

Real-time simulations of transmon systems with time-dependent Hamiltonian models

Hannes Alfred Lagemann

IAS Series

Band / Volume 56

ISBN 978-3-95806-720-2

Forschungszentrum Jülich GmbH
Institute for Advanced Simulation (IAS)
Jülich Supercomputing Centre (JSC)

Real-time simulations of transmon systems with time-dependent Hamiltonian models

Hannes Alfred Lagemann

Schriften des Forschungszentrums Jülich
IAS Series

Band / Volume 56

ISSN 1868-8489

ISBN 978-3-95806-720-2

Bibliografische Information der Deutschen Nationalbibliothek.
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der
Deutschen Nationalbibliografie; detaillierte Bibliografische Daten
sind im Internet über <http://dnb.d-nb.de> abrufbar.

Herausgeber
und Vertrieb: Forschungszentrum Jülich GmbH
Zentralbibliothek, Verlag
52425 Jülich
Tel.: +49 2461 61-5368
Fax: +49 2461 61-6103
zb-publikation@fz-juelich.de
www.fz-juelich.de/zb

Umschlaggestaltung: Grafische Medien, Forschungszentrum Jülich GmbH

Titelbild: Forschungszentrum Jülich/Sascha Kreklau

Druck: Grafische Medien, Forschungszentrum Jülich GmbH

Copyright: Forschungszentrum Jülich 2023

Schriften des Forschungszentrums Jülich
IAS Series, Band / Volume 56

D 82 (Diss. RWTH Aachen University, 2023)

ISSN 1868-8489
ISBN 978-3-95806-720-2

Vollständig frei verfügbar über das Publikationsportal des Forschungszentrums Jülich (JuSER)
unter www.fz-juelich.de/zb/openaccess.



This is an Open Access publication distributed under the terms of the [Creative Commons Attribution License 4.0](https://creativecommons.org/licenses/by/4.0/),
which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Zusammenfassung

In dieser Dissertation untersuchen wir die Aspekte von zeitabhängigen Hamiltonoperatoren (Modellen), die die zeitliche Entwicklung von Transmon Systemen beeinflussen können. Wir modellieren die Zeitentwicklung verschiedener Systeme als unitäre Echtzeitprozesse. Dazu lösen wir die zeitabhängige Schrödingergleichung (ZSG) numerisch. Die resultierenden Computermodelle bezeichnen wir als nicht-ideale-gatterbasierte Quantencomputer (NIGQC) Modelle, da Transmons in der Regel als Transmon Qubits in supraleitenden prototyp-gatterbasierten Quantencomputern (PGQC) verwendet werden.

Wir diskutieren zunächst das Modell des idealen-gatterbasierten Quantencomputers (IGQCs) und differenzieren zwischen den Begriffen IGQC, PGQC und NIGQC, welche wir in dieser Arbeit betrachten. Im Anschluss leiten wir die Schaltkreis Hamiltonoperatoren ab, welche die Zeitentwicklung von frequenzfesten und stimbaren Transmons generieren. Darüber hinaus diskutieren wir kurze und prägnante Ableitungen für effektive Hamiltonoperatoren (für beide Arten von Transmons). Wir verwenden die Schaltkreis Hamiltonoperatoren und die effektiven Hamiltonoperatoren, um zwei Mehrteilchen Hamiltonoperatoren zu definieren. Die Wechselwirkungen zwischen den verschiedenen Teilsystemen werden als Dipol-Dipol Wechselwirkungen modelliert. Im Anschluss entwickeln wir zwei Produkt-Formel Algorithmen, welche es uns erlauben die ZSG für die von uns definierten Mehrteilchen Hamiltonoperatoren numerisch zu lösen.

Anschließend verwenden wir diese Algorithmen, um zu untersuchen wie sich verschiedene Näherungen (Annahmen) auf die Zeitentwicklung von Transmon Systemen auswirken. Hier betrachten wir den effektiven Mehrteilchen Hamiltonoperator und nutzen den Schaltkreis Mehrteilchen Hamiltonoperator als einen Referenzpunkt (im übertragenen Sinne). Wir betrachten ausschließlich Szenarien bei denen ein Steuerungssignal genutzt wird. Wir finden, dass die von uns untersuchten Näherungen die Zeitentwicklung der von uns modellierten Wahrscheinlichkeitsamplituden erheblich beeinflussen können. Des Weiteren untersuchen wir, wie anfällig die numerischen Werte verschiedener Gatter-Fehler Quantifikatoren gegenüber Näherungen (Annahmen) sind. Wir

stellen fest, dass die von uns betrachteten Näherungen (Annahmen) eindeutig Gatter-Fehler Quantifikatoren wie die Diamant-Distanz und die durchschnittliche Fidelität beeinflussen. Des Weiteren sind wir oft in der Lage klare und prägnante theoretische Erklärungen für die Ergebnisse, die wir in dieser Arbeit diskutieren, zu präsentieren.

Abstract

In this thesis we study aspects of Hamiltonian models which can affect the time evolution of transmon systems. We model the time evolution of various systems as a unitary real-time process by numerically solving the time-dependent Schrödinger equation (TDSE). We denote the corresponding computer models as non-ideal gate-based quantum computer (NIGQC) models since transmons are usually used as transmon qubits in superconducting prototype gate-based quantum computers (PGQCs).

We first review the ideal gate-based quantum computer (IGQC) model and provide a distinction between the IGQC, PGQCs and the NIGQC models we consider in this thesis. Then, we derive the circuit Hamiltonians which generate the dynamics of fixed-frequency and flux-tunable transmons. Furthermore, we also provide clear and concise derivations of effective Hamiltonians for both types of transmons. We use the circuit and effective Hamiltonians we derived to define two many-particle Hamiltonians, namely a circuit and an associated effective Hamiltonian. The interactions between the different subsystems are modelled as dipole-dipole interactions. Next, we develop two product-formula algorithms which solve the TDSE for the many-particle Hamiltonians we defined.

Afterwards, we use these algorithms to investigate how various frequently applied approximations (assumptions) affect the time evolution of transmon systems modelled with the many-particle effective Hamiltonian when a control pulse is applied. Here we also compare the time evolutions generated by the effective and circuit Hamiltonian. We find that the approximations we investigate can substantially affect the time evolution of the probability amplitudes we model. Next, we investigate how susceptible gate-error quantifiers are to approximations which make up the NIGQC model. We find that the approximations (assumptions) we consider clearly affect gate-error quantifiers like the diamond distance and the average infidelity. Furthermore, we provide clear and concise theoretical explanations for many of the findings we present in this thesis.

Contents

Zusammenfassung	i
Abstract	iii
1 Introduction	1
2 An introduction to gate-based quantum computing	3
2.1 Mathematical framework	3
2.2 The single-qubit Hilbert space	6
2.3 The multi-qubit Hilbert space	8
2.4 Simulation of the ideal gate-based quantum computer	11
2.5 Algorithms	13
2.6 Prototype gate-based quantum computers	17
2.7 Non-ideal gate-based quantum computers	21
2.8 Summary and conclusions	23
3 Hamiltonian models for lumped-element circuits	25
3.1 The lumped-element approximation	25
3.2 The circuit quantisation formalism	29
3.3 Resonators and two-level systems	33
3.4 Fixed-frequency and flux-tunable transmons	36
3.5 A circuit Hamiltonian model for non-ideal gate-based transmon quantum computers	54
3.6 An effective Hamiltonian model for non-ideal gate-based transmon quantum computers	56
3.7 Summary and conclusions	58
4 Real-time simulation algorithms for time-dependent Hamiltonians	60
4.1 Solving the time-dependent Schrödinger equation numerically	60
4.2 The product-formula algorithm	62

4.3	The time-evolution operator for the effective Hamiltonian model	64
4.4	The time-evolution operator for the circuit Hamiltonian model	68
4.5	Simulations of the circuit Hamiltonian model in alternative bases	72
4.6	Implementation of the second-order time-evolution operator .	74
4.7	An overview of the simulation software	76
4.8	Summary, conclusions and outlook	80
5	Analysis of effective models for flux-tunable transmon systems	82
5.1	System specification and simulation parameters	83
5.2	Simulations of a single flux-tunable transmon	90
5.3	Simulations of suppressed transitions in the adiabatic effective two-qubit model: architecture I	96
5.4	Simulations of unsuppressed transitions in the adiabatic effective two-qubit model: architecture I	100
5.5	Simulations of suppressed transitions in the adiabatic effective two-qubit model: architecture II	110
5.6	Simulations of unsuppressed transitions in the adiabatic effective two-qubit model: architecture II	112
5.7	Summary, conclusions and outlook	119
6	On the fragility of gate-error metrics in simulations of flux-tunable transmon quantum computers	123
6.1	System specification and simulation parameters	124
6.2	Control pulses and gate implementation	127
6.3	Computation of gate-error quantifiers	129
6.4	Spectrum of the four-qubit NIGQC obtained with the circuit Hamiltonian	135
6.5	Gate metrics for the elementary gate set	139
6.6	Influence of higher states on gate-error quantifiers obtained with the circuit Hamiltonian	143
6.7	Influence of parameter changes on gate-error quantifiers obtained with the circuit Hamiltonian	146
6.8	Influence of the adiabatic approximation on gate-error quantifiers obtained with the effective Hamiltonian	150
6.9	Summary, Discussion and Conclusions	157
7	Summary, concluding remarks and outlook	162

Appendices	i
A Tables with control pulse parameters for different NIGQC models	i
B Tables with gate-error quantifiers for different NIGQC models	vi
List of figures	ix
List of tables	xii
List of acronyms	xv
List of publications	xvii
Acknowledgments	xviii
Bibliography	xx

Introduction

We begin this thesis with a fundamental distinction, namely we differentiate between a computer model and the corresponding computer hardware. Broadly speaking, a computer model consists of a collection of definitions and mathematical proofs which state that once we are provided with an ideal computing machine which is able to execute a certain number of elementary operations, we can solve a broad range of computational problems by executing sequences of these elementary operations. The computer model provides the knowledge of how to construct the sequence for a given computational problem. A famous example for such a computer model is the Turing machine, see Ref. [Tur37]. The computer hardware is tasked with the problem of realising the elementary operations from the abstract computer model in the real world and providing the necessary infrastructure for the computational process, e.g. long-term and short-term memory storage. Note that the computer model does not discriminate between different technological solutions to the problem. For example, at one time, digital computer hardware was based on vacuum tubes which were later replaced by transistors.

The ideal gate-based quantum computer (IGQC) as discussed in Ref. [NC11] is an algebraic computer model that consists of a collection of definitions and mathematical proofs, see Secs. 2.1-2.5. However, the construction of a fully functioning prototype gate-based quantum computer (PGQC) is an unsolved engineering task which requires a tremendous amount of precision in terms of control over a quantum system, see Ref. [LB13]. Many studies with a focus on PGQCs (sometimes referred to as noisy intermediate-scale quantum devices, see Ref. [Pre18]) investigate a certain type of superconducting hardware, see for example Refs. [Gu+21; Rot+17; Rol+19; Aru+19; Lac+20], so-called transmon systems. In this thesis, we model the time evolution of transmon systems as a unitary real-time process. Therefore, throughout this work we assume that the dynamics of transmon systems is governed by the time-dependent Schrödinger equation (TDSE)

$$i\partial_t |\Psi(t)\rangle = \hat{H}(t) |\Psi(t)\rangle, \quad (1.1)$$

where $\hat{H}(t)$ is the model Hamiltonian that generates the dynamics and $|\Psi(t)\rangle$ is the state vector which, by assumption, completely describes the state of the system. Note that we use $\hbar = 1$ throughout this work. Unfortunately, analytical solutions of the TDSE are rare, especially for cases where the model Hamiltonian $\hat{H}(t)$ is explicitly time dependent. Hence, we use the product formula approach, see Refs. [Rae87; HD90], to construct two algorithms that can solve the TDSE numerically, for two many-particle Hamiltonians. We use these algorithms to construct non-ideal gate-based quantum computer (NIGQC) models which describe the time evolution of superconducting PGQCs used in experiments. In this thesis, we focus on certain aspects of the model which can substantially affect the state of a NIGQC, i.e. the time evolution of transmon systems.

This thesis is structured as follows. In Chap. 2, we review the model of the IGQC and provide a distinction between the IGQC, PGQCs and NIGQCs. Next, in Chap. 3, we derive circuit and effective Hamiltonians for LC resonators, fixed-frequency and flux-tunable transmons. We use these subsystems to define two many-particle Hamiltonians, a circuit Hamiltonian and an associated effective Hamiltonian. The interactions between the different subsystems are modelled as dipole-dipole interactions. The effective Hamiltonian is supposed to mimic the dynamics of the circuit Hamiltonian. Then, in Chap. 4, we review the fundamental problems we face if we numerically solve the TDSE. Furthermore, we discuss the product-formula algorithms we use to mitigate these problems and to solve the TDSE for the Hamiltonians we define in Chap. 3. In Chap. 5, we study how various approximations affect the time evolution of transmon systems modelled by the effective Hamiltonian. Here, we consider scenarios where control pulses are applied to the transmon systems. Additionally, we compare the time evolution of the circuit Hamiltonian with the one of the effective Hamiltonian. Note that in this chapter we only consider the probability amplitudes relevant to the transitions we study. Next, in Chap. 6, we investigate how susceptible gate-error quantifiers are to the approximations which make up the NIGQC model. Gate-error quantifiers are usually used to quantify how successful a computation could be realised with a PGQC and/or with a NIGQC model. Finally, in Chap. 7, we summarise all the findings, provide concluding remarks and give an outlook for future research.

An introduction to gate-based quantum computing

In this chapter we provide a discussion of gate-based quantum computing. In Sec. 2.1, we introduce the mathematical framework we use to develop the model of the IGQC. Section 2.2 contains a discussion of the single-qubit Hilbert space. In Sec. 2.3, we define the multi-qubit Hilbert space, which is constructed by combining single-qubit Hilbert spaces. The corresponding model constitutes the IGQC. Section 2.4 contains a discussion of the simulation methods we use to numerically model the IGQC. In Sec. 2.5, we discuss a handful of algorithms which can be used to solve different computational problems with an IGQC. We discuss algorithms which are of very different nature. In Sec. 2.6 we discuss PGQCs, i.e. experimental setups which aim to realise the IGQC. The subject of Sec. 2.7 are NIGQC models, i.e. models that can be used to describe the time evolution of the PGQCs. Such a model can potentially be used to explain and/or describe some of the basic trends we find in experimental data. The first four sections of this chapter are inspired by Refs. [NC11; Lag19].

2.1 Mathematical framework

Quantum computing, on an IGQC, takes place in a finite-dimensional vector space \mathcal{H} over the field of complex numbers \mathbb{C} . The vector space is also equipped with an inner product $\langle \cdot | \cdot \rangle : \mathcal{H} \times \mathcal{H} \rightarrow \mathbb{C}$. Such a space is usually referred to as finite-dimensional Hilbert space. In the following we omit the adjective finite-dimensional but add a label D to every Hilbert space \mathcal{H}^D , where $D = \dim(\mathcal{H}^D)$ denotes the dimensionality of the vector space.

A computation on an IGQC might be understood as follows. If $|\psi\rangle \in \mathcal{H}^D$: $\langle \psi | \psi \rangle = 1$ is some arbitrary state in some Hilbert space \mathcal{H}^D , we can perform a computation by mapping this state $|\psi\rangle \mapsto |\psi'\rangle$ to another state $|\psi'\rangle \in \mathcal{H}^D$: $\langle \psi' | \psi' \rangle = 1$. We can generalise this idea by defining a function

$\hat{U}: \mathcal{H}^D \rightarrow \mathcal{H}^D$ which preserves the norm for all $|\psi\rangle \in \mathcal{H}^D$. We also require \hat{U} to be linear, which means we should speak of our function as an operator $\hat{U} \in \mathbb{L}(\mathcal{H}^D)$. Here $\mathbb{L}(\mathcal{H}^D)$ denotes the set of all linear functions, i.e. operators, in \mathcal{H}^D . Additionally, operators $\hat{U} \in \mathbb{U}(\mathcal{H}^D)$ which are norm preserving are referred to as unitary operators. Here $\mathbb{U}(\mathcal{H}^D) \subseteq \mathbb{L}(\mathcal{H}^D)$ denotes the set of all unitary operators in \mathcal{H}^D . All unitary operators \hat{U} satisfy the equation

$$\hat{U}^\dagger \hat{U} = \hat{I}, \quad (2.1)$$

where \hat{U}^\dagger denotes the so-called adjoint operator. If $\hat{L} \in \mathbb{L}(\mathcal{H}^D)$, we denote \hat{L}^\dagger as the adjoint operator if for all $|\psi\rangle, |\psi'\rangle \in \mathcal{H}^D$ the relation

$$\langle \psi' | \hat{L}^\dagger | \psi \rangle = \langle \psi | \hat{L} | \psi' \rangle^*, \quad (2.2)$$

holds true. The asterisk $*$ denotes the complex conjugate of the complex number $\langle \psi | \hat{L} | \psi' \rangle$.

The question we have to ask ourselves is: what is the relation between the algebraic formalism described above and quantum physics? The relation becomes clearer once we consider the postulates which often, see for example Refs. [Wei15; NC11; GS11], provide the foundation of the theoretical framework of quantum theory. In this thesis, we order the postulates in the same way as the authors of Ref. [NC11]. The first postulate states that an isolated physical system is completely described by a state vector $|\psi\rangle \in \mathcal{H}$: $\langle \psi | \psi \rangle = 1$ in a Hilbert space \mathcal{H} . Note that the postulate does not specify the dimensionality of the Hilbert space. However, for simplicity and notational convenience we assume that \mathcal{H} is finite-dimensional, i.e. $\mathcal{H} \rightarrow \mathcal{H}^D$. A motivation for this assumption is provided in Sec. 2.2 and 2.3. The second postulate states that the evolution of a closed system is described by a unitary operator \hat{U} which maps an initial state $|\psi\rangle$ to a final state $|\psi'\rangle$ at some point later in time. In Sec. 2.7 we discuss this postulate in more detail. Both postulates together provide us with a conceptual bridge to the computational processes described above.

Another question we find ourselves confronted with is: how can we extract information from the abstract state vectors? In the following, we assume that this extraction of information is governed by the third postulate. Here we assume that the measurement, of some unspecified discrete physical observable, is governed by a set $\{\hat{E}_i\} \subseteq \mathbb{L}(\mathcal{H}^D)$ of operators, where $i \in I \subseteq \mathbb{N}^0$ denotes a discrete variable and/or label which allows us to distinguish between different measurement outcomes, i.e. discrete events. Here I denotes an index set which contains the discrete indices i . In particular, if the system is in some arbitrary state $|\psi\rangle \in \mathcal{H}^D$: $\langle \psi | \psi \rangle = 1$, we assume that the probability

for measuring an event i is given by the expression

$$p(i) = \langle \psi | \hat{E}_i^\dagger \hat{E}_i | \psi \rangle. \quad (2.3)$$

The third postulate also determines the state vector which results from the measurement procedure

$$|\psi'\rangle = \frac{1}{\sqrt{p(i)}} \hat{E}_i |\psi\rangle, \quad (2.4)$$

where $|\psi\rangle$ is the state vector before the measurement and $|\psi'\rangle$ is the state vector after the measurement. Here we assume that event i has taken place. Since the probabilities $p(i)$ have to form a proper distribution, we require

$$\sum_i \hat{E}_i^\dagger \hat{E}_i = \hat{I}, \quad (2.5)$$

as a normalisation condition.

