

Abschlussbericht Dezember 2022 (Projektnummer 3-4332.62-IAO/71)

im Rahmen der ersten Förderphase des
»Kompetenzzentrum Quantencomputing Baden-Württemberg«



Software-Engineering industrieller, hybrider Quantenanwendungen und -algorithmen

Konsortialführung und koordinierendes Fraunhofer-Institut:



Fraunhofer-Institut für Arbeitswirtschaft und Organisation IAO

Prof. Dr.-Ing. Oliver Riedel, Prof. Dr.-Ing. Anette Weisbecker

Projektleitung und Koordination: Dr. Christian Tutschku

Autoren: Dr. Christian Tutschku, Dr. Florian Knäble, Dr. Andreas Sturm,
Dr. Bharadwaj Mummaneni, Daniel Mayer, Yeama Bangali

In Kooperation mit:



Fraunhofer-Institut für Produktionstechnik und Automatisierung IPA

Prof. Dr.-Ing. Marco Huber

Autoren: Dr. Marco Roth, Dr. Paul-Amaury Matt, Arthur Grigorjan, Philipp Wagner



Fraunhofer-Institut für Angewandte Festkörperphysik IAF

Prof. Dr. Rüdiger Quay

Autorin: PD Dr. Thomas Wellens, Kathrin König



FZI Forschungszentrum Informatik

Prof. Dr. rer. nat. Ralf H. Reussner

Autor: Oliver Denniger, Max Scheerer, Jonas Klamroth

Universität Stuttgart



Institut für Architektur von Anwendungssystemen (IAAS)

Prof. Dr. Dr. h. c. Frank Leymann

Autoren: Martin Beisel, Felix Truger



Höchstleistungsrechenzentrum Stuttgart (HLRS)

Prof. Dr.-Ing. Dr. h.c. Dr. h.c. Prof. E.h. Michael M. Resch

Autoren: Dr. Tobias Haas, Dr. Dmitry Khabi, Dr. Farida Shagieva,
Li Zhong



Eberhard Karls Universität Tübingen (EKUT),

Lehrstuhl Eingebettete Systeme

Prof. Dr. Oliver Bringmann

Autor: Simon Garhofer

Inhaltsverzeichnis

1	Projektstruktur und Zusammenfassung.....	3
2	Überblick der wissenschaftlichen Arbeiten	7
2.1	Arbeitspaket 1: Anwendungsfelder und Potenziale des Quantencomputing.....	7
2.2	Arbeitspaket 2: Spezifikation, Werkzeuge und Schnittstellen.....	10
2.3	Arbeitspaket 3: Fehlermitigation und Zuverlässigkeit	16
2.4	Arbeitspakete 4, 5: Hybride Quantenalgorithmen für industrielle Anwendung	23
2.5	Arbeitspaket 6: Quantensoftware-Engineering	50
2.6	Arbeitspaket 7: Transfer und Partnerkooperation	54
2.7	Arbeitspaket 8: Projektmanagement	60
3	Key-Performance-Indikatoren.....	61
3.1	Wissenschaftliche Publikationen	61
3.2	Im Themenfeld Quantencomputing geschulte Personen	63

1 Projektstruktur und Zusammenfassung

Das Projekt SEQUOIA steht für Software-Engineering industrieller, hybrider Quantenanwendungen und -algorithmen. Basierend auf konkreten Anwendungsfällen aus den Branchen Produktion und Entwicklung, Logistik und Mobilität, Engineering, Automotive, Energiewirtschaft, Finanzen und Versicherungen, Telekommunikation und Immobilienwirtschaft (siehe Tabelle 1) zielte das Verbundforschungsprojekt SEQUOIA auf die Erforschung, Entwicklung und Erprobung neuer Methoden, Werkzeuge und Vorgehensweisen für Quantencomputing in der industriellen Applikation.

	Optimierung	ML/KI	Simulation
Automotive	Szenario-basierte Routen-Planung zur Absicherung von Fahrfunktionen		
Produktion und Entwicklung	- Auftragssequenzierung - Optimale Schachtelung von Blechteilen - Parameteroptimierung für automatisierte Layout-Planung - Optimierung der Distributionskette von Medikamenten - Zusammensetzen einzelner Bauteile zur Toleranzminimierung - Modellvalidierung für Produktkonfigurationen - Platzierung von Arbeitsstationen - Anomalieerkennung	- Formale Verifikation neuronaler Netze - Produktqualitätssicherung durch Identifizierung von defekten Bauteilen - Klassifikation mit Kernel-Methoden - IT/OT-Sicherheit - Design Space Exploration/Edge AI	- Numerische Strömungssimulation - Prozesseffizienzsteigerung in der Tablettierung
Finanzen und Versicherungen		- Betrugserkennung - Anomalieerkennung	
Logistik und Mobilität	- Versandkartonagenauswahl - Real-Time Supply Chain Management - Verkehrsfluss-Optimierung		
Telekommunikation	Optimierung drahtloser Netzwerke		
Immobilienwirtschaft	- Standortbewertung - Planung für Kabelverlegung		
Energiewirtschaft	Ressourcenoptimierung in der Elektromobilität		

Tabelle 1: Die Anwendungsfälle des SEQUOIA-Unternehmensnetzwerkes.

Dazu gehörten Quantenalgorithmen aus dem Optimierungs-, ML/KI- und dem Simulationsbereich sowie Technologien für den Quantensoftware-Entwicklungsprozess und die (automatisierte) Ausführung von Quantenschaltkreisen. Dies beinhaltete insbesondere Verfahren zur Fehlermitigation und standardisierte Programmierschnittstellen (Application Programming Interfaces - APIs).

Diese Technologien wurden in hybriden Lösungen angewendet, welche Quantenalgorithmen mit klassischen Lösungselementen in der NISQ Ära kombinieren (noisy intermediate-scale quantum computer).

Herzstück des Forschungsprojektes war das Umsetzen und Demonstrieren von industriellen Anwendungsfällen auf dem IBM Quantum System One in Ehningen, dem ersten kommerziell nutzbaren gatterbasierten Quantencomputer in Europa. Wie in Abbildung 1 gezeigt, wurde dazu ein Anwendungszentrum bestehend aus 28 Unternehmen aufgebaut (vgl. Abbildung 3), deren Use Cases gesammelt, bewertet und klassifiziert wurden (vgl. Tabelle 1). Anschließend wurden diese Anwendungsfälle nach wirtschaftlichen und wissenschaftlichen Kriterien bewertet und im engen Austausch mit dem Unternehmensbeirat sechs sogenannte Fokus Use Cases definiert, die – als wesentliches Projektziel – mittels hybrider Quantenalgorithmen umgesetzt, getestet bzw. simuliert und auf dem IBM Quantum System One in Ehningen demonstriert wurden.

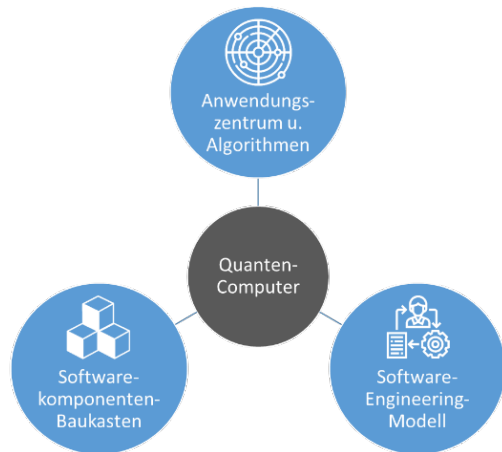


Abbildung 1 Wesentlicher Aufbau und Ziele des Verbundforschungsprojektes SEQUOIA.

Die entwickelten Algorithmen, Werkzeuge und Methoden sind im Projekt kontinuierlich in den sogenannten SEQUOIA Software-Komponenten Baukasten eingeflossen. Ziel dieses Vorgehens war es im Projektnachgang auch konventionellen Entwicklern zu ermöglichen high-level Quantenlösungen zu konstruieren ohne beispielsweise Fehlermitigationsstrategien, Schaltkreisgenerierung oder Quantenalgorithmus-Konzepte im Detail verstehen zu müssen. Weiterhin befähigt der Baukasten auch Experten dazu, schneller und effizienter Quantenlösungen für industrielle Use Cases zu konstruieren und auszuführen. Jene Toolings wurden über entsprechenden Repositorien open source zur Verfügung gestellt. Sie finden sich unter <https://github.com/UST-QuAntil>, <https://se-quoia.iao.fraunhofer.de> bzw. <https://github.com/fzi-forschungszentrum-informatik/se-quoia-gsw22>. Detaillierte Beschreibungen bzw. dezidierte Links der jeweiligen Toolings sind den Kapiteln 2.2, 2.3 bzw. 2.5 zu entnehmen.

Es ist besonders hervorzuheben, dass die entwickelten hybriden Lösungen der Fokus Use Cases sowie weitere applikationszentrierte Quantenalgorithmen als selbsterklärende, interaktive Jupyter Notebooks im »Kompetenzzentrum Quantencomputing Baden-Württemberg« (KQCBW) über die Recheninfrastruktur des Fraunhofer IAO unter der URL <https://sequoia.iao.fraunhofer.de> der breiten Öffentlichkeit niederschwellig und Open Source zur Verfügung gestellt worden sind. Eine Kurzbeschreibung zu jedem einzelnen Fokus Use Case und das jeweilige Jupyter Notebook kann unter <https://websites.fraunhofer.de/sequoia/use-cases/> aufgerufen werden.

Die in Kapitel 2.4 beschriebenen applikationszentrierten Quantenalgorithmen sowie die hierbei erarbeiteten hybriden Lösungen der Fokus Use Cases wurden unter Nutzung der am Fraunhofer IAO zur Verfügung stehenden GPU basierten Hochleistungsrechner (GPU HPC-Cluster) entwickelt. Darüber hinaus wurde für das Projekt »SEQUOIA« eine virtuelle Maschine (VM) auf einem der Fraunhofer IAO Server installiert, die im Rahmen des KQCBW Startprojekts durch Zuwendung des Landes Baden-Württemberg speziell für Quantencomputing Projekte eingerichtet wurden. Auf dieser VM ist ein JupyterHub mit entsprechenden Python-Paketen konfiguriert worden, sodass dort interaktive Jupyter Notebooks ausgeführt werden können. Über einen, sich auf der VM befindlichen, »nginx« Webserver sind unter der Subdomain <https://sequoia.iao.fraunhofer.de> die Jupyter Notebooks für die Öffentlichkeit (ab 2023) interaktiv zugänglich. Dies bedeutet, dass die im Projekt erstellten hybriden Lösungen zu Demonstrationszwecken leicht über die genannte URL online sowie open source abzurufen sind. Ferner ist durch die VM eine Software technische Kapselung sowohl im Hinblick auf die IT-Sicherheit als auch im Hinblick auf Skalierbarkeit gegeben. Dadurch können zukünftig die Ressourcen der VM in Abhängigkeit der Auslastung dynamisch angepasst werden.

Des Weiteren ist neben dem im Internet verfügbaren JupyterHub eine SEQUOIA Projektwebseite auf der Webapplikationsfarm der Fraunhofer-Gesellschaft gehostet. Diese ist gegenwärtig unter der Webadresse <https://websites.fraunhofer.de/sequoia> erreichbar und bietet der Öffentlichkeit diverse Informationen über das Projekt. Da die Webseite mithilfe des frei verfügbaren Content-Management-Systems (CMS) »WordPress« erstellt wurde, ist es aufgrund der Benutzerfreundlichkeit des Systems auch für Autoren auf einfache Art und Weise möglich Inhalte in die Internetseite einzupflegen, ohne hierzu zwingend Erfahrungen in Webdesign und -entwicklung zu haben. Auch in Zukunft wird die SEQUOIA Projektwebseite kontinuierlich um weitere Inhalte ergänzt werden.

Projektstruktur und
Zusammenfassung

Als dritte wesentliche Säule des Projektes sind alle (Teil-)Ergebnisse des Forschungsfortschrittes im SEQUOIA Software-Engineering-Modell zusammengefasst, welches Best Practices für die Entwicklung und das Umsetzen von Quantensoftware definiert. Diese Erkenntnisse wurden im Rahmen der SEQUOIA Anwenderstudie ebenfalls einem breiten Publikum vermittelt.

Jene Studie sowie alle weiteren Projektergebnisse sind dem Open Source Gedanken folgend der Öffentlichkeit über unsere eigens entwickelte SEQUOIA Homepage (<https://websites.fraunhofer.de/sequoia>) zugänglich gemacht worden.

Ziel dieses Berichtes ist es mit Abschluss der ersten Phase im KQCBW und damit einhergehend mit Abschluss des SEQUOIA Verbundvorhabens die erreichten Projektergebnisse zu präsentieren und dabei auf alle Projektmeilensteine einzugehen.

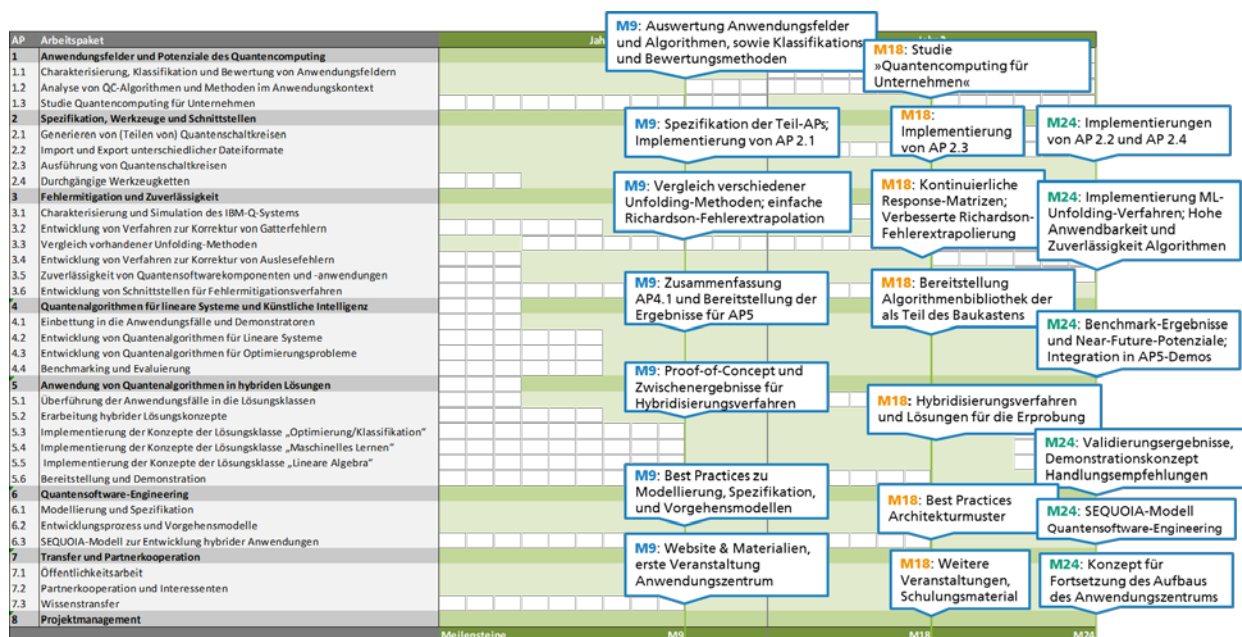


Abbildung 2 Gantt-Diagramm und Meilensteine des Verbundforschungsprojektes SEQUOIA.

Abbildung 2 zeigt das dem Projekt zugrunde liegende Gantt-Diagramm, auf dessen Arbeitspaket (AP) Ergebnisse im Folgenden explizit eingegangen wird.

Zusammenfassend sei bereits an dieser Stelle gesagt, dass alle Meilensteine wie in Abbildung 2 aufgeführt, eingehalten wurden und der Projektverlauf entsprechend der Planung durchgeführt wurde.

Es sei betont, dass alle Projektergebnisse sowohl dem SEQUOIA-Anwendungszentrum (Unternehmensnetzwerk), als auch der breiten (Fach-)Öffentlichkeit im Rahmen von zahlreichen Veranstaltungen, Fachbeiträgen, dem KQCBW Schulungsprogramm (Kapitel 2.6) und, wie bereits erwähnt, mittels der SEQUOIA-Anwenderstudie und den interaktiven Use Case Demonstratoren zugänglich gemacht worden sind (siehe SEQUOIA Webseite <https://websites.fraunhofer.de/sequoia>).

An dieser Stelle soll schon vorab auf die **im Zuwendungsbescheid geforderten Kennzahlen** eingegangen werden. Diese werden in Kapitel 2.6 (Veranstaltungen), 3.1 (Publikationen) und 3.2 (geschulte Personen) ausführlich diskutiert:

Projektbezogene wissenschaftliche Publikationen

15 veröffentlicht | **9** In Vorbereitung: Insgesamt **24 Publikationen**

Projektbezogene Patentanmeldungen:

Es wurden im Projekt keine Patente angemeldet.

Anzahl der im Themenfeld Quantencomputing geschulten Personen:

Ca. **6400** Teilnehmende bei einzelnen SEQUOIA Weiterbildungsveranstaltungen

Ca. **80** Teilnehmende im einjährigen KQCBW Schulungsprogramm (23 Termine à 3h)

Anzahl (insgesamt) der am Projekt beteiligten Unternehmen im abgelaufenen Kalenderjahr (2022):

Fokus Use Case Partner:

Simerics GmbH | Trumpf Werkzeugmaschinen GmbH + Co. KG | Kumavision AG | Bertrand AG | Schwarz IT KG

Technischer Austausch und Support

Pfizer Pharma GmbH | IBM Deutschland GmbH | StoneOne AG

Unternehmensbeirat:

Kumavision AG | Leichtbau BW GmbH | Schwarz IT KG | StoneOne AG | HQS Quantum Simulations GmbH | Trumpf Werkzeugmaschinen GmbH + Co. KG

Insgesamte Anzahl: **10 Unternehmen**, darin enthaltene KMUs: **3 Unternehmen**

Simerics GmbH | StoneOne AG | HQS Quantum Simulations GmbH

2 Überblick der wissenschaftlichen Arbeiten

2.1 Arbeitspaket 1: Anwendungsfelder und Potenziale des Quantencomputing

Charakterisierung, Klassifikation und Bewertung von Anwendungsfeldern

Zu Beginn des Projektes wurde ein Unternehmensnetzwerk an assoziierten Projektpartnern aufgebaut, das über die Projektphase kontinuierlich gewachsen ist. In der SE-QUOIA Auftaktveranstaltung mit über 270 Teilnehmern wurde der derzeitige Stand der Forschung sowie die Entwicklungspotenziale für Unternehmen erörtert. Außerdem wurden über die Projektlaufzeit insgesamt drei Unternehmensnetzwerk- sowie zwei Unternehmensbeiratsveranstaltungen durchgeführt. Im Rahmen dieser Veranstaltungen wurden die industriellen Anwendungsfälle, die in Tabelle 1 abgebildet sind, von den Konsortialpartnern gesammelt, charakterisiert, klassifiziert und in ihrer Machbarkeit bewertet. Es wurde der Projektfortschritt in die Unternehmen transferiert, sowie – mit dem Unternehmensbeirat zusammen – der Fokus des Projekts sichergestellt und gelenkt. Mitte Januar 2023 ist als Projektabschluss sowohl ein weiteres Unternehmensnetzwerk- als auch ein abschließendes -beiratstreffen in Planung, in welchem die finalen Projektergebnisse (Studie, Demonstratoren, Vorgehensmodell, Homepage, Services, ...) vorgestellt und diskutiert werden.

Abbildung 3 zeigt das SEQUOIA-Unternehmensnetzwerk bestehend aus 28 Unternehmen, welche die Wichtigkeit des Projekts und der entsprechenden Inhalte mit einem »Letter of Intent« hervorgehoben haben. Es ist anzumerken, dass basierend auf dem hiesigen Projekterfolg, jene Bestätigungen für den Projektfolgeantrag SEQUOIA End-to-End erneut mittels »Letter of Intent« unterstrichen worden sind.



Abbildung 3 SEQUOIA-Unternehmensnetzwerk – 28 LOI Partner

In zahlreichen bilateralen Unternehmensdiskussionen wurden für die in Tabelle 1 aufgeführten Use Cases Steckbriefe erstellt, die den jeweiligen Anwendungsfall übersichtlich skizzieren, die konventionelle Herangehensweise sowie Limitierung aufzeigen und potenzielle Quantenlösungen vorschlagen. Die erarbeiteten Ergebnisse wurden im Rahmen der Anwenderstudie dem industriellen Umfeld verfügbar gemacht.

Analyse von QC-Algorithmen und Methoden im Anwendungskontext

Überblick der wissenschaftlichen Arbeiten

Für die gesammelten Anwendungsfälle wurde erforscht, welche Quantenalgorithmen sich zielorientiert zur Lösung eignen. Dabei lag der Fokus vor allem auf variationellen (hybriden) Quantenalgorithmen, die während der gegenwärtigen NISQ (noisy intermediate-scale quantum computer) Ära ein hohes und erfolgsversprechendes Einsatzpotenzial aufweisen. Zusammen mit dem Unternehmensbeirat (vgl. Tabelle 2) wurden aus den konsolidierten Anwendungsfällen Exemplare ausgewählt, die ein besonders hohes und für das Land Baden-Württemberg repräsentatives Nutzpotezial aufweisen sowie alle in Abbildung 4 aufgeführten wirtschaftlichen und wissenschaftlichen Kriterien erfüllen.

Mehran Ghahremanpour	Ministerium für Wirtschaft, Arbeit und Tourismus Baden-Württemberg, Stuttgart
Martin Schlamann	Kumavision AG, Dortmund
Dr. Wolfgang Seeliger	Leichtbau BW GmbH, Stuttgart
Arno Huhn	Schwarz IT KG, Neckarsulm
Andreas Liebing	StoneOne AG, Berlin
Dr. Michael Marthaler	HQS Quantum Simulations GmbH, Karlsruhe
Dr. Frederick Struckmeier	Trumpf Werkzeugmaschinen GmbH + Co. KG, Ditzingen

Tabelle 2: Die Zusammensetzung des Unternehmensbeirats im Projekt SEQUOIA.



Abbildung 4 Wirtschaftliche und wissenschaftliche Auswahlkriterien der SEQUOIA Fokus Use Cases.

Infolgedessen wurden sechs Fokus Use Cases ausgewählt, die im weiteren Projektverlauf im Detail entwickelt und mit den Unternehmenspartnern abgestimmt wurden. Anschließend erfolgte die Implementierung, die Simulation auf den FhG-Simulatoren bzw. die Demonstration auf dem IBM Quantum System One in Ehningen. Die Fokus Use Cases sind im Einzelnen:

Optimierung von Ladeplänen einer gegebenen Ladesäuleninfrastruktur (Energy Management Problem, EMP)



Quantenbasierte **numerische Strömungssimulation**
(Computational Fluid Dynamics, CFD)



Überblick der wissenschaftlichen
Arbeiten

**Optimierung einer Auftrags-
sequenzierung** in der diskreten
Fertigung (Job Shop Problem, JSP)



Optimale Auslegung von Schnittmustern
in der Blechfertigung (Bin Packing Problem, BPP)



Szenario-basierte Routenplanung zur
Absicherung von Fahrfunktionen
(Traveling Salesperson Problem, TSP)



Routenplanung von LKW-Flotten
im Supply-Chain-Management
(Set Partitioning Problem, SPP + TSP)



Auf die konkreten Umsetzungen dieser Use Cases und der zugrundeliegenden Algorithmen sowie Methoden wird in Abschnitt 2.4 eingegangen.

Studie Quantencomputing für Unternehmen

Die im Projekt erarbeiteten Ergebnisse wurden in der geplanten Studie »Quantencomputing für Unternehmen« der breiten (Fach-)Öffentlichkeit zugänglich gemacht. Zielgruppe sind dabei sowohl Entscheider*innen, Innovationscouts und Technologieverantwortliche in Unternehmen, als auch konventionelle Entwickler*innen, die die industrielle Bedeutung des Quantencomputings verstehen und den Stand der Forschung bewerten möchten.

Im Detail behandelt die Studie die folgenden Inhalte:

- **Einführung in das Quantencomputing**
 - o NISQ Ära und Ökosystem (Stakeholder, Provider und Projekte)
 - o Funktionsweise Quantencomputing
 - o Hardwarerealisierungen
 - o Hybride Quantenalgorithmen
 - o Exkurs: Quantum Error Correction
- **SEQUOIA Anwendungsfälle**
 - o Marktanalyse und -überblick
 - o Die SEQUOIA Anwendungsfälle:
Klassifizierung, Bewertung und Herangehensweisen
 - o Wissenstransfer in die Wirtschaft und Wissenschaft
- **Der SEQUOIA Anwendungsbaukasten: Quantencomputing in der Praxis**
 - o Quantum Software Engineering:
Workflows, Vorgehensmodelle und Programmierung

- o Der SEQUOIA Werkzeugkasten: High-Level Entwickler API
- o Beispiel anhand eines Anwendungsfalls

▪ **Ausblick und Schluss**

- o Quanten-Roadmap: Handlungsempfehlungen und Transferkonzepte

2.2 Arbeitspaket 2: Spezifikation, Werkzeuge und Schnittstellen

Anpassung des Projektplans

Im Förderantrag wurde eine Diskrepanz zwischen den Meilensteinen in der detaillierten Beschreibung des Arbeitspaketes (Infobox AP2, Ergebnisse nach Meilensteinen) und der Zeitplanung im Projektantrag (vgl. Gantt-Diagramm Abbildung 2) festgestellt und aufgelöst, indem der Projektplan wie folgt angepasst wurde:

AP 2.2 und AP 2.3 waren im Zeitplan jeweils für die Projektmonate M1 bis M9 und M1 bis M24 vorgesehen. Im Widerspruch hierzu waren die Implementierungen von AP 2.2 und AP 2.3 jeweils zu den Meilensteinen M24 und M18 vorgesehen. Aus Projektsicht war es am sinnvollsten, zunächst die in AP 2.2 geplanten Import- und Export-Funktionalitäten für unterschiedliche Dateiformate von Quantenschaltkreisen umzusetzen, so dass diese bei der Ausführung der Schaltkreise (Fokus von AP 2.3) Verwendung finden konnten. Daher wurde die Implementierung von AP 2.2 für den Meilenstein M18 vorgezogen und das Meilensteinergebnis für AP 2.3 auf M24 zurückgestellt. Die in M9 für den Meilenstein vorgestellte Implementierung von AP 2.1 wurde im Projektverlauf kontinuierlich erweitert, um die erreichten Fortschritte der Fokus Use Cases zu integrieren.

Ergebnisse

Es wurde eine Schnittstelle zur **Generierung von (Teilen von) Quantenschaltkreisen (AP 2.1)** konzipiert und umgesetzt. Die Schnittstelle wurde mittels einer REST API umgesetzt, welche von einem mit Python Flask implementierten Service bereitgestellt wurde, bei dessen Implementierung ein besonderes Augenmerk auf die Erweiterbarkeit und Interoperabilität gelegt wurde. Um Benutzern einen guten Einblick in die Spezifikation der Schnittstelle zu bieten, wurden alle Endpunkte via OpenAPI integriert. Daher können alle Anfrageschemas sowie Beispielanfragen eingesehen und ausprobiert werden: Die Schnittstelle zur Generierung von Quantenschaltkreisen, implementierte Quantenschaltkreisfragmente (bspw. Basis Encoding oder Amplituden Encoding), generalisierte Quantenalgorithmen (bspw. QAOA) und benutzerdefinierte Quantenalgorithmen, die während des Projekts entstanden sind, um bestimmte Use-Cases umzusetzen. Da die meisten aktuellen Quanten SDKs in Python verfügbar sind, bietet der Service ein hohes Maß an Flexibilität und kann zur Generierung von Schaltkreisen in unterschiedlichen Sprachen verwendet werden. Die Schnittstelle ist als Open-Source Projekt auf GitHub veröffentlicht.

Relevante Werkzeuge und Schnittstellen

- Quantum Circuit Generator
 - <https://github.com/UST-QuAntiL/quantum-circuit-generator>

Im Rahmen von **AP 2.2 (Import und Export unterschiedlicher Dateiformate)** wurde die Analyse und der Vergleich von Dateiformaten für Quantenschaltkreise sowie vorhandener Cross-Compiler-Projekte zur Übersetzung zwischen diesen Formaten in Form einer umfassenden Evaluation durchgeführt. Die Resultate wurden in das vom Partner IAAS entwickelte Vergleichswerkzeug für Technologien im Bereich der Quantenanwendungsentwicklung, Qoverview, integriert. Qoverview fasst somit nun relevante Informationen zu jedem Dateiformat zusammen (Abbildung 5). Zudem werden die Übersetzungspfade zwischen Dateiformaten in einem Graphen dargestellt und es können durch Auswahl eines Übersetzungspfades weitere Informationen zu den Übersetzungsmöglichkeiten wie die nutzbaren Cross-Compiler und deren Einschränkungen angezeigt werden (Abbildung 6). Anhand dieser Resultate wurden die relevantesten Dateiformate ausgewählt und zugehörige Cross-Compiler an die Schnittstelle für den Import und Export der unterschiedlichen Dateiformate angebunden. Diese neu konzipierte Schnittstelle wurde zur Umsetzung mit dem bereits in PlanQK für die Übersetzung und Analyse von Quantenschaltkreisen entwickelten Framework *Quantum Transpiler* zusammengeführt. Dadurch ist die Kompatibilität mit dem ebenfalls in PlanQK entstandenen *NISQ-Analyzer* sichergestellt, sodass auch dieser von der in SE-QUOIA entwickelten Funktionalität profitieren kann. Der NISQ-Analyzer unterstützt Anwender bei der Auswahl geeigneter Quantencomputer für die Ausführung von Quantenschaltkreisen. Da die Hardware-Hersteller unterschiedliche Dateiformate unterstützen, müssen Schaltkreise teils entsprechend übersetzt werden. Der *Quantum Transpiler* ermöglicht Schaltkreise mittels einer API (analog zu AP 2.1) oder eines Userinterfaces hochzuladen und anschließend zu übersetzen und für bestimmte QPUs zu transpilieren. Es werden nunmehr Übersetzungen zwischen folgenden Formaten unterstützt: Bracket, Cirq, OpenQASM 2, PyQuil, Q# und Qiskit.

