

    print("\nQuadratic biases:")
    for (u, v), bias in bqm.quadratic.items():
        print(f"  ({u}, {v}): {bias}")

    print(f"\nOffset: {bqm.offset}")
    print("-" * 30)
```

Listing B.2: pretty_print_bqm