Postulate one and two, refer to closed or isolated systems but often we are interested in physical systems which consist of two or more subsystems. In the following, we assume that the composition of two or more subsystems is governed by the fourth postulate. This postulate states that the Hilbert space $\mathcal{H}^{\bar{D}}$ of the composed system has to obey the algebraic rules of a specific tensor product structure. If we assume that \mathcal{H}^D and $\mathcal{H}^{\bar{D}}$ are different Hilbert spaces in accordance with the first postulate, the fourth postulate states that the state vectors of the composed system are elements of the space

$$\mathcal{H}^{\bar{D}} = \mathcal{H}^D \otimes \mathcal{H}^{\bar{D}}, \quad (2.6)$$

where \otimes is the so-called tensor product symbol. We define the tensor product structure axiomatically, i.e. we assume that the equations

$$(|\psi\rangle_D + |\bar{\psi}\rangle_D) \otimes |\psi\rangle_{\bar{D}} = |\psi\rangle_D \otimes |\psi\rangle_{\bar{D}} + |\bar{\psi}\rangle_D \otimes |\psi\rangle_{\bar{D}}, \quad (2.7a)$$

$$|\psi\rangle_D \otimes (|\psi\rangle_{\bar{D}} + |\bar{\psi}\rangle_{\bar{D}}) = |\psi\rangle_D \otimes |\psi\rangle_{\bar{D}} + |\psi\rangle_D \otimes |\bar{\psi}\rangle_{\bar{D}}, \quad (2.7b)$$

$$c(|\psi\rangle_D \otimes |\psi\rangle_{\bar{D}}) = (c|\psi\rangle_D) \otimes (|\psi\rangle_{\bar{D}}), \quad (2.7c)$$

$$c(|\psi\rangle_D \otimes |\psi\rangle_{\bar{D}}) = (|\psi\rangle_D) \otimes (c|\psi\rangle_{\bar{D}}). \quad (2.7d)$$

are true, by definition, for all $c \in \mathbb{C}$, $|\psi\rangle_D, |\bar{\psi}\rangle_D \in \mathcal{H}^D$ and $|\psi\rangle_{\bar{D}}, |\bar{\psi}\rangle_{\bar{D}} \in \mathcal{H}^{\bar{D}}$. Every Hilbert space $\mathcal{H}^{\bar{D}}$ also needs an inner product $\langle \cdot | \cdot \rangle : \mathcal{H}^{\bar{D}} \times \mathcal{H}^{\bar{D}} \rightarrow \mathbb{C}$. This function can be defined in terms of the inner products which are defined on the spaces \mathcal{H}^D and $\mathcal{H}^{\bar{D}}$. We define the inner product on $\mathcal{H}^{\bar{D}}$ as

$$\langle \bar{\psi}_D \otimes \bar{\psi}_{\bar{D}} | \psi_D \otimes \psi_{\bar{D}} \rangle = \langle \bar{\psi} | \psi \rangle_D \langle \bar{\psi} | \psi \rangle_{\bar{D}}, \quad (2.8)$$

for all $|\psi\rangle_D, |\bar{\psi}\rangle_D \in \mathcal{H}^D$ and $|\psi\rangle_{\bar{D}}, |\bar{\psi}\rangle_{\bar{D}} \in \mathcal{H}^{\bar{D}}$. Additionally, one can show, see Ref. [NC11], that the dimensionality of the tensor product space $\mathcal{H}^{\bar{D}}$ is given by the relation

$$\dim(\mathcal{H}^{\bar{D}}) = \dim(\mathcal{H}^D) \dim(\mathcal{H}^{\bar{D}}). \quad (2.9)$$

The algebraic relations stated above enable us to consider operators, i.e. linear functions, which act on the states of the tensor-product space $\mathcal{H}^{\bar{D}}$. For example, if $\hat{A} \in \mathbb{L}(\mathcal{H}^D)$, we can define the operator $\hat{A} \otimes \hat{I}$ such that

$$\hat{A} \otimes \hat{I} (|\psi\rangle_D \otimes |\psi\rangle_{\bar{D}}) = \left(\hat{A} |\psi\rangle_D \otimes \hat{I} |\psi\rangle_{\bar{D}} \right), \quad (2.10)$$

for all $|\psi\rangle_D \in \mathcal{H}^D$ and $|\psi\rangle_{\bar{D}} \in \mathcal{H}^{\bar{D}}$. Additionally, we can define operators, in a similar manner, for all the other subspaces we model. A useful identity relation which relates such operators, e.g. $\hat{A} \otimes \hat{I}$ and $\hat{B} \otimes \hat{I}$, reads

$$\left(\hat{A} \otimes \hat{I} \right) \left(\hat{I} \otimes \hat{B} \right) = \hat{A} \otimes \hat{B}. \quad (2.11)$$

We also find that all operators of the form $\hat{A} \otimes \hat{I}$ and $\hat{I} \otimes \hat{B}$ commute, i.e. if we change the order of the product on the left-hand side of Eq. (2.11), we see that the result is still given by the right-hand side of Eq. (2.11). Note that in the following sections we also encounter operators which cannot be decomposed in the sense of Eq. (2.11).

In the end, we emphasise that so far we mainly provided definitions and stated results without mathematical prove. We refer the reader to Refs. [NC11; Wat18] for more detailed discussions of the subjects we covered in this section. Furthermore, throughout this section we assumed that the Hilbert spaces are finite dimensional and the measurement events correspond to discrete observables. We can also describe observables which are continuous, in a more general mathematical framework. Discussions of the framework which cover both cases, i.e. the discrete and the continuous case, can be found in Refs. [Wei15; GS11].

2.2 The single-qubit Hilbert space

In the previous section we introduced the algebraic calculus which is being used to model the IGQC. In this section we continue the review of the model by defining the single-qubit Hilbert space. We begin the construction process

with a discussion of the Hilbert space $\mathcal{H}^2 \subseteq \mathbb{C}^2$. This space is two dimensional. Consequently, we can express all state vectors $|\psi\rangle \in \mathcal{H}^2$: $\langle\psi|\psi\rangle = 1$ in this space as

$$|\psi\rangle = c_0 |0\rangle + c_1 |1\rangle, \quad (2.12)$$

where $|0\rangle$ and $|1\rangle$ are the basis states, whose existence is guaranteed by an existence theorem, see Ref. [HK71]. The coefficients have to satisfy the relation $|c_0|^2 + |c_1|^2 = 1$ so that the state vectors are normalised. In the following, we assume that the states $|0\rangle$ and $|1\rangle$ are given by the eigenstates of the Pauli $\hat{\sigma}^{(z)}$ operator. We define this operator in the following manner

$$\hat{\sigma}^{(z)} |z\rangle = \begin{cases} +1 |z\rangle & \text{iff } z = 0 \\ -1 |z\rangle & \text{iff } z = 1. \end{cases} \quad (2.13)$$

In addition, we can define two other Pauli operators. The Pauli $\hat{\sigma}^{(y)}$ operator is defined as

$$\hat{\sigma}^{(y)} |z\rangle = \begin{cases} +i |\neg z\rangle & \text{iff } z = 0 \\ -i |\neg z\rangle & \text{iff } z = 1, \end{cases} \quad (2.14)$$

where $\neg: \{0, 1\} \rightarrow \{0, 1\}$ is the so-called bit-flip operator which maps a zero to a one and vice versa. Similarly, the Pauli $\hat{\sigma}^{(x)}$ operator is defined as

$$\hat{\sigma}^{(x)} |z\rangle = \begin{cases} 1 |\neg z\rangle & \text{iff } z = 0 \\ 1 |z\rangle & \text{iff } z = 1. \end{cases} \quad (2.15)$$

A qubit is, by definition, a state $|\psi\rangle \in \mathcal{H}^2$: $\langle\psi|\psi\rangle = 1$ in the sense of Eq. (2.12), i.e. a normalised state in the Hilbert space \mathcal{H}^2 . It is possible to represent all physically distinctive states in a graphical way. Here we map the different states to the points of a three-dimensional unit sphere in \mathbb{R}^3 . To this end, we recast Eq. (2.12) by making use of the normalisation condition $\langle\psi|\psi\rangle = 1$. This means we introduce the angles and/or variables $\vartheta \in [0, \pi]$ and $\theta \in [0, 2\pi)$ such that the state vector reads

$$|\psi\rangle = \cos\left(\frac{\vartheta}{2}\right) |0\rangle + \sin\left(\frac{\vartheta}{2}\right) e^{i\theta} |1\rangle. \quad (2.16)$$

If we compute the vector

$$\mathbf{r} = (\langle\psi|\hat{\sigma}^{(x)}|\psi\rangle, \langle\psi|\hat{\sigma}^{(y)}|\psi\rangle, \langle\psi|\hat{\sigma}^{(z)}|\psi\rangle), \quad (2.17)$$

we find

$$\mathbf{r} = (\cos(\theta) \sin(\vartheta), \sin(\theta) \sin(\vartheta), \cos(\vartheta)). \quad (2.18)$$

Consequently, we see that the vector $\mathbf{r} \in \mathbb{R}^3$ is given by the well known expression for a cartesian vector of length one in spherical coordinates, where ϑ denotes the azimuthal angle and θ is referred to as the polar angle. In this picture, we might understand an arbitrary computation as an operation which is composed of rotations around the x-, y- and z-axes. We can express these rotations as

$$\hat{R}^{(x)}(\gamma) = e^{-i\frac{\gamma}{2}\hat{\sigma}^{(x)}}, \quad (2.19a)$$

$$\hat{R}^{(y)}(\gamma) = e^{-i\frac{\gamma}{2}\hat{\sigma}^{(y)}}, \quad (2.19b)$$

$$\hat{R}^{(z)}(\gamma) = e^{-i\frac{\gamma}{2}\hat{\sigma}^{(z)}}, \quad (2.19c)$$

where $\gamma \in [0, \pi)$ denotes the rotation angle and the orientation of the rotation is given by the right-hand rule. We find, see Ref. [NC11], that all unitary operators $\hat{U} \in \mathbb{U}(\mathcal{H}^2)$ can be expressed as

$$\hat{U} = e^{ia}R^{(\mathbf{n})}(\gamma), \quad (2.20)$$

where $a \in \mathbb{R}$ and the operator $R^{(\mathbf{n})}(\gamma)$ is given by the expression

$$\hat{R}^{(\mathbf{n})}(\gamma) = e^{-i\frac{\gamma}{2}\mathbf{n} \cdot \hat{\boldsymbol{\sigma}}}. \quad (2.21)$$

Here the vector $\mathbf{n} \in \mathbb{R}^3$ denotes a unit vector $\mathbf{n} = (n^{(x)}, n^{(y)}, n^{(z)})$ and $\hat{\boldsymbol{\sigma}} = (\hat{\sigma}^{(x)}, \hat{\sigma}^{(y)}, \hat{\sigma}^{(z)})$ is defined as a vector of Pauli operators. The operation in Eq. (2.20) can be visualised as a rotation around the vector \mathbf{n} . It is also possible to express \hat{U} in terms of rotations around two axes only. For example, we can use the x-y-x representation

$$\hat{U} = e^{ia}\hat{R}^{(x)}(\alpha)\hat{R}^{(y)}(\beta)\hat{R}^{(x)}(\gamma), \quad (2.22)$$

where $\alpha, \beta, \gamma \in \mathbb{R}$. Note that the phase factors e^{ia} have no measurable effect on the state vector.

Such decompositions might be used to implement arbitrary single-qubit operations \hat{U} on PGQC or NIGQC. In later chapters, we use such a decomposition to implement an NIGQC model.

2.3 The multi-qubit Hilbert space

As mentioned before, we can understand a computation on an IGQC as a mapping between two states which are elements of a Hilbert space \mathcal{H}^D . In this section we construct this Hilbert space from smaller single-qubit Hilbert

spaces by invoking the fourth postulate repeatedly, see Sec. 2.1. We define the Hilbert space of an N -qubit IGQC as the tensor-product space

$$\mathcal{H}^{2^N} = \bigotimes_{n=0}^{N-1} \mathcal{H}_n^2, \quad (2.23)$$

where \mathcal{H}_n^2 denotes the two-dimensional Hilbert space we discuss in Sec. 2.2. The index $n \in \{0, \dots, N-1\}$ allows us to express various mathematical objects in terms of entities which are only defined for the single-qubit spaces \mathcal{H}_n^2 , see examples below. We begin with the set

$$\mathcal{C}^N = \{|\mathbf{z}\rangle \in \mathcal{H}^{2^N} \mid \exists! \mathbf{z} \in \mathcal{B}_2^N : |\mathbf{z}\rangle = \bigotimes_{n=0}^{N-1} |z_n\rangle\}, \quad (2.24)$$

where $\mathcal{B}_2^N = \{0, 1\}^N$ denotes a set of N -tuples and $|z_n\rangle \in \mathcal{H}_n^2$ are the Pauli $\hat{\sigma}^{(z)}$ operator eigenstates we discuss in Sec. 2.2. One can show, see Ref. [NC11], that this set constitutes a basis for the space \mathcal{H}^{2^N} . In the following, we denote \mathcal{C}^N as the computational basis. An arbitrary N -qubit state $|\psi\rangle \in \mathcal{H}^{2^N}$ can be expressed as

$$|\psi\rangle = \sum_{\mathbf{z} \in \mathcal{B}_2^N} c_{\mathbf{z}} |\mathbf{z}\rangle, \quad (2.25)$$

where the coefficients of the state vector have to satisfy the relation

$$\sum_{\mathbf{z} \in \mathcal{B}_2^N} |c_{\mathbf{z}}|^2 = 1. \quad (2.26)$$

Equation (2.26) provides us with an intuitive way of looking at the states $|\psi\rangle \in \mathcal{H}^{2^N}$. We might interpret the different states as points on a high-dimensional hypersphere with radius one.

Addressing the different coefficients $c_{\mathbf{z}} \in \mathbb{C}$ by means of the bit strings $\mathbf{z} \in \mathcal{B}_2^N$ is often very convenient for analytical and/or numerical work. For example, we can define an arbitrary single-qubit operator $\hat{\mathcal{O}}_{n'} \in \mathbb{U}(\mathcal{H}^{2^N})$ for an arbitrary operator $\hat{\mathcal{O}} \in \mathbb{U}(\mathcal{H}^2)$ by means of the expression

$$\forall \mathbf{z} \in \mathcal{B}_2^N : \hat{\mathcal{O}}_{n'} |\mathbf{z}\rangle = \bigotimes_{n=0}^{N-1} \hat{\mathcal{O}}^{\delta_{n,n'}} |z_n\rangle, \quad (2.27)$$

where $\delta_{n,n'}$ denotes the Kronecker delta. This operator can also be expressed as

$$\hat{\mathcal{O}}_{n'} = \bigotimes_{n=0}^{N-1} \hat{\mathcal{O}}^{\delta_{n,n'}}, \quad (2.28)$$

see Eq. (2.11). The index n' determines which state $|z_n\rangle$ in Eq. (2.27) is affected by the operator $\hat{\mathcal{O}}$. Therefore, we refer to n' as the target and/or

qubit index. Note that an operator is fully defined once its action on a basis is determined.

In the model of the IGQC the operators $\hat{\mathcal{O}}_{n'}$ are usually referred to as single-qubit gates and not operators. In the following, we follow this convention. One usually differentiates between single-qubit gates and the so-called multi-qubit gates. As the name indicates, multi-qubit gates affect more than one qubit index at a time. We can further distinguish between multi-qubit gates which can be expressed as a composition of single-qubit gates in the sense of Eq. (2.11), e.g. $\hat{\mathcal{O}}_{n'} \otimes \hat{\mathcal{O}}_{\bar{n}}$ and multi-qubit gates which cannot be expressed in terms of a composition of single-qubit gates in the sense of Eq. (2.11). An example for the latter type of gate is given by the controlled gate $C\hat{\mathcal{O}}_{n',\bar{n}}$, where n' is the so-called control index and \bar{n} is again referred to as the target index. We define this gate as

$$\forall \mathbf{z} \in \mathcal{B}_2^N: C\hat{\mathcal{O}}_{n',\bar{n}}|\mathbf{z}\rangle = \hat{\mathcal{O}}_{\bar{n}}^{z_{n'}}|\mathbf{z}\rangle, \quad (2.29)$$

where $z_{n'} \in \{0, 1\}$ is the n' -th bit string component of the vector $|\mathbf{z}\rangle$ on the right-hand side of Eq. (2.29). Consequently, we find that this operator only acts on the basis state $|\mathbf{z}\rangle$ for bit strings where $z_{n'} = 1$.

In Sec. 2.2 we discuss the procedure of decomposing an arbitrary operator $\hat{U} \in \mathbb{U}(\mathcal{H}^2)$ into a composition of rotations around various axes. It is possible to show that something similar can be achieved in the multi-qubit space \mathcal{H}^{2^N} , with the operators $\hat{U} \in \mathbb{U}(\mathcal{H}^{2^N})$. We may express this result as follows: all unitary operators \hat{U} can be approximated, with arbitrary precision $\epsilon > 0$, by means of a composition

$$\hat{T} = \prod_{i=0}^{|I|-1} \hat{\mathcal{O}}_i, \quad (2.30)$$

of a finite sequence $(\hat{\mathcal{O}}_i)_{i \in I \subseteq \mathbb{N}^0}$ of single-qubit and two-qubit gates $\hat{\mathcal{O}} \in \mathbb{U}(\mathcal{H}^{2^N})$. The finite length $|I|$ of the sequence is determined by the precision ϵ we require. Additionally, one can show that this finite sequence $(\hat{\mathcal{O}}_i)_{i \in I \subseteq \mathbb{N}^0}$ can be constructed with only a handful of elementary gates. This set of elementary gates is usually referred to as a universal gate set. Note that there exists more than one such universal gate set. One of these sets, the so-called standard set, reads

$$G = \{C\hat{X}, \hat{H}, \hat{T}, \hat{S}\}, \quad (2.31)$$

where $\hat{X}, \hat{H}, \hat{T}, \hat{S} \in \mathbb{U}(\mathcal{H}^2)$ are defined as

$$\hat{X}|z\rangle = |\neg z\rangle, \quad (2.32a)$$

$$\hat{H}|z\rangle = \frac{1}{\sqrt{2}}((-1)^z|z\rangle + |\neg z\rangle), \quad (2.32b)$$

$$\hat{T}|z\rangle = e^{z\frac{\pi}{4}i}|z\rangle, \quad (2.32c)$$

$$\hat{S}|z\rangle = e^{z\frac{\pi}{2}i}|z\rangle. \quad (2.32d)$$

Note that $C\hat{X}$ is often referred to as CNOT gate and $\hat{X} = \hat{\sigma}^{(x)}$. Here, we simply use the letter X to address the corresponding Pauli operator, it is common practice to do the same with the $\hat{Y} = \hat{\sigma}^{(y)}$ and $\hat{Z} = \hat{\sigma}^{(z)}$ operators. The author would like to stress that the discussion presented in this section is far from exhaustive. The interested reader might consider Refs. [NC11; DN06; Bar+95] for more detailed information regarding this subject.

2.4 Simulation of the ideal gate-based quantum computer

In this section, we consider the subject of numerically modelling an IGQC. To this end, we intend to discuss the question, how can we simulate the execution of multiple computations $\hat{U}_i \in \mathbb{U}(\mathcal{H}^{2^N})$ in a computationally friendly way? Here, $i \in \{0, \dots, I-1\}$ is an index variable and $I \in \mathbb{N}$ denotes the number of gates we intend to implement. There exist two main types of algorithms for this kind of problem, namely the Schrödinger algorithm and the Feynman algorithm, see Refs. [Wil+20a; Lag19]. We discuss both algorithms and begin with the former. However, before we come to the algorithms, we first discuss the subject of computational complexity. Comparing both algorithms in terms of complexity is probably the fastest way to understand the differences between both methods.