Relevante Werkzeuge und Schnittstellen

- Qoverview
 - <https://github.com/UST-QuAntiL/Qoverview>
- NISQ-Analyzer
 - <https://github.com/UST-QuAntiL/nisq-analyzer>
- Quantum Transpiler
 - <https://github.com/UST-QuAntiL/QuantumTranspiler>

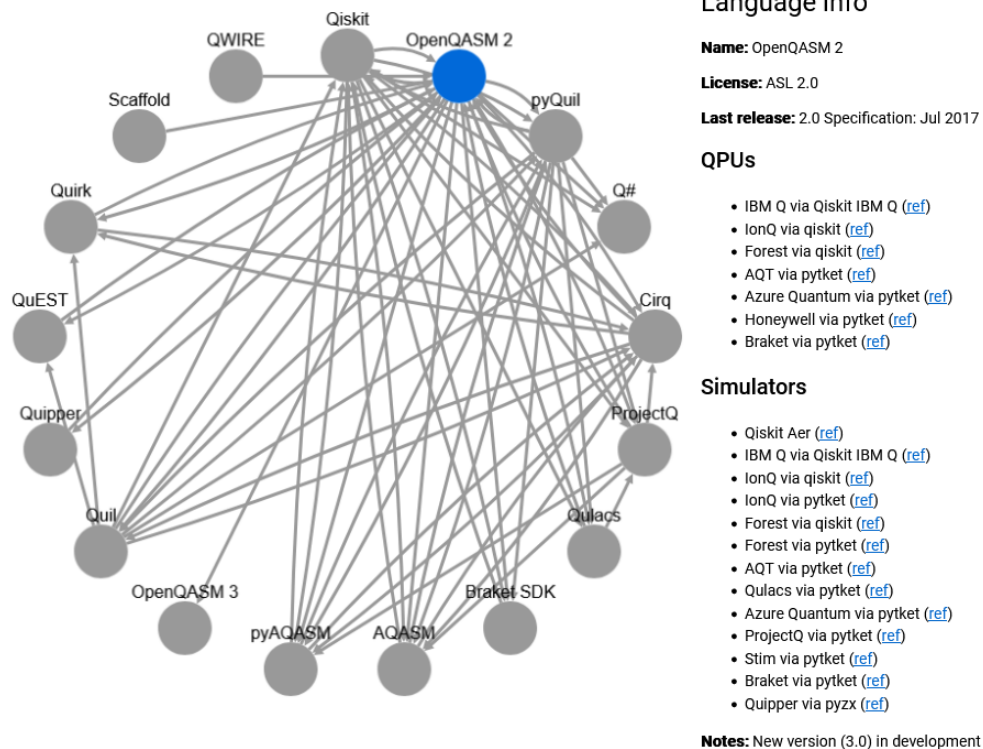


Abbildung 5 Ausschnitt aus Qview, Darstellung der Übersetzungspfade in einem Graphen (links) und Informationen zu jedem Dateiformat.

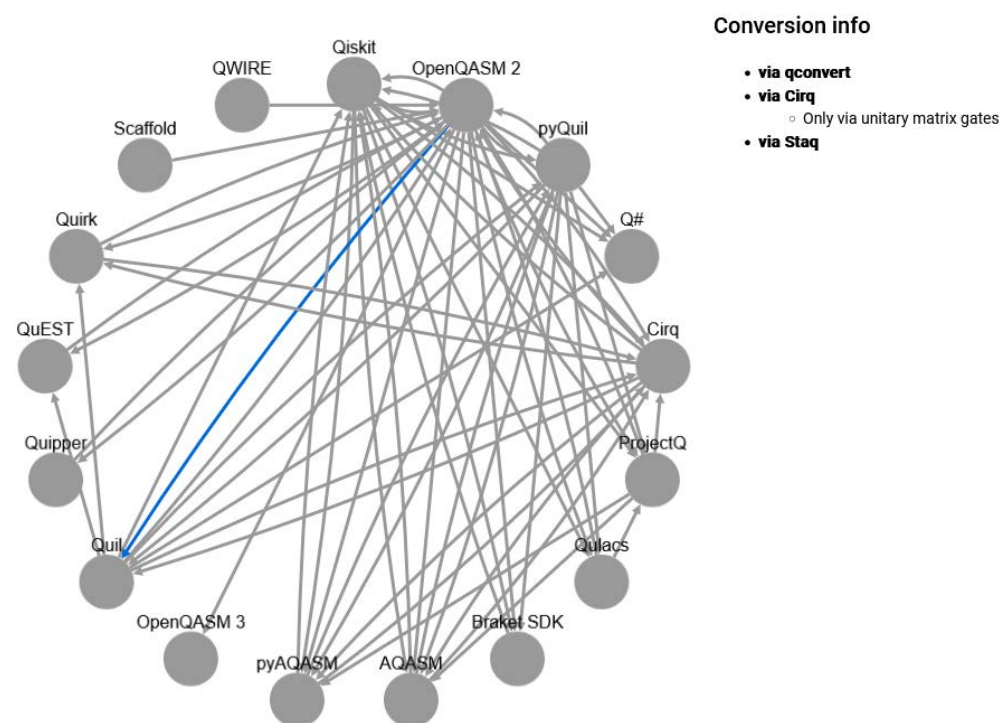


Abbildung 6 Ausschnitt aus Qview mit ausgewähltem Übersetzungspfad und Informationen zu den Übersetzungsmöglichkeiten.

Als Grundlage für die **Schnittstelle zur Ausführung von Quantenschaltkreisen (AP 2.3)** wurde eine Analyse und Übersicht über existierende Hardware zur Ausführung von Quantenschaltkreisen sowie der von dieser unterstützten Dateiformate erstellt. Diese

Informationen werden ebenfalls in Qverview bereitgestellt (Abbildung 5). Teils im Rahmen von PlanQK bestehende Implementierungen wurden hierzu erweitert und neue Implementierungen ergänzt, sodass nun Braket, Cirq, Forest, pytket, Q# und Qiskit zur Ausführung zur Verfügung stehen und mit dem NISQ-Analyzer kompatibel sind. Für jedes Framework steht ein Service mit entsprechender API (analog zu AP 2.1) zur Verfügung, über welchen die Ausführung von Quantenschaltkreisen ermöglicht wird.

Weiterhin wurde in AP 2.3 die Ausführung von Quantenanwendungen mittels Workflows untersucht. In [14] wurde ein Konzept für konfigurierbare und automatische Fehlermitigation auf Basis der Ergebnisse AP 3.4 und 3.6 entwickelt, welche mittels OpenTOSCA deployt wird.

Relevante Werkzeuge und Schnittstellen

- Ausführungs-Service für Qiskit
 - <https://github.com/UST-QuAntiL/qiskit-service>
- Ausführungs-Service für pytket
 - <https://github.com/UST-QuAntiL/pytket-service>
- Ausführungs-Service für Forest
 - <https://github.com/UST-QuAntiL/forest-service>
- Ausführungs-Service für Q#
 - <https://github.com/UST-QuAntiL/qsharp-service>
- Ausführungs-Service für Cirq
 - <https://github.com/UST-QuAntiL/cirq-service>
- Ausführungs-Service für Braket
 - <https://github.com/UST-QuAntiL/braket-service>

Im Rahmen von AP 2.4 wurden externe Werkzeuge zur Bildung einer durchgängigen Werkzeugkette gesichtet und bewertet. Der Fokus lag hier auf den Qiskit-Werkzeugen dynamical decoupling, mapomatic und mthree, welche sich allesamt mit Fehlermitigation auf verschiedenen Ebenen beschäftigen.

Dynamical Decoupling verringert die Wahrscheinlichkeit, dass Qubits dekoherieren, indem kontrollierte Pulse zur Neutralisierung der Dekohärenzeffekte eingesetzt werden. Dies erfolgt beim Transpilieren der Schaltkreise durch das Einfügen zusätzlicher logikneutraler Quantengatter zu genau diesem Zweck.

Mapomatic ist ein Werkzeug, welches der Tatsache Rechnung trägt, dass das bei der ersten Transpilation gewählte Mapping auf physikalische Qubits oftmals nicht optimal ist. Durch die Wahl möglichst fehlerfreier Qubits, die in der Topologie jedoch eventuell ungünstig gelegen sind, kann die Verwendung von vielen Swap-Gattern notwendig werden, was den Vorteil der qualitativ hochwertigen Qubits wieder zunichtemacht. Mapomatic kombiniert hier beide Informationen und kommt so im Allgemeinen zu fehlerärmeren Schaltkreisen.

Mthree steht für Matrix-free Measurement Mitigation und beruht darauf, aus einem verfälschten Ausleseergebnis durch Kenntnis der Fehlersystematik, ein Ergebnis welches näher an der richtigen Lösung liegt, zurückzurechnen.

Jede der Methoden für sich genommen verringert die Fehlerproblematik beim Quantencomputing, dies kann zum Beispiel über den Demonstrator Nummer 3 für den Use Case »**Optimierung von Ladeplänen einer gegebenen Ladesäuleninfrastruktur**« nachvollzogen werden. Daher empfiehlt SEQUOIA ihre gesamthafte Verwendung innerhalb einer effizienten Werkzeugkette.

Gegen Ende der Projektlaufzeit wurden die Methoden Mapomatic und Dynamical Decoupling ohnehin fest in Qiskit integriert. Mapomatic ist mithin automatisch im Transpiliationsprozess inbegriffen und für die (höchste) Optimierungsstufe 3 des Qiskit-Transpilers gilt dies auch für das Dynamical Decoupling.

Alle in SEQUOIA entwickelten Schnittstellen wurden in das Quokka-Ökosystem für die Service-basierte Ausführungen von Quantenalgorithmen integriert und sind über das Quokka API-Gateway erreichbar (Abbildung 7). Hierzu gehören auch zusätzliche Schnittstellen für die Auswertung von Ergebnissen, die Mitigation von Fehlern (AP 3.6 und die Optimierung von Schaltkreisparametern in variationellen Quantenalgorithmen. Die Relevanz und Anwendbarkeit des Quokka-Ökosystems für einen SEQUOIA Fokus Use Case wurde in [15] demonstriert. Ein zugehöriges Video ist auf YouTube verfügbar: [ICSOC 2022 Quokka Service Ecosystem Demonstration](https://www.youtube.com/watch?v=ICSO2022QuokkaServiceEcosystemDemonstration).

Hierzu wurde der Fokus Use Case **Routenplanung** (siehe Kapitel 2.4) wie in Abbildung 8 dargestellt als Workflow umgesetzt. Durch die Verwendung von Workflows können zusätzliche Vorteile wie Skalierbarkeit, Robustheit, Überwachbarkeit und Plattformunabhängigkeit erreicht werden.

Relevante Werkzeuge und Schnittstellen

- Quokka-Ökosystem
 - <https://github.com/UST-QuAntiL/Quokka>
- Quokka-Gateway
 - <https://github.com/UST-QuAntiL/Quokka-Gateway>
- Service für Ergebnisauswertung
 - <https://github.com/UST-QuAntiL/objective-evaluation-service>
- Service für Fehlermitigation
 - <https://github.com/UST-QuAntiL/error-mitigation-service>
- Service für die Optimierung von Schaltkreisparametern
 - <https://github.com/UST-QuAntiL/Quokka/tree/main/services/optimization-service>
- Workflowumsetzungen
 - <https://github.com/UST-QuAntiL/QuantME-UseCases>
 - <https://github.com/UST-QuAntiL/Quokka>

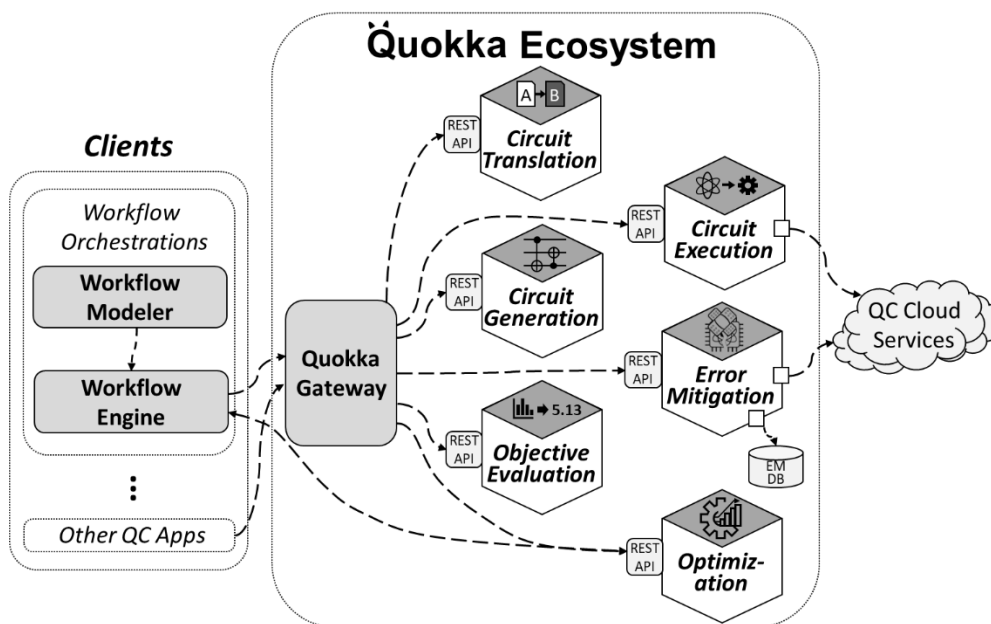


Abbildung 7 Modularisierte Schnittstellen-Architektur des im Rahmen des SEQUOIA-Werkzeugbaukastens entwickelten Quokka-Ökosystems.

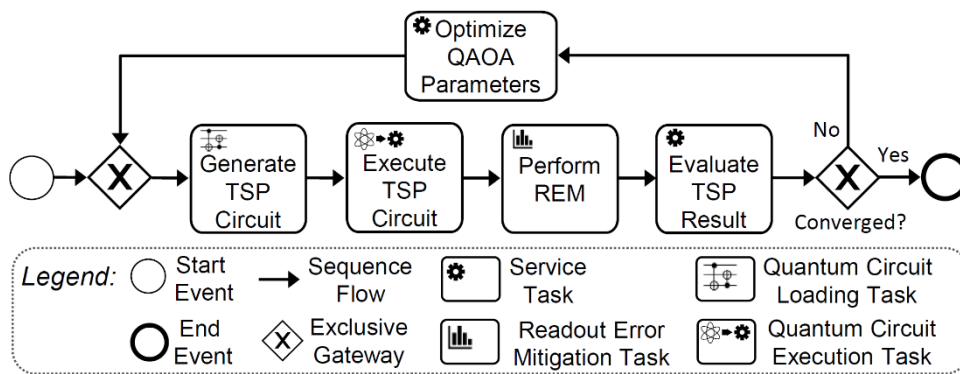


Abbildung 8 Umsetzung eines Fokus Use Cases in Form eines Workflows unter Verwendung von Quokka.

2.3 Arbeitspaket 3: Fehlermitigation und Zuverlässigkeit

Im Rahmen des **Arbeitspaketes 3.1 »Charakterisierung und Simulation des IBM-Q-Systems«** wurde der das CNOT-Gatter erzeugende Hamiltonian mit Qiskit Pulse (früher Qiskit Open Pulse) charakterisiert.

Zu Beginn des Projektes konnte festgestellt werden, dass die von IBM zur Verfügung gestellte Simulationsumgebung des Quantencomputers für die Zwecke nicht ausreicht, da das Modell nicht alle auftretenden Fehler berücksichtigt. Dazu wurde der das CNOT-Gatter erzeugende Hamiltonian

$$H = a_x ZX + a_y ZY + a_z ZZ + b_x IX + b_y IY + b_z IZ$$

(Koeffizienten a_i und b_i bezeichnen hier die Stärke der Interaktion zwischen dem Kontroll- und dem Target-Qubit. Die Operatoren, Großbuchstaben, entsprechen den Pauli-Matrizen) charakterisiert und der Einfluss der vernachlässigten Terme untersucht.

Das Vorgehen diente als Vorarbeit zu **Arbeitspaket 3.2 »Entwicklung von Verfahren zur Korrektur von Gatterfehlern«**. Da die Skalierung der Pulse extrem (Zeit-) aufwendig war und nicht als einfache Methode zur Verfügung gestellt werden konnte, wurde eine diskrete Richardson-Fehlerextrapolation, die Zero-Noise-Extrapolation (ZNE), entwickelt.

Dabei werden die Fehler zunächst durch Einfügen zusätzlicher Gatter auf kontrollierte Weise verstärkt. Wie in Abbildung 9 gezeigt, kann dann das bei verschiedenen Fehlerstärken gemessene Ergebnis durch Anfitzen einer glatten (z.B. kubischen) Funktion zum fehlerfreien Fall hin extrapoliert werden.

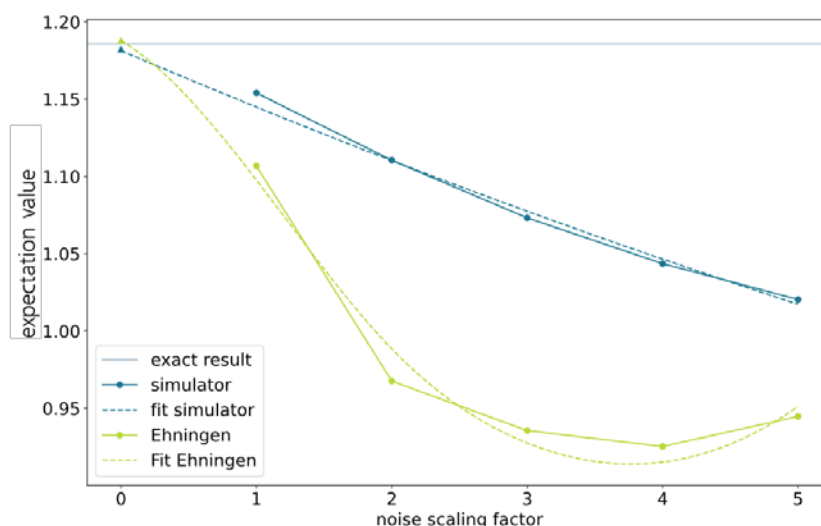


Abbildung 9 Fehlerkorrektur mit Hilfe der Zero-Noise-Extrapolation (ZNE). Aufgetragen ist der gemessene Erwartungswert (hier: die Lösung eines linearen Gleichungssystems mit dem HHL-Algorithmus) als Funktion der skalierten Fehlerstärke für den Simulator mit Fehlermodell (blau) sowie für das IBM Q System One in Ehningen mit Messfehlermitigation (grün). Die gestrichelten Linien entsprechen jeweils einem quadratischen bzw. kubischen Fit, welcher extrapoliert wird, um den gesuchten Wert bei Fehlerstärke Null zu erhalten. In der Tat

kommen die extrapolierten Werte dem exakten Ergebnis (hier: 1,18585) viel näher als die ohne Fehlerskalierung (d.h. bei Fehlerstärke 1) gemessenen Werte.

In diesem Beispiel betrachten wir die Lösung eines linearen Gleichungssystems mit dem HHL-Algorithmus. Der gesuchte fehlerfreie Wert beträgt hier 1,18585. Das IBM Q System One in Ehningen liefert stattdessen zunächst (d.h. bei skaliertem Fehlerstärke 1) Werte in der Nähe von 1,10. Nach Skalierung der Fehlerstärke mit anschließender Extrapolation erhält man hingegen den Wert 1,1879, welcher dem exakten Ergebnis sehr nahekommt.

Im nächsten Schritt wurde mittels Fehlerrandomisierung nach Wallman [<https://arxiv.org/abs/1511.00727>] eine verbesserte Version der Fehlerextrapolation implementiert. Dabei werden zufällige Einzelgatter vor und nach CNOT-Gattern sowie auf angrenzenden Qubits hinzugefügt. Die Fehler werden dadurch zu stochastischen Pauli-Fehlern vereinfacht und Crosstalk zwischen angrenzenden Qubits kann unterdrückt werden. In Abbildung 10 ist ein Schaltkreis mit drei Qubits und zwei CNOT-Gattern, sowie mehreren Einzelgattern zu sehen. Bei der Fehlerrandomisierung werden in den ursprünglichen Schaltkreis (oben) zufällige Einzelgatter hinzugefügt (Mitte). Durch anschließendes Vereinfachen und zusammenfassen der Einzelgatter ändert sich die Gesamtgatter-Anzahl nicht (unten).

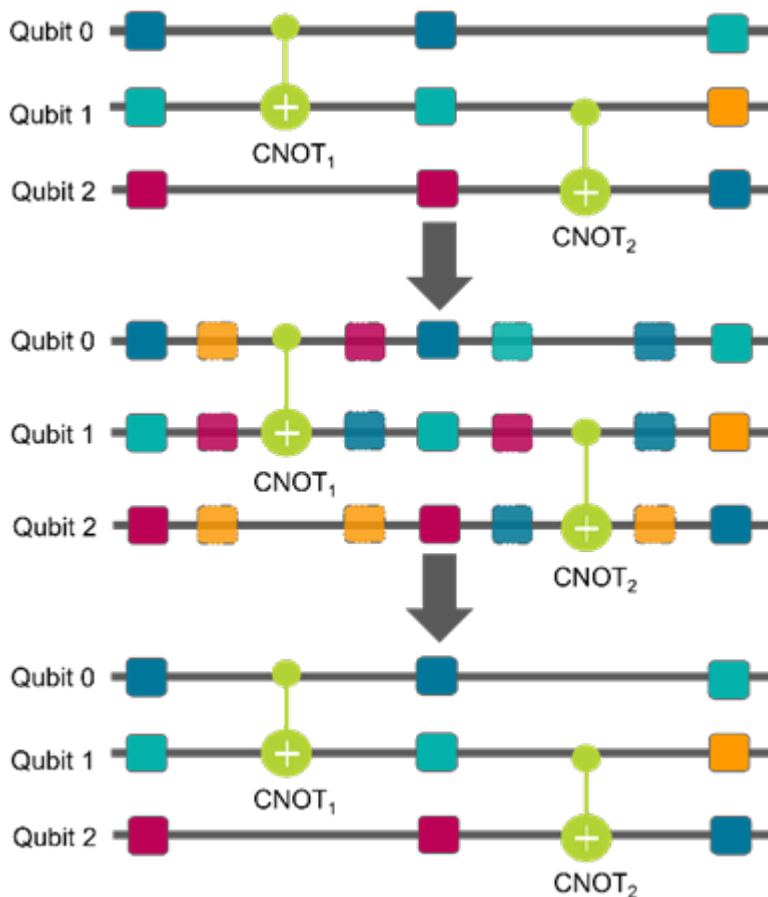


Abbildung 10 Einfügen zufälliger Einzelgatter vor und nach jedem CNOT-Gatter, sowie bei angrenzenden Qubits wandelt die Fehler in stochastische Pauli-Fehler um. Durch anschließendes Vereinfachen und zusammenfassen der Einzel-Gatter, ist die Gesamtgatteranzahl ähnlich groß wie zu Beginn

Der Vergleich mit der ursprünglichen Version, sowie mit einem nutzbaren Packet (Mitiq [<https://mitiq.readthedocs.io/en/stable/>]) zeigt, dass die hier im Projekt entwickelte ZNE deutlich geringere Abweichungen vom exakten Wert zeigt, sowie die Varianz verringert ist (Abbildung 11). Dadurch werden die Werte häufiger zum korrekten Erwartungswert mitigiert.

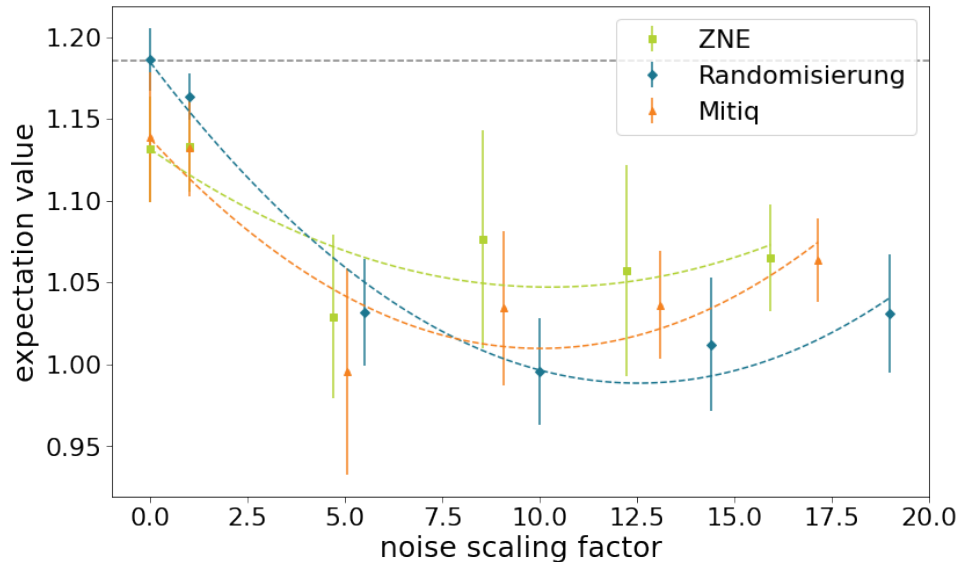


Abbildung 11 Vergleich der ZNE (grün) mit dem Softwarepaket Mitiq (orange) sowie der erweiterten ZNE mit Fehlerrandomisierung (blau). Die Fehlerrandomisierung zeigt eine geringere Varianz und eine höhere Genauigkeit der Mitigation.

Der in **Arbeitspaket 3.3** geplante **Vergleich vorhandener Unfolding-Methoden** zur Auslesefehlerkorrektur wurde mit dem Bericht zum Meilenstein M9 abgeschlossen. Auf Basis einer initialen Literaturrecherche [14] wurden vielversprechende Methoden zur Reduktion von Auslesefehlern implementiert und hinsichtlich ihrer Ergebnisqualität, Laufzeit und Komplexität untersucht. Die in Abbildung 12 und Abbildung 13 dargestellten Graphen zeigen die Entwicklung der Fehlerrate und Laufzeit der untersuchten Methoden für eine steigende Anzahl an Qubits. Der Vergleich hat aufgezeigt, dass Auslesefehlermitigationsverfahren die Qualität von auf dem Quantencomputer berechneten Lösungen, abhängig von der gewählten Methode, deutlich verbessern können. Allerdings ist insbesondere bei Schaltkreisen mit einer hohen Breite auf die Skalierbarkeit der gewählten Methode zu achten.

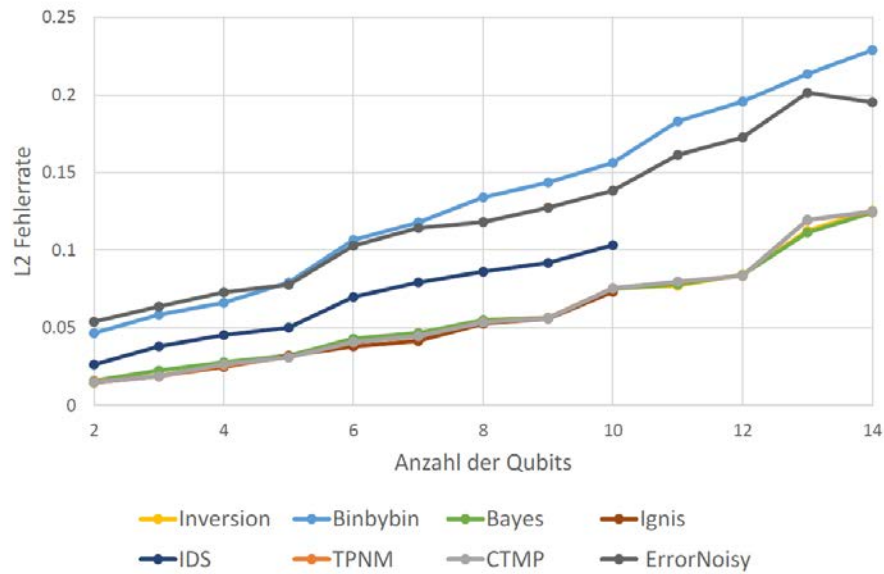


Abbildung 12 Entwicklung der Fehlerrate bei verschiedenen Auslesefehlermitigationsverfahren im Vergleich zum Rohergebnis (ErrorNoisy).

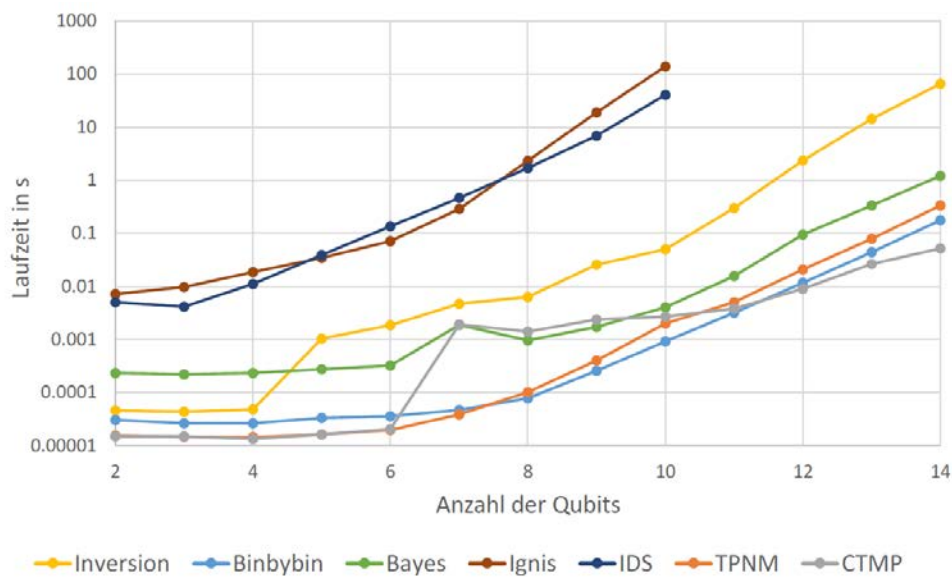


Abbildung 13 Entwicklung der Laufzeit verschiedener Auslesefehlermitigationsverfahren in Abhängigkeit von der Anzahl verwendeter Qubits.