Finding the complexity of an algorithm amounts to analytically determining bounds for the memory and the runtime requirements that a code instance is bounded by. Such an analysis is usually based on several assumptions, i.e. we usually make assumptions about the algorithm and/or program parameters. In this work, we are only interested in the worst case running time complexity and the worst case memory requirements. To this end, we make use of so-called asymptotic notation. The symbols $\mathcal{O}^{\text{Space}}(g(n))$ and $\mathcal{O}^{\text{Time}}(f(n))$ denote the space and time complexity, respectively. These symbols can be understood as follows, for some $n_0 \in \mathbb{N}$, we find that the runtime $r(n)$ and the memory requirement $m(n)$ obey the inequalities $0 \leq m(n) \leq c_{\text{Space}}g(n)$ and $0 \leq r(n) \leq c_{\text{Time}}f(n)$, where $c_{\text{Space}}, c_{\text{Time}} \in \mathbb{R}^+$ and $n \geq n_0$. Here $n \in \mathbb{N}$ denotes a discrete variable, the reader may consider n to be the only relevant variable for the algorithm

we investigate, i.e. the fictitious algorithm we talk about while introducing the notation. For an extensive discussion of this subject we refer the reader to Refs. [Cor+09; AB06].

If we use an implementation of the Schrödinger algorithm to simulate the IGQC, we are required to store all state vector coefficients $c_{\mathbf{z}}$, see Eq. (2.25), during the execution of the program. We often use double precision for our computations, in such a case we find that the data structure for the state vector has the size 2^{N+4} Byte, where N denotes the number of qubits we simulate. Note that it is also possible to use encoding schemes which require a little bit less memory, see Ref. [Rae+19]. However, the memory requirements of the Schrödinger algorithm still scale exponentially with the number of qubits. In the following, we assume that the state vector data structure dominates the memory requirements completely, therefore we can write $\mathcal{O}^{\text{Space}}(2^N)$ for the space complexity. The memory requirements of the Schrödinger algorithm are a consequence of the tensor-product structure, see Sec. 2.1, we use to model multi-qubit systems.

We now turn our attention to the time complexity. Here we assume that the different \hat{U}_i are either single-qubit gates or two-qubit gates of the controlled type. Two-qubit or multi-qubit gates which can be expressed as compositions of single-qubit gates are also simulated as a series of single-qubit gates, see Sec. 2.3 for more details. Single-qubit gates require at most updates of two state vector coefficients $c_{\mathbf{z}}$ at the same time. Two-qubit gates of the controlled type on the other hand might require updates of four state vector components $c_{\mathbf{z}}$ at the same time. This means we can partition the state vector data structure into groups of two and four coefficients $c_{\mathbf{z}}$ which we update together, sequentially. Consequently, we have to perform $2^N/2$ and $2^N/4$ updates in total and each update rule comes with a fixed number of floating-point operations. Therefore, we find that the total number of operations scales with 2^N and therefore we write $\mathcal{O}^{\text{Time}}(2^N)$. If we make use of parallel computing, we can reduce the run time for the simulation of a single gate to a certain extent. However, ultimately we cannot compensate for the exponential scaling of the runtime with N , i.e. this would require nearly unlimited ideal hardware resources.

If we use an implementation of the Feynman algorithm to simulate the IGQC, we are only required to store a fraction of all state vector coefficients $c_{\mathbf{z}}$, see Eq. (2.25), during the execution of the program. Additionally, since the computations of the different coefficients $c_{\mathbf{z}}$ are completely independent, we usually compute the coefficients one at a time.

The idea of the Feynman algorithm is to decompose certain two-qubit gates of the controlled type, which are part of the sequence of \hat{U}_i , in such a way that we can compute the different coefficients $c_{\mathbf{z}}$ by simulating a series

of smaller qubit systems with sizes $P = \{P_0, \dots, P_{|P|-1}\} \subseteq \mathbb{N}$. Here $P \subseteq \mathbb{N}$ denotes the set of subsystem sizes and $|P|$ refers to the cardinality of P . The simulations of the smaller qubit systems are performed with the Schrödinger algorithm and since we are in control of the sizes P , we find that we can overcome the memory bottleneck of a pure Schrödinger simulation. In the end, we combine the simulation results of the different subsystems and compute a single coefficient $c_{\mathbf{z}}$. However, we also pay a price for the reduced memory requirements. The amount of computations required to perform the Feynman simulation increases exponentially with the number of decomposed gates $S \in \mathbb{N}$ in the sequence \hat{U}_i .

If we consider the Feynman algorithm discussed in Ref. [Lag19], we find that the space complexity of the algorithm is given by $\mathcal{O}^{\text{Space}}(2^{\max(P)})$, where $\max(P)$ denotes the maximum over all subsystem qubit sizes. The corresponding time complexity reads $\mathcal{O}^{\text{Time}}(2^S |P| 2^{\max(P)})$. A thorough discussion of this algorithm, where the complexity results are derived properly, can be found in Ref. [Lag19], which contains previous academic work by the author. This work is concerned with the development and implementation of a simulator for the IGQC. Here, a Feynman algorithm was used.

For this thesis the simulation code discussed in Ref. [Lag19] was extended so that one has the choice between an implementation of the Schrödinger and the Feynman algorithm. The IGQC simulator is integrated into a larger simulation software framework, see Sec. 4.7.

2.5 Algorithms

So far, we mainly considered computations in terms of abstract mappings $\hat{U} \in \mathbb{U}(\mathcal{H}^{2^N})$ between two states of the Hilbert space \mathcal{H}^{2^N} , for some fixed number of qubits N . In this section, we extend the picture of the computational process. To this end, we discuss various algorithms which at least require the implementation of a mapping \hat{U} as part of a larger computational process. Note that this discussion is non-exhaustive, we simply intend to provide the reader with a better understanding of how the computation in terms of \hat{U} fits into the bigger picture of things. We try to focus on algorithms and/or subjects which are relevant for the results of this thesis. Furthermore, the following discussion is rather general in the sense that we avoid a detailed mathematical analysis of the different algorithms. This lack of detail allows us to cover more ground. We refer the reader to specific references if necessary.

Broadly speaking, we might define, see Ref. [Cor+09], algorithms as sequences of computational rules and/or steps which take well-defined data

structures as inputs in order to produce well-defined output data structures. Furthermore, an algorithm is usually associated with a specific computational problem so that the input data structure represents a problem instance and the output data structure is considered to be a potential solution. With this in mind, we might classify algorithms as follows. An algorithm belongs to the class of correct algorithms if and only if for every input, the algorithm stops and the output is the correct solution to the computational problem, with absolute certainty. Algorithms which belong to the class of incorrect algorithms possibly do not terminate, i.e. the output is simply not guaranteed to be the correct solution for the given problem after a finite number of computational steps. Obviously, we can classify algorithms by means of different criteria. For example, a classification system which takes into account the complexity, see Sec. 2.4, of algorithms is a very useful tool to better understand the relations between different classes of algorithms and/or problems, see Refs. [Cor+09; AB06] for more details. In the following we discuss three different algorithms: the quantum Fourier transformation algorithm, Shor's factoring algorithm and variational hybrid algorithms.

We begin with the quantum Fourier transformation (QFT) algorithm. The input for this algorithm is some state vector $|\psi\rangle \in \mathcal{H}^{2^N}$: $\langle\psi|\psi\rangle = 1$. For the moment we ignore the input state so that we can focus on the action of the QFT on the basis states $|\mathbf{z}\rangle \in \mathcal{C}^N$. The action of the QFT on these states reads

$$\text{QFT} |\mathbf{z}\rangle = \frac{1}{\sqrt{2^N}} \sum_{\mathbf{z}' \in \mathcal{B}_2^N} e^{2\pi i \frac{\mathbf{z}\mathbf{z}'}{2^N}} |\mathbf{z}'\rangle, \quad (2.33)$$

where $z \in \mathbb{N}$ denotes the integer $z = \sum_{n=0}^{N-1} z_n 2^n$. Consequently, the action of the operator $\hat{\text{QFT}}$ on an arbitrary state vector $|\psi\rangle$ can be expressed as

$$\sum_{\mathbf{z} \in \mathcal{B}_2^N} c_{\mathbf{z}} |\mathbf{z}\rangle \mapsto \sum_{\mathbf{z}' \in \mathcal{B}_2^N} d_{\mathbf{z}'} |\mathbf{z}'\rangle,$$

where the state vector coefficients $d_{\mathbf{z}'}$ are given by

$$d_{\mathbf{z}'} = \frac{1}{\sqrt{2^N}} \sum_{\mathbf{z} \in \mathcal{B}_2^N} c_{\mathbf{z}} e^{2\pi i \frac{\mathbf{z}\mathbf{z}'}{2^N}}. \quad (2.34)$$

We can use Eq. (2.34) as the definition for the discrete Fourier transformation, i.e. by definition the discrete Fourier transformation maps the components $c_{\mathbf{z}}$ to the components $d_{\mathbf{z}'}$. While in theory the quantum Fourier transformation algorithm exhibits an advantage in terms of time complexity, over classical Fourier transformation algorithms, we find that extracting the complex coefficients $d_{\mathbf{z}'}$ and preparing the state $|\psi\rangle$ is a major practical problem, see

Ref. [NC11]. In principle, information extraction and state preparation are governed by postulate three, see Sec. 2.1, which means we have no direct access to the complex coefficients $c_{\mathbf{z}}$ and $d_{\mathbf{z}'}$. However, we can use this algorithm as an intermediate computational step as part of another algorithm, namely Shor’s factoring algorithm, see Refs. [Sho94; Sho97]. This is the algorithm we discuss next.

The computational problem Shor’s algorithm aims to solve can be expressed as follows. Assume we are given an integer $M = pq$, where $p, q \in \mathbb{N}$ are prime numbers. The question then is what are the prime factors p, q of the integer M . The algorithm can be split into two parts. The first part can be executed on a classical computer. Here we transform the factoring problem into an order-finding problem. If $m \in \mathbb{N}$: $m < M$ and m and M have no common factors, we say that the order of m modulo M is the smallest integer $r \in \mathbb{N}$ which satisfies $m^r = 1$ modulo M . The second part is meant to be executed on a quantum computer. Here we solve the order-finding problem by applying another algorithm which in turn relies on the QFT algorithm. The crux of Shor’s algorithm is that the factoring problem can potentially be solved in polynomial time, i.e. with a time complexity which is given by a polynomial. To the best knowledge of the author, no classical state-of-the-art implementable algorithm can solve this problem in polynomial time. This is why many cryptography systems are built on the assumption that, in the foreseeable future, no algorithm can be implemented which solves this problem in polynomial time. Shor’s algorithm provides an example where many algorithms are concatenated in the sense that the computational rules are algorithms themselves. Therefore, Shor’s algorithm provides us with a natural example of a problem where the original computational problem gets transformed, subproblems are solved with various algorithms and in the end the processes are aligned in the correct order. This is probably the best way to think about computing in general. We are allowed to creatively combine all resources at our disposal, the only important criterion is that the end result is the correct one.

The next algorithm or rather the next class of algorithms we discuss combines the generation of a specific state vector $|\psi(\mathbf{x})\rangle \in \mathcal{H}^{2^N}$: $\langle\psi|\psi\rangle = 1$ with classical optimisation algorithms. Here $\mathbf{x} \in \mathbb{R}^{D'}$ denotes a parameter vector and for large N we assume that D' is substantially smaller than $D = 2^N$. The class of variational hybrid algorithms may be characterised as follows. Assume we are given the following inputs: an initial state $|\mathbf{0}\rangle \in \mathcal{H}^{2^N}$, a set of gates which form the unitary operator $\hat{U}(\mathbf{x}) \in \mathbb{U}(\mathcal{H}^{2^N})$ and a Hermitian operator $\hat{H} \in \mathbb{H}(\mathcal{H}^{2^N})$. Here $\mathbb{H}(\mathcal{H}^{2^N}) \subseteq \mathbb{L}(\mathcal{H}^{2^N})$ denotes the set of all Hermitian operators, i.e. $\hat{H} \in \mathbb{H}(\mathcal{H}^{2^N})$ implies $\hat{H} = \hat{H}^\dagger$. We can then use $\hat{U}(\mathbf{x})$

to compute the parameterised state vector

$$|\psi(\mathbf{x})\rangle = \hat{U}(\mathbf{x}) |\mathbf{0}\rangle, \quad (2.35)$$

on an IQQC. Then we use this state to determine the function value

$$C(\mathbf{x}) = \langle \psi(\mathbf{x}) | \hat{H} | \psi(\mathbf{x}) \rangle, \quad (2.36)$$

for a fixed \mathbf{x} . The function $C: \mathbb{R}^{D'} \rightarrow \mathbb{R}$ is the so-called cost function. Note that the cost-function values are always real, this is a consequence of the fact that \hat{H} is Hermitian. We might think about this function as some kind of subroutine which runs on an IQQC. A classical optimisation algorithm can call this subroutine in order to minimise the cost function C .

The procedure described above is motivated by the variational principle. This principle states that

$$C(\mathbf{x}) \geq E_0, \quad (2.37)$$

is true for all $\mathbf{x} \in \mathbb{R}^{D'}$, see Ref. [Wei15]. Here E_0 denotes the minimum of the eigenvalues of \hat{H} . Since \hat{H} is defined on a finite-dimensional Hilbert space, we find that the existence of the minimum is guaranteed. Note that the states $|\psi(\mathbf{x})\rangle$ are normalised. The variational principle guarantees that we cannot surpass E_0 while minimising $C(\mathbf{x})$. Therefore, we might understand this class of algorithms as a method to find the ground state, i.e. the eigenstate which corresponds to the eigenvalue E_0 of a Hermitian operator \hat{H} . For a general $\hat{U}(\mathbf{x})$ there is no guarantee that this method converges to E_0 . Therefore, we denote this method as a heuristic method to find the ground state of a specific Hermitian operator \hat{H} .

We can define a generic operator \hat{H} in terms of the Pauli operators we discuss in Sec. 2.2. The full operator reads

$$\hat{H} = \sum_{m=0}^{M-1} J_m \bigotimes_{n=0}^{N-1} \hat{O}_{m,n}, \quad (2.38)$$

where $\hat{O}_{m,n} \in \{\hat{I}, \sigma^{\hat{x}}, \sigma^{\hat{y}}, \sigma^{\hat{z}}\}$. This operator is sufficiently general to cover a variety of interesting cases, e.g. the Ising model or the Heisenberg model. In principle, we can choose $\hat{U}(\mathbf{x})$ at random. However, we prefer to generate parameterised states $|\psi(\mathbf{x})\rangle$ which, for some reason, are likely to converge to the ground state for a given \hat{H} . It is sometimes the case that different states $|\psi(\mathbf{x})\rangle$ are associated with different algorithm names, see for example Ref. [FGG14] for the quantum approximate optimisation algorithm.

2.6 Prototype gate-based quantum computers

With this section we have reached a crucial juncture in this thesis. In the previous sections we talked about a static computer model, i.e. all changes which affect the state vector $|\psi\rangle \in \mathcal{H}^{2^N}$ are in principle instantaneous. The IGQC is then simply an algebraic mathematical model and we connect this model to the physical theory and/or the experiment by means of the postulates we discussed in Sec. 2.1. In principle, we could formulate this computer model without considering the physical postulates.

In this section, we discuss what we call are prototype gate-based quantum computers (PGQCs). There exist a variety of different PGQCs which aim to realise an IGQC, see Ref. [DiV13]. We focus our discussion on a specific type of PGQC: low-temperature superconducting integrated circuits and the experimental setups which operate these circuits. References [Rot+17; Kri+20; Aru+19; Ben+20] discuss various experiments which make use of three different types of circuit architectures. Figure 2.1 shows a photograph of a superconducting integrated circuit, i.e. a chip, which is supposed to form a five-qubit system. The five large circles in the middle of the photograph are meant to manifest the qubits. The four squares which connect the different qubits, via straight lines which form the letter X, are the so-called coupler elements. These elements provide an indirect coupling between the qubits. We can also see wave-like structures which are connected to the five qubits. These structures are used to read out the qubit state. The remaining structures on the chip can be used to apply control pulses in terms of currents and voltages, we denote these as feed lines. Therefore, we find that we might have to consider more system components than just the qubits themselves. In this case, we should probably at least take into account the couplers, read-out elements and feed lines.

Experiments, see Ref. [Nag19], show that the qubits on chips like the one in Fig. 2.1 are different from the qubits we define for the IGQC model. For example, if we initialise the qubit in state $|1\rangle$ at time t_0 and wait some time $t > t_0$, we usually find $p_1(t) < 1$ for various times t . Here $p_1(t)$ denotes the probability for finding the qubit in the first-excited state. Additionally, if we consider the limit $t \gg t_0$, we often find that the qubit mainly resides in the ground state $|0\rangle$. The relaxation process can be described by an exponential function of the form $p_1(t) = a + be^{-t/T_1}$, where a is a real-valued constant which describes the population that resides in the state $|1\rangle$ after $t \rightarrow \infty$. Furthermore, in this model the real-valued constants b and T_1 describe the decay into the state $|0\rangle$. We denote the time T_1 as the relaxation

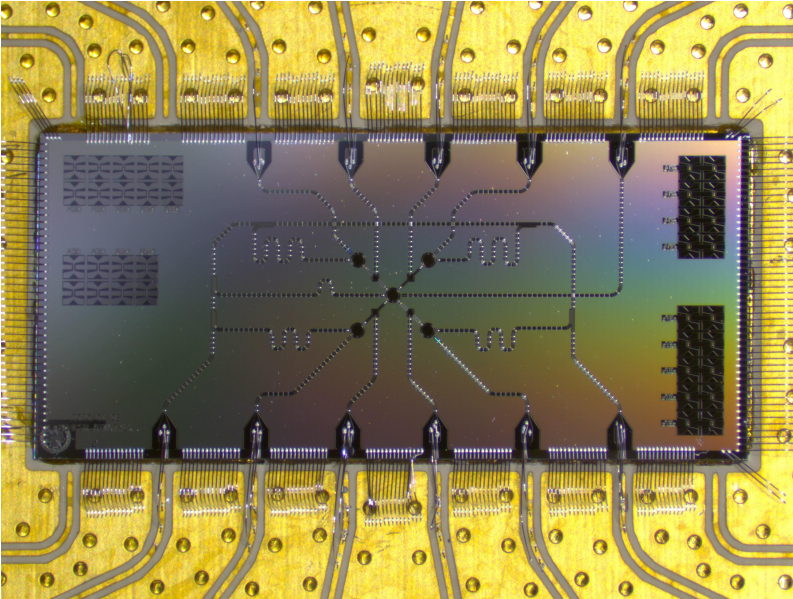


Figure 2.1: Photograph of a chip which contains structures such that five superconducting qubits might be realised in an experiment. The five dominant, i.e. larger, circles in the centre of the photograph are supposed to be the qubits. The four small squares which connect the different qubits are the so-called coupler elements. The coupler elements are supposed to provide an indirect coupling between the different qubits. We can also see wave-like structures which are connected to all five qubits. These are used to read out the qubit state. The remaining structures are used to apply control pulses in terms of currents and voltages. With permission of Jonas Bylander from Chalmers University of Technology.

time. State-of-the-art T_1 times are of the order of microseconds. Experimental qubits are usually characterised by more than just the relaxation law parameters. One of the most important parameters is the real-valued qubit frequency $\omega^{(01)}$. This parameter is determined by means of a spectroscopy experiment. Broadly speaking, we use one of the feed lines mentioned in our discussion of Fig. 2.1 to send a harmonic signal of frequency $\omega^{(D)}$ to the qubit. Then, the qubit frequency $\omega^{(01)}$ is the drive frequency $\omega^{(D)}$ which optimally excites the qubit from the ground $|0\rangle$ to the excited state $|1\rangle$ and vice versa. Another parameter of interest is the so-called dephasing time T_2 . However, we do not intend to discuss this parameter here. Reference [Nag19] provides an introduction to the topic of qubit characterisation and measurement, including the relaxation time T_1 , the qubit frequency $\omega^{(01)}$ and the dephasing time T_2 .

Superconducting integrated circuits are usually part of a much larger experimental setup. Figure 2.2 shows two photographs of an experimental setup to operate a chip at around 10 mK. Both images show the same setup from different perspectives. The left photograph in Fig. 2.2(a) shows different electronic devices, see the boxes in the rolling shelf carts. These devices are connected to a closed cryostat, see the white cylinder with the name tag Bluefors. The right photograph in Fig. 2.2(b) shows the open cryostat from the bottom. One can see different horizontal layers at the top of the image and a silver cylinder at the bottom of the image. The different layers correspond to different temperature zones within the setup, assuming the device is in operation. The silver cylinder is used as a magnetic shield. Note that such shields are usually made of superconducting materials. A much more detailed discussion of cryogenic setups is provided by Ref. [Kri+19].

The discussion so far should make one thing clear, namely that a complete, detailed, mathematical description of such a system is most likely an impossible task. The vast number of interdependent variables we have to consider is simply too great. For example, we might consider the variables material, see Ref. [Mur21; MM14], control signals, see Ref. [Rol+19; Wer+21], temperatures at various points in the experimental setup, see Ref. [Kri+19], temporal stability of experimental parameters like T_1 , $\omega^{(01)}$ and T_2 , see Ref. [Bur+19] and maybe even cosmic radiation, see Ref. [McE+22]. Consequently, it seems to be impossible that we can formulate a model which describes all these aspects in one combined mathematical framework. However, we might be able to describe and/or explain some trends we see in the experimental data and gain some insight into the dynamics of certain system variables of importance.

Additionally, so far we have ignored another important issue, namely how we extract the qubit state data from the experimental setup. If we perform

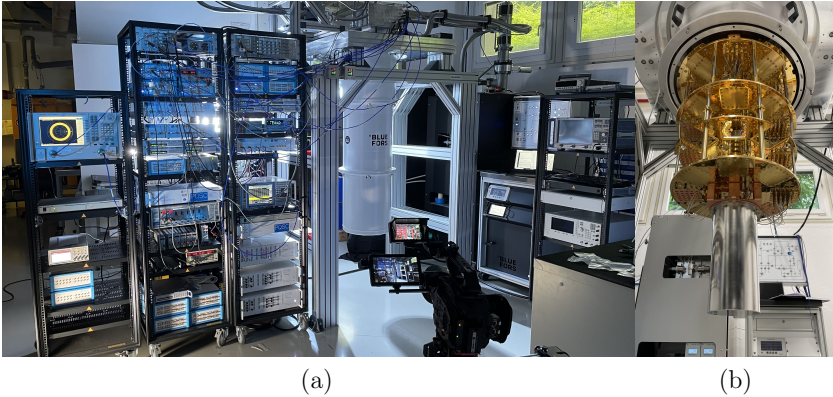


Figure 2.2: Photographs of an experimental setup to operate a superconducting integrated circuit at around 10 mK. Note that Fig. 2.1 shows a photograph of such a superconducting integrated circuit. The images show the same setup from different perspectives. The left photograph (a) shows a variety of control electronics which is connected to a closed cryostat, see white cylinder with the name tag Bluefors. The right photograph (b) shows the open cryostat from the bottom. We can see that the cryostat consists of different layers. Each layer usually corresponds to its own temperature zone. Additionally, at the bottom we can see a silver cylinder. This part of the setup is used to shield the chip from electromagnetic fields which might disturb the chip operation. See Ref. [Kri+19] for more detailed information about the setup. With permission of Pavel Bushev from Forschungszentrum Jülich.

measurements with a PGQC, we find that we have to map the measurement data to discrete events, see Ref. [Nag19]. These discrete events correspond to measuring the qubit states $|0\rangle$ and $|1\rangle$ and sometimes even other higher excited states like $|2\rangle$, see Ref. [Kri+22]. Note that these states are not part of the IGQC model. This is a consequence of the fact that the measurement protocols, see Refs. [Nag19], do not naturally lead to perfectly differentiable discrete event data. This might be explained as follows, we repeat every experiment several times in the hope that we generate the exact same state vector or system state every time we perform the experiment. This is the state we intend to sample from, in the sense of the IGQC model $|\psi\rangle \in \mathcal{H}^{2^N}$. However, our discussion of the experimental setup suggests that this might not be the case. It is probably more realistic to assume that we sample from state vectors which are similar but not equal, due to all the environmental factors which play a role in the measurement process. Assuming, that this qualitative assessment is correct, we might even expect that the experimental data scatters to a certain extent and we have to employ data processing protocols which generate discrete event data. Note that not all experiments, see for example Ref. [Kri+22] and Refs. [And+20; Hei+18], use the same data processing protocols, this makes a comparison of experiments and/or devices more difficult. Furthermore, the coupler states are usually not measured or only considered during the calibration of the control pulses, see Ref. [Ben+20]. Consequently, we have to take into account the specific measurement protocol used if we compare the PGQC data to some mathematical model.

2.7 Non-ideal gate-based quantum computers

In Secs. 2.1-2.5 we discuss the IGQC model and in Sec. 2.6 we discuss superconducting PGQC. In this section we discuss another type of model: non-ideal gate-based quantum computers (NIGQCs). These models might be used to describe certain aspects of the PGQC. As discussed in Sec. 2.6, superconducting PGQC are very complex devices and we should not be ignorant and attempt to describe the full device with one model. A NIGQC model should be used to study certain aspects of the PGQC in question.

We begin our discussion of NIGQC with the second postulate, i.e. the postulate we discussed in Sec. 2.1. From here on, we assume that the time evolution of a closed physical system is governed by the time-dependent

Schrödinger equation (TDSE)

$$i\partial_t |\Psi(t)\rangle = \hat{H}(t) |\Psi(t)\rangle, \quad (2.39)$$

where $\hat{H}(t) = \hat{H}^\dagger(t)$ for all $t \in \mathbb{R}^+$, i.e. $\hat{H}(t)$ is a Hermitian operator for every point in time. This operator generates the dynamics of the system and in the following we denote the operator $\hat{H}(t)$ as a Hamiltonian and the model, i.e. the Hamiltonian in combination with the TDSE, as a Hamiltonian model. The formal solution of the TDSE can be expressed as

$$\hat{\mathcal{U}}(t, t_0) = \mathcal{T} \exp \left(-i \int_{t_0}^t \hat{H}(t') dt' \right), \quad (2.40)$$

where \mathcal{T} denotes the time-ordering symbol. Since $\hat{H}(t)$ is Hermitian as defined above, we find that $\hat{\mathcal{U}}$ is unitary. Therefore, the assumption that the system dynamics is governed by a Hamiltonian model provides us with a way to determine an explicit operator $\hat{\mathcal{U}}$, once $\hat{H}(t)$ is fixed. Here we assume that we can determine $\hat{\mathcal{U}}$ analytically and/or numerically.

The science of physics, see for example Refs. [Wei15; Dir25; Bla+21], might provide us with systematic rules for determining operators $\hat{H}(t)$ whose predictions agree, to some extent, with most experiments performed in laboratories all over the world. However, these rules are not necessarily perfect and/or the corresponding models are so complex that we cannot solve the TDSE for the $\hat{H}(t)$ we derive. Therefore, we might consider simply guessing an operator $\hat{H}(t)$ or rather its form based on experience. Alternatively, we can also apply approximations to Hamiltonian models, i.e. to the models which are too complex. The problem with this approach is that usually we do not know to what extent the results of the original and the approximate model deviate. This may lead to some kind of false confidence in one or both models.

The physical systems and/or experiments we intend to model are usually characterised by a set of parameters $\mathbf{R} \in \mathbb{R}^{D^*}$, where $D^* \in \mathbb{N}$, such that $\hat{H}(t)$ actually reads $\hat{H}(\mathbf{R}, t)$. Here \mathbf{R} contains the parameters which characterise the static properties of a device as well as the dynamic features like the explicit form of $\hat{H}(t)$ at time $t > t_0$. Consequently, the solution of the TDSE should be expressed as $|\Psi(\mathbf{R}, t)\rangle$. It might be the case that the solutions $|\Psi(\mathbf{R}, t)\rangle$ and $|\Psi(\mathbf{R}', t)\rangle$ for two close by parameter vectors \mathbf{R} and \mathbf{R}' deviate quite strongly, at least after some time $t > t_0$. Therefore, we make this distinction here. We can make the scenario even more complex, by assuming that $\mathbf{R} \rightarrow \mathbf{R}(t)$ the parameter vectors are time dependent too. Experiments show that several quantities which are usually assumed to be constant actually fluctuate over time, see for example Ref. [Bur+19]. Whether or not such

a time dependence is relevant for an accurate description of the experiment and/or device has to be verified by considering the experimental data, i.e. we have to study how the experimental data changes on different time scales.

This way of modelling computations $\hat{\mathcal{U}}$ in a finite-dimensional Hilbert space is quite different from the one we discuss in Secs. 2.1-2.5. First, if we use a Hamiltonian model to implement $\hat{\mathcal{U}}$, we model a dynamic process which transforms the state vector continuously in case of an analytical model or as a real-time process in case of a numerical model. In the IGQC model the variable time does not play an explicit role, i.e. here changes to the state vector occur instantaneously. Second, the qubit states we use for the Hamiltonian model are not necessarily decoupled from the extended state space, i.e. the model we use is not necessarily a natural two-level model. If this is the case, we have to define a projection \hat{P} to the computational subspace which is formed by the different qubit states. Note that we ourselves define what is considered to be a qubit state in the NIGQC model. We can then determine the operator $\hat{M} = \hat{P}\hat{\mathcal{U}}\hat{P}$ and by definition, consider the operator \hat{M} to be our computational map, which we might compare to computational maps $\hat{U} \in \mathbb{U}(\mathcal{H}^{2^N})$ which are defined for the IGQC model. Third, we usually do not know how to choose the system parameters \mathbf{R} such that the projection \hat{M} perfectly fits a desired unitary operator \hat{U} .

In the following chapters we intend to make these issues more concrete. First we have to define a Hamiltonian which generates the dynamics of the system, see Chap. 3. Then we can build a numerical NIGQC model, see Chap. 4 and determine the parameters \mathbf{R} which realise the computational map \hat{M} , see Chap. 6. Additionally, we intend to study how the NIGQC models affect the implementation of simple algorithms, in comparison to the IGQC model, see Chap. 6. Note that in principle we can relax the assumption that the dynamics is generated by a Hamiltonian model and consider other models, for example a Lindblad master equation model, see Refs. [Lin76; GKS76].

2.8 Summary and conclusions

In Secs. 2.1-2.5 we developed the model of the ideal gate-based quantum computer (IGQC) and discussed how to simulate the model. In this model, we describe the state of the IGQC by means of a time-independent state vector in a high-dimensional Hilbert space.

In Sec. 2.6 we discussed a superconducting prototype gate-based quantum computer (PGQC) and some problems which arise once we attempt to transfer the idea of a time-independent IGQC into the real world which is in-

herently dynamic. For example, we reported on experiments which show that even subtle issues like cosmic radiation affect state-of-the-art superconducting PGQCs. Note that there are more issues which need to be considered.

In Sec. 2.7 we introduced non-ideal gate-based quantum computer (NIGQC) models. With these models we attempt to describe different aspects of PGQCs. The state of a NIGQC is described by a time-dependent state vector or a time-dependent density matrix depending on which equation generates the dynamics of the system. In the former case we use the TDSE to generate a unitary time evolution. Similarly, in the latter case we use the Lindblad master equation to generate a non-unitary time evolution. Furthermore, we discussed some general issues regarding NIGQC models.

In conclusion, if we speak of gate-based quantum computers, we should at least clearly separate between the three different terms IGQCs, NIGQCs and PGQCs introduced in this chapter. Obviously, they are not one and the same.

Hamiltonian models for lumped-element circuits

In this chapter, we introduce the Hamiltonians we use to model NIGQCs, see Sec. 2.7. In Sec. 3.1, we discuss the lumped-element circuit model we use to describe different electromagnetic systems. Next, in Sec. 3.2, we review how one can derive a Hamiltonian for a given lumped-element circuit. In Sec. 3.3, we apply this procedure to a so-called LC circuit without any power sources. This circuit is the electromagnetic equivalent of the harmonic oscillator. Furthermore, in this section we also discuss a bath model which can be formulated in terms of the LC circuit Hamiltonian. Then, in Sec. 3.4, we derive and discuss two circuit Hamiltonians, one for the fixed-frequency and one for the flux-tunable transmon. Later on, we use the bare eigenstates of both systems to model qubits in a NIGQC model. Furthermore, here we also discuss effective Hamiltonians which might be used to describe fixed-frequency and flux-tunable transmons, instead of the circuit Hamiltonians. In Sec. 3.5, we define a many-particle circuit Hamiltonian which can be used to model a complete NIGQC. Similarly, in Sec. 3.6, we define an associated many-particle effective Hamiltonian which can be used to achieve the same goal. Note that throughout this chapter we use $\hbar = 1$.

3.1 The lumped-element approximation

In Sec. 2.6, we discussed superconducting PGQCs and we found that a complete description of such devices might be impossible, i.e. there are simply too many variables which are relevant to the problem at hand. Consequently, a description in terms of Hamiltonian or NIGQC models, see Sec. 2.7, forces us to make some simplifying approximations. In this section, we discuss the so-called lumped-element approximation. Ultimately, we intend to describe multi-qubit systems as electromagnetic systems, in a lumped-element model formulation.

Broadly speaking, we may describe the lumped-element approximation as follows. A circuit consists of different conductive (superconductive) electromagnetic structures and strip conductors (superconductors). The function of the strip conductors is to connect the different electromagnetic structures. In our model, we lump the more or less complex electromagnetic structures into fictitious two-terminal elements and the strip conductors become ideal wires. The two-terminal elements and ideal wires then form a network, i.e. the different terminals are the nodes n of the network, we connect the nodes via ideal wires and the two-terminal elements become branches b . Additionally, we assume that all elements are fully characterised by a constitutive relation. The corresponding constitutive relations connect the currents $I_b(t)$ following through the different two-terminal elements (branches) with the voltage differences $V_b(t) = V_n(t) - V_{n'}(t)$ between the two nodes n and n' . Here $V_n(t)$ denotes the voltage at node n . In our network model, currents and voltages become directed edges of the network, i.e. a change in direction corresponds to a sign flip in terms of currents and voltages. In the following, we assume that the constitutive relations for linear capacitive and inductive elements are given by the equations

$$I_b(t) = C\dot{V}_b(t) \quad (3.1)$$

and

$$\dot{I}_b(t) = \frac{1}{L}V_b(t) \quad (3.2)$$

respectively. Here C denotes the capacitance and L is the so-called inductance. Furthermore, we assume that the so-called Josephson junctions are described by a constitutive relation of the form

$$I_b(t) = I_c \sin\left(\frac{2\pi}{\Phi_0}\Phi_b(t)\right), \quad (3.3)$$

where the constant I_c denotes the critical current and $\Phi_0 = h/2e$ is the so-called magnetic flux quantum, which is of the order of 10^{-15} Weber. With Eq. (3.3) we also introduced the branch flux variable

$$\Phi_b(t) = \int_{-\infty}^t V_b(t')dt', \quad (3.4)$$

where we assume that $V_b(t) \rightarrow 0$ for $t \rightarrow -\infty$. Josephson junctions can be understood as some kind of non-linear inductors. The capacitance C , the inductance L and the critical current I_c are the parameters we use to characterise the different elements in our model.

So far, we can only describe individual elements in terms of currents and voltages. However, we are interested in the case where the different elements interact with one another. To this end, we assume that the different elements or rather the currents and voltages of the different elements are connected by Kirchhoff's laws. The first law or the current law reads

$$\sum_{b \in \mathcal{N}_i} I_b(t) = 0, \quad (3.5)$$

for all nodes $i \in \mathbb{N}^0$ in the network. Here $\mathcal{N}_i \subseteq \mathbb{N}^0$ denotes an index set and this set contains all the branches b which are connected to the node i . The second law or Kirchhoff's voltage law states that

$$\sum_{b \in \mathcal{L}_j} V_b(t) = \sum_{e \in \mathcal{E}_j} \mathcal{EMF}_e(t), \quad (3.6)$$

for all loops j in the network. Here $\mathcal{L}_j \subseteq \mathbb{N}^0$ denotes the index set of all two-terminal element branches b which are part of the loop j . The functions $\mathcal{EMF}_e(t)$ denote so-called electromotive forces (EMFs), we consider EMFs to be model functions which allow us to describe idealised power sources. The index set $\mathcal{E}_j \subseteq \mathbb{N}^0$ contains all EMF branches e which are part of the loop j .

Up to this point, we formulated our model in terms of assumptions, without actually having considered the physical scenario in question. There are two reasons for this approach. First, it is beyond the scope of this thesis to properly discuss all the relevant issues. Second, there are some questions which cannot be resolved completely. The remainder of this section is devoted to the issues just mentioned.

With our model we a priori assume that the two quantities, voltage $V_b(t)$ and current $I_b(t)$, are well defined for all branches b in the network. From the point of view of Maxwell's theory of electromagnetism, see Refs. [Max10; Zan13; Wen+58], we define currents as

$$I(t) = \int_S \mathbf{j}(\mathbf{r}, t) d\mathbf{S}, \quad (3.7)$$

where $\mathbf{j}: \mathbb{R}^4 \rightarrow \mathbb{R}^3$ denotes the so-called current density and S is a finite surface in three-dimensional space \mathbb{R}^3 . This leads to the question, how can we determine the density $\mathbf{j}(\mathbf{r}, t)$ and the obvious answer would be to solve Maxwell's equations in combination with the Coulomb-Lorentz force for the physical system in question. However, it is unlikely that this approach yields actual results in terms of an analytical function $\mathbf{j}(\mathbf{r}, t)$, i.e. exact solutions

to this mathematical problem are the exception. Furthermore, once we consider $I_b(t)$, we find that there is no reference to the finite surface S which determines the right-hand side of Eq. (3.7). Therefore, we might assume that $I_b(t)$ is constant throughout the two-terminal element we consider. A more detailed discussion of this assumption is provided by Ref. [Wen+58, Chapter II].