Ein Konzept zur Bereitstellung von Kalibrierungsmatrizen wurde im Rahmen des **AP 3.4 Entwicklung von Verfahren zur Korrektur von Auslesefehlern** erarbeitet. Ein mit Python Flask implementierter Service ermöglicht die kontinuierliche Generierung der Kalibrierungsdaten für die Quantenprozessoren unterschiedlicher Anbieter. Bei der Implementierung wurde insbesondere auf die Erweiterbarkeit und Plattformunabhängigkeit geachtet, sodass problemlos weitere Methoden und Anbieter integriert werden können. Aufgrund ihrer Größe wird für die Speicherung und Bereitstellung der Kalibrierungsdaten auf eine skalierbare, hoch performante Objekt-Datenbank zurückgegriffen.

Weiterhin wurden in AP 3.4 Machine-Learning-Methoden (ML-Methoden) zur Korrektur von Auslesefehlern untersucht. Hierzu wurde zunächst in einer umfassenden Literaturrecherche ein Überblick geschaffen [14]. Zum Vergleich der ML-Methoden mit den zuvor untersuchten Unfolding-Methoden wurde die Auslesefehlermitigation auf Basis neuronaler Netzwerke implementiert. Wie aus Abbildung 16 ersichtlich wird, kann ein vollständig verbundenes neuronales Netzwerk (siehe Abbildung 14) nach ausreichend langem Training vergleichbare Ergebnisse wie die zuvor untersuchten Unfolding-Methoden erreichen. Aufgrund der in der Anzahl verwendeter Qubits exponentiell vielen Zustände, die in der Eingabe und Ausgabe des Netzwerks berücksichtigt werden müssen, ist jedoch die Skalierbarkeit dieser Methode stark eingeschränkt. Daher wurden Möglichkeiten zur Verbesserung der Skalierbarkeit untersucht, insbesondere durch das Clustern von Qubits für die Eingabe ins Netzwerk (siehe Abbildung 15). Beispielweise können im Falle von 5 Qubits zwei Cluster mit jeweils 2 und 3 Qubits gebildet und die Messungen auf diese heruntergerechnet werden. Dadurch fallen statt 2^5 nur noch $2^2 + 2^3$, also 12 statt 32, zu unterscheidende Zustände an. Außerdem wird die innere Struktur des Netzwerks reduziert, sodass weniger Parameter anfallen und das Training des Modells schneller konvergiert. Da die gemessenen Zustände hierbei nur auf Clustern von Qubits betrachtet werden, gehen Informationen über die ursprünglich gemessenen Zustände verloren, was sich in einer schlechteren Mitigationsleistung im Vergleich zum vollständig verbundenen Netzwerk niederschlägt. Dennoch wird eine signifikante Fehlerreduktion erreicht und zugleich der Aufwand reduziert und die Skalierbarkeit verbessert.

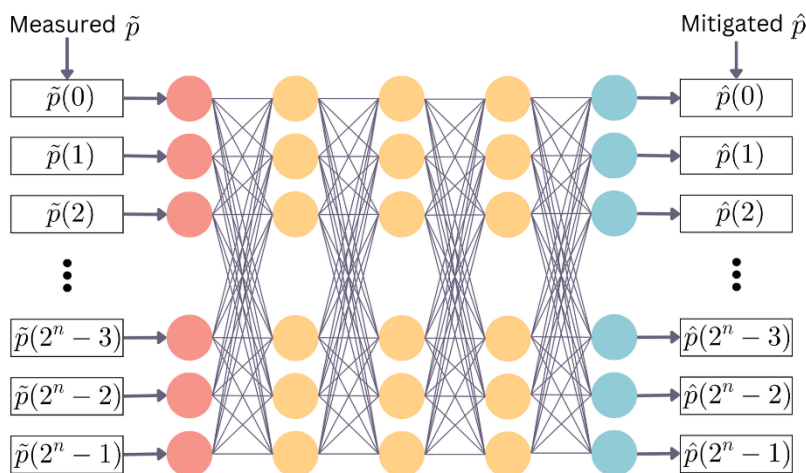


Abbildung 14 Struktur des vollständig verbundenen neuronalen Netzkes zur Auslesefehlermitigation.

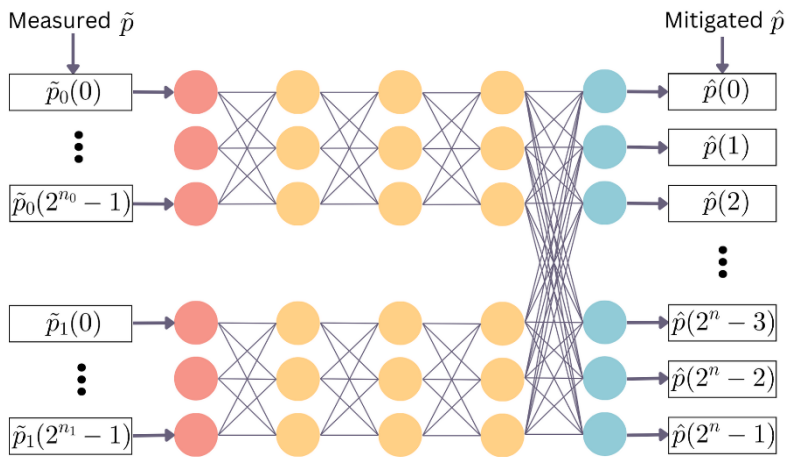


Abbildung 15 Struktur des adaptierten neuronalen Netzes zur Auslesefehlermitigation.

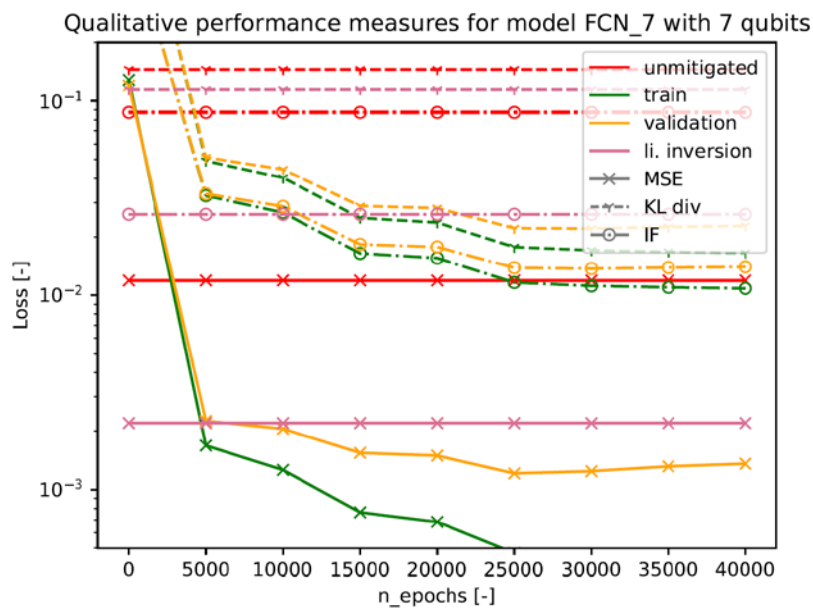


Abbildung 16 Vergleich Fully Connected Neural Network (FCN) mit Unfolding mittels Matrix-Invertierung

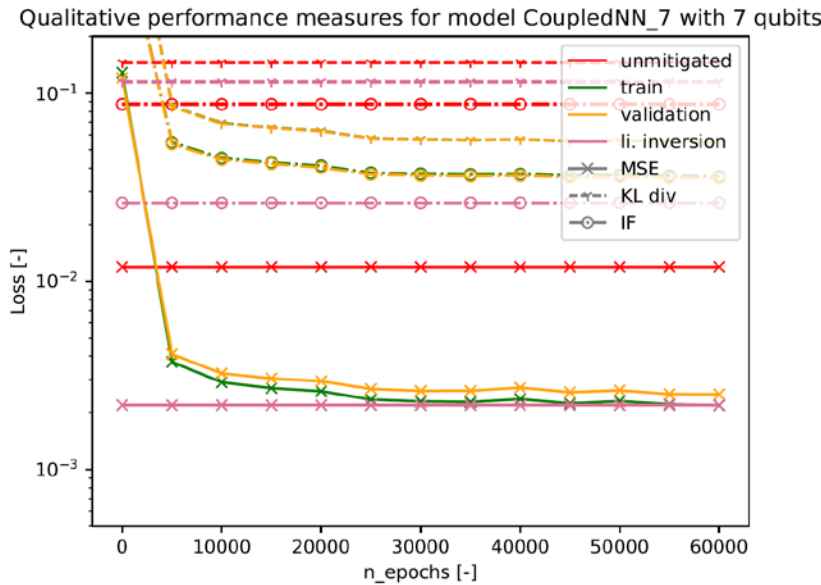


Abbildung 17 Vergleich adaptiertes Neural Network mit Unfolding mittels Matrix-Invertierung

Arbeitspaket 3.5 behandelte die **Zuverlässigkeit und Absicherung von Quantensoftwarekomponenten und -anwendungen**. Während der Arbeiten wurden die engen Anknüpfungspunkte bei der Analyse von Zuverlässigkeit zum Thema Modellierung (AP 6.1) und bei der Absicherung mit dem Thema durchgängige Werkzeugketten (AP 2.4) offensichtlich. Entsprechend wurde die Analyse von Zuverlässigkeitseigenschaften bei durch Quantencomputing induzierten Unsicherheiten im Kontext von Modellierung in AP 6.1 betrachtet. Für das Thema Absicherung wurde eine Stärkung des Proof-of-Concept in Richtung Open-Source-Bereitstellung angestrebt, um dem Thema durchgängige Werkzeugketten (AP 2.4) stärker Rechnung zu tragen.

Insbesondere wurde untersucht wie durch eine zielgerichtete Übersetzung von Quantenschaltkreisen in klassische Programmiersprachen ein breites Spektrum von klassischen Verifikations-, Test- sowie Spezifikationsmethoden auf Quantencomputing übertragen werden können. Diese grundlegende Idee, die bereits auf der TAV im Rahmen der GI 2021 als Work-In-Progress-Paper veröffentlicht wurde [6], erlaubt es gängige Tools von klassischen Sprachen (gezeigt am Beispiel Java) auf Quantenschaltkreise anzuwenden. Mit Hilfe dieser Werkzeugkette konnten bereits bekannte Quantenalgorithmen, wie der Deutsch-Josza-Algorithmus sowie der Grover-Algorithmus für kleine Anzahlen an Qubits als korrekt bewiesen werden. Der Reiz dieser Methodik ist einerseits die hohe Automatisierbarkeit, die auch weniger erfahrenen Entwicklern die Anwendung erlaubt, sowie die hohe Flexibilität (nahezu jedes Werkzeug der Host-Sprache ist dank der Übersetzung auch auf Quantenschaltkreise anwendbar). Nachteil der präsentierten Vorgehensweise ist die mangelnde Skalierbarkeit, da die Übersetzungen exponentiell in der Anzahl der Qubits wachsen. Eine wissenschaftliche Diskussion des Proof-of-Concept fand im Rahmen der QPL'22 [9] und auf der KQCBW Developer Conference statt. Damit wurde der Meilenstein M18 erreicht.

In **AP 3.6 Entwicklung von Schnittstellen für Fehlermitigationsverfahren** wurden die wichtigsten Fehlermitigationsverfahren in den Komponentenbaukasten integriert. Hierbei wurden die über **AP 3.4** bereitgestellten Kalibrierungsmatrizen verwendet. Die Verfahren können dann über Schnittstellen verwendet werden und somit einfach und

plattformunabhängig in Quantenanwendungen eingebunden werden (siehe Beschreibung in **AP 2**). Die in **AP 3.2** entwickelte Fehlermitigation (ZNE) kann direkt in Quantenanwendungen integriert werden. Diese wurde im Rahmen des Projektes den Partnern als Python-Methode zur Verfügung gestellt und via des ZNE Demonstrators auf der Projekthomepage <https://websites.fraunhofer.de/sequoia/use-cases/> auch der Öffentlichkeit zugänglich gemacht.

2.4 Arbeitspakete 4 und 5: Quantenalgorithmen für industrielle Anwendung mittels hybrider Lösungen

Wie man aus dem Titel bereits erkennen kann, sind die Arbeitspakete 4 und 5 besonders eng verzahnt. Daher werden die Projektergebnisse der hiesigen Arbeitspakete im Folgenden gemeinsam dargestellt. Grundlage aller Quantenalgorithmen und Lösungsimplementierung sowie Benchmarking sind die in AP1 erörterten, vom Unternehmensnetzwerk eingespielten, Anwendungsfälle. Insbesondere erläutern wir die Erreichung aller Teil-AP-Ziele im folgenden Abschnitt anhand der unter AP1 vorgestellten sechs SEQUOIA Fokus Use Cases.

Am Beginn steht mit AP 5.1 die Überführung eines Anwendungsfalles in seine Lösungskategorie. Dann geht es gemäß AP 5.2 darum, ein hybrides Lösungskonzept zu erarbeiten; also um die passende Auswahl eines Teilproblems in diesem Anwendungsfall, welches sich für aktuelle NISQ-Quantencomputer eignet.

In AP 4.2 und 4.3 werden Implementierungen von Quantenalgorithmen entwickelt (lineare Systeme und Optimierung). Dies unter dem Gesichtspunkt, dass sie sich gemäß AP 4.1 besonders für die Einbettung in die Anwendungsfälle und Demonstratoren eignen. Aus jenen muss dann ein geeigneter Quantenalgorithmus ausgewählt werden, der das entsprechende Teilproblem löst. Dazu findet gemäß AP 4.4 ein Benchmarking und eine Evaluierung der in Frage kommenden Alternativen statt. Außerdem ist eine Anpassung des Quantenalgorithmus an den jeweiligen Anwendungsfall möglich. In AP 5.3, 5.4 oder 5.5 schließt sich dann je nach Lösungskategorie die Implementierung der in 5.2 erarbeiteten gesamthaften hybriden Lösung an (Optimierung, ML, Lineare Algebra). Wenn diese hinreichend ausgereift ist, kann schließlich in AP 5.6 die Bereitstellung und Demonstration erfolgen.

Die Algorithmen QAOA (sowohl der Quantum Approximate Optimization Algorithm als auch der Quantum Alternating Operator Ansatz) und VQLS (Variational Quantum Linear Solver) wurden als sehr relevante Algorithmen für die vorliegenden Anwendungsfälle identifiziert. Für jeden Fall wurde der jeweilige Algorithmus in die spezifische Hybridisierungslösung eingebettet. Da die Anpassung hochgradig problemabhängig ist, werden wir, wie schon eingangs erwähnt, die Ergebnisse aus AP 4 und AP 5 hier gemeinschaftlich darstellen, orientiert an den jeweiligen Anwendungsfällen.

Fokus Use Case »Optimierung von Ladeplänen einer gegebenen Ladesäuleninfrastruktur« (Fraunhofer IAO und LamA)

Im Zuge des Verbundprojekts »**LamA** (Laden am Arbeitsplatz)« wird seit Juli 2018 bundesweit an 37 Fraunhofer-Instituten **Ladeinfrastruktur für Elektrofahrzeuge** errichtet [lama1]. **Ziel dieses Use Cases** war es **Ladepläne** für diese Infrastruktur **zu optimieren**. Zwei beispielhafte Optimierungen wären zum einen basierend auf Wetterprognosen die Elektrofahrzeuge dann zu laden, wenn viel erneuerbare Energie (Sonne, Wind, etc.) verfügbar ist. Oder zum anderen basierend auf Strompreisprognosen zu

den Tageszeiten zu laden, wann der Strom am günstigsten ist. Beide Beispielen basieren wesentlich darauf, dass die E-Autos am Arbeitsplatz geladen werden und somit Ankunfts- und Abfahrtszeiten (ungefähr) bekannt sind und eine große Zeitspanne (ca. 8 Stunden) zum Laden zur Verfügung steht. Viele weitere Optimierungen sind denkbar und wurden mit den Domain-Experten von LamA diskutiert. In Anbetracht der vielen **Einschränkungen von NISQ-Computern** und dem Projektziel Quanten-Anwendungen für diese zu entwickeln sowie zu erproben, wurde gemeinsam entschieden als einen **ersten Schritt** in Richtung Ladeplanoptimierung die **Lastspitzenreduktion** zu betrachten. Diese reduziert Kosten und schont die Lade- und Netzinfrastruktur.

Als konkrete Aufgabe wurde definiert die Spitzenlast, welche aus dem Stromnetz gezogen wird, zu minimieren, wobei gegeben ist wie viele Elektrofahrzeuge an einer Ladesäule sind, wann sie ankommen sowie abfahren und wie viel Energie sie laden müssen. Abbildung 18 veranschaulicht die Situation. Man kann erkennen, dass noch **weitere Vereinfachungen** wie z.B. eine Diskretisierung der Zeitachse, die Beschränkung auf diskrete Ladestufen und kleine Problemstellungen getroffen wurden. Dies hat den Hintergrund NISQ-kompatible Probleme zu erhalten, welche jedoch noch den Kern des zugrunde liegenden Use Cases modellieren. Es handelt sich somit um ein **Proof-of-Concept-Modell**, das als Quanten-Anwendung umgesetzt wurde.

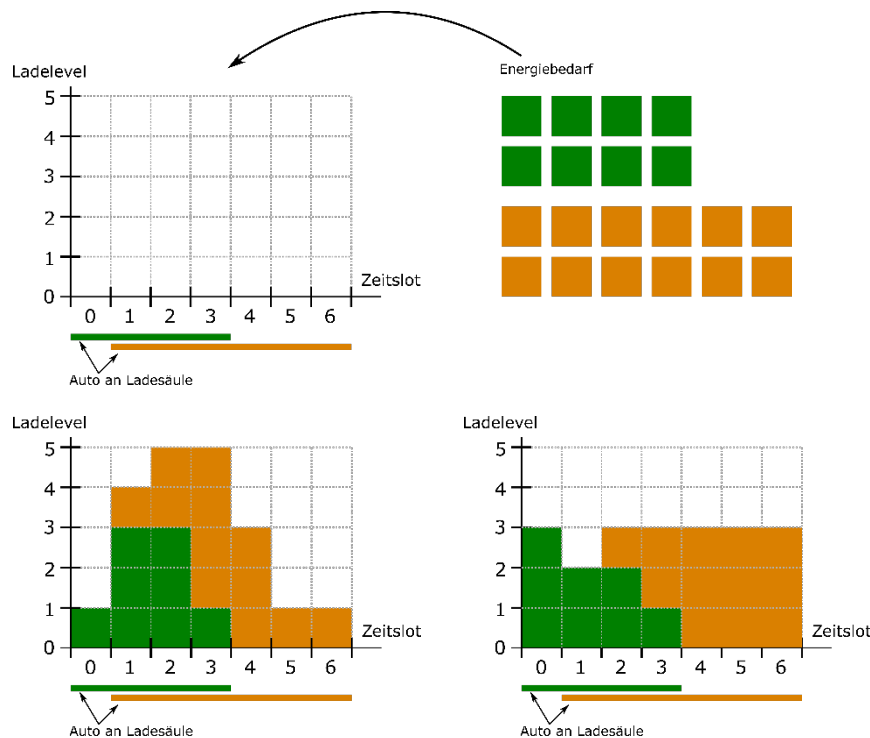


Abbildung 18 Proof-of-Concept-Modell zum Fokus Use Case »Optimierung von Ladeplänen«. (Beispiel für abstrakte Beschriftung: 1 Zeitslot = 1 Stunde, 1 Ladelevel = 5 kW).

Als ersten Schritt zur Umsetzung als Quanten-Anwendung wurde eine **mathematische Modellierung** vorgenommen. Diese resultierte in einem quadratischen, ganzzahligen Optimierungsproblem mit Nebenbedingungen. Zugehörig wurde eine **objekt-orientierte Implementierung** in der Programmiersprache **Python** erarbeitet, welche Verständlichkeit, Wartbarkeit und insbesondere Erweiterbarkeit auf spätere komplexere Modelle sicherstellt. Basierend auf dem Modul **Qiskit Optimization** wurde eine Schnittstellenfunktion programmiert, welche Instanzen dieser Implementierung in die

Welt des Quanten-Software-Framework **Qiskit** überträgt. Ein weiterer notwendiger Schritt, um für den Quanten-Algorithmus QAOA (s.u.) zugänglich zu sein, ist die Transformation des obigen mathematischen Modells in ein quadratisches, binäres Problem ohne Nebenbedingung (**QUBO**, englisch für quadratic unconstrained binary optimization problem). Alle diese klassischen Vorarbeiten sind im **ersten Demonstrator** zu diesem Use Case dokumentiert [lama2].

Als nächster Schritt wurde der Quanten-Algorithmus QAOA (englisch für Quantum Approximate Optimization Algorithm) zuerst als geeigneter Algorithmus für das hier betrachtete Problem identifiziert, anschließend im Detail analysiert sowie auf **verschiedenen Abstraktionsstufen implementiert** und auf Simulatoren ausgeführt. Ein wichtiges Resultat aus diesem Arbeitsschritt ist die Erkenntnis, dass eine Implementierung **hoher Abstraktion**, wie sie die `qiskit`-Klasse `QAOA` [lama3] bietet, einen einfachen Zugang zu Quantenalgorithmen erlaubt, jedoch nur mäßig effizient für konkrete Probleme implementiert ist und vor allem viele Pre- und Postprocessing-Schritte ausführt, welche zwar das Ergebnis verbessern, jedoch auch sehr rechenintensiv sind. Ein **fairer Vergleich** verschiedener (klassischer und Quanten-) Algorithmen und **verlässliche Aussagen zur Effizienz und Skalierung** von QAOA sind somit **nicht möglich**. Genau andersherum verhält es sich mit eigenen **low-level Implementierungen** von QAOA. Für den hiesigen Use Case wurden sie basierend auf `qiskit.opflow` und `qiskit.quantum_info` vorgenommen. Hier ist eine **performante Umsetzung** des Quanten-Algorithmus angepasst auf den Use Case möglich und man hat **volle Kontrolle** über die verschiedenen Schritte bis zum Endresultat. Eine solche Umsetzung erfordert jedoch **umfangreiches Expertenwissen** sowohl in der Theorie des Quanten-Algorithmus als auch in der Implementierung in **Python** und **Qiskit**. Als **guter Mittelweg** hat sich die `qiskit`-Klasse `QAOAAnsatz` [lama4] erwiesen, welche die QAOA-Schaltkreise generiert, während die restliche Funktionalität des Algorithmus vom Nutzer bereitgestellt wird. Eine detaillierte Dokumentation dieses Wissens zu QAOA und dem Vorgehen und der Implementierung ist in [lama5] – [lama8] und im **zweiten Demonstrator** [lama9] verfügbar.

Aufbauend auf diesen Arbeiten ist die Ausführung der QAOA-Schaltkreise auf realen Quantencomputern (insbesondere dem **IBM Quantum System One Ehningen**) angegangen worden. Als erste große Herausforderung stand hier die **Transpilation der Schaltkreise** auf die IBMQ Systeme an. Es stellte sich heraus, dass die naive Verwendung des **Qiskit**-Transpiler keine zufriedenstellenden Ergebnisse erzielen kann. Ein **genaues Wissen** über die **verschiedenen Transpilationsschritte** und ihre Steuerung (z.B. über die Parameter des Transpilers) als auch Techniken zur **Mitigation von Hardwarefehlern** (z.B. mittels Dynamical Decoupling und Auslesefehlermitigation) sind zwingend notwendig. Im Projekt wurden diese teilweise selbst in *Arbeitspaket 3: Fehlermitigation und Zuverlässigkeit* erarbeitet aber auch durch Workshops mit IBM Experten [lama12] erlangt. In [lama7] und im **dritten Demonstrator** [lama10] wurden diese Techniken dokumentiert und demonstriert sowie Big Data Tools vorgestellt, mit denen die Resultate der Quantencomputer verarbeitet werden können. Es muss jedoch darauf hingewiesen werden, dass das gesamte Framework **Qiskit** und insbesondere auch der Transpiler sehr volatil sind und eine stete Integration bzw. Entfernung von Features stattfindet. So müssen bereits einige im Projekt erarbeiteten Techniken nicht mehr manuell implementiert werden und auch der aktuell dokumentierte Stand kann in nächster Zeit bereits überholt sein. Derselbe Effekt ist auch während der Projektlaufzeit mit dem Service **Qiskit Runtime** aufgetreten, welches sich zum Zeitpunkt der Erstellung dieses Berichts in einem so großen Umbau befindet, dass von einer Integration in den Demonstrator abgesehen worden ist.

Durch all diese Vorarbeiten war es möglich eine **systematische Untersuchung** des Fokus Use Cases als Quanten-Anwendung in Form zahlreicher Experimente insbesondere auf dem Quantencomputer **IBMQ Ehningen** durchzuführen. Um den Rahmen dieses

Berichts nicht zu sprengen, werden an dieser Stelle nur einige wenige dieser Experimente in kurzen Worten dargestellt. Alle Experimente zusammen mit ausführlichen Details und Interpretationen sind in [lama8] und im **vierten Demonstrator** [lama11] dokumentiert. Um den **Einschränkungen der NISQ-Ära** Rechnung zu tragen, wurden kleine Beispiele wie sie Abbildung 19 zeigt durchgeführt. Dafür werden **acht Qubits** benötigt, was einer Größe entspricht, bei der man von IBMQ Ehningen (Quantenvolumen 64) noch aussagekräftige Resultate erhält (16 Qubit Beispiele sind in [lama11] dokumentiert). Als **variationeller Algorithmus** besteht QAOA aus zwei Teilen, einer wird auf dem Quantencomputer ausgeführt und einer auf dem klassischen Computer. Beim **klassischen Anteil** handelt es sich um eine **Optimierung der freien Parameter** in QAOA. Abhängig von dem QAOA-Hyperparameter p hat man dabei $2p$ Parameter zu optimieren. In der Literatur wird oft berichtet, dass es sich hierbei um eine **sehr herausfordernde Aufgabe für klassische Optimierer** handelt. Dies konnte in den vorliegenden Experimenten bestätigt werden. So zeigt Abbildung 20 eine typische Optimierungslandschaft, durch welche ein Optimierer ausgehend von einem vom Nutzer festzulegenden Startpunkt navigieren muss. Dies gestaltet sich als schwierig, da die Landschaft stark oszillatorisch mit vielen lokalen Extrema ist. Auch ein Verschwinden der Gradienten (der sogenannte Effekt der Barren Plateaus) ist bereits auszumachen. Auch Abbildung 21 bestätigt nochmals die **Schwierigkeit optimale Parameter** zu erhalten. Hier sieht man das Ergebnis des klassischen Optimierers, wenn er von verschiedenen Startpunkten ausgeführt wird. Die vielen lokalen Extrema spiegeln sich hier in der **großen Varianz der Ergebnisse** wider.

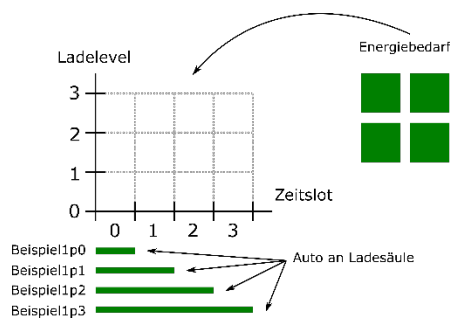


Abbildung 19: Vier Beispiele für Proof-of-Concept Modelle. Die Problemstellung besteht immer aus einem E-Auto, einer Ladesäule, vier verfügbare Zeitslots, vier mögliche Ladelevel und einer zu ladenden Energie. Die Beispiele unterscheiden sich lediglich in Ankunfts- und Abfahrtszeiten. Je kürzer das E-Auto an der Ladesäule ist, desto flacher werden später die QAOA-Schaltkreise.

Beispiel1p3

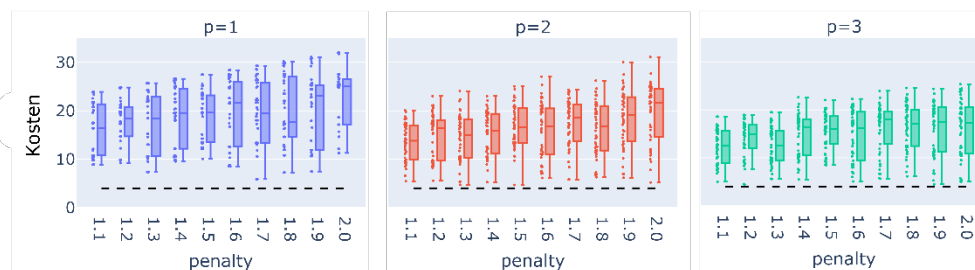


Abbildung 21 Ergebnis des klassischen Optimierers COBYLA für 50 zufällig gewählte Startpunkte $\vec{\beta}_0 \in [0, \pi]^p$ und $\vec{\gamma}_0 \in [0, 2\pi]^p$ mit einer exakten State-Vector-Simulation. Der Parameter **penalty** auf der x-Achse geht in der Transformation des Ausgangsproblems zu einem QUBO ein.

Durch die oben genannten Experimente konnten sehr gute bis optimale Wahlen für die Parameter in QAOA gefunden werden. Im nächsten Schritt wurden QAOA-Schaltkreise

mit genau diesen Parametern auf dem IBMQ Ehningen ausgeführt, um die Leistungsfähigkeit unter diesen bestmöglichen Bedingungen zu evaluieren. Es war zu erwarten, dass neben der Qubitanzahl vor allen Dingen die Tiefe der Schaltkreise eine entscheidende Rolle für die Qualität des Resultates spielt. Dies wurde im Design der Experimente zentral berücksichtigt und kann über die Verweildauer des E-Autos an der Ladesäule gesteuert werden (s.