The voltage $V(t)$ is a quantity which is more difficult to grasp. Voltages are usually introduced in the context of electrostatics, where the electric field $\mathbf{E}(\mathbf{r}, t): \mathbb{R}^4 \rightarrow \mathbb{R}^3$ and the magnetic field $\mathbf{B}(\mathbf{r}, t): \mathbb{R}^4 \rightarrow \mathbb{R}^3$ decouple. This case occurs once both fields are constant for all times t . Here we find that the electric field $\mathbf{E}(\mathbf{r}) = -\nabla\varphi(\mathbf{r})$ can be expressed in terms of a scalar potential $\varphi(\mathbf{r}): \mathbb{R}^3 \rightarrow \mathbb{R}$ alone. Therefore, in electrostatics we can define voltages as differences

$$V = \int_{\mathbf{r}_n}^{\mathbf{r}_{n'}} \mathbf{E}(\mathbf{r}) d\mathbf{l} = \varphi(\mathbf{r}_n) - \varphi(\mathbf{r}_{n'}), \quad (3.8)$$

in terms of the scalar potential $\varphi(\mathbf{r})$. Here we make use of the fact that the line integral in Eq. (3.8) is path independent, see Ref. [Zan13]. This definition does not naturally translate to all non-static cases, i.e. in some cases it is not possible to express the electric field with such ease. However, for quasi-static scenarios, where changes in the state of the system occur instantaneously, we might still be able to add some meaning to the right-hand side of Eq. (3.8), see Ref. [Wen+58, Chapter II].

Some authors, see Ref. [FCA60, Section 6.10], promote the view that voltages are not defined outside of electrostatics, i.e. if the path independence in the computation of the voltage is not given. If this is the case, we have to ask ourselves whether or not the constitutive relations and Kirchhoff's laws have a well-defined physical meaning in the context of Maxwell's theory of electromagnetism. The author of Ref. [McD12] advocates the use of so-called retarded electric scalar potentials in the Lorenz gauge. Another author, see Ref. [Bal12, Section 1.4], provides a detailed discussion of the relation between Maxwell's field equations and the circuit equations introduced in this section, see Eqs. (3.1)–(3.6). Additionally, Ref. [Bal12, Section 1.4] also provides a detailed discussion of the relation between field quantities and the corresponding circuit quantities, e.g. the electric field intensity $\mathbf{E}(\mathbf{r}, t)$ and the voltage $V(t)$.

In conclusion, to the best knowledge of the author, there exists no standard definition of the quantity voltage $V(t)$ in time-dependent electromagnetism. Therefore, we might take the stand, in a qualitative manner, that time-dependent voltages can be associated with quasi-static potential differences $\varphi(\mathbf{r}_n, \mathbf{t}) - \varphi(\mathbf{r}_{n'}, \mathbf{t})$ in the sense of Eq. (3.8). The condition for the quasi-static or quasi-stationary state is often, see Refs. [Wen+58; Zan13],

expressed as

$$\omega s \ll c, \quad (3.9)$$

where $\omega \in \mathbb{R}$ denotes the frequency of the oscillating currents and/or voltages in the circuit, $s \in \mathbb{R}$ refers to the distance which characterises the circuit and $c \in \mathbb{R}$ is the speed of light, which is of the order 10^8 meter per second.

In this thesis, we prefer the axiomatic view presented at the beginning of this section. To this end, we treat EMFs as model functions, i.e. EMFs allow us to describe idealised power sources. Furthermore, the constitutive relations in Eqs. (3.1)–(3.3) allow us to include energy conserving two-terminal elements. In the end, we can construct arbitrary networks and compare the predictions made by the circuit theory with the pointer readings, on different meters, we find in the laboratory, this view is motivated by Ref. [Wen+58].

3.2 The circuit quantisation formalism

In this section, we discuss how one can derive so-called circuit Hamiltonians. Circuit Hamiltonians, sometimes in combination with the TDSE, provide us with quantum theoretical models, which allow us to make predictions for electromagnetic circuits in the quantum regime, i.e. in the regime where quantum effects dominate the physical scenario in question. In Sec. 3.5, we use different circuit Hamiltonians to define a many-particle Hamiltonian, which might describe some aspects of superconducting PGQCs, see Sec. 2.6, we are interested in. The derivation procedure for circuit Hamiltonians is usually referred to as circuit quantisation.

The first step in the quantisation of an electromagnetic circuit corresponds to choosing a set of independent variables such that all voltages $V_b(t)$ and currents $I_b(t)$ of the circuit can be expressed in terms of these variables and the corresponding derivatives. Furthermore, we also require that the Euler-Lagrange equations, which are associated with a certain Lagrangian function $\mathcal{L}: \mathbb{R}^N \rightarrow \mathbb{R}$, enforce Kirchhoff's current and voltage law.

To the best knowledge of the author, there exist two well-studied methods to systematically quantise a non-dissipative electromagnetic circuit. The method of nodes and the method of loops, see Refs. [BKD04; YD84; Dev97; VD17; UH16]. Broadly speaking, both methods deviate in their choice of variables. The method of nodes makes use of the fact that if we use the generalised branch fluxes

$$\Phi_b(t) = \int_{-\infty}^t V_b(t') dt', \quad (3.10)$$

as our dynamic variables, where $V_b(t) \rightarrow 0$ for $t \rightarrow -\infty$, we find that Kirchhoff's voltage law can be satisfied without further ado by expressing $\Phi_b(t)$ in terms of independent generalised node fluxes $\Phi_n(t)$. Similarly, the method of loops makes use of the fact that if we use the generalised branch charges

$$Q_b(t) = \int_{-\infty}^t I_b(t') dt', \quad (3.11)$$

as our dynamic variables, where $I_b(t) \rightarrow 0$ for $t \rightarrow -\infty$, Kirchhoff's current law can be satisfied by choosing another set of independent variables, see Ref. [UH16]. Which method to choose depends on the problem at hand. For an extensive discussion of this issue see Ref. [Ulr17]. In the end, we have to construct the Lagrangian \mathcal{L} such that the remaining second law is satisfied too. In this thesis, we are interested in circuits, which contain linear capacitors, linear inductors, Josephson junctions, i.e. non-linear inductors, voltage sources and time-dependent external fluxes. For this case, i.e. the case where no non-linear capacitors are present, the method of nodes is well suited.

We can determine the node fluxes Φ_n with the following procedure. First, we choose a designated node, the so-called ground node $\Phi_g = 0$. Then, we choose a spanning tree \mathcal{S} , i.e. an undirected subgraph, which contains all nodes of the network but no loops. We use the discrete variable $j \in \mathbb{N}^0$ to refer to the different loops of the network. Branches, which close the loops are denoted as chords or closure branches. Similarly, the branches of the spanning tree are simply called tree branches. Next, we give the tree branches an orientation, they should point away from the ground node. In addition, we give each loop j an orientation, this orientation is chosen randomly. In the end, we align the orientation of the closure branches with the ones of the loops. Consequently, at this point all branches possess an orientation. In the end, we define the node fluxes

$$\Phi_n = \sum_{b \in \mathcal{S}} S_{n,b} \Phi_b, \quad (3.12)$$

where $S_{n,b} = 1$ if branch b is part of the path $\mathcal{P}(0, n)$ from the ground node to node n . If branch b is not part of path $\mathcal{P}(0, n)$, we set $S_{n,b} = 0$. Note that all branches on the paths $\mathcal{P}(0, n)$ have the same orientation because we defined it this way, i.e. this is a convenient choice. Finally, we can use the node fluxes Φ_n to write

$$\Phi_b = \Phi_n - \Phi_{n'}, \quad (3.13)$$

for every branch b . The branch b points from node n' to node n . This choice of variables ensures that if we form the sum of all branch fluxes Φ_b , for an

arbitrary loop j , the sum equals zero. Consequently, the time derivatives of these sums also add up to zero and this in turn means that Kirchhoff's voltage law is satisfied. A more detailed discussion of this topic with emphasis on graph theory, which goes beyond what we discuss in this thesis, can be found in Ref. [BKD04].

So far, we have ignored the presence of voltage sources $V_g(t)$ and external fluxes $\Phi_e(t)$. If we model both as EMFs, we can include $V_g(t)$ and $\Phi_e(t)$ by means of Kirchhoff's voltage law. Assuming, the external flux Φ_e present in loop j is time-independent, we add a corresponding term

$$\Phi_b = \Phi_n - \Phi_{n'} \pm \Phi_e, \quad (3.14)$$

to the closure branch b of this loop. The sign ± 1 is chosen according to the right-hand rule. The case of time-dependent external fluxes is subject of ongoing research, see Refs. [YSK19; RD22]. We discuss this issue in Sec. 3.4 in more detail. If a voltage source $V_g(t)$ is part of a branch b which contains another circuit element whose constitutive relation is expressed in terms of the variable Φ_e , we write

$$\Phi_b = \Phi_e \pm \int_{-\infty}^t V_g(t') dt'. \quad (3.15)$$

The time derivative of this relation states that the voltage drop across the whole branch V_b is given by the sum of the voltage drop across the element $V_e(t) = \dot{\Phi}_e(t)$ and the voltage source $V_g(t)$. We can solve this equation for Φ_e and add the corresponding term for the element e to the Lagrangian, see Sec. 3.4.

Kirchhoff's current law can be enforced by constructing the Lagrangian \mathcal{L} accordingly, i.e. such that the Euler-Lagrange equations yield this law. The first construction rule states that every branch b with an inductive element, which is characterised by a constitutive relation of the form $I_b(t) = g(\Phi_b(t))$, results in an additional potential term

$$U(\Phi_b) = \int_{-\infty}^t \dot{\Phi}_b(t') g(\Phi_b(t')) dt', \quad (3.16)$$

of the Lagrangian $\mathcal{L} = T - U$. The second construction rule states that every branch b with a capacitive element, which is characterised by a constitutive relation of the form $Q_b(t) = f(V_b(t))$, results in an additional kinetic term

$$T(\dot{\Phi}_b) = \int_{-\infty}^t \dot{\Phi}_b(t') f(\dot{\Phi}_b(t')) dt', \quad (3.17)$$

of the same Lagrangian \mathcal{L} . We have to perform variable substitutions in Eqs. (3.16) and (3.17) to see that the energies U and T are functions of Φ_b and $\dot{\Phi}_b$, respectively. The results read

$$U(\Phi_b) = \int_{\lim_{t \rightarrow -\infty} \Phi_b(t)}^{\Phi_b(t)} g(\Phi_b) d\Phi_b, \quad (3.18)$$

and

$$T(\dot{\Phi}_b) = f(\dot{\Phi}_b) \dot{\Phi}_b - \int_{\lim_{t \rightarrow -\infty} \dot{\Phi}_b(t)}^{\dot{\Phi}_b(t)} f(\dot{\Phi}_b) d\dot{\Phi}_b. \quad (3.19)$$

For the case we are interested in, we can verify that the Euler-Lagrange equations

$$\frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{\Phi}_n} - \frac{\partial \mathcal{L}}{\partial \Phi_n} = 0, \quad (3.20)$$

for the different node fluxes Φ_n yield Kirchhoff's current law. If we have a linear capacitive element, we find a contribution $\pm C \ddot{\Phi}_b$ to the sum of currents, see first term on the left-hand side of Eq. (3.20). Similarly, an inductive element contributes the term $\pm g(\Phi_b)$ to the sum of currents, see second term on the left-hand side of Eq. (3.20). The sign is determined by the orientation of the branch $\Phi_b = \pm(\Phi_n - \Phi_{n'})$. Therefore, changing the orientation results in a sign flip of both currents. This ensures that the in and outgoing currents carry the correct sign with regard to Kirchhoff's current law.

In a next step, we perform a Legendre transformation, see Ref. [Fen49], to derive the Hamiltonian function

$$H = \sum_{n=0}^{N-1} Q_n \dot{\Phi}_n - \mathcal{L}, \quad (3.21)$$

where

$$Q_n = \frac{\partial \mathcal{L}}{\partial \dot{\Phi}_n}, \quad (3.22)$$

is the so-called n^{th} conjugate variable. Here we assume that \mathcal{L} is a differential, convex function.

In the end, we perform the canonical quantisation. This means we make the same fundamental assumptions as the author of Ref. [Dir25], namely we replace the variables Φ_n and Q_n with the operators $\hat{\Phi}_n$ and \hat{Q}_n and map the Poisson brackets $\{\Phi_n, Q_{n'}\} = \delta_{n,n'}$ to the commutators $[\hat{\Phi}_n, \hat{Q}_{n'}] = i\hbar \delta_{n,n'}$.

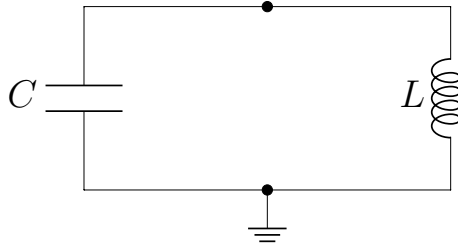


Figure 3.1: Circuit diagram of a network containing a linear capacitor with capacitance C and a linear inductor with inductance L . The ground node is marked by a dashed triangle.

3.3 Resonators and two-level systems

In circuit quantum electrodynamics there exists a circuit equivalent to the well-studied model of a mechanical harmonic oscillator. This is the first system we quantise with the formalism discussed in Sec. 3.2. Figure 3.1 shows a circuit diagram of exactly this system. As one can see, the left branch contains a linear capacitor with capacitance C and the right branch contains a linear inductor with inductance L . The ground node is marked by a dashed triangle. Here we choose $\Phi_g = 0$ such that the left branch flux is given by $\Phi_C = \Phi_n$ and the right one reads $\Phi_L = -\Phi_n$. Consequently, the Lagrangian has two terms, the kinetic energy term of the capacitor

$$T(\dot{\Phi}_n) = \frac{C}{2} \dot{\Phi}_n^2 \quad (3.23)$$

and the potential energy term

$$U(\Phi_n) = \frac{1}{2L} \Phi_n^2. \quad (3.24)$$

The Lagrangian then reads

$$\mathcal{L} = \frac{C}{2} \dot{\Phi}_n^2 - \frac{1}{2L} \Phi_n^2, \quad (3.25)$$

and the conjugate variable can be expressed as

$$Q = \frac{\partial \mathcal{L}}{\partial \dot{\Phi}_n} = C \dot{\Phi}_n. \quad (3.26)$$

If we perform the Legendre transformation as well as the canonical quantisation, we find the Hamiltonian

$$\hat{H} = \frac{\hat{Q}^2}{2C} + \frac{\hat{\Phi}^2}{2L}. \quad (3.27)$$

Note that in this last step we removed the node label. We can express this Hamiltonian as

$$\hat{H} = E_C \hat{n}^2 + \frac{E_L}{2} \hat{\varphi}^2, \quad (3.28)$$

where

$$\hat{n} = \frac{1}{2e} \hat{Q}, \quad (3.29)$$

$$\hat{\varphi} = \frac{2\pi}{\Phi_0} \hat{\Phi}, \quad (3.30)$$

$$E_C = \frac{(2e)^2}{2C}, \quad (3.31)$$

$$E_L = \left(\frac{\Phi_0}{2\pi} \right)^2 \frac{1}{L}. \quad (3.32)$$

Here E_C and E_L denote the capacitive and inductive energy of the systems, respectively. The operator given by Eq. (3.28) and the corresponding eigenvalues and eigenstates are discussed in almost all quantum theory textbooks, see for example Ref. [Wei15, Section 2.5]. The eigenstates $|\psi^{(z)}\rangle$ in φ -space read

$$\psi^{(z)}(\varphi) = \frac{1}{\sqrt{2^z z!}} \sqrt[4]{\frac{\xi}{\pi}} e^{-\frac{(\sqrt{\xi}\varphi)^2}{2}} H_z(\sqrt{\xi}\varphi), \quad (3.33)$$

where $z \in \mathbb{N}^0$, $\xi = \sqrt{E_L/(2E_C)}$ and $H_z(\sqrt{\xi}\varphi)$ denotes the Hermite polynomial of order z . We can express the Hamiltonian in its eigenbasis as

$$\hat{H}_{\text{Res.}} = \omega^{(R)} \left(\hat{a}^\dagger \hat{a} + \frac{\hat{I}}{2} \right), \quad (3.34)$$

where the $\omega^{(R)}$ is the resonance frequency

$$\omega^{(R)} = \sqrt{2E_C E_L}, \quad (3.35)$$

and \hat{a} and \hat{a}^\dagger are the bosonic number operators which can be defined in terms of their action

$$\hat{a}^\dagger |\psi^{(z)}\rangle = \sqrt{z+1} |\psi^{(z+1)}\rangle, \quad (3.36)$$

$$\hat{a} |\psi^{(z)}\rangle = \sqrt{z} |\psi^{(z-1)}\rangle, \quad (3.37)$$

on the discrete eigenstates $|\psi^{(z)}\rangle$. In analogy with the classical AC circuit theory we call this system simply a resonator. This is why we changed the label $\hat{H} \rightarrow \hat{H}_{\text{Res.}}$ of the Hamiltonian. Furthermore, from here on we omit the term $\hat{I}/2$ which only contributes a non-measurable phase to the dynamics of the system.

The resonator model in its formulation in terms of bosonic operators, see Eq. (3.34), is the basis of many quantum theoretical models, see for example Refs. [AM76; Bal07]. In the context of superconducting PGQCs we are interested in one particular model namely the two-level system (TLS) model, see Ref. [MCL19] for a review of this subject. We define the generic Hamiltonian

$$\hat{H} = \hat{H}_{\text{Sys.}} + \hat{H}_{\text{TLS},\Sigma} + \hat{W}_{\text{Int.}}, \quad (3.38)$$

where $\hat{W}_{\text{Int.}}$ is some arbitrary interaction and $\hat{H}_{\text{Sys.}}$ is a Hamiltonian which describes some arbitrary system. Furthermore, the Hamiltonian

$$\hat{H}_{\text{TLS},\Sigma} = \sum_{l \in L} \omega_l^{(T)} \hat{b}_l^\dagger \hat{b}_l, \quad (3.39)$$

describes a collection of non-interacting TLSs. Here we simply changed the notation, i.e. $\omega^{(R)} \rightarrow \omega^{(T)}$, $\hat{a} \rightarrow \hat{b}$, $\hat{a}^\dagger \rightarrow \hat{b}^\dagger$ and $L \subseteq \mathbb{N}^0$ denotes an index set. The reason for this change in notation is that in the TLS model we only consider the lowest two levels of $\hat{H}_{\text{Res.}}$, for every TLS. We can look at this model from the following point of view, \hat{H}_{TLS} constitutes a bath for the system $\hat{H}_{\text{Sys.}}$ and the interactions between bath and system are governed by the operator $\hat{W}_{\text{Int.}}$. Reference [Wil+20b] contains a numerical study where TLSs are used to include temperature effects into a model. Obviously, once we consider a specific physical scenario, we have to fix the TLS parameters $\omega_l^{(T)}$ and the operator $\hat{W}_{\text{Int.}}$.

TLS models are used to describe and/or explain various phenomena. For example, in experiments we find that relaxation times T_1 , dephasing times T_2 and qubit frequencies $\omega^{(01)}$, see Sec. 2.6 for a definition of these parameters, fluctuate around a mean value on different time scales. Such fluctuations might be explained by the presence of TLSs, see Ref. [Bur+19]. TLSs are also used to describe other types of solid state phenomena, see Ref. [Esq98].

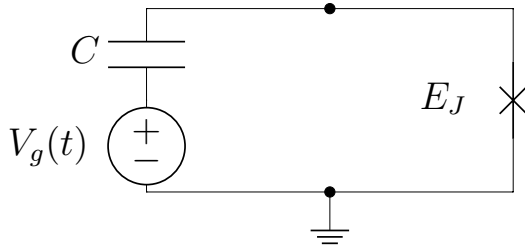


Figure 3.2: Circuit diagram of a network containing a linear capacitor with capacitance C , a Josephson junction with Josephson energy E_J and a voltage source $V_g(t)$. The ground node is marked by a dashed triangle.