Abbildung 19). Durch viele Experimente kann folgendes bestätigt werden: Für flache Schaltkreise im Bereich 20 – 30 CNOT Gates (nach Transpilation) können gute Resultate von IBMQ Ehningen erhalten werden (Abbildung 22 links). Für mitteltiefe Schaltkreise (40 – 70 CNOT Gates) geht die Qualität merklich zurück, kann jedoch durch Fehlermitigationstechniken wieder entschieden verbessert werden (Abbildung 22 Mitte). Für noch tiefere Schaltkreise (über 80 CNOT Gates) ist ohne Fehlermitigationstechniken keine sinnvolle Lösung mehr zu erwarten und selbst wenn diese angewandt werden, ist die Varianz in der Qualität drastisch (Abbildung 22 rechts). An dieser Stelle muss unbedingt erwähnt werden, dass die Ergebnisqualität von IBMQ Ehningen stark über die Zeit variiert (s. Abbildung 23) und somit sämtliche Aussagen nur eine Momentaufnahme sind. Auch ist in Abbildung 23 zu sehen, dass die übliche Beurteilung eines Quantenschaltkreises vor allem aufgrund der CNOT Anzahl nicht immer eine valide Einschätzung der erzielten Qualität der Quantenlösung entspricht. Hier bedarf es noch weiterer Forschung.

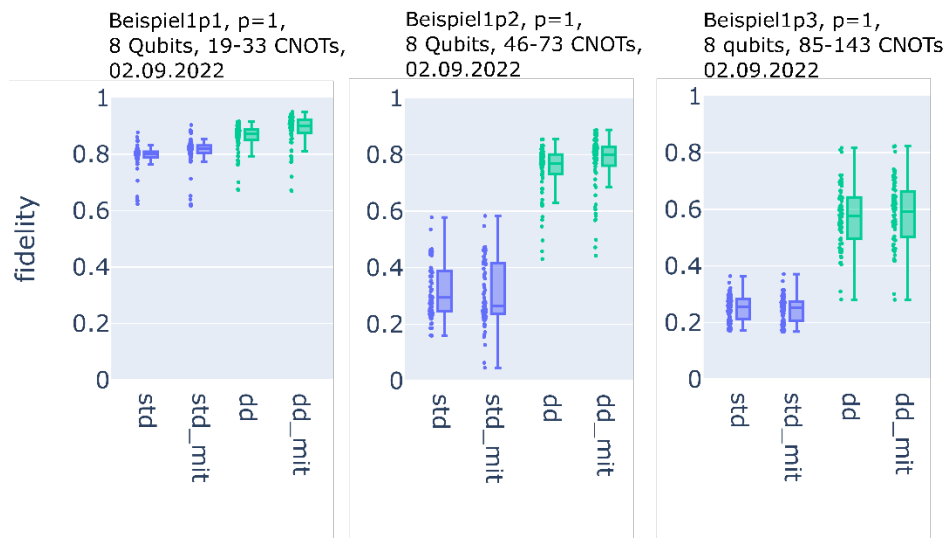


Abbildung 22 Qualität der von **IBMQ Ehningen** erhaltenen Lösung im Vergleich zu einer exakten Simulation (gemessen als **fidelity**, wobei 0 sehr schlecht und 1 sehr gut ist). Von links nach rechts sind drei verschiedene Beispiele, wobei das E-Auto links am kürzesten an der Ladesäule ist und rechts am längsten, s.

Abbildung 19. Die **Tiefe der QAOA-Schaltkreise nimmt dadurch von links nach rechts zu**. Es wurden 75 verschiedene Transpilationen durchgeführt, daher die variierende CNOT Gate Anzahl. Beschriftung der x-Achse: std=standard Transpilation mit **Qiskit**-Transpiler und **optimization_level=3**, dd=standard Transpilation und zusätzlich Dynamical Decoupling, xx_mit=xx und zusätzlich Auslesefehlerkorrektur.

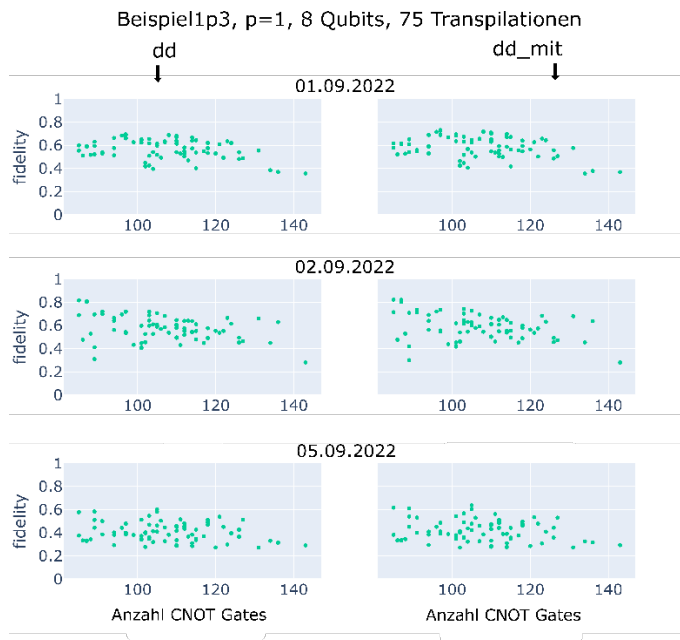


Abbildung 23 Qualität der vom **IBMQ Ehningen** erhaltenen Lösung im Vergleich zu einer exakten Simulation des gleichen Beispiels an **verschiedenen Tagen**. Zur Bedeutung dd und dd_mit und zum Begriff fidelity, s. Abbildung 22.

Die Herleitung und Modellierung des Problems, sowie die erhaltenen Ergebnisse wurden in diversen Großveranstaltungen wie z.B. dem »Quantum Village Ehningen« und in weiteren internen und externen Entwicklertreffen präsentiert (siehe Veranstaltungsliste AP7). Des Weiteren wurden zahlreiche Diskussion zu den fachlichen Herausforderungen des Use Cases mit Experten z.B. von IBM [lama12], des Fraunhofer Programmer Teams, in den SEQUOIA-PlanQK Entwickelndenworkshops und mit weiteren Fachexperten geführt.

[lama1] <https://www.lama.zone/>

[lama2] A. Sturm, SEQUOIA DEMONSTRATOR Energy Use Case, notebook_1, [LINK](#)

[lama3] `qiskit.algorithms.minimum_eigen_solvers.qaoa.QAOA`, Version `qiskit 0.39.2`, `qiskit-terra 0.22.2`

[lama4] `qiskit.circuit.library.n_local.qaoa_ansatz.QAOAAnsatz`, Version `qiskit 0.39.2`, `qiskit-terra 0.22.2`

[lama5] A. Sturm, QAOA: intensive course, [IAO-IAF-Schulungsprogramm](#), 18.01.2022,

[lama6] D. Pranjic, Theory of QAOA, [IAO-IAF-Schulungsprogramm](#), 24.05.2022

[lama7] A. Sturm, QAOA Implementation, [IAO-IAF-Schulungsprogramm](#), 24.05.2022

[lama8] A. Sturm, Energy Use Case: Optimization of Charging Schedules for Electric Cars, [IAO-IAF-Schulungsprogramm](#), 27.09.2022

[lama9] A. Sturm, SEQUOIA DEMONSTRATOR Energy Use Case, notebook_2, [LINK](#)

[lama10] A. Sturm, SEQUOIA DEMONSTRATOR Energy Use Case, notebook_3, [LINK](#)

[lama11] A. Sturm, SEQUOIA DEMONSTRATOR Energy Use Case, notebook_4, [LINK](#)

[lama12] 13.12.2021: IBM-Workshop on Transpilation, 05. / 06.04.2022: IBM-Workshop on circuit mapping, dynamical decoupling and QAOA, 11.22: IBM Quantum Summit 2022, Practitioners Forum, Yorktown Lab Visit

Im zweiten Fokus Use Case wird das Thema »**Quantenbasierte numerische Strömungssimulation**« behandelt. Ziel ist es, heutige Computational Fluid Dynamics

(CFD) Rechnungen mit Quantencomputern zu beschleunigen und/oder in ihrer Approximationsgüte zu verbessern. Der hier betrachtete Use Case wurde in enger Zusammenarbeit zwischen dem **Fraunhofer IAO und der Firma Simerics** entwickelt.

Überblick der wissenschaftlichen Arbeiten

Das Grundprinzip heutiger CFD-Rechnungen basiert auf dem Lösen (gekoppelter) Navier-Stokes-Gleichungen in drei Raum- und einer Zeitdimension. Um ein für die NISQ-Ära angemessenes Zielproblem zu definieren, wurde zusammen mit der Firma Simerics abgeklärt, initial die dreidimensionale Navier-Stokes-Gleichung unter bestimmten Randbedingungen zu lösen: Dabei wurde auf zwei Raum- und eine Zeitdimension fokussiert, wobei wir in longitudinale Richtung (in Strömungsrichtung) der Einfachheit halber zuerst Translationsinvarianz annehmen (d.h. effektiv nur zwei Dimensionen analysieren, eine Zeit- und eine Raumdimension). Dies ist schematisch für einen fixen Zeitpunkt in Abbildung 24 gezeigt.

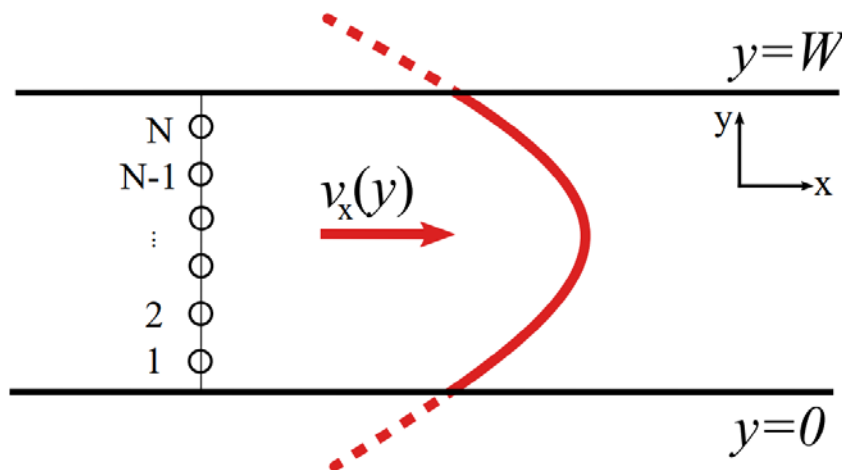


Abbildung 24 Schematisches Diskretisieren und Flussprofil einer zweidimensionalen Strömung zu einem gegebenen Zeitpunkt $t = t_0$.

Die Navier-Stokes Gleichung für den hydrodynamischen Fluss einer inkompressiblen Flüssigkeit mit Translationsinvarianz in die $\vec{e}_x$ -Richtung und unter verschwindender Fluidgeschwindigkeit in $\vec{e}_y$ -Richtung ist in zwei Raum- und einer Zeitdimension gegeben als

$$\frac{\partial u_x}{\partial t} = \nu \frac{\partial^2 u_x}{\partial y^2},$$

wobei $u_{x_0}(t, y)$ die Strömungsgeschwindigkeit an der Stelle $x = x_0$ als Funktion der Koordinaten y und t definiert. Die kinematische Viskosität ist durch ν gegeben. Diese spezielle Art der Gleichung ist auch als zweidimensionale Wärmeleitungsgleichung bekannt (eine Dimension Raum, eine Dimension Zeit), wobei die Viskosität als Temperaturleitfähigkeit interpretiert werden kann.

Eine populäre Methode zur Berechnung von approximativen Lösungen der Wärmeleitungsgleichung besteht darin die Differenzialgleichung mittels der Finite Differenzen Methode im Ort und mit dem impliziten Eulerverfahren in der Zeit zu diskretisieren. Hierbei handelt es sich um ein numerisch sehr stabiles Verfahren, welches jedoch mit dem Preis kommt, dass in jedem Zeitschritt ein lineares Gleichungssystem gelöst werden muss. Genau dies soll im hiesigen Use Case durch passende (Quanten-) Algorithmen geschehen. Es geht also primär um das Lösen linearer Gleichungssysteme $A\vec{x} = \vec{b}$. Die detaillierte Herleitung dieser Gleichung hinsichtlich der initialen Differentialgleichung (DGL) ist unserem Demonstrator zu entnehmen (<https://websites.fraunhofer.de/sequoia/use-cases/>).

Insbesondere ist die hier diskutierte DGL in Ihrer diskretisierten Form durch eine tri-diagonale Matrix beschrieben, das heißt eine Matrix mit lediglich besetzter Diagonale und Nebendiagonalen.

Das Lösen von linearen Gleichungssystemen mittels des »reinen« Quantenalgorithmus HHL [QCFD_1] hat sich aufgrund der zu hohen Schaltkreistiefe als nicht zielführend auf aktueller Quantenhardware erwiesen (vgl. Schaltkreistiefe und Qubit-Skalierung in Abbildung 25). Daraufhin wurde der Variational Quantum Linear Solver [QCFD_2] (VQLS) als variationelle, hybride Alternative in einer eigenen Implementierung untersucht. Dies geschah sowohl in Simulationen mit und ohne Noise-Modell als auch auf realer Quantenhardware (insbesondere auf dem IBM Q System One in Ehningen).

VQLS:			HHL:		
Matrix	Qubits	Tiefe	Matrix	Qubits	Tiefe
4x4	3	11	4x4	6	1096
8x8	4	15	8x8	8	36962

Abbildung 25 Schaltkreistiefe und Qubitanzahl des VQLS- vs. HHL Algorithmus in Abhängigkeit der Größe des zu lösenden linearen Gleichungssystems.

Eine zentrale Rolle im VQLS Algorithmus spielt die Wahl der Kostenfunktion. Anhand deren Wert bemisst sich die Güte des iterativen Lösungsvorschlags im hybriden quanten-klassischen Optimierungsprozess. Eine hohe Kostenfunktion bedeutet eine schlechte Approximation der Lösung des linearen Gleichungssystems und umgekehrt.

Im VQLS Algorithmus gibt es zwei Alternativen die Kostenfunktion zu definieren, welche konkret Abbildung 26 zu entnehmen sind: Die globale Kostenfunktion C_G und Ihr lokales Pendant C_L .

Während die globale Kostenfunktion C_G keine lokalen Minima in der Optimierungslandschaft aufweist, zeigt sie jedoch sogenannte Barren Plateaus bei skalierender Qubitanzahl. Die lokale Kostenfunktion C_L hingegen umgeht jenen Effekt und geht ebenfalls mit deutlich geringeren Schaltkreistiefen einher (insbesondere mit nur zwei Hadamard Tests pro Evaluierung). Sie scheint daher in der NISQ Ära präferiert, kommt jedoch mit einer deutlichen Herausforderung: Man kann im klassischen Optimierungsschritt leicht in lokalen Minima ‚hängenbleiben‘. Dies definiert eine besondere Herausforderung an die zu wählenden klassischen Optimierer.

Der Übersichtlichkeit halber haben wir in Abbildung 26 eine performante Lösung eines linearen Gleichungssystems sowohl mit der lokalen als auch der globalen Kostenfunktion dargestellt. Man sieht, dass zwar in beiden Ansätzen die klassische Lösung reproduziert werden kann, die lokale Methode allerdings besser konvergiert. Dieses Verhalten ist insbesondere bei größeren Systemen deutlich zu erkennen, kommt aber mit dem Problem lokaler Minima, bei denen eine konvergierte Lösungen erreicht werden kann, welche jedoch einen völlig falschen Lösungsvektor darstellt.

$$C_G = 1 - \frac{\sum_{l,l'} c_l c_{l'}^* \langle 0|V^\dagger A_{l'}^\dagger U|0\rangle \langle 0|U^\dagger A_l V|0\rangle}{\sum_{l,l'} c_l c_{l'}^* \langle 0|V^\dagger A_{l'}^\dagger A_l V|0\rangle}$$

$$C_L = \frac{1}{2} - \frac{1}{2n} \frac{\sum_{j=0}^{n-1} \sum_{l,l'} c_l c_{l'}^* \langle 0|V^\dagger A_{l'}^\dagger U Z_j U^\dagger A_l V|0\rangle}{\sum_{l,l'} c_l c_{l'}^* \langle 0|V^\dagger A_{l'}^\dagger A_l V|0\rangle}$$

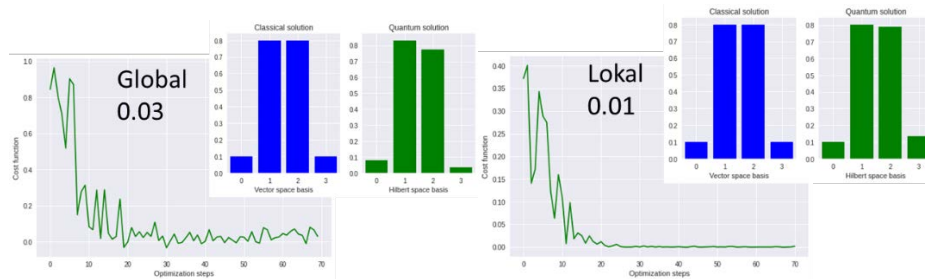


Abbildung 26 Iteratives Lösen des linearen Gleichungssystems $A\vec{x} = \vec{b}$ mittels der globalen Kostenfunktion C_G und lokalen Kostenfunktion C_L mittels QASM Simulator. Die Definition der einzelnen Variablen und Operatoren sind [QCFD_2] zu entnehmen.

Bei systematischer Analyse der gängigen klassischen Qiskit-Optimierer hat sich herausgestellt, dass die Performanz dieser stark vom Noise-Modell oder dem (echten) Rauschen des IBM Quantum System One abhängig ist. Ohne Fehlerquellen kristallisierte sich bei der Analyse der COBYLA als bester Optimierer heraus, während im Beisein von Fehlerquellen der SPSA Optimierer das beste Ergebnis lieferte.

In Abbildung 27 sind zwei Beispiele des VQLS Algorithmus auf dem IBM Quantum System One Ehningen mit lokaler Kostenfunktion und unterschiedlichen klassischen Optimierern dargestellt, welche ein schlechtes Konvergenzverhalten zeigen, während Abbildung 28 eine erfolgreiche Minimierung der Kostenfunktion zeigt, allerdings erst nach Kalibrierung.

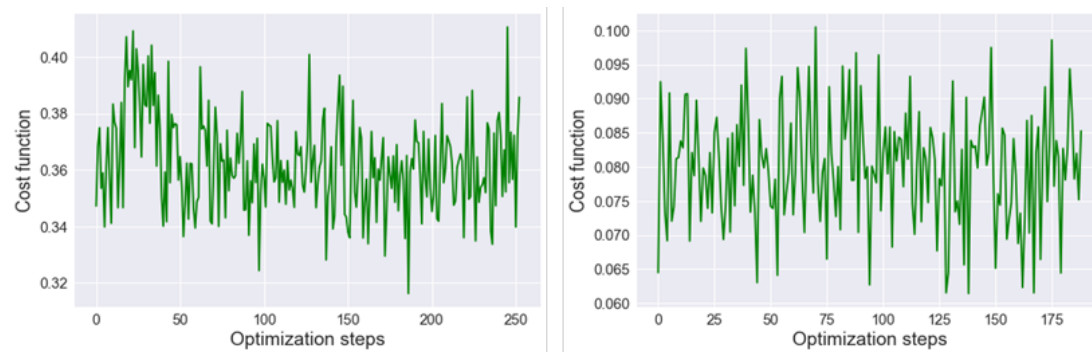


Abbildung 27 Konvergenzverhalten der klassischen Optimierer »Conjugate Gradient« (links) und »Truncated Newton« (rechts) im VQLS Algorithmus auf dem IBM Quantum System One Ehningen mit lokaler Kostenfunktion. Es ist keine Konvergenz zu erkennen.

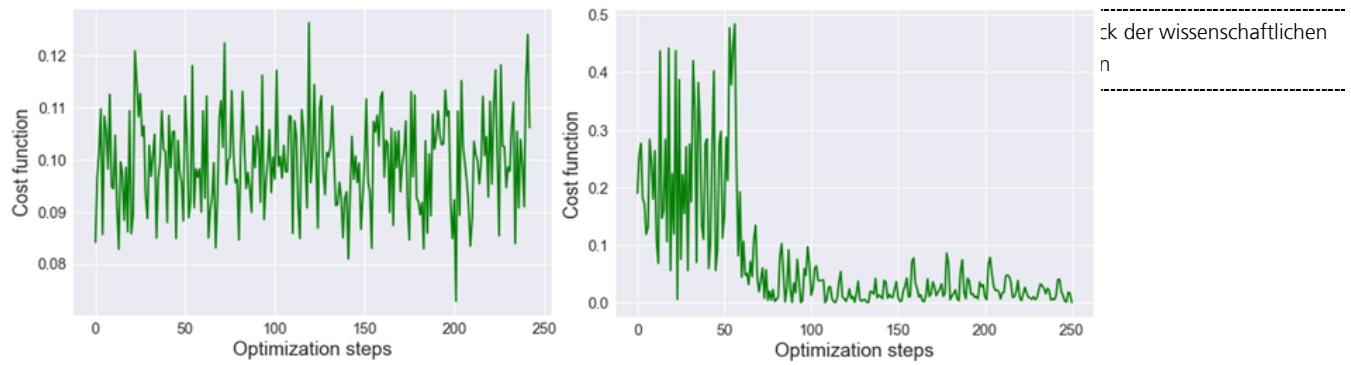


Abbildung 28 Konvergenzverhalten des klassischen Optimierers SPSA im VQLS Algorithmus auf dem IBM Quantum System One Ehningen mit lokaler Kostenfunktion ohne Kalibrierung (links) und mit Kalibrierung (rechts). Nach Kalibrierung ist eine deutliche Konvergenz zu erkennen.

Wie erwähnt ist neben der Konvergenz des klassischen Optimierers alleine die zugehörige Qualität der Lösung im VQLS Algorithmus von besonderem Interesse, da gerade auch lokale Minima in der lokalen Kostenfunktion approximiert werden können. Dies wurde in zahlreichen Beispielen (für verschiedene Matrizen und Vektoren) auf dem IBM Quantum System One untersucht, wobei aufgrund der Beschränkungen der NISQ-Ära nur kleine lineare Systeme betrachtet werden konnten. Abbildung 29 stellt das Ergebnis des VQLS Algorithmus für den Fall einer 4×4 Identitätsmatrix dar. Hier ist eine gute Qualität der Lösung zu erkennen. Wird die Schwierigkeit auf eine 8×8 Identitätsmatrix erhöht, ist bereits eine Verschlechterung des Resultates zu sehen (siehe Abbildung 30). Für den Fall einer Tridiagonalmatrix, wie sie oft bei Diskretisierungen von partiellen Differentialgleichungen auftritt (siehe obige Diskussion), ist in Abbildung 30 zu erkennen, dass die Lösung des VQLS Algorithmus selbst für 4×4 Systeme nur noch von mäßiger Qualität ist.

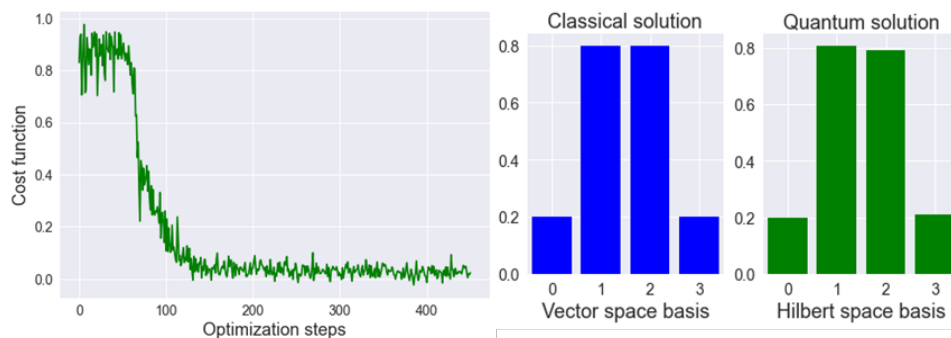


Abbildung 29 Konvergenzkurve (links) und Resultat (rechts) des VQLS Algorithmus auf dem IBM Quantum System One in Ehningen. Als Optimierer kam SPSA zum Einsatz, als Matrix A die 4×4 Identitätsmatrix und als rechte Seite $\vec{b}$ der Vektor $[0.2, 0.8, 0.8, 0.2]$. In der Mitte ist die exakte Lösung dargestellt.

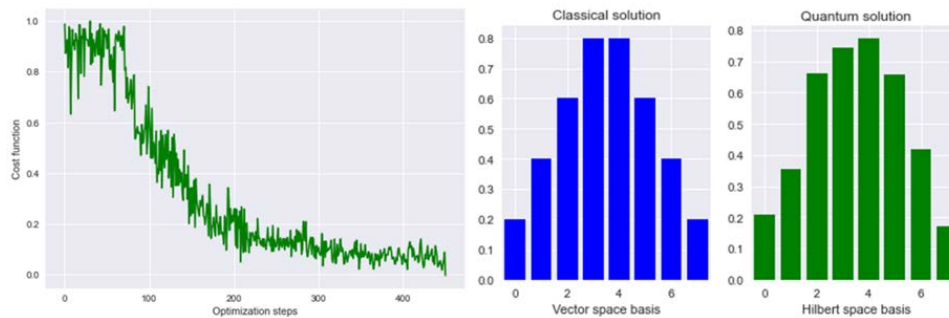


Abbildung 30 Situation wie in Abbildung 29, aber als Matrix A kam die 8 x 8 Identitätsmatrix zum Einsatz und als rechte Seite $\vec{b}$ der Vektor $[0.2, 0.4, 0.6, 0.8, 0.8, 0.6, 0.4, 0.2]$.

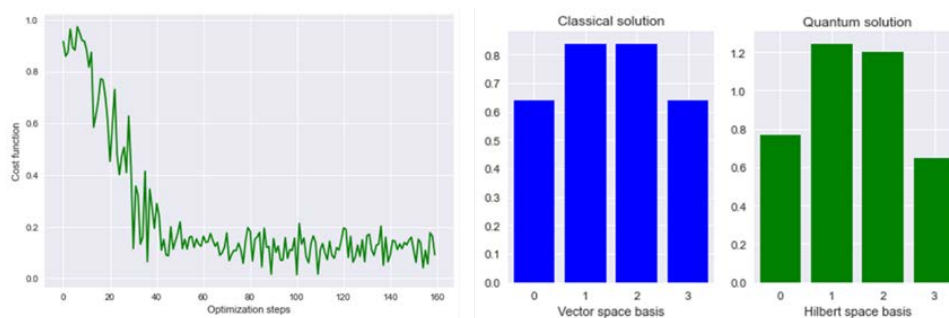


Abbildung 31 Situation wie in Abbildung 29 aber als Matrix A wurde eine tridiagonale Matrix verwendet sowie als rechte Seite $\vec{b}$ der Vektor $[1, 1, 1, 1]$.

Um die Performanz des Algorithmus hinsichtlich der Lösung auch komplexerer Strömungsprobleme zu erhöhen, wurden im letzten Schritt in unsere VQLS Algorithmen die in AP3 entwickelten Fehlermitigations-Strategien inkludiert. Jene Lösung, sowie unser finaler Demonstrator wurde als interaktives Jupyter Notebook auf der KQCBW Recheninfrastruktur deployt und ist dort ab Anfang Januar 2023 der breiten Öffentlichkeit open source zugänglich.

Link: <https://websites.fraunhofer.de/sequoia/use-cases/>

Abschließend sei anzumerken, dass noch weitere Forschung für den Dateninput (Encoding des $\vec{b}$ Vektors und Pauli-Zerlegung der Matrix A) sowie den Datenoutput (Auslesen des Lösungsvektors) notwendig ist, welche sonst bei Skalierung der Systemgröße mögliche Quantenvorteile wieder revidieren könnten.

Die Herleitung und Modellierung des Problems, sowie erste Ergebnisse wurden in zwei »**Quantum Village Ehningen**« Großveranstaltungen und in diversen internen und externen Entwicklertreffen präsentiert (siehe Veranstaltungsliste AP7). Diskussion der fachlichen Herausforderungen erfolgte u.a. mit Experten von IBM, dem Fraunhofer Programmierer Team, in PlanQK Entwickelndenworkshops und mit weiteren Experten.

[QCFD_1] Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd: Quantum Algorithm for Linear Systems of Equations. In: Phys. Rev. Lett. 103, 2009

[QCFD_2] C. Bravo-Prieto, R. LaRose, M. Cerezo, Y. Subasi, L. Cincio, and P. Coles: Variational Quantum Linear Solver. <https://arxiv.org/abs/1909.05820>

Der dritte Use Case »**Optimierung einer Auftragssequenzierung in der diskreten Fertigung**« behandelt die Verteilung von Produktionsaufträgen auf unterschiedlichen Produktionsressourcen und wurde gemeinsam vom **Fraunhofer IPA mit der KUMAVISION AG** erarbeitet. Diese Problemstellung existiert in nahezu allen fertigen Branchen jedoch variiert sie meist im Detail bei den abzubildenden Einflussfaktoren (z.B. Materialverfügbarkeit, Mitarbeiterfähigkeiten, Anlagenkapazitäten etc.), Randbedingungen oder zu optimierenden Zielgrößen.

Überblick der wissenschaftlichen Arbeiten

Eine in der Literatur weit verbreitete Definition dieser Problemart ist das sogenannte Job Shop Scheduling Problem (JSP). Dabei handelt es sich um eine kondensierte Variante des Problems, die bei einem klassischen JSP folgende Einflussfaktoren definiert: Abzuarbeiten sind eine Reihe von Produktionsaufträgen (Jobs), bei denen jeder Job aus beliebig vielen Fertigungsschritten (Operationen) bestehen kann, die in einer vorgegebenen Reihenfolge abgearbeitet werden müssen. Jeder Fertigungsschritt muss auf einer spezifischen Anlage (Maschine) durchgeführt werden. Die Anlagen unterliegen dabei einer Reihe von Einschränkungen. So müssen die vorangegangenen Fertigungsschritte eines jeden Jobs vollendet sein, bevor der aktuelle Fertigungsschritt bearbeitet werden kann. Eine Anlage kann außerdem zu jedem Zeitpunkt nur einen Fertigungsschritt bearbeiten. Fertigungsaufträge können nicht pausiert werden. Ziel des klassischen JSP ist die Minimierung der Gesamtproduktionsdauer (engl. Makespan).