3.4 Fixed-frequency and flux-tunable transmons

In this section, we quantise and analyse two circuits. These systems are usually referred to as fixed-frequency and flux-tunable transmons. The eigenstates of these systems are often used as qubit states in NIGQC models. The so-called transmission line shunted plasma oscillation qubit or transmon for short was first introduced by the authors of Ref. [Koc+07]. The circuit Hamiltonian model for NIGQC we introduce in Sec. 3.5 makes use of two types of transmons, namely fixed-frequency and flux-tunable transmons. Note that the term transmon is actually reserved for a specific parameter regime in which both systems operate. We first quantise both circuits and then discuss the corresponding eigenproblems together. Additionally, we introduce effective Hamiltonians for both systems.

Figure 3.2 shows the circuit diagram of a fixed-frequency transmon system. The left branch contains a linear capacitor with capacitance C as well as a voltage source $V_g(t)$. The right branch contains a Josephson junction with Josephson energy E_J . We choose $\Phi_g = 0$ such that the left branch flux is given by

$$\Phi_n = \Phi_C + \int_{-\infty}^t V_g(t') dt', \quad (3.40)$$

where Φ_C is the branch flux variable associated with the capacitor. Therefore, the right branch flux reads $\Phi_{E_J} = -\Phi_n$. We find that the kinetic energy of the Lagrangian is given by

$$T(\dot{\Phi}_n) = \frac{C}{2} \left(\dot{\Phi}_n - V_g(t) \right)^2, \quad (3.41)$$

and the potential energy can be expressed as

$$U(\Phi_n) = -E_J \cos\left(\frac{2\pi}{\Phi_0}\Phi_n\right). \quad (3.42)$$

The full Lagrangian then reads

$$\mathcal{L} = \frac{C}{2}(\dot{\Phi}_n - V_g(t))^2 + E_J \cos\left(\frac{2\pi}{\Phi_0}\Phi_n\right). \quad (3.43)$$

Therefore the conjugate variable is given by the expression

$$Q = \frac{\partial \mathcal{L}}{\partial \dot{\Phi}_n} = C(\dot{\Phi}_n - V_g(t)). \quad (3.44)$$

After performing the Legendre transformation and the canonical quantisation, we find the Hamilton operator

$$\hat{H} = \frac{\hat{Q}^2}{2C} + V_g(t)\hat{Q} - E_J \cos\left(\frac{2\pi}{\Phi_0}\hat{\Phi}\right). \quad (3.45)$$

Note that here we removed the node label n from the operator $\hat{\Phi}_n$. After completing the square and removing some terms which only contribute non-measurable phases to the dynamics of the system, the Hamilton operator reads

$$\hat{H} = \frac{(2e)^2}{2C} \left(\frac{\hat{Q}}{2e} + \frac{CV_g(t)}{2e} \right)^2 - E_J \cos\left(\frac{2\pi}{\Phi_0}\hat{\Phi}\right). \quad (3.46)$$

If we perform the variable substitutions

$$\hat{n} = \frac{\hat{Q}}{2e}, \quad (3.47a)$$

$$\hat{\varphi} = \frac{2\pi}{\Phi_0}\hat{\Phi}, \quad (3.47b)$$

$$n_g(t) = -\frac{CV_g(t)}{2e}, \quad (3.47c)$$

we arrive at the final Hamilton operator for a fixed-frequency transmon

$$\hat{H}_{\text{Fix.}} = E_C(\hat{n} - n_g(t))^2 - E_J \cos(\hat{\varphi}), \quad (3.48)$$

where $E_C = (2e)^2/(2C)$ denotes the capacitive energy of the system.

Figure 3.3 shows a circuit diagram of a flux-tunable transmon system. The network contains linear capacitors with capacitances C_l (left) and C_r

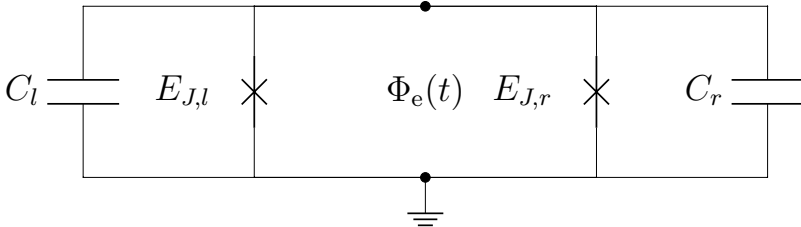


Figure 3.3: Circuit diagram of a network containing linear capacitors with capacitances C_l (left) and C_r (right), two Josephson junctions with Josephson energies $E_{J,l}$ (left) and $E_{J,r}$ (right) as well as an external flux $\Phi_e(t)$ threading through the center loop. The ground node is marked by a dashed triangle.

(right), two Josephson junctions with Josephson energies $E_{J,l}$ (left) and $E_{J,r}$ (right) as well as an external flux $\Phi_e(t)$ threading through the center loop. We model the external flux as an EMF and not as an actual physical implementation in terms of additional circuit elements. Therefore, we include the external flux by means of Kirchhoff's voltage law. The following derivation is motivated by the work in Ref. [YSK19]. The authors performed a critical analysis of the standard quantisation formalism which is often used in the literature. Note that in the meantime the work in Ref. [YSK19] was extended by the authors of Ref. [RD22].

Kirchhoff's voltage law for the center loop yields the time-dependent constraint

$$\Phi_l + \Phi_r = \Phi_e(t). \quad (3.49)$$

We can define a new degree of freedom

$$\Phi_n = m_l \Phi_l + m_r \Phi_r, \quad (3.50)$$

such that the left

$$\Phi_l = \frac{(\Phi_n - m_r \Phi_e(t))}{m_\Delta}, \quad (3.51)$$

and the right

$$\Phi_r = -\frac{(\Phi_n - m_l \Phi_e(t))}{m_\Delta}, \quad (3.52)$$

branch flux satisfy Eq. (3.49) for all $m_l, m_r \in \mathbb{R}$. Here $m_\Delta = m_l - m_r$. We use the two relations $V_l = V_{C_l} = V_{E_{J,l}}$ and $V_r = V_{C_r} = V_{E_{J,r}}$ given by Kirchhoff's

voltage law for the left and right loop to express the Lagrangian

$$\mathcal{L} = \frac{C_l}{2} \left(\frac{(\dot{\Phi}_n - m_r \dot{\Phi}_e(t))}{m_\Delta} \right)^2 + \frac{C_r}{2} \left(\frac{(\dot{\Phi}_n - m_l \dot{\Phi}_e(t))}{m_\Delta} \right)^2 - U(\Phi_n), \quad (3.53)$$

solely in terms of the variable Φ_n . Here the potential energy is given by

$$\begin{aligned} U(\Phi_n) = & -E_{J_l} \cos \left(\frac{2\pi}{\Phi_0} \frac{(\Phi_n - m_r \Phi_e(t))}{m_\Delta} \right) \\ & - E_{J_r} \cos \left(\frac{2\pi}{\Phi_0} \frac{(\Phi_n - m_l \Phi_e(t))}{m_\Delta} \right). \end{aligned} \quad (3.54)$$

After evaluating the squares and neglecting all factors proportional to $\dot{\Phi}_e(t)^2$, which in the end only contribute a non-measurable phase to the dynamics of the system, the Lagrangian reads

$$\mathcal{L} = \frac{C_\Sigma}{2m_\Delta^2} \dot{\Phi}_n^2 - \frac{(C_l m_r + C_r m_l)}{m_\Delta^2} \dot{\Phi}_e(t) \dot{\Phi}_n - U(\Phi_n). \quad (3.55)$$

The conjugate variable Q can be expressed as

$$Q = \frac{C_\Sigma}{m_\Delta^2} \dot{\Phi}_n - \frac{(C_l m_r + C_r m_l)}{m_\Delta^2} \dot{\Phi}_e(t), \quad (3.56)$$

such that the Hamiltonian

$$H = \frac{m_\Delta^2}{2C_\Sigma} Q^2 + \frac{(C_l m_r + C_r m_l)}{C_\Sigma} \dot{\Phi}_e(t) Q + U(\Phi), \quad (3.57)$$

can be determined by means of the Legendre transformation. Note that here we removed the node label n from the variable Φ_n . If we promote the conjugate variables Φ and Q to the conjugate operators $\hat{\Phi}$ and \hat{Q} and perform the substitutions

$$\hat{\varphi} = \frac{2\pi}{\Phi_0} \hat{\Phi}, \quad (3.58a)$$

$$\hat{n} = \frac{1}{2e} \hat{Q}, \quad (3.58b)$$

$$\varphi(t) = \frac{2\pi}{\Phi_0} \Phi_e(t), \quad (3.58c)$$

we find the Hamilton operator

$$\hat{H} = E_{C_\Sigma} m_\Delta^2 \hat{n}^2 + \frac{(C_l m_r + C_r m_l)}{C_\Sigma} \dot{\varphi}(t) \hat{n} + U(\hat{\varphi}), \quad (3.59)$$

where the second term on the right-hand side contains the factor $\hbar = 1$. We now simplify the parameterisation by assuming $m_\Delta = 1$ and $C_l = C_r$. Additionally, we define $m_r = -\beta$ and $C_\Sigma = C$ such that the Hamiltonian for a flux-tunable transmon can be expressed as

$$\begin{aligned} \hat{H}_{\text{Tun.}} = & E_C \hat{n}^2 + \left(\frac{1}{2} - \beta \right) \dot{\varphi}(t) \hat{n} - E_{J_l} \cos(\hat{\varphi} + \beta \varphi(t)) \\ & - E_{J_r} \cos(\hat{\varphi} + (\beta - 1)\varphi(t)). \end{aligned} \quad (3.60)$$

In the following we impose the so-called irrotational constraint, see Ref. [YSK19], onto the variable Φ . This means we use $\beta = 1/2$ as a specific choice for the variable Φ .

The flux-tunable transmon Hamiltonian for $\beta = 1/2$ can be expressed as

$$\hat{H}_{\text{Tun.}} = E_C \hat{n}^2 - E_{J,\text{eff.}}(t) \cos(\hat{\varphi} - \varphi_{\text{eff.}}(t)), \quad (3.61)$$

where

$$E_{J,\text{eff.}}(t) = E_\Sigma \sqrt{\cos\left(\frac{\varphi(t)}{2}\right)^2 + d^2 \sin\left(\frac{\varphi(t)}{2}\right)^2}, \quad (3.62)$$

denotes the effective Josephson energy and

$$\varphi_{\text{eff.}}(t) = \arctan\left(d \tan\left(\frac{\varphi(t)}{2}\right)\right), \quad (3.63)$$

refers to the effective external flux. Here we also define the auxiliary system parameters

$$E_\Sigma = (E_{J_l} + E_{J_r}), \quad (3.64)$$

and

$$d = (E_{J_r} - E_{J_l}) / (E_{J_l} + E_{J_r}). \quad (3.65)$$

The latter is usually referred to as the asymmetry factor. The obvious structural similarity with Hamiltonian Eq. (3.48) gives this system its name: the flux-tunable transmon. Both systems, the fixed-frequency and flux-tunable transmon are usually operated in the regime $E_J/E_C \gg 1$ or $E_{J,\text{eff.}}(t)/E_C \gg 1$, see Ref. [Koc+07].

The next aim is to add a driving term of the form $n_g(t)\hat{n}$ to the model of a flux-tunable transmon. Figure 3.4 shows the circuit of a flux-tunable transmon, see Fig. 3.3, with an additional branch which contains a voltage source characterised by the real-valued function $V_g(t)$ and an additional capacitor with the capacitance C_g . In the following we make use of Kirchhoff's voltage law for the center loop

$$V_{C_g} + V_g(t) = V_{C_l}, \quad (3.66)$$

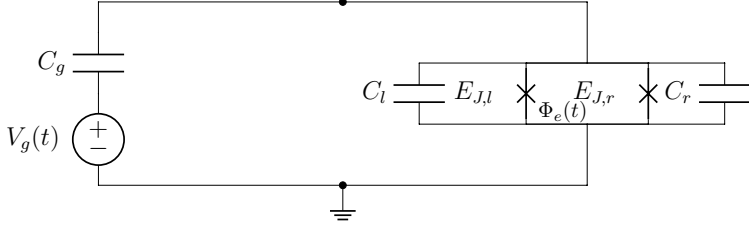


Figure 3.4: Circuit diagram of a network. The right branch contains linear capacitors with capacitances C_l (left) and C_r (right), two Josephson junctions with Josephson energies $E_{J,l}$ (left) and $E_{J,r}$ (right) as well as an external flux $\Phi_e(t)$ threading through the loop which contains the two Josephson junctions. The left branch contains a linear capacitor C_g and a voltage source $V_g(t)$. The ground node is marked by a dashed triangle.

to derive the circuit Hamiltonian \hat{H}^* for this system. The Lagrangian for this system reads

$$\mathcal{L}^* = \mathcal{L} + \frac{C_g}{2} \left(\dot{\Phi}_l - V_g(t) \right)^2. \quad (3.67)$$

Consequently, we can express \mathcal{L}^* in terms of Φ_n

$$\mathcal{L}^* = \mathcal{L} + \frac{C_g}{2m_\Delta^2} \dot{\Phi}_n^2 - \frac{C_g}{m_\Delta^2} \dot{\Phi}_n \left(m_r \dot{\Phi}_e(t) + m_\Delta V_g(t) \right), \quad (3.68)$$

where we dropped all terms proportional to $\dot{\Phi}_e(t)^2$, $V_g(t)^2$ and $\dot{\Phi}_e(t)V_g(t)$. In the end, these terms only contribute a non-measurable phase to the dynamics of the system. Again, we simplify the parametrisation by assuming $m_\Delta = 1$, $m_r = -\beta$, $C_l = C_r$ and $C = C_\Sigma$, the result reads

$$\begin{aligned} \mathcal{L}^* = & \frac{(C + C_g)}{2} \dot{\Phi}_n^2 - \left(\frac{C}{2} - \beta (C + C_g) \right) \dot{\Phi}_e(t) \dot{\Phi}_n \\ & - C_g V_g(t) \dot{\Phi}_n - U(\Phi_n), \end{aligned} \quad (3.69)$$

where as before we neglect all terms proportional to $\dot{\Phi}_e(t)^2$. In an ad hoc manner, we assume that $C + C_g \rightarrow C$ such that the system's dynamic behaviour is described by the Lagrangian

$$\mathcal{L}^* = \frac{C}{2} \dot{\Phi}_n^2 - C \left(\frac{1}{2} - \beta \right) \dot{\Phi}_e(t) \dot{\Phi}_n - C_g V_g(t) \dot{\Phi}_n - U(\Phi_n). \quad (3.70)$$

Therefore, the conjugate variable reads

$$Q = C \dot{\Phi}_n - C \left(\frac{1}{2} - \beta \right) \dot{\Phi}_e(t) - C_g V_g(t), \quad (3.71)$$

and the first term of the Hamiltonian function can be expressed as

$$Q\Phi_n = \frac{Q^2}{C} + \left(\frac{1}{2} - \beta\right) \dot{\Phi}_e(t)Q + \frac{C_g}{C}V_g(t)Q. \quad (3.72)$$

After expressing the Lagrangian in terms of the variables Φ_n and Q , the function reads

$$\mathcal{L}^* = \frac{Q^2}{2C} - U(\Phi_n), \quad (3.73)$$

where we neglect all terms which only contribute non-measurable phases to the dynamics of the system. Consequently, the Hamiltonian can be expressed as

$$H^* = E_C \left(\frac{Q}{2e}\right)^2 + \left(\frac{1}{2} - \beta\right) (2e)\dot{\Phi}_e(t) \left(\frac{Q}{2e}\right) - 2E_C n_g(t) \left(\frac{Q}{2e}\right) + U(\Phi_n), \quad (3.74)$$

where the real-valued function $n_g(t)$ is defined as

$$n_g(t) = -\frac{C_g V_g(t)}{2e}. \quad (3.75)$$

After performing the canonical quantisation, completing the square with regard to the term $n_g(t)\hat{n}$, using the substitutions in Eq. (3.58)(a-c) with $\hbar = 1$ and dropping some terms which only contribute non-measurable phases to the dynamics of the system, we find the Hamilton operator

$$\begin{aligned} \hat{H}_{\text{Tun.}}^* = & E_C (\hat{n} - n_g(t))^2 + \left(\frac{1}{2} - \beta\right) \dot{\varphi}(t)\hat{n} - E_{J,I} \cos(\hat{\varphi} + \beta\varphi(t)) \\ & - E_{J,r} \cos(\hat{\varphi} + (\beta - 1)\varphi(t)). \end{aligned} \quad (3.76)$$

Note that here we removed the node label n from the operator $\hat{\Phi}_n$. Furthermore, as before the second term on the right-hand side of Eq. (3.76) contains the factor $\hbar = 1$. We use the Hamiltonian in Eq. (3.76) to describe flux-tunable transmons with an additional charge drive component.

In this thesis, almost all results are obtained numerically. Therefore, we have to express the Hamiltonians in Eqs. (3.48) and (3.59) in some discrete basis. The basis we use is the so-called charge basis $\{|n\rangle\}_{n \in \mathbb{Z}}$. We define the basis states by

$$|n\rangle = \frac{1}{2\pi} \int_0^{2\pi} e^{-in\varphi} |\varphi\rangle d\varphi, \quad (3.77)$$

where $n \in \mathbb{Z}$, such that the charge operator can be expressed as

$$\hat{n} = \sum_{n=-\infty}^{\infty} n |n\rangle\langle n|. \quad (3.78)$$

The states $|\varphi\rangle$ are the eigenstates of the operator $\hat{\varphi}$. Here $\varphi \in [0, 2\pi)$. Both operators are Hermitian and Fourier transform duals of each other. The charge operator \hat{n} can be expressed as a differential operator

$$\hat{n} = -i\partial_{\varphi}, \quad (3.79)$$

in the flux basis. Furthermore, the flux basis states read

$$|\varphi\rangle = \sum_{n=-\infty}^{\infty} e^{in\varphi} |n\rangle. \quad (3.80)$$

If we consider the action of the operator

$$e^{i\hat{\varphi}} |n\rangle = \frac{1}{2\pi} \int_0^{2\pi} \langle\varphi|n\rangle e^{i\varphi} |\varphi\rangle d\varphi, \quad (3.81)$$

on the charge states $|n\rangle$, we find

$$e^{i\hat{\varphi}} |n\rangle = |n-1\rangle. \quad (3.82)$$

Consequently, we can express the cosine operator as

$$\cos(\hat{\varphi}) = \frac{1}{2} \sum_{n=-\infty}^{\infty} (|n\rangle\langle n+1| + |n+1\rangle\langle n|). \quad (3.83)$$

The statements made regarding the charge basis states are not as accurate as required. However, a detailed discussion of the mathematical subtleties would require us to extend the mathematical framework considerably. Therefore, we abstain from doing so and refer the interested reader to Ref. [Wil16, Section 2.1.3], which provides a summary of mathematical issues that appear upon closer inspection. For the simulations in this thesis, we always use a finite number $2N_c + 1$, where $N_c \in \mathbb{N}^0$, of basis states $\{|-N_c\rangle, \dots, |0\rangle, \dots, |N_c\rangle\}$ to express the Hamiltonians Eqs. (3.48) and (3.59) as matrices in a computer program.

If we consider qubits, which are abstract states by definition, see Sec. 2.2, as our object of study, we usually assume that the dynamics of the system can, in some basis, be confined to a small subspace. An even smaller subspace is used as the computational basis, see Eq. (2.24), for NIGQC models. Since this is a rather subtle issue, it is best to further clarify this statement. The aim is to control the system such that the state vector can be expressed in terms of the computational basis at certain points in time, i.e. the moments in time where a measurement would take place. The qubit states themselves

Table 3.1: Parameters for a fixed-frequency (row $i = 0$) and a flux-tunable transmon (row $i = 1$). The first column contains indices i for the different parameter sets. The second and third column show the qubit frequency $\omega^{(Q_0)}$ and anharmonicity $\alpha^{(Q_0)}$. The fourth, fifth and sixth column show the capacitive energy E_C , the left Josephson energy E_{J_l} and the right Josephson energies E_{J_r} , respectively. The seventh column shows the flux offset parameter φ_0 which is determined by $\varphi(0)$, see Eq. (3.59). All units are in GHz except the flux offset parameter φ_0 which is without units. The parameters are motivated by Refs. [Gan+20; Lac+20].

i	$\omega_i^{(Q_0)}/2\pi$	$\alpha_i^{(Q_0)}/2\pi$	$E_{C_i}/2\pi$	$E_{J_{l,i}}/2\pi$	$E_{J_{r,i}}/2\pi$	$\varphi_{0,i}/2\pi$
0	6.200	-0.285	1.027	20.371	n/a	n/a
1	5.200	-0.295	1.036	4.817	9.633	0

are usually associated with the eigenstates $|\phi^{(z)}(t)\rangle$ and eigenvalues $E^{(z)}(t)$ of some time-dependent Hamiltonian $\hat{H}(t)$, given by

$$\hat{H}(t) |\phi^{(z)}(t)\rangle = E^{(z)}(t) |\phi^{(z)}(t)\rangle, \quad (3.84)$$

where $z \in \mathbb{N}^0$. Note that we usually use the states at $t = 0$ as qubit states.

In this thesis, we use the eigenstates of the fixed-frequency and flux-tunable transmons as qubit states. In this context, we define the circuit Hamiltonian NIGQC model qubit frequency $\omega^{(Q_0)} = E^{(1)}(0) - E^{(0)}(0)$ and anharmonicity $\alpha^{(Q_0)} = E^{(2)}(0) - E^{(0)}(0) - 2\omega^{(Q_0)}$ in terms of the eigenvalues $E^{(z)}(t)$ of the corresponding systems. Additionally, in both cases we can express $|\phi^{(z)}(t)\rangle$ and $E^{(z)}(t)$ in terms of the Mathieu functions \mathcal{M}_A , \mathcal{M}_C and \mathcal{M}_S . Here we use the same notation as the author of Ref. [Cot02]. This is a consequence of the fact that the Hamiltonian in Eq. (3.48) and the Hamiltonian in Eq. (3.61) for $\beta = 1/2$ are structurally equivalent, i.e. the only difference between both Hamiltonians is that they show different time dependencies, with regard to the variables $n_g(t)$ and $\varphi(t)$. Despite the fact that both systems share this particular kind of equivalence, we find that their behaviour, with respect to the variables $n_g(t)$ and $\varphi(t)$ is rather different. Therefore, we discuss and compare both systems, with respect to their corresponding dynamic variables, by analysing the eigenstates and eigenvalues.

The eigenstates of a fixed-frequency transmon in the φ -space read

$$\phi^{(z)}(\varphi, t) = \frac{e^{in_g(t)\varphi}}{\sqrt{2\pi}} \left(\mathcal{M}_C(a(t), b, \frac{\varphi}{2}) + i^{1+2(z+1)} \mathcal{M}_S(a(t), b, \frac{\varphi}{2}) \right), \quad (3.85)$$

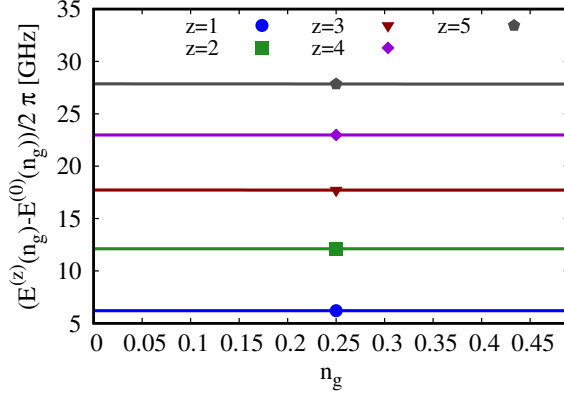


Figure 3.5: The lowest five energy levels of a fixed-frequency transmon as a function of the effective offset charge n_g . We use the Hamiltonian Eq. (3.48), $N_c = 50$ charge states and the parameters listed in Table 3.1, row $i = 0$, to obtain the results.

where we introduced the auxiliary functions

$$a(t) = \frac{4E^{(z)}(t)}{E_C}, \quad (3.86a)$$

$$b = \frac{-2E_J}{E_C}, \quad (3.86b)$$

and the eigenvalues

$$E^{(z)}(t) = E_C \mathcal{M}_A(z + 1 - (z + 1)[\text{mod}2] + 2n_g(t)(-1)^z, b). \quad (3.87)$$

The variable $n_g(t)$ enters the eigenstates twice: explicitly through the global phase $e^{in_g(t)\varphi}$ and implicitly through $E^{(z)}(t)$. We emphasise that eigenstates and eigenvalues are taken from Ref. [Cot02].

Figure 3.5 shows the lowest five states of the energy spectrum of a fixed-frequency transmon as a function of effective offset charge n_g . Here we use the Hamiltonian given by Eq. (3.48), $N_c = 50$ charge states and the parameters listed in Table 3.1, row $i = 0$, to diagonalise the corresponding matrix representation. As one can see, Fig. 3.5 shows five quasi-constant energies. To the best knowledge of the author, the contributors of Ref. [Koc+07] were the first to notice that if the system is operated in the transmon limit $E_J/E_C \gg 1$, the eigenvalues $E^{(z)}(t)$ become quasi constant with regard to the variable

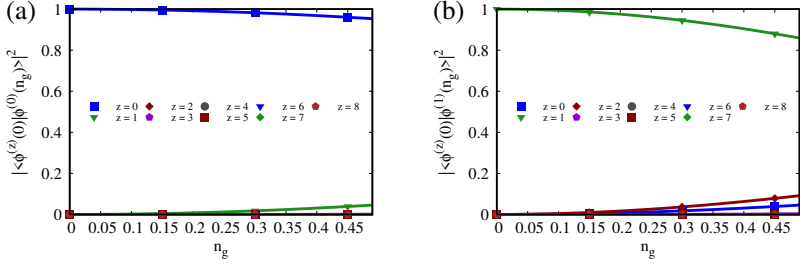


Figure 3.6: Overlaps $|\langle \phi^{(z)}(0) | \phi^{(0)}(n_g) \rangle|^2$ (a) and $|\langle \phi^{(z)}(0) | \phi^{(1)}(n_g) \rangle|^2$ (b) between the instantaneous eigenstates $|\phi^{(0/1)}(\varphi)\rangle$ and various measurement basis states $|\phi^{(z)}(0)\rangle$, where $z \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$, as functions of the external charge n_g . The basis states $|\phi^{(z)}(\varphi)\rangle$ are obtained by numerical diagonalisation. We use Hamiltonian Eq. (3.48), $N_c = 50$ charge states and the parameters listed in Table 3.1, row $i = 0$, to obtain the results.

n_g . For the remainder of this discussion, we assume that the eigenvalues are actually constant. Therefore, in the transmon limit the eigenstates only vary in time by means of a global phase. If we assume that $n_g(0) = 0$, the eigenstates read

$$\phi^{(z)}(\varphi, t) = e^{in_g(t)\varphi} \phi^{(z)}(\varphi, 0). \quad (3.88)$$

We define that the transmon basis states are the eigenstates of $\hat{H}_{\text{Fix.}}(t)$ for $t = 0$. The overlap between the transmon basis states and $|\phi^{(z)}(t)\rangle$ then reads

$$\langle \phi^{(z)}(t) | \phi^{(z)}(0) \rangle = \int_0^{2\pi} e^{-in_g(t)\varphi} |\phi^{(z)}(\varphi, 0)|^2 d\varphi. \quad (3.89)$$

For practical purposes, i.e. the implementation of quantum gates, it is fair to assume that $n_g(t) \leq 0.05$, see for example Refs. [Wil+17; Wil20a]. In such cases, we find that the argument $n_g(t)\varphi$ only varies slowly. Therefore, one may assume that $\langle \phi^{(z)}(t) | \phi^{(z)}(0) \rangle \simeq \langle \phi^{(z)}(0) | \phi^{(z)}(0) \rangle$ for all t . Note that the overlap cannot be fully time independent, otherwise we would not be able to implement any gates with a fixed-frequency transmon.

To study this relation for various values of n_g , Fig. 3.6(a-b) show the overlaps $|\langle \phi^{(z)}(0) | \phi^{(0)}(n_g) \rangle|^2$ (a) and $|\langle \phi^{(z)}(0) | \phi^{(1)}(n_g) \rangle|^2$ (b) between the instantaneous eigenstates $|\phi^{(0)}(n_g)\rangle$ and $|\phi^{(1)}(n_g)\rangle$ and various transmon basis states $|\phi^{(z)}(0)\rangle$, where $z \in \{0, 1, \dots, 9\}$, as functions of the external charge n_g . We use Hamiltonian Eq. (3.48), $N_c = 50$ charge states and the parameters listed in Table 3.1, see $i = 0$, to compute the overlap.

Figures 3.6(a-b) show that for $n_g \leq 0.05$ the overlaps barely deviate from one or zero. However, these small deviations are sufficient to implement single-qubit gates. For $n_g > 0.05$, the state $|\phi^{(0)}(n_g)\rangle$ can possibly be expressed as a superposition of the states $|\phi^{(0)}(0)\rangle$ and $|\phi^{(1)}(0)\rangle$. Similarly, the state $|\phi^{(1)}(n_g)\rangle$ can possibly be expressed as a superposition of the lowest three transmon basis states.

As a consequence, we find that for small n_g fixed-frequency transmons behave like artificial atoms, i.e. like systems with eigenvalues and eigenstates which are approximately time independent. These properties allow us to activate transitions, by means of Rabi oscillations, see Ref. [CGD92], between different transmon basis states, potentially without leaving the lowest three or four states. The dynamics of certain circuit architectures, which solely rely on fixed-frequency transmons and resonators as coupler elements, might be modelled with four transmon basis states only, see Ref. [Wil20a].

Fixed-frequency transmons are often described by effective models, see Refs.[Rot+17; McK+16; Koc+07; Bla+21; Gu+21] and many more. We can develop such an effective model in a step-wise manner. Note that for simplicity, we set $n_g(0) = 0$ in the beginning. If we expand the cosine in Eq. (3.48) to the second order, we find the Hamiltonian given by Eq. (3.28) with $E_L = E_J$ such that the harmonic oscillator frequency given by Eq. (3.35) reads $\omega = \sqrt{2E_C E_J}$. Note that here we neglect all terms which only contribute a non-measurable phase to the dynamics of the system. If we expand the cosine in Eq. (3.48) to the quartic order, we find the Hamiltonian

$$\hat{H}_{\text{quartic}} = \omega \hat{c}^\dagger \hat{c} + -\frac{E_C}{48} (\hat{c}^\dagger - \hat{c})^4, \quad (3.90)$$

where \hat{c}^\dagger and \hat{c} are the bosonic operators, see Sec. 3.3, which act on the harmonic basis states $|\psi^{(z)}\rangle$. We can split the operator

$$(\hat{c}^\dagger - \hat{c})^4 = \hat{D} + \hat{V}, \quad (3.91)$$

into an operator \hat{D} which is diagonal in the harmonic basis and one \hat{V} which is non-diagonal in the same basis. We neglect the operator \hat{V} and define another Hamiltonian

$$\hat{H}_{\text{fix.}} = \omega^{(q_0)} \hat{c}^\dagger \hat{c} + \frac{\alpha^{(q_0)}}{2} \left(\hat{c}^\dagger \hat{c} \left(\hat{c}^\dagger \hat{c} - \hat{I} \right) \right), \quad (3.92)$$

where $\omega^{(q_0)} = \omega + \alpha^{(q_0)}$ and $\alpha^{(q_0)} = -E_C/4$. The model parameters $\omega^{(q_0)}$ and $\alpha^{(q_0)}$ determine the first and second energy levels of the system $\hat{H}_{\text{fix.}}$. The former $\omega^{(q_0)}$ denotes the transmon qubit frequency and the latter $\alpha^{(q_0)}$ is referred to as the transmon qubit anharmonicity. Note that the derivation

of Hamiltonian Eq. (3.92) is similar but not equivalent to one which can be found in Ref. [DiV13, Section B 4.1.3].

We can also incorporate the driving term in Eq. (3.48) by noting that we can factor out the term $-2E_C n_g(t) \hat{n}$. The corresponding driving term reads

$$\hat{D}_{\text{Dri.}}(t) = \Omega(t) (\hat{c}^\dagger + \hat{c}), \quad (3.93)$$

where $\Omega(t) \propto -2E_C n_g(t)$. In the end, we define the effective Hamiltonian for a fixed-frequency transmon as

$$\hat{H}_{\text{fix.}}(t) = \omega^{(q_0)} \hat{c}^\dagger \hat{c} + \frac{\alpha^{(q_0)}}{2} \left(\hat{c}^\dagger \hat{c} (\hat{c}^\dagger \hat{c} - \hat{I}) \right) + \Omega(t) (\hat{c}^\dagger + \hat{c}). \quad (3.94)$$

It is often the case that only the lowest two or three basis states are used to model these systems. If we use two basis states only, we can express the effective Hamiltonian in terms of Pauli spin operators as

$$\hat{H}_{\text{fix.,TLS}}(t) = -\frac{\omega}{2} \hat{\sigma}^{(z)} + \Omega(t) \hat{\sigma}^{(x)}, \quad (3.95)$$

where we omit the contribution of all higher-order terms in the cosine expansion, see $\omega^{(q_0)}$.

We emphasise that one cannot claim that the circuit Hamiltonian given by Eq. (3.48) and any of the effective Hamiltonians discussed so far generate the same dynamics in terms of the TDSE. Since we introduce more effective Hamiltonians in this section, we address this issue in a general manner at the end of this section.

Next, we turn our attention to the flux-tunable transmon. As mentioned before, the Mathieu functions are also the eigenstates of the flux-tunable transmon Hamiltonian given by Eq. (3.61). The Hamiltonians in Eq. (3.48) and Eq. (3.61) are structurally equivalent, only the parameter dependencies are different, i.e. we have to perform the substitutions $E_J \rightarrow E_{J,\text{eff}}(t)$ and $\varphi \rightarrow \varphi - \varphi_{\text{eff}}(t)$. However, this means that flux-tunable transmons have time-dependent eigenvalues and eigenstates.

Figure 3.7 shows the lowest five energy levels of a flux-tunable transmon as a function of the flux offset φ . Here we use the Hamiltonian given by Eq. (3.59), $N_c = 50$ charge states and the parameters listed in Table 3.1, row $i = 1$, to obtain the results. As one can see, all energies decrease with increasing flux φ . If one uses the lowest two states as qubit states, one might associate the tunable energy levels with a tunable qubit frequency.

Figures 3.8(a-b) show the overlaps $|\langle \phi^{(z)}(0) | \phi^{(0/1)}(\varphi) \rangle|^2$ between the instantaneous eigenstates $|\phi^{(0)}(\varphi)\rangle$ (a) and $|\phi^{(1)}(\varphi)\rangle$ (b) and various transmon basis states $|\phi^{(z)}(0)\rangle$, where $z \in \{0, 1, \dots, 9\}$, as functions of the external flux

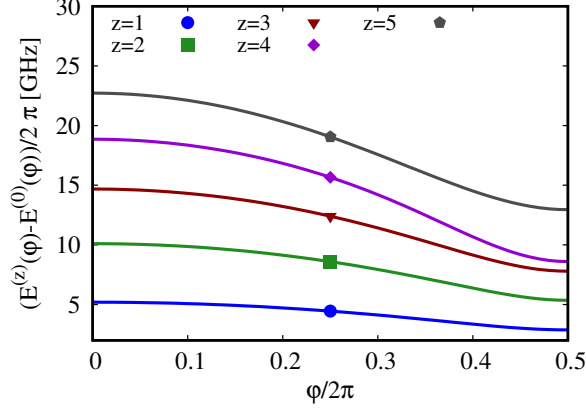


Figure 3.7: The lowest five energy levels of a flux-tunable transmon as a function of the flux offset φ . We numerically diagonalise Hamiltonian Eq. (3.59) with $N_c = 50$ charge states and the parameters listed in Table 3.1, row $i = 1$.

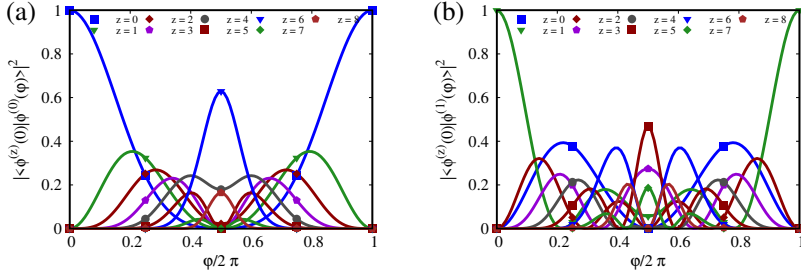


Figure 3.8: Overlaps $|\langle \phi^{(z)}(0) | \phi^{(0)}(\varphi) \rangle|^2$ (a) and $|\langle \phi^{(z)}(0) | \phi^{(1)}(\varphi) \rangle|^2$ (b) between the instantaneous eigenstates $|\phi^{(0/1)}(\varphi)\rangle$ and various measurement basis states $|\phi^{(z)}(0)\rangle$, where $z \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$, as functions of the external flux φ . The basis states $|\phi^{(z)}(\varphi)\rangle$ are obtained by numerical diagonalisation of Hamiltonian Eq. (3.59). We use $N_c = 50$ charge states and the parameters listed in Table 3.1, row $i = 1$, to obtain the results.

$\varphi/2\pi \in [0, 1]$. Here we use the parameters listed in Table 3.1, row $i = 1$, the Hamiltonian given by Eq. (3.59) and $N_c = 50$ charge states to compute the overlaps between the different states. Figures 3.8(a-b) show that the overlaps between the instantaneous basis states and the transmon basis states $z = 0$ and $z = 1$ vanish as $\varphi/2\pi$ approaches the point 0.5, afterwards they return back to one. Hamiltonian Eq. (3.61) has two symmetry points, one at $\varphi/2\pi = 0.5$ and one at $\varphi/2\pi = 1.0$. The two symmetry points explain this behaviour.

In accordance with the adiabatic theorem, see Refs. [Wei15; Ami09], the external flux might allow us to move the system from a state $|\phi^{(z)}(\varphi_0)\rangle$, where $\varphi_0 = \varphi(0)$ is an arbitrary flux offset which defines the transmon basis, with energy $E^{(z)}(\varphi_0)$ to a state $|\phi^{(z)}(\varphi(t))\rangle$ with energy $E^{(z)}(\varphi(t))$, see Fig. 3.7, as long as we vary the external flux slowly enough. Since in general $\langle \phi^{(z)}(\varphi(t)) | \phi^{(z)}(\varphi_0) \rangle \neq 1$, we expect that the description of these systems in the transmon basis probably requires more than just two or three basis states. The instantaneous basis states are linear combinations of the lowest transmon basis states, see Figs. 3.8(a-b) and the amplitude of $\varphi(t)$ determines the overlap or how many basis states are needed to describe the system.