Abbildung 32 zeigt eine beispielhafte Lösung eines JSP auf drei Maschinen. Die Produktionsschritte, die zu ein und demselben Job gehören sind hier farblich zusammengefasst. Die Anordnung führt zu einer Gesamtproduktionsdauer von 5 Zeiteinheiten.

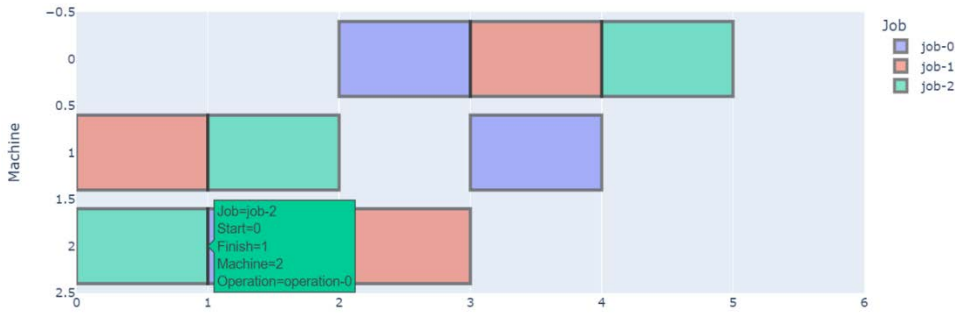


Abbildung 32: Exemplarische Lösung für ein Job Shop Scheduling Problem

$$\begin{aligned}
& \min_x \hat{C}_{max} \\
& + p_4 \cdot \sum_{j=1}^N \sum_{m=1}^M \left(\sum_{t=0}^T x_{j,m,t} - 1 \right)^2 + p_5 \cdot \sum_{j=1}^N \sum_{(m_1, m_2) \in P_j} \sum_{t_1 + d_{j,m_1} > t_2} x_{j,m_1,t_1} \cdot x_{j,m_2,t_2} \\
& + p_6 \cdot \sum_{m=1}^M \sum_{j_1 \neq j_2} \sum_{t_1 \leq t_2 < t_1 + d_{j_1,m}} x_{j_1,m,t_1} \cdot x_{j_2,m,t_2} + p_7 \cdot \sum_{j=1}^N \left(\sum_{t=0}^T (t \cdot x_{j,\sigma_M^j,t}) + d_{j,\sigma_M^j} + \hat{s}_j - \hat{C}_{max} \right)^2 \\
& \hat{C}_{max} \geq 0, \hat{s}_j \in \mathbb{Z}, x_{j,m,t} \in \{0, 1\}, \\
& \forall j = 1, \dots, N, \forall m = 1, \dots, M, \forall t = 0, \dots, T.
\end{aligned}$$

Abbildung 33: Formulierung des Time-Indexed-JSP-QUBO

In AP5 wurde das JSP in Form des sogenannten Time-Indexed-JSP-QUBO [JSP1] erfolgreich implementiert. Es wurden diverse Anpassungen des QUBOs evaluiert. Dabei wurden alternative Formulierungen für einzelne Penalty Terms der Lösungsfunktion implementiert. Es konnte vor allem festgestellt werden, dass eine Anpassung am Penalty Term - der dazu führen soll, dass alle Operationen tatsächlich starten - zu besseren Ergebnissen führt. Außerdem wurden Heuristiken implementiert, die die benötigten Binärvariablen und den damit verbundenen Lösungsraum reduzieren (engl. Variable Pruning). Beispielsweise wurden absurde Startzeiten für Fertigungsschritte aus dem Lösungsraum entfernt. Dies führte für eine Minimalinstanz des QUBOs zu einer Formulierung, die 9-Qubits benötigt, um das Problem zu beschreiben.

Mithilfe der Qiskit Implementierung des QAOA-Algorithmus in Verbindung mit einem MinimumEigenOptimizer konnten somit minimale JSP-Probleminstanzen auf dem IBM Q System One in Ehningen erfolgreich gelöst werden. Die implementierte Lösung wurde in Form eines Demonstrators unter https://sequoia.iao.fraunhofer.de/user/job-shopscheduling_ipa/lab veröffentlicht.



Abbildung 34 Exemplarische Wahrscheinlichkeitsverteilung der Lösungen bei Nutzung des QAOA-Ansatzes auf dem IBM System One (optimale Lösung rot markiert)

Erste Tests zeigten, dass der weitere Fokus verstärkt auf intelligente Heuristiken zur Findung von passenden Penalty Term Faktoren und Ober- sowie Untergrenzen für die ganzzahligen Variablen einer gegebene JSP-Probleminstanz gelegt werden muss, um gute Ergebnisse für größere Probleminstanzen zu erhalten. Hierfür wäre die Anwendung von maschinellen Lernverfahren ein denkbarer Ansatz.

Des Weiteren wurden Tests durchgeführt, um die Ergebnisse des QAOA-Ansatzes mithilfe von unterschiedlichen klassischen Optimierern und verschiedenen Transpilierungen des Schaltkreises weiter zu verbessern.

Abschließend wurde versucht durch weitere Heuristiken den JSP-Use-Case für größere Probleminstanzen mithilfe des QAOA-Algorithmus zu lösen. Hierzu wurde z.B. für Probleminstanzen, welche eine hohe Bearbeitungsdauer aufweisen, durch sogenanntes *Operation Duration Squashing* eine Variablenreduktion erzielt. Mittels eines iterativen Ansatzes wurde versucht das JSP-Problem in kleinere Teilprobleme aufzubrechen, die wiederum sukzessive mithilfe des QAOA-Algorithmus auf dem IBMQ Backend in Ehningen gelöst werden sollten. Eine Veröffentlichung, die die beiden beschriebenen Verbesserungen mithilfe von Quantum Annealing auf D-Wave Systemen implementiert hat und vielversprechende Ergebnisse aufweist [JSP2] sollte als Grundlage dienen und auf

dem gatter-basierten IBM Backend angepasst werden, um dort große JSP-Probleminstanzen zu lösen. Hierbei wurden allerdings keine präsentierbaren Ergebnisse erzielt.

[JSP1] <https://ojs.aaai.org/index.php/ICAPS/article/view/19826> | Citation: Zhang, J., Lo Bianco, G. and Beck, J. C. (2022) »Solving Job-Shop Scheduling Problems with QUBO-Based Specialized Hardware«, *Proceedings of the International Conference on Automated Planning and Scheduling*, 32(1), pp. 404-412. doi: 10.1609/icaps.v32i1.19826.

[JSP2] https://link.springer.com/chapter/10.1007/978-3-030-50433-5_39 | Citation: Kurowski, K., Węglarz, J., Subocz, M., Różycki, R., Waligóra, G. (2020). Hybrid Quantum Annealing Heuristic Method for Solving Job Shop Scheduling Problem. In: ICCS 2020. Lecture Notes in Computer Science, vol 12142. Springer. https://doi.org/10.1007/978-3-030-50433-5_39

Der vierte Fokus Use Case »**Optimale Auslegung von Schnittmustern in der Blechfertigung**« wurde gemeinsam **vom Fraunhofer IPA mit der Trumpf-Gruppe** erarbeitet. Gegenstand des Use Cases ist es, in der Blechfertigung beim Schneiden von Blechtafeln mit Laser- oder Stanzmaschinen die Materialausnutzung pro Blechtafel zu maximieren, um eine ressourcenoptimierte Fertigung zu ermöglichen.

Der Use Case lässt sich auf das *unregelmäßige Packproblem auf einem Streifen in 2D* reduzieren. Hierbei geht es um das Auffinden von Position und Orientierung von N Polygonen beliebiger Form – im Folgenden Teile genannt - auf einer rechteckigen Fläche mit bekannter und fester Höhe H und variabler Länge L , sodass alle Teile komplett in die Fläche passen, keine Teile überlappen und die Länge L der Fläche minimal ist. In der folgenden Abbildung 35 ist ein manuell angefertigtes und nichttriviales Beispiel mit 12 Teilen (linkes Bild) und einer quasi-optimalen Lösung (rechtes Bild) mit entsprechender Position und Orientierung der 12 Teile zu sehen.

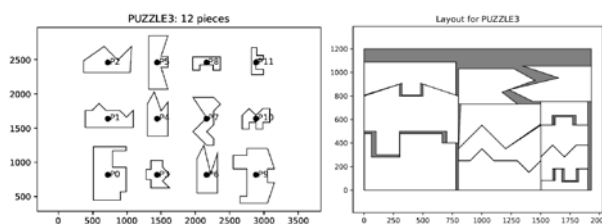


Abbildung 35 Beispielhafte Darstellung typischer Teile, die in unseren Berechnungen platziert werden (links). Resultat eines erfolgreichen Packvorgangs (rechts).

Neben der unmittelbaren Bedeutung für den vorliegenden Use Case ist dieses Problem darüber hinaus auch von erheblicher wirtschaftlicher Bedeutung. Durch die Lösung des Problems kann die verarbeitende Industrie ihre Kosten für Materialien wie Stoff, Leder, Karton, Holz, Kunststoff, Glas, Keramik oder Metall minimieren. Diese werden für die Herstellung von Einzelteilen verwendet, aus denen sich Endprodukte zusammensetzen, wie z. B. Kleidung, Autos, Schiffe, Elektrogeräte, Maschinen und Verpackungen. Es sei darauf hingewiesen, dass dieses Problem auch von großer ökologischer Bedeutung ist, da es die Einsparung knapper Ressourcen ermöglicht und zur Abfallvermeidung beiträgt.

Das Packproblem ist ein komplexes kombinatorisches Optimierungsproblem mit exponentieller Komplexität. Unsere Lösung ist ein hybrider quanten-klassischer Algorithmus:

1. **Analyse der geometrischen Kompatibilität zwischen Teilen.** Für jedes beliebige Paar der Teile P und P' bestimmen wir die optimale Position und Orientierung von P' relativ zu P , in der Art, dass der verschwendete Raum zwischen den Teilen minimiert wird. Der verschwendete Raum, auch Verschnitt genannt, korrespondiert dabei mit der konvexen Hülle der Anordnung beider Teile.
2. **Bildung von Gruppen geometrisch kompatibler Teile.** Wir nennen zwei Teile *geometrisch kompatibel*, wenn ihre geometrische Inkompatibilität hinreichend gering ist, z. B. wenn sie unter einem Schwellenwert von 20 % liegt. Wir bilden dann viele Gruppen, die wir *Cluster* nennen, von zufälliger Größe, in denen die Teile geometrisch kompatibel sind. Jedes Teil innerhalb eines Clusters muss mindestens mit einem anderen Teil im selben Cluster geometrisch kompatibel sein.
3. **Bildung kompakter Rechtecke aus jedem Cluster.** Wir ordnen die Teile jedes Clusters räumlich nacheinander an, indem wir die folgende »greedy« Heuristik verwenden. Gegeben eine Folge von Teilen $P_{i_1}, P_{i_2}, \dots, P_{i_n}$ wird das erste Teil am Ursprung mit dem Orientierungswinkel 0 Grad platziert. Für jedes neu zu platzierende Teil P_{i_k} umkreise mit dem nächsten Teil $P_{i_{k+1}}$ das zuletzt platzierte Teil und probiere alle nicht überlappenden Positionen und Ausrichtungen aus. Behalte die Platzierung bei, die die Fläche des minimalen rechteckigen Begrenzungsrahmens minimiert, der die Teile $P_{i_1}, P_{i_2}, \dots, P_{i_k}, P_{i_{k+1}}$ umschließt. Mit den nächsten Teilen wird in ähnlicher Weise verfahren, bis alle Teile der Folge platziert sind. Ziel ist es, die Teile in jedem Cluster so zu platzieren, dass die Gesamtfläche innerhalb des Begrenzungsrahmens der Teile, die nicht von den Teilen besetzt ist (der Verschnitt), minimal ist. Für einen Cluster mit n Teilen gibt es a zunächst $n!$ mögliche Reihenfolgen. Bei einem Cluster mit 4 Teilen gibt es beispielsweise 24 solcher Sequenzen. Bei einem Cluster mit 5 Teilen wächst die Anzahl der Sequenzen auf 120 und bei 6 Teilen auf 720... Die Anordnung der Teile gemäß einer gegebenen Sequenz ist sehr rechenaufwändig, daher wollen wir dies nicht für jede mögliche Sequenz tun. Wir brauchen also ein einfaches Verfahren, um eine gute Reihenfolge direkt zu bestimmen. Wir können beobachten, dass die gesamte verschwendete Fläche ungefähr durch die Summe der verschwendeten Flächen zwischen jedem Paar von Teilen entlang der Sequenz begrenzt wird. Auf der Grundlage dieser Beobachtung können wir die Platzierungsreihenfolge wählen, die den Gesamtverschnitt minimiert. Sie entspricht der Summe $\sum_{k=1}^{n-1} d(P_{i_k}, P_{i_{k+1}})$. Die Suche nach dieser besten »Reihenfolge« für das Packen einer Menge von Teilen ist mathematisch äquivalent zur Lösung des so genannten *Traveling Salesman Problems* (TSP), das im Allgemeinen wie folgt formuliert wird. Gegeben ein Set von n Städten, benannt durch ihre Indices $1, 2, \dots, n$, finde eine Route $i_1, i_2, \dots, i_n$, die jede Stadt exakt einmal besucht und die Reisedistanz $\sum_{k=1}^{n-1} d(i_k, i_{k+1})$ minimiert. Um das TSP auf einem Quantencomputer zu lösen, kodieren wir das Optimierungsproblem zunächst als QUBO Problem (Lucas, 2014). $x_{i,p}$ sei eine binäre Variable, die gleich 1 ist, wenn der Pfad im Schritt p durch die Stadt i führt. Das TSP ist äquivalent zur Minimierung der folgenden Kostenfunktion, mit dem Bestrafungskoeffizienten A unter der Bedingung $A > \max_{i,j} d(i,j)$:

$$C(x) = \sum_{i=1}^n \sum_{j=1}^n d(i,j) * \sum_{p=1}^{n-1} x_{i,p} x_{j,p+1} + A \sum_{p=1}^n \left(1 - \sum_{i=1}^n x_{i,p} \right)^2 + A \sum_{i=1}^n \left(1 - \sum_{p=1}^n x_{i,p} \right)^2$$

Der QAOA von (Fahri, 2014), erlaubt es, eine beliebige Funktion $C(z)$ auf Bit-Strings $z = z_1 z_2 \dots z_n$ mit einer Länge von n Bits zu maximieren. Wenn wir die Lösungen von QAOA als Sequenzen verwenden und die Teile pro Cluster »greedy« in der Reihenfolge entsprechend der Sequenzen anordnen, erhalten wir räumliche Anordnungen, die kompakt in den Begrenzungsrahmen passen.

4. **Bildung zufälliger Partitionen aus der Menge der Teile, indem die besten rechteckigen Anordnungen verwendet werden.** In der Mathematik ist eine *Partition* einer Menge eine Gruppierung ihrer Elemente in nicht leere und sich nicht überschneidende Teilmengen, und zwar so, dass jedes Element in genau einer Teilmenge enthalten ist. Wir interessieren uns nun für die Erzeugung von Partitionen der Menge von N zu verpackenden Stücken, oder äquivalent, der Menge der Indizes $\{0, 1, \dots, N - 1\}$ der Stücke. Wir werden nur Partitionen betrachten, bei denen die Gruppen aus Clustern von Teilen bestehen, die wir zuvor zu kompakten Rechtecken angeordnet haben.
5. **Packe die Rechtecke.** Für jede Partition lässt sich das ursprüngliche Packproblem auf ein einfacheres Problem reduzieren, bei dem nur Rechtecke aus den Clustern der Partition auf der Fläche platziert werden müssen. Das Problem, Rechtecke in einen rechteckigen Behälter zu packen, wird als Rechteckpacken bezeichnet, ein NP-schweres Optimierungsproblem, das wir heuristisch mithilfe des MAXRECTS-Algorithmus (Jylänki, 2010) lösen.
6. **Finale Schritte.** Die gepackten Rechtecke werden durch die Clusterteile ersetzt. So erhalten wir ein Layout, das die Position und die Ausrichtung der einzelnen Teile angibt. Dieses Layout wird durch lokale Optimierung weiter verbessert. Dabei werden alle Teile so weit wie möglich nach links verschoben, wobei Bewegungen nach oben und unten erlaubt sind. Wir optimieren die Layouts, die wir erhalten haben, lokal und behalten das beste Layout unter allen erhaltenen Layouts, d. h. das Layout, das die geringste Länge L erfordert. Dieses beste Layout wird einem abschließenden globalen Optimierungsverfahren unterzogen, bei dem das am weitesten rechts liegende Teil in den am weitesten links liegenden unbesetzten Raum oder »Hohlraum« verschoben wird. Das neue Teil ganz rechts im Layout wird, wenn möglich, ebenfalls nach links verschoben. Der Verschiebungsprozess wird so lange wiederholt, bis das aktuelle Teil ganz rechts nicht mehr verschoben werden kann.

Wir haben unseren Ansatz experimentell an 3 »puzzleartigen« Pack-Instanzen und 5 Instanzen SHAPES1, SHAPES2, SHIRTS, TROUSERS und SWIM getestet, die in (Oliveira, 2000) definiert sind und in der Literatur häufig zum Benchmarking der Leistung von Pack-Algorithmen verwendet werden. Die Versuchsanordnung sieht folgendermaßen aus. Wir erlauben 8 Winkel für die Ausrichtung der Stücke (Vielfache von 45 Grad). Der QAOA-Schaltkreis besitzt $p=5$ Wiederholungen und die Schaltungsparameter β und γ werden mit dem COBYLA-Optimierer ermittelt. Die Heuristik ist in Python implementiert und basiert auf den Python-Modulen shapely, qiskit und rectpack. Wir verwenden 16 Qubits, die es erlauben, TSPs mit einer maximalen Anzahl von 4 Städten zu lösen. Der qasm-Simulator von qiskit wird zur Optimierung der Schaltkreisparameter β und γ verwendet. Die Schaltkreise werden dann auf dem Quantum System One von IBM in Ehningen ausgeführt. Die Anzahl der Shots wurde auf 1000 pro Schaltkreis festgelegt. Für jeden Schaltkreis wurde der Zustand mit der höchsten gemessenen Anzahl als Lösung ausgewählt, und die Lösungen, die die Beschränkungen des Optimierungsproblems verletzen, wurden nachbearbeitet, um sicherzustellen, dass der kodierte Pfad jede Stadt genau einmal besucht. Prototypische Ergebnisse sind in Abbildung 36 zu sehen.

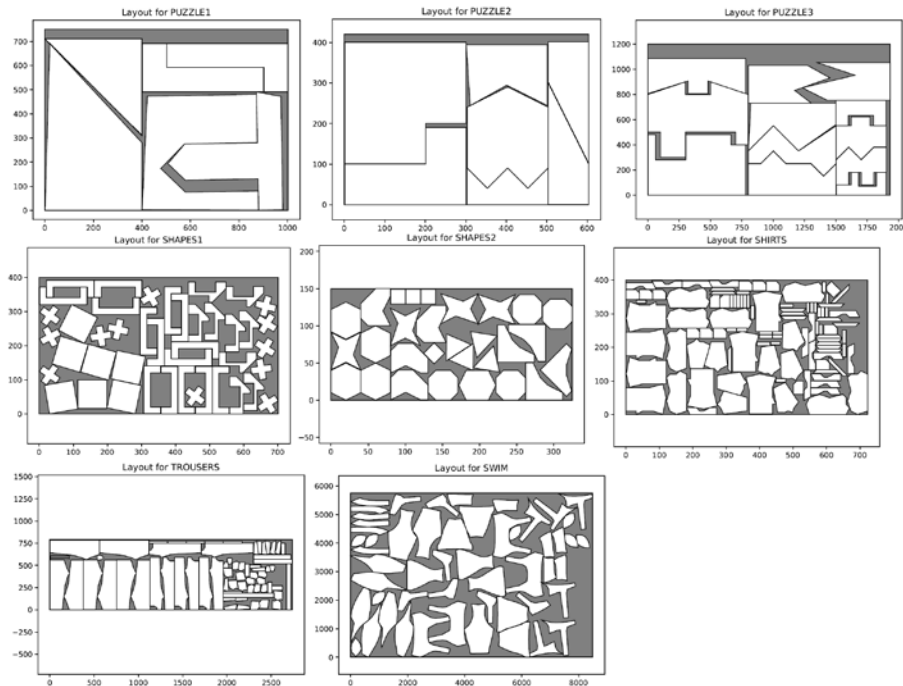


Abbildung 36: Exemplarische Resultate des quanten-hybriden Packvorgangs verschiedener Beispiele – gerechnet auf dem IBM Quantum System One in Ehningen.

Quantencomputing wird erst seit kurzem für die Lösung von Packproblemen in Betracht gezogen. Es wurde zunächst zur Lösung des 1D-Bin-Packing-Problems, dann zur Lösung des 2D-Rechteck-Packproblems (Terada, 2018) und mit unserem Ansatz nun auch zur Lösung des 2D-unregelmäßigen Packproblems eingesetzt. Diese Arbeit unterstreicht die aktuelle Bedeutung der Entwicklung und des Einsatzes von Dekompositionsmethoden bei der Lösung von realen Problemen größeren Umfangs auf NISQ-Hardware mit begrenzter Anzahl von Qubits. Auch wenn der aktuelle Ansatz vielversprechende Ergebnisse liefert, weist er einige Limitierungen wie z.B. eine unvorteilhafte Skalierung auf größere Probleminstanzen und eine schwierige Optimierung des variationalen QAOA auf. Fortschritte könnten z.B. durch neue klassische Optimierer, die besser in der Lage sind mit Barren Plateaus umzugehen erzielt werden. Trotz der aufgetretenen Schwierigkeiten konnten wir feststellen, dass der Entwurf von quantenklassischen Hybridalgorithmen ein zielführender Ansatz ist, der die Entwicklung neuer Konzepte und Methoden zur Lösung rechenintensiver Probleme ermöglicht. Die Ergebnisse dieses Use-Cases werden derzeit in einer Publikation aufbereitet [20].

- Fahri, Edward, Goldstone, Jeffrey and Gutmann, Sam (2014). »**A quantum approximate optimization algorithm**«. In: *arXiv preprint arXiv:1411.4028*.
- Jylänki, Jukka (2010). »**A thousand ways to pack the bin – a practical approach to two-dimensional rectangle bin packing**«. In: *retrieved from <http://clb.de-mon.fi/files/RectangleBinPack.pdf>*.
- Lucas, Andrew (2014). »**Ising formulations of many NP problems**«. In *Frontiers in physics* 2, p5.
- Oliveira, José F., A Miguel Gomes and J. Soeiro Ferreira (2000). »**TOPOS – A new constructive algorithm for nesting problems**«. In: *OR-Spektrum* 22.2, p 263-284.
- Terada, Kotaro et al. (2018). »**An Ising model mapping to solve rectangle packing problem**«. In: *2018 International Symposium on VLSI Design, Automation and Test (VLSI-DAT)*. IEEE, p1-4.

Im fünften Fokus Use Case »**Szenario-basierte Routenplanung zur Absicherung von Fahrfunktionen**« von der **Universität Tübingen, dem FZI und der Firma Bertrandt** wurde die Routenplanung für ein Testfahrzeug betrachtet. Dabei sollen Testszenarien für automatisierte Fahrfunktionen in der realen Umgebung möglichst effizient durchlaufen werden können. Beispielsweise soll die Erkennung und Interpretation von Ampelanlagen getestet werden. Zu diesem Zweck soll, vom Standpunkt des Testfahrzeugs aus, eine Strecke auf Basis von Kartendaten ermittelt werden, auf der in einer vorgegebenen Zeit möglichst viele Ampelanlagen bzw. bestimmte Varianten von Ampelanlagen passiert werden. Durch Berücksichtigung von Witterung, Verkehrsdichte und vorgegebene Fahrtrichtungen steigt die Komplexität dieses Problems erheblich an. Es wurde deshalb zum einen in einer wirklichkeitsnäheren Variante als Orienteering Problem (OP) und auf den kombinatorischen Problemerkern reduziert als Traveling Salesperson Problem (TSP) formuliert. Bei beiden handelt es sich also um kombinatorische Optimierungsprobleme¹.

Ein erster Ansatz zur Lösung des eigentlichen Orienteering Problems auf dem Quantencomputer, der sich leider als wenig effizient erwiesen hat, besteht in der Formulierung als ILP und Überführung in ein QUBO, wobei für die nicht binären Variablen und Randbedingungen zusätzliche Variablen eingeführt werden müssen. Die automatische Abbildung des ILP durch Qiskit auf ein Quadratic Program führt bereits für ein Minimalbeispiel zu großen Schaltkreisen, die auf dem Quantencomputer in Ehningen nicht sinnvoll ausgeführt werden können.

Alternativ können für viele Optimierungsprobleme Quantenschaltkreise »manuell« erstellt werden, die die konkreten Eigenschaften des Problems ausnutzen und somit kleinere/effizientere Schaltkreise zur Folge haben. Die grundlegende Idee ist hierbei Randbedingungen, die für das Optimierungsproblem gelten müssen, statt in die Kostenfunktion in den sogenannten Mixer des QAOA-Algorithmus (Quantum Alternating Operator Ansatz) einzubetten. Es ist bekannt, dass es eine solche optimierte Formulierung für das TSP gibt.² Für das OP bleibt dies eine offene Fragestellung, da die im Gegensatz zu TSP nicht vorab bekannte Tourlänge eine entsprechende Formulierung stark erschwert.

Ein Algorithmus zur approximativen Lösung des TSP wurde mittels QAOA – Algorithmus auf dem Quantenrechner und Simulator umgesetzt und getestet. Dies führte zu Tage, dass selbst kleine Instanzen des Problems zwar lauffähig sind, aber hohe Fehlerraten auf dem Quantenrechner in Ehningen zur Folge haben. Die Implementierung wurde daher hinsichtlich der Schaltnetzgröße optimiert. Mithilfe des in AP 2 entwickelten Frameworks *Quokka*, das eine komfortable Einbindung eines Schaltkreises in einen Workflow erlaubt, konnten Tools zur Fehlermitigation aus AP 3 die TSP-Lösung aufarbeiten.

Die Schaubilder unten illustrieren beispielhaft die Ergebnisse auf dem Quantenrechner in Ehningen für die TSP-Implementierung mit drei Städten. Beide Bilder zeigen Histogramme der Messungen, nachdem der QAOA-Schaltkreis mit optimierten Parametern ausgeführt wurde. Das gewünschte Ergebnis ist zusätzlich rot markiert. Links sind die häufigsten Messergebnisse ohne weitere Nachbearbeitung dargestellt. Daraus ist zu erkennen, dass das gesuchte Ergebnis durch die verrauschten Berechnungen nicht klar als solches zu erkennen ist. Das wahrscheinlichste Messergebnis (höchster Balken) ist dar-

¹ Vansteenwegen, Pieter, Wouter Souffriau, and Dirk Van Oudheusden. »The orienteering problem: A survey.« *European Journal of Operational Research* 209, no. 1 (2011): 1-10.

² Hadfield S, Wang Z, O’Gorman B, Rieffel EG, Venturelli D, Biswas R. »From the Quantum Approximate Optimization Algorithm to a Quantum Alternating Operator Ansatz.« *Algorithms*. 2019; 12(2):34.

über hinaus eine ungültige TSP-Tour, verletzt also mindestens eine der Randbedingungen der Problemstellung. Trotz dem, dass eine QAOA-Variante implementiert wurde, die theoretisch ungültige Touren gar nicht im Zustandsraum zulässt, sind durch Gatter- und Auslesefehler solche Zustände gemessen worden.

Filtert man alle ungültigen Zustände aus den Messergebnissen heraus, ist das optimale Ergebnis als solches erkennbar, was im rechten Bild dargestellt ist. Für drei Städte gibt es nur 6 Möglichkeiten, sie in eine Reihe zu bringen – aus diesen Möglichkeiten wurde die optimale am häufigsten gemessen. Demnach muss für die sinnvolle Verwendung von Quantenrechnern großer Fokus daraufgelegt werden, Fehler zu reduzieren, zu mitigieren oder zu korrigieren.

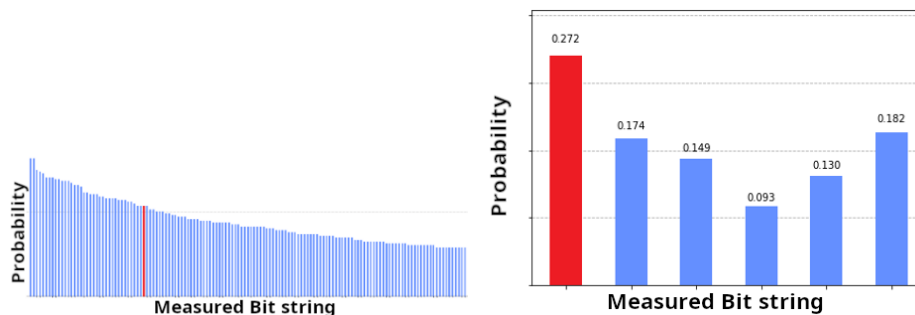


Abbildung 37: Verteilung der Häufigkeit der gemessenen Bitstrings direkt nach der Ausführung auf dem QC in Ehningen (links) und Verteilung der Häufigkeit der Strings, nachdem ungültige Touren herausgefiltert wurden

Die bisher existierenden, in ihrer Größe stark eingeschränkten Quantenrechner sollen möglichst früh Probleme lösen, deren Komplexität praktische Relevanz haben. Deshalb wurde nach Möglichkeiten gesucht, größere Probleminstanzen in kleinere aufzuteilen, um sie so auf kleinen Quantenrechnern berechenbar zu machen. Da durch schrumpfende individuelle Problemgrößen die auszuführenden Schaltkreise kleiner werden, kann so auch das entstandene Rauschen auf die Unterprobleme verteilt werden. Es wurden dabei Ansätze erarbeitet, die sich Elementen von dynamischer Programmierung und dem A*-Algorithmus³ bedienen. Eine entscheidende Rolle spielt dabei ein Machine Learning – Verfahren, mit dem die voraussichtliche Länge einer optimalen TSP – Tour abgeschätzt werden kann. Dies wird dazu benötigt, die Generierung der Unterprobleme zu führen, da andernfalls exponentiell viele Unterprobleme generiert würden, deren Lösung zur Lösung des ursprünglichen Problems nicht beitragen, weshalb jeder Laufzeitvorteil verschwände. In Kombination mit dem optimierten Schaltnetz zur Lösung des TSP auf dem Quantenrechner wird diese Methode in [21] publiziert (bereits in Vorbereitung).

In der Abbildung 38 unten sind zwei verschiedene Lösungen für eine zufällig generierte TSP-Instanz zu sehen. Links ist die optimale Lösung dargestellt. Die Lösung rechts zeigt das Ergebnis des oben beschriebenen Algorithmus. Die generierten Teilprobleme bzw. deren Lösungen sind farblich voneinander abgegrenzt. Es wird daraus auch ersichtlich, dass es sich hierbei um ein Approximationsverfahren handelt, da Heuristiken und Schätzungen maßgeblich für die Güte des Ergebnisses verantwortlich sind. Gleichzeitig wird für die Lösung der Teilprobleme QAOA verwendet, was an sich nur approximative Ergebnisse liefert.

³ P. E. Hart, N. J. Nilsson and B. Raphael, "A Formal Basis for the Heuristic Determination of Minimum Cost Paths," in IEEE Transactions on Systems Science and Cybernetics, vol. 4, no. 2, pp. 100-107, July 1968, doi: 10.1109/TSSC.1968.300136.

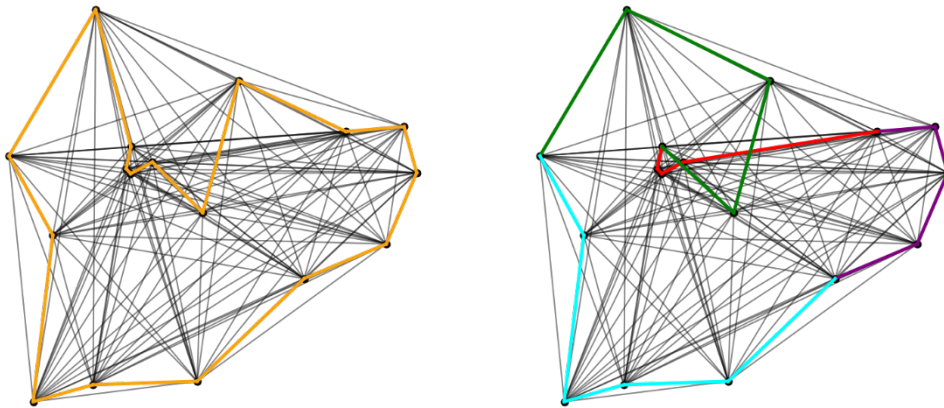


Abbildung 38: Darstellung einer optimalen TSP-Tour (links) und die approximative Lösung, die durch Problemaufteilung und Lösung auf dem Quantenrechner entstanden ist.

Eine genauere Beschreibung der Funktionsweise dieses Algorithmus und dessen Implementierung sind in einem Demonstrator (<https://sequoia.iao.fraunhofer.de/user/bertrandt/lab>) festgehalten worden.

Die Ergebnisse wurden zusammen mit Bertrandt ausgearbeitet, dokumentiert und bereits auf mehreren Veranstaltungen vorgestellt (siehe Veranstaltungsliste AP7).

Im sechsten Fokus Use Case »**Routenplanung von LKW-Flotten im Supply-Chain-Management**« vom **Fraunhofer IAO mit der Firma Schwarz IT** wurde der Anwendungsfall als Vehicle Routing Problem (VRP) identifiziert, welches eine Verallgemeinerung des klassischen Traveling Salesperson Problems (TSP) darstellt.

Es geht dabei darum, eine große Zahl durchzuführender Lieferungen optimal auf eine kleinere Zahl von Lastwagen aufzuteilen. Sobald einem Lastwagen eine Auswahl von Lieferungen zugewiesen ist, entsteht ein kleineres Optimierungsproblem, welches man in erster Näherung als klassisches TSP auffassen kann, genauer handelt es sich um ein Pickup Delivery Problem (PDP). Dieses modifiziert das TSP in der Hinsicht, dass nicht mehr beliebige Abfolgen der Orte möglich sind, sondern gewisse Reihenfolgen eingehalten werden müssen (für jedes Paar von Abhol- und Lieferorten muss der Abholort vor dem Lieferort besucht werden).

Zur Komplexitätsreduktion wurde in einem ersten Schritt eine Einschränkung auf ein ebensolches PDP als Teilproblem durchgeführt. Dieses wurde in ein QUBO Problem umgeformt, welches mit dem Algorithmus QAOA gelöst wurde. Hierbei wurde aufgrund der ähnlichen Problemstellung eng mit den Use Cases »Szenario-basierte Routenplanung zur Absicherung von Fahrfunktionen« und »Optimierung von Ladeplänen einer gegebenen Ladesäuleninfrastruktur« zusammengearbeitet.

In einem ersten Ansatz zur Lösung des gesamten VRP, also die gesamt-optimale Aufteilung der Lieferungen auf einzelne Lastwagen, wurde versucht das VRP als Set Partitioning Problem (SPP) aufzufassen, welches sich durch eine Formulierung als QUBO ebenfalls für eine Lösung auf dem Quantencomputer anbietet. Dieser Ansatz hat sich leider nicht als sinnvoll erwiesen, da zur Formulierung der globalen Kostenfunktion die Lösung von sehr vielen kleinen PDPs nötig wäre und keine gute Heuristik zur Umgehung dieser Problematik gefunden wurde.

In Abbildung 39 ist ein VRP mit 9 durchzuführenden Lieferungen skizziert. Ein Pickup und Delivery Orte-Paar ist hier durch Gleichfarbigkeit und die Buchstaben P und D gekennzeichnet. Eine mögliche Partitionierung in drei Untermengen mit jeweils drei Lieferpaaren ist skizziert. Für eines der dadurch entstehenden PDPs ist eine optimale Lösung angegeben. Man beachte, dass innerhalb der gleichen Farbe immer P vor D durchlaufen wird. Alle Lastwagen starten vom selben Depot aus (schwarzer Punkt).

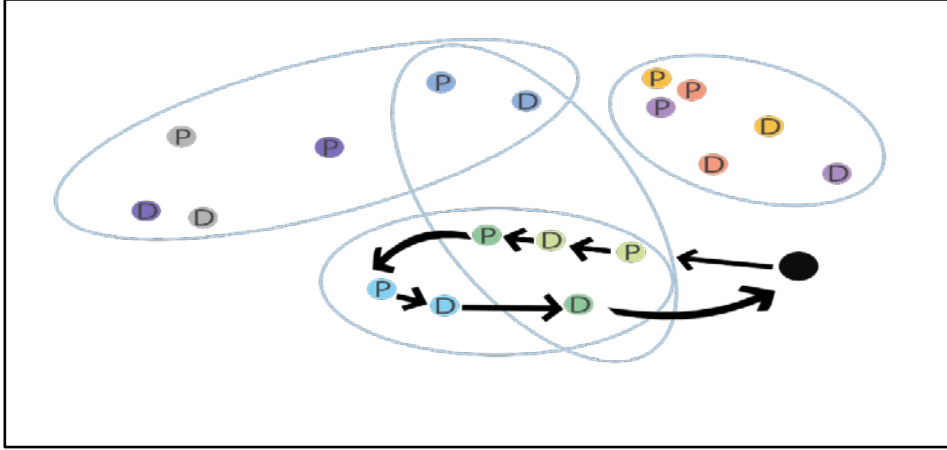


Abbildung 39 Ein Vehicle Routing Problem mit 9 Lieferungen als Set Partitioning Problem und die beispielhafte Lösung eines der drei dadurch entstehenden Pickup and Delivery Probleme.

Anstelle des Set-Partitioning-Ansatzes wurde das VRP daher als Überlagerung einzelner PDPs dargestellt. Ein einzelnes PDP kann (unter der zusätzlichen, laut Industriepartner gegebenenfalls gerechtfertigten Vereinfachung, dass Abholung und Lieferung direkt aufeinander folgen müssen) als Traveling-Salesperson-Problem (TSP) aufgefasst werden. Zur Enkodierung eines TSP mit n Orten sind n^2 Qubits nötig, wenn man die Besuchsreihenfolge als Permutationsmatrix repräsentiert. Die entsprechende Kostenfunktion f , die eine binäre Matrix $x_{i,j}$ aufnimmt, und diese gemäß der zu fahrenden Strecken bewertet, stellt sich wie folgt dar:

$$f(x) = \sum_{i=1}^n x_{i,1} D_{n+1,i} + \sum_{t=1}^n \sum_{i,j=1}^n x_{i,t} x_{j,t+1} D_{i,j} + \sum_{i=1}^n x_{i,n} D_{i,n+1} + A \sum_{j=1}^n \left(1 - \sum_{i=1}^n x_{i,j} \right)^2 + A \sum_{i=1}^n \left(1 - \sum_{j=1}^n x_{i,j} \right)^2$$

$x_{i,j} = 1$ bedeutet hier, dass Tour i an Stelle j innerhalb der Abfertigungsreihenfolge steht. $D_{i,j}$ stellt die (bekannten) Kosten für die Leerfahrt dar, um vom Ende der Tour i zum Anfang der Tour j zu gelangen. $D_{n+1,i}$ entspricht der Entfernung vom Depot zum Beginn von Tour i . Der Parameter A ist ein Strafparameter, der sicherstellen soll, dass Binärmatrizen, die keine Permutationsmatrizen sind, nicht als Lösung in Frage kommen. Denn diese würden Konfigurationen entsprechen, die entweder Touren auslassen oder mehrere Touren gleichzeitig bedienen. Man beachte, dass die Kostenfunktion quadratisch in den $x_{i,j}$ ist, damit haben wir das Problem als Quadratic Unconstrained Binary Optimization (QUBO) Problem formuliert, was uns die Nutzung von QAOA zur Lösung ermöglicht.

In Abbildung 40 ist beispielhaft gezeigt wie eine Abfertigungsreihenfolge als Permutationsmatrix kodiert werden kann. Man beachte, dass hier aufgrund der Vereinfachung,

dass Pickup und Delivery direkt aufeinander folgen müssen, eine Tour, welche vorher aus zwei Orten bestand, nun auf einen Punkt abstrahiert wurde.

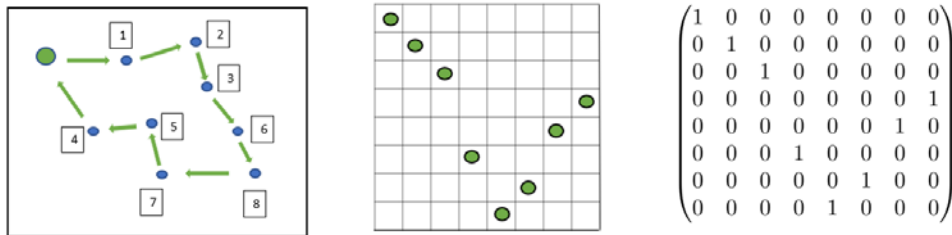


Abbildung 40 Eine mögliche Lösung eines Pickup and Delivery Problems eines LKW mit 8 Lieferungen und seine Enkodierung als Permutationsmatrix $x_{t,j}$.

Um ein VRP mit k Lastwagen und n Touren als QUBO formulieren zu können, kann man eine mögliche Aufteilung in Subtours über eine Partition einer Permutationsmatrix in k binäre Matrizen darstellen. Dafür sind dann $k \cdot n^2$ Qubits notwendig. Die entsprechende Kostenfunktion lässt sich ähnlich wie oben im Falle von einem LKW aufstellen und sie ist immer noch quadratisch in den Matrixelementen. Abbildung 41 zeigt eine mögliche Aufteilung für zwei LKWs und die entsprechende Partition der Permutationsmatrix.

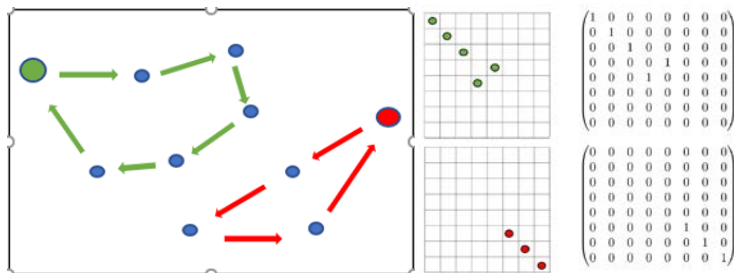


Abbildung 41 Eine mögliche Lösung eines Vehicle Routing Problems mit zwei LKWs und 8 Lieferungen und seine Enkodierung als zerlegte Permutationsmatrix.

In diesem Beispiel sind also bereits 128 Qubits notwendig. Ein auf dem Quantum System One theoretisch noch darstellbarer Fall sind etwa zwei Lastwagen und drei Touren mit 18 notwendigen Qubits. Im Folgenden wird eine Wahrscheinlichkeitsverteilung für einen LKW mit 3 Touren dargestellt, was 9 Qubits erforderte. Die entsprechenden Schaltkreise wurden mit Hilfe der Qiskit-Klasse QAOA-Ansatz erzeugt.

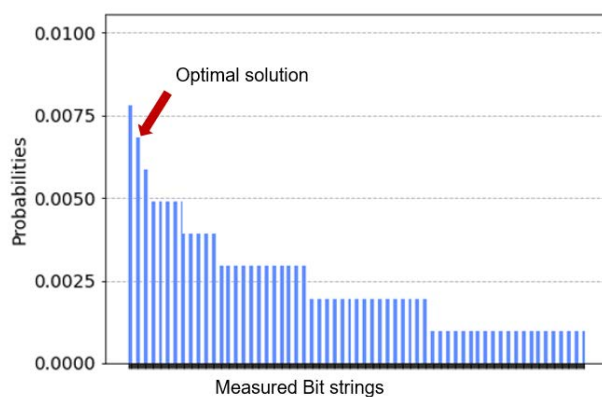


Abbildung 42 Wahrscheinlichkeitsverteilung der QAOA-Lösung eines Vehicle Routing Problems mit einem LKW und 3 Lieferungen gerechnet mit 1000 shots auf dem Quantum System One in Ehningen. Die Optimale Lösung ist gekennzeichnet.

Das Ergebnis steht exemplarisch für die aktuellen Möglichkeiten auf dem Quantum System One in Ehningen. Die Wahrscheinlichkeitsverteilung der 1000 Shots ist sehr breit verteilt, das optimale Ergebnis rangiert zumindest an vorderer Stelle. Der Bitstring mit maximaler Wahrscheinlichkeit repräsentiert allerdings keine zulässige Lösung. Im Gegensatz zu Simulationen kann man aufgrund von Dekohärenzeffekten in echter Hardware auch mit passend gewähltem Strafparameter unzulässige Lösungen nicht immer vermeiden.

Abbildung 43 zeigt ein Simulationsergebnis mittels des QASM-Simulators. Man sieht, dass die Wahrscheinlichkeitsverteilung innerhalb einer Simulation viel gedrängter ist, da hier im Normalfall keine unzulässigen Lösungen auftreten. Auch wenn der favorisierte Bitspring mit 33% Wahrscheinlichkeit nur einer leicht suboptimalen Lösung entspricht, ragen die optimalen Lösungen mit 5% - 6% noch substanziell heraus. Zudem treten hier keine unzulässigen Lösungen auf.

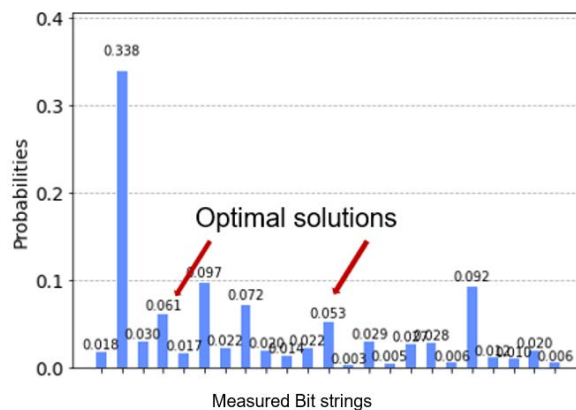


Abbildung 43 Wahrscheinlichkeitsverteilung der QAOA-Lösung eines Vehicle Routing Problems mit einem LKW und 4 Lieferungen gerechnet mit 1000 shots mit dem QASM-Simulator. Die beiden optimalen Lösungen sind gekennzeichnet.

Die Problemgrößen mit zwei LKWs, die 18 Qubits erfordern, lassen sich auf dem System One in Ehningen leider nicht mehr sinnvoll berechnen. Die relativ hohe Qubitanzahl in Zusammenhang mit der starken Kopplung (jedes der 18 Qubits muss mit bis zu 12 anderen verschränkt werden) führt zu einer prohibitiven Anzahl an CNOT Gattern, die im entsprechenden Schaltkreis bei den derzeitigen Fehlerraten zu so viel Rauschen führen, dass es auch von den aktuellsten Fehlermitigationstechniken nicht mehr beherrscht werden kann.

Die Ergebnisse dieses Use Cases wurden bereits im Rahmen des Quantum Village Ehningen gemeinsam mit dem Industriepartner Schwartz IT der Öffentlichkeit vorgestellt. Eine abschließende Veröffentlichung ist in Vorbereitung. Die verwendeten Algorithmen sind auch in einem Demonstrator unter https://sequoia.iao.fraunhofer.de/user/schwarzit/lab/tree/vehicle_routing/Logistics_Use_Case_multiple_Trucks.ipynb dargestellt. Hier wurden Beispielinstanzen von Vehicle Routing Problemen unterschiedlicher Komplexität in Python modelliert auf welche QAOA in unterschiedlicher Parametrisierung angewendet werden kann.

Gemischt-ganzzahlige lineare Programme sind eine bedeutende Klasse von Optimierungsproblemen. Im Rahmen der betrachteten Use Cases treten diese z.B. bei der Tourenplanung auf, aber auch andere Probleme können als solche dargestellt werden, beispielsweise das in dem Use Case **Optimierung von Ladeplänen einer gegebenen Ladesäuleninfrastruktur** betrachtete Problem der Optimierung der Ladung von Elektroautos unter den gegebenen Einschränkungen an verfügbaren Ladeplätzen, Netzbelastung, Ladegeschwindigkeit der Autos etc.

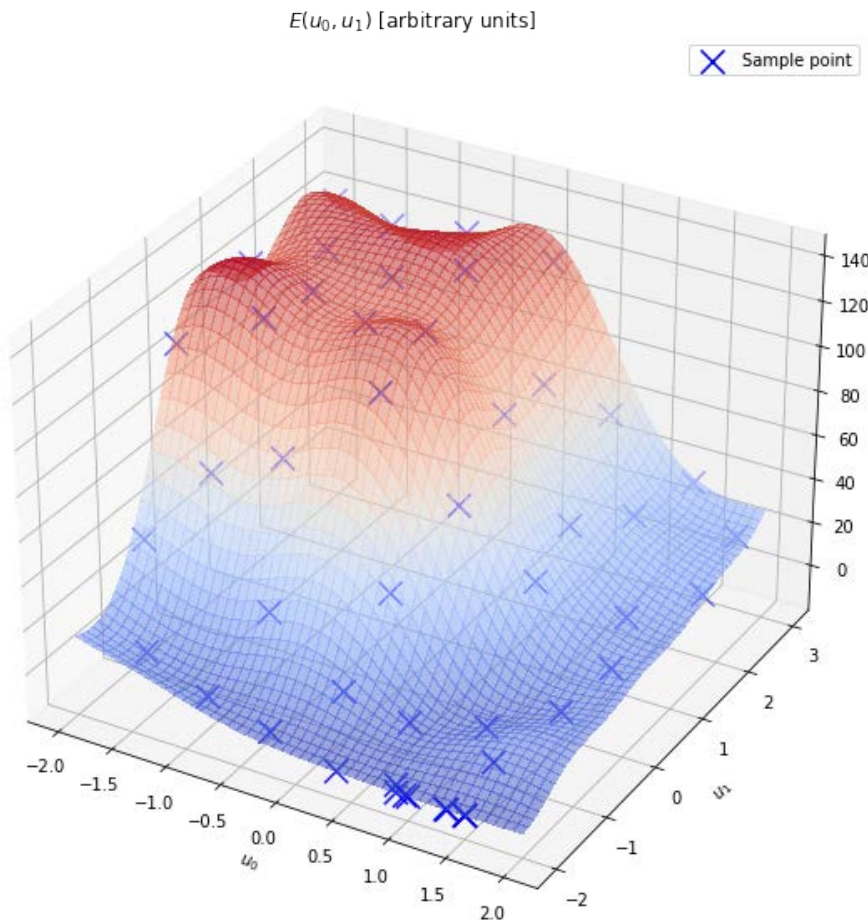
Während in den Minimalmodellen der Use Cases diese Problematik in ihrer Komplexität (noch) nicht mit einbezogen wurde, hat sie das **HLRS der Universität Stuttgart** unabhängig davon eigenständig untersucht.

Während für gewöhnliche lineare Optimierungsprobleme, also mit rein reellen Variablen, klassische, effiziente Lösungsalgorithmen existieren, ist dies für ganzzahlige Probleme, und somit auch für gemischt-ganzzahlige Probleme, im Allgemeinen nicht gegeben. Wie im Use Case **Optimierung von Ladeplänen einer gegebenen Ladesäuleninfrastruktur** erläutert, kann das rein ganzzahlige Problem jedoch als QUBO-Problem formuliert werden und mit einem Quantencomputer mittels beispielsweise QAOA gelöst werden. In der Literatur finden sich derzeit zwei verschiedene Wege, wie nun ein gemischt-ganzzahliges Problem mit Hilfe eines klassischen Computers und eines Quantencomputers gelöst werden kann. Unter geeigneten Voraussetzungen kann das Problem mittels Alternating Direction Method of Multipliers (ADMM) gelöst werden, in dem das (für feste reelle Werte) ganzzahlige Problem als eine »Richtung« aufgefasst wird und bei fixierten reellen Werten auf dem Quantencomputer gelöst wird, z.B. mittels QAOA oder VQE. Anschließend wird analog der reellwertige Teil als eine andere Richtung aufgefasst und mit fixierten binären Variablen klassisch gelöst. Dieser Algorithmus ist in Qiskit für eine bestimmte Problemklasse mit Konvexitätsbedingungen implementiert.

Als Alternative wurde in der Literatur auch vorgeschlagen, das ganze Problem als großes Optimierungsproblem zu sehen, indem das Problem als QUBO (mit festen reellen Koeffizienten) formuliert wird und die Optimierungsparameter von QAOA (oder VQE) zusammen mit den kontinuierlichen Koeffizienten der Zielfunktion den gesamten Parameterraum aufspannen. In diesem Szenario wird der Quantencomputer nur noch zum Auswerten des QUBOs bei festen Parametern verwendet.

Als weitere Methode schlagen wir die Bildung eines Kriging-Surrogate-Modells vor, bei dem an festgelegten »Stützstellen« ebenfalls bei festen reellen Werten die entstehenden QUBO-Probleme auf dem Quantencomputer optimiert werden und anschließend mit diesen Werten ein reellwertiges Modell gebildet wird, das klassisch optimiert werden kann. Für jedes neue Optimum des klassischen Optimierers im Surrogate-Modell fügen wir dann den »echten« optimalen Wert ins Modell ein, indem wir wieder die binären Variablen auf dem Quantencomputer für fixierte reellwertige Variablen bestimmen und verbessern so schrittweise das Modell. Als Beispiel nachfolgend eine grafische Darstellung, welche mittels Aer-Simulator (statevector) erstellt wurde. Die blauen Kreuze geben die Stützstellen an. An der Stelle mit den gehäuften blauen Kreuzen wurden wie zuvor beschrieben Verfeinerungen des Surrogate-Modells vorgenommen

(neue Optima als zusätzliche Stützstellen hinzugefügt).



Überblick der wissenschaftlichen
Arbeiten

Abbildung 44 Oberflächenplot des Surrogate-Modell mit Stützstellen. Durch Verfeinerung (die markierten Punkte, welche dicht beieinanderliegen) wird das Modell in der Nähe des Minimums präzisiert.

In der zweiten Phase des Projekts haben wir eine Version des 2-ADMM-Modells, welche die Äquivalenzbedingungen sicherstellt und das beschriebene Kriging-Surrogate-Modell implementiert (<https://websites.fraunhofer.de/sequoia/use-cases/>). Unsere eigene Implementierung des 2-ADMM-Algorithmus erlaubt uns eine bessere Kontrolle über die Parameter als die in Qiskit implementierte Version des 3-ADMM-Algorithmus. Die Implementierung unserer beiden Algorithmen basiert auf dem VQE-Algorithmus und der estimator-qiskit-runtime-primitive, welche gemäß Dokumentation auch die Interaktion zwischen klassischem Computer (zur klassischen Optimierung) und Quantencomputer verbessert.

Getestet wurden die Implementierungen mit einem kleinen (2 binäre und 2 kontinuierliche Variablen) und einem größeren Problem (4 binäre und 2 kontinuierliche Variablen).

Ergebnis ist, dass beide Implementierungen für geeignete Probleme funktionieren. Folgende, teils erwartete, Beobachtungen wurden gemacht:

- Die Zwangsbedingungen, die als Strafterm (»soft constraint«) eingefügt wurden, wurden insbesondere bei zu geringem Strafgewicht nicht zwingend erfüllt.

- Aktuell gibt es scheinbar viel Overhead beim Ausführen des VQE-Algorithmus per Qiskit-runtime-primitive. Das Problem mit 2 binären Variablen, 512 shots und zwei klassischen Variablen hatte eine Laufzeit von ca. 9 s pro VQE-Schritt (resilience-level 1, optimization-level 1). Eine Änderung auf 4 binäre Variablen ergab eine Laufzeit von 10 s (Backend ibmq_quito). Mit resilience-level 0, optimization-level 0 reduziert sich die Zeit auf 4 s bzw. 5 s. Die TREX-Fehlermitigation ist anscheinend so implementiert, dass bei jedem Schritt entsprechende Messungen zur Fehlermitigation unternommen werden, was zum großen Overhead beiträgt. Dies macht sich offensichtlich insbesondere bei den sehr kleinen Beispielen stark bemerkbar.
- Nicht konvexe Probleme sind, wie bekannt, schwer zu optimieren. Eine Möglichkeit zum Umgang mit lokalen Optima etc. besteht bei ADMM im Wählen von verschiedenen Startwerten, im Surrogate-Modell durch Nutzung genügend vieler Stützstellen und anschließender Nutzung von klassischen Techniken für nicht notwendigerweise konvexe Optimierungsprobleme.
- Die Laufzeit des gesamten Algorithmus ist teils sehr lang. Ein Lauf des 2-ADMM-Algorithmus auf dem Backend ibmq_quito betrug bei nur zwei Schritten bis zum Ende des Algorithmus mehr als 2068 s, wobei diese Zeit im Wesentlichen für die VQE-Schritte benötigt wurde (ca. 2068 s für die VQE-Schritte und ca. 0.004 s für die Optimierung des rein reellwertigen Problems).
- Auf dem IBM-System in Ehningen haben wir grundsätzlich dasselbe Verhalten festgestellt, nur dauerten dort die VQE-Berechnungen etwa doppelt so lang, z.B. etwa 19 s für einen VQE-Schritt (resilience-level 1, optimization-level 1, 2 binäre Variablen, 512 shots für den VQE circuit).

Derzeit scheint es im Allgemeinen nicht angeraten zu sein, das Surrogate-Modell zu nutzen, da die Auswertung genügend vieler Stützstellen auf dem Quantencomputer sehr viel Zeit beansprucht, die bei ADMM schon zur Optimierung genutzt werden kann. Jedoch könnte dies bei größeren Problemen interessanter werden. Die von uns getesteten kleinen Probleme leiden auch sehr deutlich unter dem aktuellen Overhead der in den runtime-primitives implementierten Fehlermitigation. Bei unseren kleinen Testproblemen haben wir wenig Probleme mit dem Rauschen der Werte des Quantencomputers gesehen. Dies liegt vermutlich einerseits daran, dass die Ansätze vergleichsweise wenig CNOT-Gates haben, andererseits macht ein verrauschter (falscher) Punkt nicht so viel aus, da wegen des iterativen Vorgehens (Wechsel zwischen Optimierung des binären Teils auf dem Quantencomputer und reellwertigen Teils auf dem klassischen Computer) in den Algorithmen dies im nächsten Schritt korrigiert werden kann.

QDNN PDE Solver:

In den Natur- und Ingenieurwissenschaften werden viele äußerst detaillierte mathematische Modelle entwickelt, um physikalische Phänomene zu beschreiben. Viele dieser Modelle werden natürlicherweise in Form von Differentialgleichungen dargestellt, meistens als zeitabhängige partielle Differentialgleichungen (PDEs). Das Lösen solcher Differentialgleichungen ist von enormer Bedeutung für viele Probleme, z.B. der Wettervorhersage, von astronomischen Simulationen, der molekularen Modellierung oder der Triebwerkskonstruktion. Die meisten wichtigen Gleichungen lassen sich allerdings nur schwer oder gar nicht analytisch lösen und es werden daher normalerweise numerische Lösungsverfahren angewandt, so wie im zweiten Fokus Use Case »**Quantenbasierte numerische Strömungssimulation**«. Während man sich dort den Quantenrechner zunutze macht, um diese Numerik besser zu bewältigen, soll in dieser algorithmischen Betrachtung alternativ das Zusammenspiel von Quantencomputing mit Machine Learning zur Anwendung kommen.

Die numerische Lösung partieller Differentialgleichungen (PDEs) ist schwierig und stellt seit langem eine rechnerische Herausforderung dar. Um genaue Lösungen mit geringem Fehler bei minimalem Rechenaufwand zu erhalten, sind handgefertigte Löser erforderlich, die immer auf die vorliegende Gleichung zugeschnitten sind. In jüngster Zeit wurde viel Forschung zu Lösern auf Basis von tiefen neuronalen Netzwerken betrieben, bei denen tiefe neuronale Netze mittels eines stochastischen Gradientenabstiegsverfahren an zufällig ausgewählten Punkten trainiert werden, um die Differentialgleichung sowie die Anfangs- und die Randbedingungen zu erfüllen. Damit wird das Problem eine PDE zu lösen in ein Problem des Maschinenlernens umgewandelt, das in der Regel nicht von der Numerik getrieben, sondern datengetrieben ist. Das Training tiefer neuronaler Netze ist daher aufgrund der großen Datenmengen rechenintensiv, hat einen hohen Speicherbedarf und erfordert viel Interprozesskommunikation. Das Aufkommen von Quantencomputern eröffnet jedoch neue, aufregende Möglichkeiten, die die Gesetze der Quantenmechanik direkt ausnutzen, um die technologischen und thermodynamischen Grenzen der Numerik auf klassischen Computern zu umgehen.

Das HLRS der Universität Stuttgart hat daher eine neuartige Methode zum Lösen partieller Differentialgleichungen vorgeschlagen und getestet, die auf einem quantenneuronalen Netz (QDNN) basiert und eine schnellere Optimierung bei reduziertem Speicherbedarf gegenüber einem klassischen DNN ermöglicht. Das entworfene QDNN ist von einem Quantum Convolutional Network inspiriert, bei dem als Faltungsschicht ein einziges quasi-lokales unitäres Gatter auf translationsinvariante Weise angewendet wird. Für das Pooling wird ein Teil des Netzes gemessen und die Messergebnisse legen die Parameter unitärer Rotationen fest, die auf die jeweils benachbarten Qubits angewendet werden. Nichtlinearitäten im QCNN entstehen durch die Verringerung der Anzahl von Freiheitsgraden bei den Messungen der Pooling-Schicht. Faltungs- und Pooling-Schichten werden so lange angewandt, bis die Anzahl der übrigen Qubits die gewünschte Größe erreicht. Abschließend wird eine vollständig verbundene Schicht auf alle verbleibenden Qubits angewendet, indem eine einzige unitäre Operation auf allen Qubits ausgeführt wird. Das Endergebnis der Schaltung wird durch Messen einer festgelegten Anzahl von Ausgangs-Qubits erhalten. Dieses QDNN wurde mittels auf einem HPC-System generierten Daten trainiert.

Wir haben mit unserem Ansatz Experimente zur Lösung der 1D-Burger-Gleichung und 2D-Poisson-Gleichung durchgeführt. Nachfolgende Grafiken legen dar, dass die Anwendung des QDNN für beide Probleme gute Näherungslösungen ergibt.

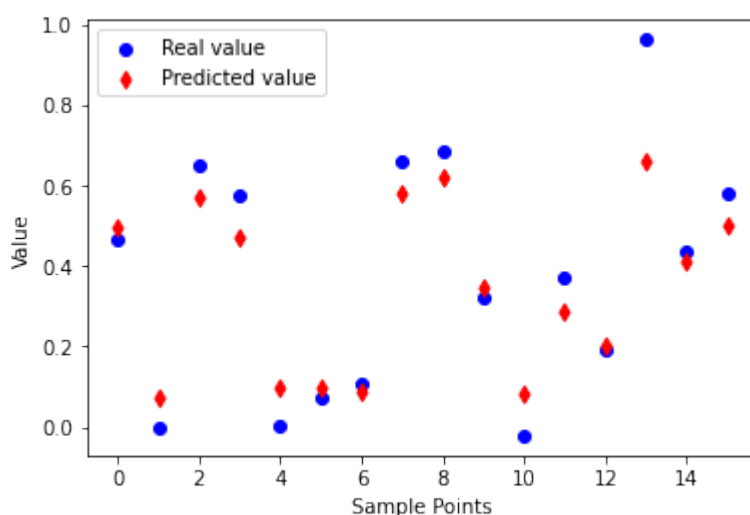


Abbildung 45 Vergleich zwischen der klassischen numerischen Lösung und der durch das QDNN vorhergesagten Lösung für eine viskose 1D-Burger-Gleichung. Die Punkte sind zufällig gewählte Datenpunkte zu jeweils einem festen Zeitpunkt und nicht notwendigerweise räumlich beieinanderliegend. Die Grafik illustriert,

dass die Vorhersage des QDNN an zufällig gewählten Punkten ähnliche Werte liefert wie eine numerische Simulation.

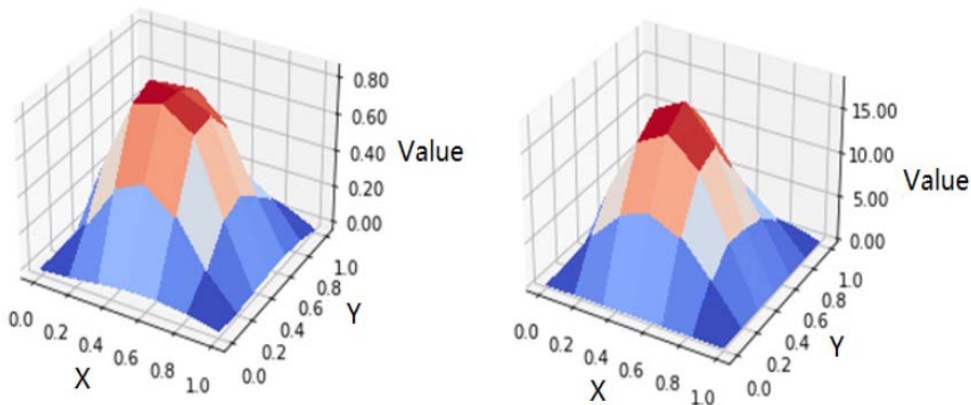


Abbildung 46 Vergleich zwischen der klassischen numerischen Lösung (links) und der vom QDNN vorhergesagten Lösung (rechts) der 2D-Poisson-Gleichung als Heat-Map dargestellt

Abschließend soll an dieser Stelle noch einmal hervorgehoben werden, dass durch die bereitgestellten Demonstratoren (<https://websites.fraunhofer.de/sequoia/use-cases/>), die SEQUOIA Studie (<https://websites.fraunhofer.de/sequoia/>), die Use Case Publikationen (Kapitel 3.1), den SEQUOIA Werkzeugkasten (<https://github.com/UST-QuAntil>) und den (internen) SEQUOIA Algorithmenkatalog alle Meilensteine wie im Gantt-Diagramm Abbildung 2 aufgeführt, ebenfalls in den APs 4 und 5 gehalten worden sind.

2.5 Arbeitspaket 6: Quantensoftware-Engineering

Die Literaturrecherche und die konzeptionellen Arbeiten haben gezeigt, dass bei der Modellierung von Quantenanwendungen auf Architekturebene vor allem die Aspekte Deployment und Ausführung hybrider Anwendungen sowie der Umgang mit durch Quantencomputing induzierten Unsicherheiten im Fokus stehen. Das Thema Deployment und Ausführung wird in AP 2.3 adressiert, so dass **AP 6.1 »Modellierung und Spezifikation«** maßgeblich das Thema des Umgangs mit Unsicherheiten im Hinblick auf die Korrektheit der Ergebnisse adressiert. Es werden somit auch Aspekte aufgegriffen, die im Antrag in AP 3.5 dargestellt sind.

Zuerst wurde erarbeitet, wie eine Modellierung hybrider Quantenanwendungen auf Architekturebene erfolgen kann, so dass Aussagen über Laufzeiteigenschaften möglich werden. Hierzu wurde eine bestehende Architektur-Beschreibungssprache erweitert: Das Palladio Component Model (PCM). Die PCM-Modellierungssprache erlaubt die Beschreibung von klassischen Softwarearchitekturen. PCM ist Teil eines Frameworks: Dem Palladio-Ansatz. Das Palladio-Ökosystem bietet neben PCM noch weitere Simulations- und Analysewerkzeuge, mit denen PCM-Modelle bzgl. diversen Qualitätsattributen untersucht werden können. Im Rahmen von AP 6.1 wurde die PCM-Modellierungssprache dahingehend erweitert, dass auch Quantenalgorithmen sowie ihre Ausführung auf Quantencomputern modelliert und in ein bestehendes PCM-Modell integriert werden können. Diese Erweiterung erlaubt folglich die Modellierung von hybriden Quantensoftwaresystemen, d.h. die Modellierung von klassischen Softwarekomponenten, Quantenkomponenten sowie ihre Abhängigkeiten bzw. Kommunikation. Die Spracherweiterung bildet weiterhin die Grundlage für Analyserweiterungen. Somit wird zusätzlich ein bestehendes Analysewerkzeug zur Vorhersage von Zuverlässigkeitsattributen im

Palladio-Kontext erweitert, so dass nicht nur Zuverlässigkeitsvorhersagen von klassischen Softwarearchitekturmodellen getätigt werden können, sondern auch für Architekturen, die hybride Quantensoftwaresysteme beschreiben. Hierbei wurde für die Berechnung der Zuverlässigkeit einer Quantenkomponente die Estimated Success Probability als Metrik verwendet. Zuletzt wurde der Ansatz in ein weiteres Palladio-Werkzeug integriert: PerOpteryx. PerOpteryx bildet ein Werkzeug, welches Softwareentwickler auf Basis von PCM-Modellen in der Entscheidungsfindung unterstützt. Da jedes Softwaresystem mit einer großen Menge an Entwurfsentscheidungen assoziiert ist, erlaubt PerOpteryx die automatisierte Exploration des Entwurfsraums und liefert eine Menge von möglichen Architekturkandidaten zurück. Aus dieser Menge kann der Kandidat ausgewählt werden, welcher die Nicht-funktionalen Anforderung des Systems am besten erfüllt. PerOpteryx wurde dahingehend erweitert, dass auch Quanten-spezifische Entwurfsentscheidungen berücksichtigt und analysiert werden können. Dieser Ansatz wurde in [5] publiziert.

Im Anschluss an das Thema Modellierung von hybriden Quantensoftwaresystemen wurde das Thema Architekturmuster adressiert. Grundlegende Muster für den Entwurf hybrider Quantenalgorithmen und -anwendungen wurden in Zusammenarbeit mit PlanQK in der Publikation »Patterns For Hybrid Quantum Algorithmus« [1] beschrieben. Da hierbei das Thema Architekturmuster nur am Rande aufgegriffen wird, wurden Architekturmuster mit Fokus auf den Umgang mit Zuverlässigkeit im Projekt separat adressiert und publiziert.

Im Zusammenhang mit fehlertoleranten hybriden Quantensoftwaresystemen wurden bestehende Architekturmuster aus dem (allgemeinen) Themenkomplex der fehlertoleranten Systeme betrachtet. Hierbei wurden drei verschiedene Architekturmuster ausgewählt und die Anwendbarkeit der Muster im Quantencomputing bewertet. Konkret wurden das Voter-, Comparison- und Sparing-Muster diskutiert. Während die Anwendbarkeit des Comparison und Sparing-Muster nur diskutiert wurden, so wurde das Voter-Muster als konkrete Variante implementiert. Für die Implementierung mussten zwei wesentliche Fragestellungen erörtert werden: (i) Wie wird die Voting-Komponente umgesetzt? (ii) Wie manifestiert sich (heterogene und homogene) Redundanz bei Quantenkomponenten.

Die erste Fragestellung bezieht sich auf die Problematik, dass das Resultat einer Quantenkomponente einem Zufallsprozess entspricht, d.h. einer Wahrscheinlichkeitsverteilung über mögliche Lösungen bzw. Messungen. Während beim klassischen Voter mehrere Werte mittels (z.B.) einem Mehrheitsvotum oder einer einfachen Durchschnittswertberechnung aggregiert werden, müssen bei Quantenkomponenten mehrere Wahrscheinlichkeitsverteilungen kombiniert werden. Für die Zusammenführung von Messungen einzelner Quantenkomponenten wurde die Voter-Komponente als Linear Opinion Pool implementiert. Ein Linear Opinion Pool definiert die mathematischen Konzepte, um aus mehreren Wahrscheinlichkeitsverteilungen eine höherwertige bzw. verfeinerte Verteilung zu synthetisieren.

Für die zweite Fragestellung wurden mehrere Möglichkeiten skizziert, Quanten-Komponenten redundant auszulegen. Beispielweise durch das Benutzen von unterschiedlichen Transpilations-Seeds (zur Generierung von unterschiedlichen Varianten desselben Quantenalgorithmus) oder die Verwendung von unterschiedlichen Backends bzw. Quantencomputern.

Zuletzt wurde das angepasste Quantencomputing-Voter-Muster auf Basis von mehreren (zufällig) generierten Quantenalgorithmen evaluiert. Hierbei wurden im Wesentlichen, die einzelnen Lösungen mit den Lösungen des Voter-Musters verglichen. Die Evaluierung hat gezeigt, dass die Instanzen des Voter-Musters für einen großen Teil der generierten Quantenschaltkreise bessere/genauere Lösungen erzielt haben. Die erarbeiteten Konzepte sowie die Evaluierung wurden publiziert [8].

Der Meilenstein M9 wurde maßgeblich mit der genannten Publikation [5] erreicht. Der optional vorgesehene Proof-of-Concept bzgl. Anpassung von Modellierungswerkzeugen wurde nicht für M9 adressiert, da eine Verknüpfung mit den fehlertoleranten Architekturmustern als sinnvoll erachtet wurde. Der Meilenstein M18 wurde durch die Publikationen [1] und [8] zu (Architektur-)Mustern erreicht. Ergänzend zu [8] wurde der zugehörige Code als Open Source bereitgestellt <https://github.com/fzi-forschungszentrum-informatik/sequoia-gsw22>.

In der Anwenderstudie beschreibt Kapitel 3 das Thema Quantum Software Engineering. Hierzu wurde der aktuelle Stand der Forschung zu diesem Thema recherchiert und es wurden die Einflüsse von Quantencomputing auf die unterschiedlichen Teildisziplinen des Software Engineerings und insbesondere die üblicherweise unterschiedenen Phasen im Softwareentwicklungsprozess (siehe Abbildung 1) herausgearbeitet. Im Hinblick auf **AP 6.2 »Entwicklungsprozess und Vorgehensmodelle«** sind insbesondere der Entwurf und das Testen von Quantenanwendungen hervorzuheben.



Abbildung 47 Typische Phasen des Softwareentwicklungsprozesses ohne die Berücksichtigung von Praktiken, wie iteratives Vorgehen.

Da gegenwärtig noch eine eher geringe Abstraktion von Charakteristiken konkreter QC-Hardware gegeben ist, sind Anpassungen von Quantenalgorithmen an konkrete Anwendungsfälle und Hardware in einem Ausmaß notwendig, das weit über die Notwendigkeiten bei konventionellen Rechnern hinausgeht. Durch den großen Bedarf, sich in der Tiefe mit Quantenalgorithmen auseinanderzusetzen, gestaltet sich auch die Qualifizierung von Quantensoftwareentwicklern schwierig, da sowohl fundierte Kenntnisse der Informatik als auch der Physik benötigt werden.

Durch die immanent probabilistische Funktionsweise des Quantencomputing fällt auch das Testen und die Validierung der entwickelten Software schwerer. Eine Möglichkeit ist hier, mit Simulatoren statt mit realer Hardware zu arbeiten, damit beschränkt man sich allerdings auf geringe Problemkomplexitäten, von denen aus nicht unbedingt auf eine generelle Validität geschlossen werden kann. Bei Tests auf realer Hardware können keine perfekt reproduzierbaren Ergebnisse erwartet werden und das Festsetzen einer tolerierbaren Abweichung muss damit Bestandteil eines Testkonzepts sein.

Die Phasen Entwicklung und Verteilung werden durch AP 2 adressiert und insbesondere beim Thema Verteilung an den Arbeiten von PlanQK ausgerichtet.

Die für den Meilenstein M9 vorgesehene Publikation von »Best Practices Vorgehensmodell« wurde durch die Anwenderstudie erreicht und umfasst sowohl die Ausführungen zum Thema Quantum Software Engineering als auch zum SEQUOIA Anwendungsbaukasten, die vor allem die Arbeiten aus AP 2 umfassen.

Das AP **6.3 »SEQUOIA-Modell für Quantensoftware-Engineering hybrider Anwendungen«** umfasst die systematische Erfassung und Aufbereitung der Erfahrungen aus den Forschungs- und Entwicklungsarbeiten von SEQUOIA sowie dem Entwurf und der Realisierung der Demonstratoren. Auf dieser Basis wurde ein SEQUOIA-Modell erarbeitet und zur Veröffentlichung vorbereitet.

Die Inhalte des SEQUOIA-Modell stammen maßgeblich aus den Arbeiten des AP 6, Gesprächen mit den Verantwortlichen für die Umsetzung der Demonstratoren aus AP 5 und Diskussionen bei der KQCBW Developer Conference im Oktober 2022. Konkret wurde ein Phasenmodell erarbeitet. Es wurden fünf Entwicklungsphasen extrahiert, die ein Quanten-Software-Ingenieur bei der Implementierung eines Quantenalgorithmus durchläuft:

- **Problemdefinition:** In der Phase der Problemdefinition geht es hauptsächlich darum, das zugrundeliegende Domänenproblem in ein mathematisches Problem zu überführen, das durch einen Quantenalgorithmus gelöst werden kann.
- **Selektion des Quantenalgorithmus:** Die Phase beschreibt die Auswahl eines konkreten Quantenalgorithmus auf Basis des zugrundeliegenden mathematischen Problems (aus der letzten Phase).
- **Implementierung:** Skizziert die Implementierungsphase des Quantenalgorithmus, d.h. welche Teile des Quantenalgorithmus automatisch generiert werden und welche manuell implementiert werden müssen.
- **Fine-tuning:** Da die heutigen Quantencomputer durch kurze Dekohärenzzeiten, Gatter- und Messfehler gekennzeichnet sind, sind die erzeugten Messwerte fehlerbehaftet und ungenau. Die Fine-tuning-Phase umfasst daher alle Verfeinerungs- und Optimierungsschritte einer Quantenalgorithmus-Implementierung, die zur Erzeugung von akkuraten Messungen führen (z.B. Error-Mitigation, Hyperparameter-tuning, etc.)

Das Phasenmodell ist in Abbildung 48 zu sehen.

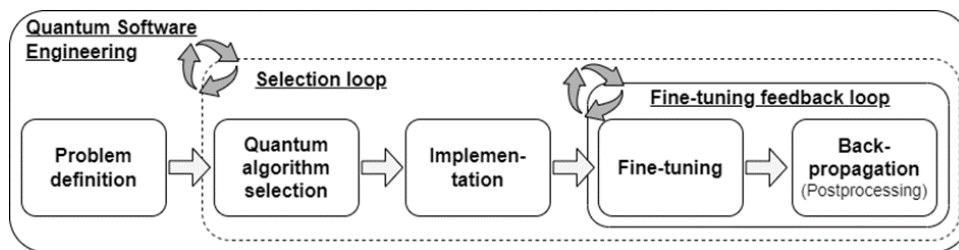


Abbildung 48: Phasenmodell von Quantenalgorithmen

Zu den Phasen werden jeweils die Erfahrung, die in den Phasen gemacht wurden, in verschiedenen Anwendungsfällen diskutiert.

Im Hinblick auf Meilenstein M24 ist die Publikation des SEQUOIA-Modells zwar vorbereitet, allerdings noch nicht erfolgt, da in den letzten Wochen der Projektlaufzeit keine Einreichungsfristen geeigneter Konferenzen oder Workshops liefen. Die Veröffentlichung wird nach Projektende nachgeholt. Sofern die bevorzugte Veröffentlichung über wissenschaftliche Konferenzen oder Workshops nicht erfolgreich ist, wird im Rahmen der geplanten Weiterführung des Projektes als SEQUOIA End-to-End eine Bereitstellung über die Projektwebseite geklärt.

2.6 Arbeitspaket 7: Transfer und Partnerkooperation

Überblick der wissenschaftlichen
Arbeiten

Öffentlichkeitsarbeit, Veranstaltungen und Wissenstransfer

Wie geplant wurden alle Projektergebnisse kontinuierlich der breiten (Fach-) Öffentlichkeit zugänglich gemacht. Zusammenfassend ist dies in Abbildung 49 Die SEQUOIA Wissenstransfer- und Öffentlichkeitsarbeit zusammengefasst. visualisiert.

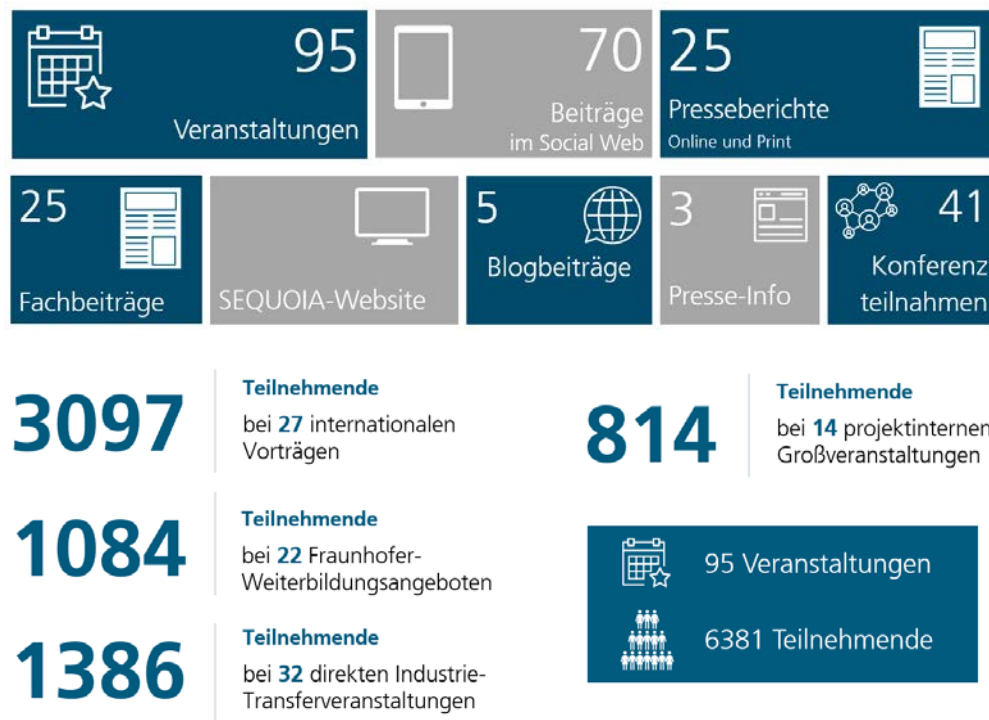


Abbildung 49 Die SEQUOIA Wissenstransfer- und Öffentlichkeitsarbeit zusammengefasst.

Neben der in Kapitel 2.1 diskutierten Studie, wurden die Projektergebnisse in insgesamt 95 Transferveranstaltungen sowohl der breiten Öffentlichkeit kommuniziert als auch der Fachöffentlichkeit verständlich gemacht. Insgesamt sind dabei ca. 6400 Teilnehmende erreicht worden.

Neben den **projektinternen Veranstaltungen**

- **02.03.21** | SEQUOIA Auftaktveranstaltung (279 Teilnehmende)
- **21.04.21** | **25.06.21** | **18.10.21** | **02.08.22** | **01.23** | Unternehmensnetzwerk- und Beiratstreffen (50 Teilnehmende)
- **03.21** | Pilotschulung Quantencomputing im Kompetenzzentrum QC Baden-Württemberg (30 Teilnehmende)
- **30.07.21** | **27.08.21** | QC-Hardware Talks (40 Teilnehmende inkl. PlanQK)
- **23.02.22** | **24.02.22** | SEQUOIA-PlanQK Entwicklertreffen (25 Teilnehmende)
- **29.02.22** | KQCBW-Spitzentreffen: Projektupdate (15 Teilnehmende)
- **18.10.22** | Fraunhofer Kompetenznetzwerktreffen QC (40 Teilnehmende)
- **20 & 21.10.22** | Developer Conference KQCBW (70 Teilnehmende)

wurden die erarbeiteten Projektergebnisse in etablierte **Fraunhofer Weiterbildungsangebote** integriert

- **24.06.21** | **24.03.22** | Institutsfrühstück IAO (50 Teilnehmende)
- **15.07.21** | Office 21 - Quantencomputing auf dem Weg in die Praxis (14 Teilnehmende)
- **22.07.21** | Future-Car: Post-Quantum Cryptography (100 Teilnehmende)
- **27.01.21** | **12.03.21** | **20.04.21** | **10.06.21** | **19.10.21** | **02.11.21** | **25.11.21** | **08.02.22** | **20.09.22** | DigitalDialog: Mythos und Realität (50 Teilnehmende)
- **24.03.21** | **13.07.21** | **21.10.21** | **09.11.21** | **07.12.21** | **21.02.22** | **11.10.22** | DigitalDialog QC lernen, wo fange ich an? (50 Teilnehmende)
- **06.07.21** | **30.09.21** | DigitalDialog Post-Quanten-Kryptographie (40 Teilnehmende)

Überblick der wissenschaftlichen Arbeiten

sowie direkt in die **Industrie transferiert**

- **19.04.21** | Quantencomputing Q&A mit BMAS (15 Teilnehmende)
- **18.05.21** | Lernreise Industrie 4.0: Quantencomputing (150 Teilnehmende)
- **08.06.21** | Video-Statement [Quantencomputer lassen Supercomputer blass aussehen. - Wirtschaftskraft Plus](#)
- **23.06.21** | **28.04.22** | **04.10.22** | Quantum Computing - Basics und Use Cases für Finanzdienstleister LBBW (70 Teilnehmende)
- **30.06.21** | Eco KG Sicherheit – Quantencomputing (19 Teilnehmende)
- **12.07.21** | **12.10.21** | **05.11.21** | **19.11.22** | **03.12.21** | Projektgruppensitzung Quantencomputing für den Leichtbau BW (10 Teilnehmende)
- **24.09.21** | Vortrag IHK Konstanz: QC für Innovationsberater und KMU-Unternehmen (40 Teilnehmende)
- **19.10.21** | MCC Kongress IT-Optionen für Versicherungen (100 Teilnehmende)
- **24.11.21** | [S-TEC - Spitzentreffen »Transformation: NOW & NEXT!«](#) (120 Teilnehmende)
- **09.12.21** | [5. AI Systems Tübingen Meetup: Quantum & AI](#) (50 Teilnehmende)
- **26.01.22** | Quantencomputing in der Praxis: Manufacturing, Bayern Innovativ (100 Teilnehmende)
- **18.02.22** | Future Hotel Digital Business Breakfast: Quantum Computing – Industrial Application Outlook (15 Teilnehmende)
- **24.02.22** | FZI Open House Workshop: Hands-on Quantum Computing – Programmierung und Anwendungen verstehen (29 Teilnehmende)
- **01.03.22** | IBM Besuch Ana Paula de Jesus Assis (10 Teilnehmende)
- **10.03.22** | Lama-Transferveranstaltung Oberhausen: Ressourcenoptimierung durch den Einsatz von Quantencomputern (100 Teilnehmende)
- **17.03.22** | Fraunhofer IAO Kuratoriumssitzung: Quantencomputing in der industriellen Anwendung (30 Teilnehmende)
- **17.03.22** | ZVEI Sondierungsworkshop »Quantentechnologien«, Entwicklung und Demonstration von anwendungsnahen Quantenapplikationen (20 Teilnehmende)
- **21.03.22** | Stifterverband in Ehningen (10 Teilnehmende)
- **28.03.22** | Besuch BDI-Präsident Ehningen (5 Teilnehmende)
- **28.03.22** | Projektsitzung Bundesdruckerei (5 Teilnehmende)
- **06.05.22** | QoolCamp: Start-Up Generator und Absolvententransfer Uni Stuttgart im AIXpress Böblingen (80 Teilnehmende)
- **10.08.22** | IAO-Bundesdruckerei IBM Quantum System One Ehningen Erfahrungssession und Implementierungshilfe (8 Teilnehmende)
- **26.09.22** | Layout-Fachgruppentreffen in Ehningen (20 Teilnehmende)
- **10.10.22** | Arbeitssitzung Wirtschaftsministerium BW: Quantencomputing in der industriellen Applikation (20 Teilnehmende)
- **16.11.22** | Simerics UserMeeting: QCED (100 Teilnehmende)
- **13.12.22** | EFFEKTE Wissenschaftsfestival Karlsruhe - Themenabend Quantencomputing (30 Teilnehmende)

Dabei war das Projekt nicht nur auf nationaler, sondern insbesondere auch auf der **internationalen Bühne vertreten**

Überblick der wissenschaftlichen
Arbeiten

- **24.02.21** | [QCTalks: The Bitter Truth About Quantum Algorithms in the NISQ Era](#) (935 Aufrufe)
- **26.05.21** | **25.04.22** Bitkom Quantum Summit (50 Teilnehmende)
- **01.07.21** | ISC High Performance 2021 Digital Post: Quantum Algorithms for Linear Systems of Engineering (10 Teilnehmende)
- **29.09.21** | [Fraunhofer Italia Trend Dialog - Die Zukunft des Quantencomputings](#) (192 Aufrufe)
- **19.10.21** | [BCCG Expert Circle Business applications for using Quantum technology](#) (100 Teilnehmende)
- **21.10.21** | BaWü International: [Science Meets Industry – Quantum Technologies](#) (80 Teilnehmende)
- **18.11.21** | SC21 Virtual Booth: Quantum algorithms complexity for linear systems of engineering (10 Teilnehmende)
- **13.07.21** | **01.12.21** | **17.03.22** | **12.07.22** | **01.12.22** Quantum Village Ehnningen – QC-Perspektiven für Unternehmen (150 Teilnehmende)
- **01.12.21** | SICOS BW 5. Industrial HPC User Round Table (40 Teilnehmende)
- **03.02.22** | OOP Conference: Quantencomputing in der Anwendung – State-of-the-Art und Future Roadmap (100 Teilnehmende)
- **07.02.22** | Inderprastha Engineering College: Quantum Computing Fundamentals (20 Teilnehmende)
- **31.03.22** | Fraunhofer International Day: Hardware Software Codesign. Application-based Quantum Algorithms in the NISQ Era (150 Teilnehmende)
- **26. & 27.04.22**
World of Quantum Messe: Projektvorstellung (100 Teilnehmende)
- **18.05.22** | Besuch Brasilianische Delegation (20 Teilnehmende)
- **01.06.22** | Besuch Kanadische Delegation (40 Teilnehmende)
- **04.07.22** | Besuch Vizepräsident des EU-Parlaments (10 Teilnehmende)
- **28. & 29.09.22** | Futuras in Res (Fraunhofer Quantentechnologiemesse)
- **05.10.22** | Photonics 4 Future: »Quantum Computing in Industrial Applications: Status Quo and Future Roadmap« (100 Teilnehmende)
- **24. & 25.10.22** | 34th Workshop on Sustained Simulation Performance (50 Teilnehmende)
- **17.11.22** | Innovation Park AI Heilbronn: Innovation QC: Von der Technologie hin zum Aufbau eines vielschichtigen Ökosystems (40 Teilnehmende)
- **23.11.22** | ICDK Innovation Talk on Quantum Ecosystems: Quantum Use Cases (100 Teilnehmende)
- **28.11.22** | House of Logistics and Mobility (HOLM): Quantencomputing in der Industriellen Applikation: Status Quo und Future Roadmap (150 Teilnehmende)

Mithilfe all jener fachlichen Vorträgen auf nationaler und internationaler Ebene konnte ein erfolgreicher Wissenstransfer für unterschiedliche Zielgruppen gewährleistet werden. Sowohl Unternehmen als auch andere Forschungsinstitutionen konnten vom im Projekt erarbeiteten Fachwissen profitieren.

Obgleich das vom Fraunhofer IAO und IAF koordinierte Schulungsprogramm Quantencomputing hier ebenfalls aufgezählt werden müsste, gehen wir der Übersichtlichkeit halber hierauf erst in Abschnitt 3.2 ein.

Wie in Abbildung 49 verdeutlicht, beschränkte sich die Projekt-Öffentlichkeitsarbeit aber nicht nur auf Veranstaltungen, sondern beinhaltete außerdem eine Vielzahl an:

Presseinformationen

Überblick der wissenschaftlichen
Arbeiten

- **10.05.21** | [Unternehmensnetzwerk »Quantencomputing«: Von der Forschung in die Praxis](#)
- **25.11.21** | [Quantencomputing: Know-how in Unternehmen bringen](#)
- **21.07.22** | [Von Quantum Awareness zur Quantum Readiness mit erfolgreichem Know-how-Transfer](#)

Presseberichte

- **01.02.21** | [Saxony - The High-Tech Network](#) | [Zeitenvogel](#) | [Reinraum Online](#) | Quantencomputing - Hohes Zukunftspotenzial für die Industrie
- **01.03.21** | Sensorik Magazin | [Quantencomputing: Hohes Zukunftspotenzial für die Industrie](#)
- **02.03.21** | Deutsche Initiative für Netzwerkinformation e.V. | [Projektstart Quantencomputing: von der Forschung ins Unternehmen](#)
- **April 21** | Gauß Allianz | [Quantencomputing für den Einsatz in der Wirtschaft](#)
- **15.05.21** | it-daily.net | [Unternehmensnetzwerk Quantencomputing](#)
- **08.06.21** | WirtschaftsKraftplus | [Quantencomputer lassen Supercomputer blass aussehen](#)
- **08.06.21** | Datacenter Insider | [Quantentechnologie-Gipfel von Bitkom](#)
- **27.07.21** | all-electronics | [Der Weg zum Quantencomputer: Herausforderungen und Potenzial](#)
- **26.10.21** | bw-l.de (Newsroom) | [Wissenschaft trifft Wirtschaft: Quantencomputing in Baden-Württemberg](#)
- **12.11.21** | Quantum Computing Report | [Classiq Joins the Fraunhofer Led Sequoia Project Which Focuses on Industrial Hybrid Quantum Applications and Algorithms](#)
- **15.11.21** | Quantum Zeitgeist | [Classiq to Collaborate with The Fraunhofer Institute and other Leading Universities on New Methods for Industrial Quantum Use](#)
- **15.11.21** | Semiconductor Digest | [Classiq to Collaborate with The Fraunhofer Institute](#)
- **02.12.21** | all-electronics | [Wie Fraunhofer Quantencomputing-Knowhow in Unternehmen bringt](#)
- **Spezialausgabe 2021** | Heise Magazine iX | [Quantencomputer in Deutschland](#)
- **01.05.22** | Die Magaziniker 6.0 | [Die deutsche Industrie gibt sich die Quanten](#)
- **02.06.22** | Merton Onlinemagazin des Stifterverbandes | [Wir müssen Quantentechnologie in die Schulen und Universitäten bringen](#)
- **07.07.22** | Think-ing | [Irgendwas zwischen Null und Eins](#)
- **08.09.22** | InnoVisions | [Die Industrie rechnet mit Quantenbits](#)
- **17.09.22** | Industry of Things | [Durch Know-how-Transfer zur Quantum Readiness](#)
- **01.08.22** | Datacenter Insider | [Die Quantencomputing-Schulungen von Fraunhofer IAO und Fraunhofer IAF](#)
- **18.11.22** | Heilbronner Stimme | [KI-Park Heilbronn: Ökosysteme wachsen nicht von alleine](#)
- **30.11.22** | all-electronics | [Das plant das Projekt SEQUOIA für Quantencomputer](#)
- Deutschlandfunk | Quantencomputing im Projekt SEQUOIA (in Vorbereitung)

Blogbeiträge

Überblick der wissenschaftlichen
Arbeiten

- **12.03.21** | [Quantencomputing – Hype oder Quantensprung für Unternehmen?](#)
- **27.05.21** | [Post-Quanten-Kryptographie – wie man trotz Quantencomputer noch sichere IT-Systeme betreiben kann](#)
- **01.12.21** | [Was macht einen Quantencomputer so mächtig? Teil 1: Qubits, Superposition und weitere Grundlagen](#)
- **02.12.21** | [Was macht einen Quantencomputer so mächtig? Teil 2: Der Quantenparallelismus](#)
- **08.07.22** | [Quantencomputing: Drei Erfolgsfaktoren für den Weg in die industrielle Anwendung](#)

Veröffentlichungen in Fachmagazinen (peer-review siehe unten)

- **20.12.21** | [Quantum Computing – The Path Towards Industrial Applications](#)

Webpräsenz:

- [Software-Engineering industrieller, hybrider Quantenanwendungen und -algorithmen \(SEQUOIA\) \(fraunhofer.de\)](#)
- [SEQUOIA Webseite](#)
- LinkedIn Fokussseite: [The Quantum Future](#)
- LinkedIn Gruppe: [Quantum Village Ehningen](#)
- [Initiative Wirtschaft 4.0 Baden-Württemberg \(IW4.0\) | LinkedIn Eventserie](#)
- Podcast: [Nachwuchs gesucht! Neue Köpfe für Quantencomputing](#)

Des Weiteren war es zentrale Aufgabe des Arbeitspaketes 7, das SEQUOIA Projekt hinsichtlich **Vernetzung und Kooperationen** im Deutschen und Baden-Württembergischen Quantenökosystem zu etablieren sowie **Folgeprojekte** zu motivieren, um das erarbeitete Wissen weiter zu verwerten und zu verstetigen.

■ SEQUOIA Unternehmensnetzwerk

Wie im Kontext von Abbildung 3 gezeigt und diskutiert, wurde im Projekt ein Unternehmensnetzwerk von 28 Firmen aus unterschiedlichen Branchen aufgebaut und vernetzt. Im Fokus stand dabei der Austausch über aktuelle Entwicklungen des Quantencomputings sowie die Erarbeitung von konkreten Nutzungsszenarien für Quantencomputer in der Wirtschaft. Die Unternehmen bekamen dabei die Möglichkeit, sich untereinander zu vernetzen und gemeinsam mit dem Projektkonsortium ihre spezifischen Nutzungsszenarien für einen potenziellen Einsatz des Quantencomputers im eigenen Unternehmen zu erarbeiten. Im Rahmen von regelmäßigen Netzwerktreffen wurden erste konkrete Nutzungsszenarien der Quantencomputer präsentiert, welche die Wirtschaft in Baden-Württemberg voranbringen werden. Es sei anzumerken, dass bereits in der Antragsphase des Folgeprojektes SEQUOIA End-to-End jenes Netzwerk mit neuen LOI-Partnern erweitert werden konnte.

■ Kooperation mit weiteren Forschungsvorhaben

Der Wissensaustausch zwischen den Projekten **PlanQK**, **AutoQML** und **SEQUOIA** war (und ist weiterhin) zentraler Bestandteil der Projektbemühungen. Gemeinsam wurden hier Entwicklerworkshops durchgeführt (siehe Veranstaltungen) und Expertise ausgetauscht, um gemeinsam die Forschungslandschaft in Deutschland effizient voranzubringen.

Um dies insbesondere im Rahmen des **Kompetenzzentrums Quantencomputing BW** zu gewährleisten, führte das Fraunhofer IAO und IAF die erste gemeinsame Entwickelndenkonferenz im Land Baden-Württemberg in enger Kooperation mit der **IBM** in Ehningen durch (Developerkonferenz siehe Veranstaltungsliste). Mit über 70 interdisziplinären Wissenschaftlern aus dem Kompetenzzentrum Quantencomputing Baden-Württemberg, des AutoQML Konsortiums und des PlanQK Projektes (inkl. Unternehmensvertretern) wurden hier Synergien geschaffen, das Ökosystem vernetzt und gemeinsame Quantenlösungen diskutiert. Es ist geplant jenes Format im Land regelmäßig weiterzuführen.

Es sei außerdem angemerkt, dass im Rahmen des **Fraunhofer IAO-IAF Quantencomputing Schulungsprogramms** die Zusammenarbeit und Vernetzung zwischen den Projekten SEQUOIA, EFFEKTIF, QORA, QC4BW, QuEst und SiQuRe (siehe Kapitel 3.2) signifikant gefördert worden ist. In zahlreichen Workshops und einer Schulungskonferenz wurde hier Wissen vermittelt, ausgetauscht und gemeinsame Lösungsstrategien befeuert.

■ Folgeprojekte

Zuletzt haben sich aus dem Projekt SEQUOIA heraus ebenfalls mehrere Forschungsvorhaben / Projektanträge als Zusammenschluss von mehreren Projektpartnern (und Unternehmen) mit gleichen Interessen ergeben:

- o **AutoQML** (BMWK, bewilligt und im Januar 2022 begonnen)
Fraunhofer IAO und IPA, Zeppelin, Trumpf, GFT, IAV, USU, KEB
- o **Leichtbau QC** (DLR, Skizze eingereicht)
Fraunhofer IAO, DLR, KENBUN IT AG, EDAG Group, CSI Entwicklungstechnik, EnDes Gruppe, GFT, DYNAMore, Leichtbau BW
- o **QuASy** (EU-Projekt, abgelehnt)
Atos, ColibriTD, QBee Uni Stuttgart HLRS, Fraunhofer IAO, et al.
- o **European Digital Innovation Hub** – Quantentechnologien (Konzept im nationalen Vorentscheid erfolgreich; keine Antragstellung auf europäischer Ebene, da finales Finanzierungskonzept der Europäischen Kommission von den Konsortialpartnern nicht abgebildet werden konnte) DLR, Fraunhofer, FZI
- o **QuAX** – Quantum und AI Experience Center (Direktauftrag erteilt)
Fraunhofer IAO (in enger Kooperation mit IBM, Quantum Village)
- o **KODIS 2** - Forschungs- und Innovationszentrum Kognitive Dienstleistungssysteme (Antrag eingereicht)
Fraunhofer IAO (in enger Kooperation mit der Dieter Schwarz Stiftung)

Teilnahme an (weiteren) Fachkonferenzen, Workshops und Schulungen:

Zuletzt hat das Konsortium ebenfalls an (inter-)nationalen Fachkonferenzen, Workshops und Schulungen teilgenommen, um aktuelle state-of-the-art Forschung betreiben zu können, unser Ökosystem in der Quantencommunity zu repräsentieren und nachhaltig zu vernetzen (peer-reviewed Konferenzen siehe 3.1 Publikationen)

- **14.01.21** | PlanQK-Webinar: Einführung TensorflowQuantum
- **22.03.21** | Workshop on Quantum Software Architecture (QSA) im Rahmen der IEEE International Conference on Software Architecture (ICSA)
- **26.05.21** | Bitkom Quantum Summit 2021
- **10.06.21** | PlanQK-Webinar: Verschränkung und das Verschieben von Schranken
- **07.21** | Qiskit Global Summer School on Quantum Machine Learning
- **07.07.21** | PlanQK-Webinar: Hybride Quantenanwendungen
- **07.10.21** | PlanQK-Webinar: »Messen ist einfach!« ist vermessen
- **11.11.21** | IBM Quantum Webinar: Quantifying the Impact of Precision Errors on Quantum Approximate Optimization Algorithms
- **02.12.21** | 2nd European Quantum Technologies Virtual Conference (EQTC)
- **07.12.21** | PlanQK-Webinar: Parity Quantum Computing
- **13.12.21** | IBM-Workshop for Fraunhofer on advanced transpilation methods
- **10.12.21** | ETP4HPC webinar: Quantum for HPC
- **2022** | Fraunhofer IAO-IAF Quantencomputing Schulungsprogramm
- **18.01.22** | IBM Quantum Webinar: Quantum kernels for real-world predictions based on electronic health records
- **04.05.22** | IBM-Workshop for Fraunhofer on circuit mapping and QAOA
- **13.04.22** | PlanQK Industry Day: Financial Services
- **25.04.22** | Fujitsu Workshop on Digital Annealing: Combinatorial Optimization
- **11.05.22** | Bitkom Quantum Summit 2022
- **07.22** | Qiskit Global Summer School on Quantum Error Correction
- **07.22** | Qiskit Global Summer School on Quantum Simulations
- **22.09.22** | PlanQK Industry Day for Energy
- **14.10.22** | PlanQK-Webinar: Heuristisches zur Bell-Ungleichung: Hintergrund zum Physik-Nobelpreis 2022
- **11.22** | IBM Quantum Summit 2022, Practitioners Forum, Yorktown Lab Visit
- **03.11.22** | PlanQK-Webinar: Quanten-Topologische Datenanalyse
- **30.11.22** | MDPI Entropy Webinar: Advances in Quantum Computing
- **07.12.22** | 1. Quantum Meetup @ KIT

Überblick der wissenschaftlichen Arbeiten

2.7 Arbeitspaket 8: Projektmanagement

Neben den Aufgaben des Projektmanagements:

- Einrichten einer JF-Struktur (AP-Lead und Entwickler)
- Koordination der Projektpartner (insb. Berichte und Veranstaltungen)
- Sicherstellen der (Teil-)AP Ergebnisse und Meilensteine
- Sicherstellen der Ticketplanung und des Ticketabrufs
- Organisation einer Entwickelndenkonferenz (siehe Veranstaltungsliste)
- Organisation von Unternehmensnetzwerktreffen

ist hervorzuheben, dass alle Partner den Projekt-Kooperationsvertrag sowie im Herbst 2021 die Nutzungs- und Lizenzverträge zum Rechnen auf dem IBM Quantum System One in Ehningen unterschrieben haben.

3.1 Wissenschaftliche Publikationen

Die im Rahmen des SEQUOIA-Forschungsprojekts gewonnenen Erkenntnisse und erzielten Forschungsergebnisse wurden in einer Reihe von Publikationen für die wissenschaftliche Fachwelt aufbereitet. Hierbei lag ein besonderer Fokus auf der Anwendung und Umsetzung von hybriden Quantenalgorithmen, deren Bestandteile teils auf klassischer Hardware und teils auf Quantencomputern ausgeführt werden. Verschiedene Varianten hybrider Quantenalgorithmen und der dabei verwendeten Konzepte wurden untersucht und als Patterns aufbereitet [1]. Als Spezialfall wurde das »Warmstarten« von Quantenalgorithmen erforscht [7]. Die Modellierung und Automatisierung hybrider Quantenalgorithmen einschließlich der Orchestrierung der notwendigen Komponenten wurden konzipiert und prototypisch umgesetzt [2, 11]. Methoden, Werkzeuge und Technologien zur Realisierung von hybriden Quantenanwendungen in der Cloud wurden in Hinblick auf die daraus entstehenden Herausforderungen für Softwareentwickler untersucht [3, 4, 10]. Best Practices zur Verbesserung der Qualität von fehlerbehafteten Ergebnissen bei der Ausführung auf Quantencomputern wurden analysiert und als Patterns aufbereitet [12]. Die Übertragung von klassischen Verifikations-, Test- sowie Spezifikationsmethoden auf Quantencomputing wurde in [6] untersucht. [5] bietet auf der Basis von Optimierungswerkzeugen Hilfe zur Entscheidungsfindung für Quantensoftwareingenieure in komplexen Entwurfsräumen. In [16,17] werden schließlich die gesammelten Ergebnisse der Use Cases zusammengetragen.

• Wissenschaftliche Publikationen

- [1] M. Weigold, J. Barzen, F. Leymann, and D. Vietz, »Patterns for Hybrid Quantum Algorithms« in *Proceedings of the 15th Symposium and Summer School on Service-Oriented Computing (SummerSOC 2021)*, Sep. 2021, pp. 34–51, doi: 10.1007/978-3-030-87568-8_2
- [2] B. Weder, J. Barzen, F. Leymann, and M. Zimmermann, Hybrid Quantum Applications Need Two Orchestrations in Superposition: »A Software Architecture Perspective« in *Proceedings of the 18th IEEE International Conference on Web Services (ICWS 2021)*, 2021, pp. 1–13, doi: 10.1109/ICWS53863.2021.00015
- [3] D. Vietz, J. Barzen, F. Leymann, and K. Wild, »On Decision Support for Quantum Application Developers: Categorization, Comparison, and Analysis of Existing Technologies« in *Computational Science -- ICCS 2021*, 2021, pp. 127–141, doi: 10.1007/978-3-030-77980-1_10
- [4] D. Vietz, J. Barzen, F. Leymann, B. Weder, and V. Yussupov, »An Exploratory Study on the Challenges of Engineering Quantum Applications in the Cloud« in *Proceedings of the 2nd Quantum Software Engineering and Technology Workshop (Q-SET 2021) co-located with IEEE International Conference on Quantum Computing and Engineering (QCE21)*, 2021, pp. 1–12
- [5] M. Scheerer, J. Klamroth, Denninger, »Engineering Reliable Hybrid Quantum Software: An Architectural-driven Approach« 2nd International Workshop on Quantum Software Engineering & Technology (QSET 2021)
- [6] J. Klamroth, M. Scheerer, O. Denninger, »Holistische Verifikation von Hybriden Quantenprogrammen durch Software Bounded Model Checking« Workshop Herausforderungen beim Testen moderner Softwaresysteme @ GI Jahrestagung 2021

- [7] F. Truger, M. Beisel, J. Barzen, F. Leymann, and V. Yussupov, »Selection and Optimization of Hyperparameters in Warm- Quantum Optimization for the MaxCut Problem« in *Electronics*, MDPI, 2022
- [8] M. Scheerer, J. Klamroth and O. Denninger, »Fault-tolerant Hybrid Quantum Software Systems«, 2022 IEEE International Conference on Quantum Software (QSW), 2022, pp. 52-57, doi: 10.1109/QSW55613.2022.00023
- [9] J. Klamroth, M. Scheerer and O. Denninger, »QIn: Enabling Formal Methods to Deal with Quantum Circuits«, Quantum Physics and Logic 2022 (QPL'22) Poster
- [10] B. Weder, J. Barzen, F. Leymann, and D. Vietz, »Quantum Software Development Lifecycle« in *Quantum Software Engineering*, Springer International Publishing, 2022 ([arXiv](#))
- [11] B. Weder, J. Barzen, M. Beisel, and F. Leymann, »Analysis and Rewrite of Quantum Workflows: Improving the Execution of Hybrid Quantum Algorithms« in *Proceedings of the 12th International Conference on Cloud Computing and Services Science (CLOSER 2022)*, 2022
- [12] M. Beisel, J. Barzen, F. Leymann, F. Truger, B. Weder, and V. Yussupov, »Patterns for Quantum Error Handling« in *Proceedings of the Fourteenth International Conference on Pervasive Patterns and Applications (PATTERNS 2022)*, 2022
- [13] Vietz, Daniel; Barzen, Johanna; Leymann, Frank; Weder, Benjamin: Splitting Quantum-Classical Scripts for the Generation of Quantum Workflows. In: Proceedings of the 26th Conference on Enterprise Design, Operations, and Computing (EDOC 2022), Springer International Publishing, 2022
- [14] Beisel, Martin; Barzen, Johanna; Leymann, Frank; Truger, Felix; Weder, Benjamin; Yussupov, Vladimir: Configurable Readout Error Mitigation in Quantum Workflows. In: Electronics. Vol. 11(19), MDPI, 2022
- [15] M. Beisel, S. Garhofer, F. Leymann, F. Truger, B. Weder, and V. Yussupov, »Quokka: A Service Ecosystem for Workflow-Based Execution of Variational Quantum Algorithms«, 20th International Conference on Service Oriented Computing, 2022

• In Vorbereitung

- [16] F. Shagieva, B. C. Mummaneni, C. Tutschku »Quantum Computational Fluid Dynamics«, *in prep.*
- [17] A. Sturm, F. Knäble, C. Tutschku »Quantum Optimization in Mobility and Logistics«, *in prep.*
- [18] G. Koßmann, A. Sturm, C. Tutschku, »Quantum Optimization for Job Shop Scheduling«, *in prep.*
- [19] N. Schillo, »Quantum Algorithms and Quantum Machine Learning for Differential Equations«, Masterarbeit betreut von A. Sturm, voraussichtliche Fertigstellung November 2023
- [20] P. A. Matt, M. Huber, M. Roth, »A quantum-inspired heuristic for solving the irregular strip packing problem«, *in prep.*
- [21] S. Garhofer, O. Bringmann »Using an A*-based framework for decomposing combinatorial optimization problems to employ NISQ computers«, *in prep.*
- [22] F. Truger, J. Barzen, M. Bechtold, M. Beisel, F. Leymann, A. Mandl, and V. Yussupov »A Systematic Mapping Study on Warm-Starting Techniques in the Quantum Computing Domain«, work in progress, *in prep.*
- [23] L. Zhong, Q. Pan, D. Hoppe »Quantum Deep Neural Network for Solving PDEs«, *in prep.*
- [24] L. Zhong, Q. Pan, D. Hoppe »Is QNN Really Beneficial in Image Processing? A Comparative Study«, *in prep.*

3.2 Im Themenfeld Quantencomputing geschulte Personen

Wie bereits in der Beschreibung von Arbeitspaket 7 erwähnt, wurden im Rahmen von SEQUOIA-Transferveranstaltungen ca. 6400 Teilnehmende erreicht bzw. in verschiedenen fachlichen Tiefen geschult. Von der Management Sicht, bis zum Entwicklerwissen wurde die erarbeitete Expertise in zahlreichen Veranstaltungen mit unterschiedlichem Tiefgang vermittelt.

Darüber hinaus, um möglichst vielen Interessierten das nötige Know-How zur effizienten Nutzung von Quantencomputern wie dem IBM-Quantencomputer in Ehningen zu vermitteln, entwickelte das Fraunhofer IAO und das Fraunhofer IAF im Rahmen des Kompetenzzentrums »Quantencomputing Baden-Württemberg« ein [Schulungsangebot Quantencomputing \(fraunhofer.de\)](https://schulungsangebot.quantencomputing.fraunhofer.de), welches dies über das Jahr 2022 erstmals systematisch getan hat. Da jenes Konzept in Phase 0 des KQCBWs entwickelt wurde und die Inhalte signifikant auf der Expertise des Forschungsprojektes SEQUOIA beruhen, wird dies im Folgenden beschrieben.

Vermittelt wurden in diesem Programm fundierte Kenntnisse in Quantentechnologien und deren Programmierung sowie erste Erfahrungen mit realen Quantencomputern wie dem IBM Quantum System One in Ehningen (Grundlagenkurs Quantencomputing). Angefangen bei den naturwissenschaftlichen Grundprinzipien der Physik, Informatik und Mathematik führte das Schulungsprogramm hin zur praxisnahen Implementierung fundamentaler Quantenalgorithmen sowohl auf der hauseigenen IAO / IAF-Simulationsumgebung (KQCBW Phase 0) als auch auf dem IBM Quantum System One in Ehningen. Der Umgang mit diesem Quantensystem und das Kennenlernen seiner Charakteristika definierten einen zentralen Aspekt der Schulungsinhalte.

Die Teilnehmenden aus Forschung und Industrie erhielten außerdem einen Einblick in eine Vielzahl von Spezialthemen, wie etwa die (hybriden) Quantenalgorithmen Deutsch-Jozsa, Bernstein-Vazirani, Grover, HHL, Shor, QFT, QPE, VQE, VQLS und QAOA. Diese wurden in Form von dreistündigen Webinaren bis hin zu Halbtagesseminaren vermittelt. Weiter wurden die Konzepte Fehlermitigation, effizientes Transpiling, sowie Qiskit Pulse- oder Runtime-Methoden erläutert und diskutiert. Die Teilnehmenden konnten bei den Vertiefungen und den Anwendungsgebieten ihre Interessen und ihr Vorwissen mit einbringen.

Auf Wunsch der Teilnehmenden wurde insbesondere eine extra Session zur Post-Quantum Kryptographie durch das Fraunhofer AISEC in das Programm inkludiert. Abschließend wurden die erlernten QC-Werkzeuge zur Realisierung konkreter Anwendungsfälle verwendet (Use-Case Sessions).

Die Teilnehmenden lernten insbesondere die unterschiedlichen QC-Hardware- und Software-Ansätze sowie deren Vor- bzw. Nachteile für ausgewählte Use Cases kennen. Dabei standen hybride QC-Infrastrukturen und deren cloudbasierte Nutzung im Fokus. Dies geschah zum einen durch Webinare von Fraunhofer und Technologieanbietern (IBM, QuTech, Quantum Brilliance, D-Wave, QuiX, Pasqal, Fujitsu, Anaqor, IoSQuantum, HQS, Classiq, Quantinuum, AWS) als auch durch Gastbeiträge von bedeutenden nationalen Förderprojekten (European Quantum Industry Consortium, European Photonics Industry Consortium, Infineon Technologies AG, QBN, Bitkom, PlanQK, AQUAS, EniQma, AutoQML, Q.AX). Zentrales Ziel war hierbei ein proaktiver Austausch der Quantencommunity sowie eine realistische Einschätzung der aktuellen Wettbewerbslandschaft und der perspektivischen Entwicklung des QC-Ökosystems.

Im Detail sind die konkreten Schulungsinhalte dem angehängten PDF zu entnehmen (Anlage 1). Ergänzend ist mit Unterstützung von IBM eine Schulungskonferenz mit meet-the-expert Sessions sowie ein Vor-Ort-Besuch beim Quantencomputer in Ehningen durchgeführt worden (ca. 40 Teilnehmende). Ein Teil der angebotenen Netzwerk-

und Austauschveranstaltungen wurde außerdem für die Öffentlichkeit zugänglich gemacht, um so die Vernetzung des Ökosystems in Baden-Württemberg aktiv voranzutreiben.

Key-Performance-Indikatoren

Im Rahmen dieses Programms wurden insgesamt 81 Teilnehmende aus 41 Organisationen geschult, wobei eine Interessensbekundung (erster Kontakt) von 52 Unternehmen zu verzeichnen war. Davon nahmen 20 Unternehmen mit insgesamt 24 Personen teil. Hinzu kamen 21 FuE-Partner mit insgesamt 57 Teilnehmenden. Die konkret teilnehmenden Unternehmen und Partner sind Abbildung 50 zu entnehmen.

20 Unternehmen mit 24 Teilnehmenden	21 FuE-Partner mit 57 Teilnehmenden
- Porsche Digital GmbH - Schwarz IT KG - IBM Deutschland Research & Development GmbH - Trumpf Werkzeugmaschinen GmbH und Co. KG - Deutsche Bank AG - Landesbank Baden-Württemberg (LBBW) - Rohde&Schwarz GmbH & Co. KG - BASF SE - Airbus Defence and Space GmbH - Adesso SE - USU Software GmbH - Frank Arendt Software Engineering - Zeppelin GmbH - SIMERICS GmbH - Deutsches Elektronen-Synchrotron DESY - tiri GmbH - Evatec GmbH - EDAG Engineering GmbH - EurA Consult AG - Xitaso GmbH	- FZI Forschungszentrum Informatik - Fraunhofer IAO - Fraunhofer IAF - Fraunhofer EMI - Fraunhofer EMFT - Fraunhofer IPA - Fraunhofer ICT - Fraunhofer IWM - DLR-Institut für Technische Thermodynamik - DLR-Institut für Quantentechnologien - Duale Hochschule Baden-Württemberg (DHBW) - Universität Stuttgart - FMQ TI HLRS - Institut für komplexe Quantensysteme der Universität Ulm - Karlsruhe Instituts für Technologie KIT - TKM und PHI - Physikalisches Institut der Universität Konstanz - Eberhard Karls Universität Tübingen (EKUT) - Theoretical Quantum Optics Embedded Systems - Hochschule Furtwangen University (HFU)

Abbildung 50 Teilnehmer am Schulungsprogramm Quantencomputing 2022 des Fraunhofer IAO und Fraunhofer IAF.

Jenes Schulungsprogramm wird 2023 modularisiert weitergeführt und sukzessive weiterentwickelt.

Es ist besonders hervorzuheben, dass jenes Programm insbesondere dem Wissenstransfer (Verwertung und Verstetigung) der KQCBW Projekte dient, sowie (auch in Zukunft) jenen Projekten eine Bühne im QC-Ökosystem bietet.