As for fixed-frequency transmons, we find that flux-tunable transmons are often modelled by means of effective models, see for example Refs. [McK+16; Rot+17; Yan+18; Did+18; Rol+19; Bla+21; Gu+21] and we can develop such an effective model in a step-wise manner. If we expand the Hamiltonian given by Eq. (3.61) to the second order in the cosine, we can define the Hamiltonian

$$\hat{H}_{\text{second}}(t) = E_C \hat{n}^2 + \frac{E_{J,\text{eff.}}(t)}{2} (\hat{\varphi} - \varphi_{\text{eff.}}(t))^2, \quad (3.96)$$

such that the structural similarity to the harmonic oscillator Hamiltonian in Eq. (3.28) should be obvious. The difference between the Hamiltonians in Eqs. (3.28) and (3.96) is that the eigenstates and eigenvalues of the Hamiltonian in Eq. (3.96) are now time dependent. The time-dependent eigenstates in φ -space read

$$\psi^{(z)}(x(t)) = \frac{1}{\sqrt{2^z z!}} \sqrt{\frac{\xi(t)}{\pi}} e^{-\frac{x(t)^2}{2}} H_z(x(t)), \quad (3.97)$$

where $z \in \mathbb{N}^0$, $\xi(t) = \sqrt{E_{J,\text{eff.}}(t)/(2E_C)}$, $x(t) = \sqrt{\xi(t)}(\varphi - \varphi_{\text{eff.}}(t))$ and $H_z(x(t))$ denotes the Hermite polynomial of order z . The eigenvalues are characterised by the time-dependent frequency

$$\omega(t) = \sqrt{2E_C E_{J,\text{eff.}}(t)}. \quad (3.98)$$

We use the time-dependent basis states $|\psi^{(z)}(t)\rangle$ to model the dynamics of the system. Consequently, we work with the transformed state vector

$$|\Psi^*(t)\rangle = \mathcal{W}(t) |\Psi(t)\rangle, \quad (3.99)$$

where $\mathcal{W}(t)$ is a unitary transformation which maps the basis states $|\psi^{(z)}(0)\rangle$ at time $t = 0$ to the states $|\psi^{(z)}(t)\rangle$ at time t . The TDSE for the state vector $|\Psi^*(t)\rangle$ should retain its original form, see Refs. [Wil20b; Wei15]. Therefore, we have to transform the Hamiltonian as

$$\hat{H}^*(t) = \mathcal{W}(t) \hat{H}(t) \mathcal{W}^\dagger(t) - i\mathcal{W}(t) \partial_t \mathcal{W}^\dagger(t). \quad (3.100)$$

We denote the second term in Eq. (3.100) as the driving term

$$\hat{\mathcal{D}}_{\text{Dri.}}(t) = -i\mathcal{W}(t) \partial_t \mathcal{W}^\dagger(t), \quad (3.101)$$

and we can determine this term to be

$$\hat{\mathcal{D}}_{\text{Dri.}}(t) = -i\sqrt{\frac{\xi(t)}{2}} \dot{\varphi}_{\text{eff.}}(t) (\hat{c}^\dagger - \hat{c}) + \frac{i}{4} \frac{\dot{\xi}(t)}{\xi(t)} (\hat{c}^\dagger \hat{c}^\dagger - \hat{c} \hat{c}), \quad (3.102)$$

where we assume that $\xi(t) \neq 0$ for all times t . Here we adjusted the derivation in Ref. [Wil20b, Section 5.1.2] which considers a similar model. Additionally, we find

$$\dot{\varphi}_{\text{eff.}}(t) = \dot{\varphi}(t) \frac{d}{2 \left(\cos\left(\frac{\varphi(t)}{2}\right)^2 + d^2 \sin\left(\frac{\varphi(t)}{2}\right)^2 \right)}, \quad (3.103)$$

and

$$\frac{\dot{\xi}(t)}{\xi(t)} = \dot{\varphi}(t) \frac{(d^2 - 1) \sin(\varphi(t))}{8 \left(\cos\left(\frac{\varphi(t)}{2}\right)^2 + d^2 \sin\left(\frac{\varphi(t)}{2}\right)^2 \right)}. \quad (3.104)$$

The transformed Hamiltonian in the time-dependent harmonic basis can be expressed as

$$\hat{H}_{\text{second}}^*(t) = \omega(t) \hat{c}^\dagger \hat{c} - i\sqrt{\frac{\xi(t)}{2}} \dot{\varphi}_{\text{eff.}}(t) (\hat{c}^\dagger - \hat{c}) + \frac{i}{4} \frac{\dot{\xi}(t)}{\xi(t)} (\hat{c}^\dagger \hat{c}^\dagger - \hat{c} \hat{c}), \quad (3.105)$$

where we neglect all terms which only contribute a non-measurable phase to the dynamics of the system. Note that so far we only considered a time-dependent harmonic oscillator model.

If we use the lowest two eigenstates only, we can use the effective Hamiltonian

$$\hat{H}_{\text{second, TLS}}^*(t) = -\frac{\omega(t)}{2} \hat{\sigma}^{(z)} + -\sqrt{\frac{\xi(t)}{2}} \dot{\varphi}_{\text{eff.}}(t) \hat{\sigma}^{(y)}, \quad (3.106)$$

to model the dynamics of the system. Some authors, see for example Refs. [Rot+17; McK+16; Yan+18], use the term $-(\omega(t)/2)\hat{\sigma}^{(z)}$ only to model flux-tunable transmons as two-level systems. If we consider the Hamiltonian given by Eq. (3.61) and compare it with the term $-(\omega(t)/2)\hat{\sigma}^{(z)}$, we find that with such an approximation all higher-order terms in the cosine expansion do not contribute to the dynamics of the system. Additionally, the driving term in Eq. (3.102) and all higher states are ignored completely.

We can include some contributions of the higher-order terms in the cosine expansion as for the case of the fixed-frequency transmon, see Eq. (3.91). First, we expand the cosine in Eq. (3.61) to the quartic order and define the Hamiltonian

$$\hat{H}_{\text{quartic}}^*(t) = \omega(t)\hat{c}^\dagger\hat{c} + \frac{E_C}{48}(\hat{c}^\dagger - \hat{c})^4 + \hat{\mathcal{D}}_{\text{Dri.}}(t). \quad (3.107)$$

Second, we split the operator

$$(\hat{c}^\dagger - \hat{c})^4 = \hat{D} + \hat{V}, \quad (3.108)$$

into a diagonal \hat{D} and non-diagonal \hat{V} part, in the time-dependent harmonic basis. Third, we define the two effective Hamiltonians

$$\hat{H}_{\text{tun.,I}}^*(t) = \omega^{(q)}(t)\hat{c}^\dagger\hat{c} + \frac{\alpha^{(q_0)}}{2}(\hat{c}^\dagger\hat{c}(\hat{c}^\dagger\hat{c} - \hat{I})) + \hat{\mathcal{D}}_{\text{Dri.}}(t), \quad (3.109)$$

and

$$\hat{H}_{\text{tun.,II}}^*(t) = \omega^{(q)}(t)\hat{c}^\dagger\hat{c} + \frac{\alpha^{(q_0)}}{2}(\hat{c}^\dagger\hat{c}(\hat{c}^\dagger\hat{c} - \hat{I})), \quad (3.110)$$

where $\omega^{(q)}(t) = \omega(t) + \alpha^{(q_0)}$ and $\alpha^{(q_0)} = -E_C/4$. With the definition of the Hamiltonian given by Eq. (3.109) we neglect all non-diagonal contributions \hat{V} . Similarly, with the definition of the Hamiltonian in Eq. (3.110) we neglect all non-diagonal contributions \hat{V} and the driving term $\hat{\mathcal{D}}_{\text{Dri.}}(t)$.

To the best knowledge of the author, one often finds that the tunable frequency $\omega^{(q)}(t)$ is modelled with the function

$$\omega^{(q)}(t) = \omega^{(q_0)} \sqrt{\cos\left(\frac{\varphi(t)}{2}\right)^2 + d^2 \sin\left(\frac{\varphi(t)}{2}\right)^2}, \quad (3.111)$$

only, where $\omega^{(q_0)} = \text{const.}$ or $\omega^{(q_0)} = \sqrt{2E_C E_\Sigma}$ and flux-tunable transmons are modelled with the effective Hamiltonian given by Eq. (3.110), see for example Refs. [McK+16; Rot+17; Gan+20; Gu+21].

We now turn to the question how the dynamics generated by an original model Hamiltonian $\hat{H}_M(t)$ and the corresponding effective Hamiltonian

$\hat{H}_E(t)$ deviate in time. The formal solutions of the TDSE for both Hamiltonians can be expressed as

$$\hat{\mathcal{U}}_M(t, t_0) = \mathcal{T} \exp \left(-i \int_{t_0}^t \hat{H}_M(t') dt' \right) \quad (3.112)$$

and

$$\hat{\mathcal{U}}_E(t, t_0) = \mathcal{T} \exp \left(-i \int_{t_0}^t \hat{H}_E(t') dt' \right), \quad (3.113)$$

where $\hat{\mathcal{U}}_M$ and $\hat{\mathcal{U}}_E$ denote the time-evolution operators for both Hamiltonians. Consequently, a quantitative assessment of the deviations between both models should be based on the time-evolution operators in combination with an appropriate operator norm. Unfortunately, we find that in almost all cases the solution of the TDSE is not known such that a detailed comparison between both solutions is a non-trivial problem in itself. A general discussion of this issues and explicit examples can be found in Ref. [Bur+22].

Comparing the time evolution of two Hamiltonian models can pose a rather difficult problem alone. However, we have to consider additional complications once we consider larger systems, i.e. interacting many-particle systems. For example, by defining the effective Hamiltonian in Eq. (3.110), we neglect the driving term given by Eq. (3.102). For simplicity, we consider the effective Hamiltonian in Eq. (3.109) to be the original model. This approximation constitutes some type of adiabatic approximation. The Hamiltonian in Eq. (3.110) is diagonal so that the formal solution of the TDSE can be obtained. The structure of this solution is similar but not equivalent to the one of a state vector under the adiabatic approximation, see Ref. [Wei15]. In case of the Hamiltonian in Eq. (3.110), we additionally neglect the so-called geometric phase. This well-studied approximation is based on a formulation of the problem in terms of the instantaneous eigenstates. Consequently, if we use the Hamiltonian in Eq. (3.110) to define an interacting many-particle system, we find that the corresponding error bounds are not valid any more. Furthermore, moving from the Hamiltonian given by Eq. (3.61) to the Hamiltonian in Eq. (3.110) requires a series of redefinitions of the model of a flux-tunable transmon. Additionally, it is usually the case that only the lowest states of the system are used to model the dynamics. This step constitutes an additional approximation. To the best knowledge of the author, none of these steps comes with an explicit analysis of the error in terms of the time-evolution operators involved in the process, see Ref. [Bur+22] for explicit examples. Consequently, we should not claim that the different models generate the same dynamics.

3.5 A circuit Hamiltonian model for non-ideal gate-based transmon quantum computers

We can use fixed-frequency transmons, flux-tunable transmons, resonators and TLS, see Sec. 3.3 and 3.4, to define a many-particle circuit Hamiltonian model. The generic model Hamiltonian reads

$$\hat{H}(t) = \hat{H}_{\text{Fix.},\Sigma}(t) + \hat{H}_{\text{Tun.},\Sigma}(t) + \hat{H}_{\beta,\Sigma}(t) + \hat{H}_{\text{Res.},\Sigma} + \hat{H}_{\text{TLS},\Sigma} + \hat{V}_{\text{Int.}}, \quad (3.114)$$

where the individual terms are defined as

$$\hat{H}_{\text{Fix.},\Sigma}(t) = \sum_{i \in I} E_{C_i} (\hat{n}_i - n_{g,i}(t))^2 - E_{J_i} \cos(\hat{\varphi}_i), \quad (3.115a)$$

$$\begin{aligned} \hat{H}_{\text{Tun.},\Sigma}(t) = \sum_{j \in J} & (E_{C_j} (\hat{n}_j - n_{g,j}(t))^2 + (\frac{1}{2} - \beta_j) \dot{\varphi}_j(t) \hat{n}_j \\ & - E_{J_{l,j}} \cos(\hat{\varphi}_j + \beta_j \varphi_j(t)) \\ & - E_{J_{r,j}} \cos(\hat{\varphi}_j + (\beta_j - 1) \varphi_j(t))), \end{aligned} \quad (3.115b)$$

$$\hat{H}_{\text{Res.},\Sigma} = \sum_{k \in K} \omega_k^{(R)} \hat{a}_k^\dagger \hat{a}_k, \quad (3.115c)$$

$$\hat{H}_{\text{TLS},\Sigma} = \sum_{l \in L} \omega_l^{(T)} \hat{b}_l^\dagger \hat{b}_l, \quad (3.115d)$$

$$\begin{aligned}
\hat{V}_{\text{Int.}} = & \sum_{(i,i') \in I \times I'} G_{i,i'}^{(0)} \left(\hat{n}_i \otimes \hat{n}_{i'} \right) \\
& + \sum_{(j,i) \in J \times I} G_{j,i}^{(1)} \left(\hat{n}_j \otimes \hat{n}_i \right) \\
& + \sum_{(j,j') \in J \times J'} G_{j,j'}^{(2)} \left(\hat{n}_j \otimes \hat{n}_{j'} \right) \\
& + \sum_{(k,i) \in K \times I} G_{k,i}^{(3)} \left(\hat{a}_k + \hat{a}_k^\dagger \right) \otimes \hat{n}_i \\
& + \sum_{(k,j) \in K \times J} G_{k,j}^{(4)} \left(\hat{a}_k + \hat{a}_k^\dagger \right) \otimes \hat{n}_j \\
& + \sum_{(l,i) \in L \times I} G_{l,i}^{(5)} \left(\hat{b}_l + \hat{b}_l^\dagger \right) \otimes \hat{n}_i \\
& + \sum_{(l,j) \in L \times J} G_{l,j}^{(6)} \left(\hat{b}_l + \hat{b}_l^\dagger \right) \otimes \hat{n}_j \\
& + \sum_{(k,k') \in K \times K'} G_{k,k'}^{(7)} \left(\hat{a}_k + \hat{a}_k^\dagger \right) \otimes \left(\hat{a}_{k'} + \hat{a}_{k'}^\dagger \right) \\
& + \sum_{(l,k) \in L \times K} G_{l,k}^{(8)} \left(\hat{b}_l + \hat{b}_l^\dagger \right) \otimes \left(\hat{a}_k + \hat{a}_k^\dagger \right) \\
& + \sum_{(l,l') \in L \times L'} G_{l,l'}^{(9)} \left(\hat{b}_l + \hat{b}_l^\dagger \right) \otimes \left(\hat{b}_{l'} + \hat{b}_{l'}^\dagger \right).
\end{aligned} \tag{3.115e}$$

The full Hamiltonian $\hat{H}(t)$ describes a set of interacting fixed-frequency transmons, flux-tunable transmons, resonators and TLSs. Here $I \subseteq \mathbb{N}^0$ denotes the index set of the fixed-frequency transmons, $J \subseteq \mathbb{N}^0$ denotes the index set of the flux-tunable transmons, $K \subseteq \mathbb{N}^0$ denotes the index set of the resonators and $L \subseteq \mathbb{N}^0$ denotes the index set of the TLSs. We use the term $\hat{V}_{\text{Int.}}$ to model dipole-dipole interactions, see Refs. [Zan13; CGD92], between the different subsystems. The interaction term $\hat{V}_{\text{Int.}}$ has ten terms in total, i.e. each type of circuit element can interact with the remaining circuit element types. For simplicity and notational convenience we express the interaction operator as

$$\hat{V}_{\text{Int.}} = \sum_{n \in \{0, \dots, 9\}} \sum_{(i,j) \in I_n \times J_n} G_{i,j}^{(n)} \left(\hat{v}_{n,i} \otimes \hat{v}_{n,j} \right), \tag{3.116}$$

where the different $\hat{v}_{n,i}$, $\hat{v}_{n,j}$, I_n and J_n are defined in terms of the interaction operator given by Eq. (3.115)(e). In the following, we use the interaction strength constants $G_{i,j}^{(0-9)}$ to address the different terms.

The generic form of the Hamiltonian Eq. (3.114) allows us to model a wide range of physical scenarios. First, we can model different circuit architectures and quantum gates, see Ref. [Bla+21] for a review of different circuit architectures and gate strategies. Second, we are able to describe small circuits in a thermal bath environment of TLSs, see Ref. [Wil+20b] and Sec. 3.3.

In practice, we find that even if we are able to solve the TDSE for $\hat{H}(t)$, the work effort to set up simulations for these very different scenarios is immense, i.e. each physical scenario comes with its own special needs. Furthermore, the capabilities of the simulation code in terms of the number of circuit elements which can be simulated, are determined by the simulation algorithm we introduce in Chap. 4, the exact trajectories of $n_{g,i/j}(t)$ and $\varphi_j(t)$ and the system parameters we introduce later. This issue is further discussed in Chap. 5 and 6.

3.6 An effective Hamiltonian model for non-ideal gate-based transmon quantum computers

We can use the resonator model defined in Sec. 3.3 and the effective models for fixed-frequency and flux-tunable transmons defined in Sec. 3.4 to define a many-particle effective Hamiltonian model in the harmonic basis. Note that here we do not explicitly take into account the time-dependent harmonic basis which is used to formulate the effective flux-tunable transmon model. The generic model Hamiltonian reads

$$\hat{H}(t) = \hat{H}_{\text{Tra},\Sigma}(t) + \hat{H}_{\text{Res},\Sigma} + \hat{D}_{\text{Dri},\Sigma}(t) + \hat{\mathcal{D}}_{\text{Dri},\Sigma}(t) + \hat{W}_{\text{Int.}}(t). \quad (3.117)$$

The first term

$$\hat{H}_{\text{Tra},\Sigma}(t) = \sum_{i \in I} \omega_i^{(q)}(t) \hat{c}_i^\dagger \hat{c}_i + \frac{\alpha_i^{(q_0)}}{2} \left(\hat{c}_i^\dagger \hat{c}_i \left(\hat{c}_i^\dagger \hat{c}_i - \hat{I} \right) \right), \quad (3.118)$$

describes a collection of non-interacting transmon qubits, fixed frequency and/or flux tunable. Here $I \subseteq \mathbb{N}^0$ denotes an index set for the different transmons. If we choose $\omega_i^{(q)}(t) = \text{const.}$ for all times t and a certain transmon term $i \in I$, we describe a fixed-frequency transmon. Otherwise, we model a flux-tunable transmon, see Sec. 3.4.

As in Sec. 3.5, we use the second term

$$\hat{H}_{\text{Res},\Sigma} = \sum_{j \in J} \omega_j^{(R)} \hat{a}_j^\dagger \hat{a}_j, \quad (3.119)$$

to model a set of non-interacting resonator elements. Here $J \subseteq \mathbb{N}^0$ denotes an index set for the different resonator elements.

The third term

$$\hat{D}_{\text{Dri},\Sigma}(t) = \sum_{i \in I} \Omega_i(t) \left(\hat{c}_i^\dagger + \hat{c}_i \right), \quad (3.120)$$

is defined as the sum of individual charge driving terms which act on the different transmon subspaces. Here $\Omega_i(t) \propto n_{g,i}(t)$ denotes a real-valued function which allows us to model quantum gates.

The fourth term

$$\hat{\mathcal{D}}_{\text{Dri},\Sigma}(t) = \sum_{i \in I} -i \sqrt{\frac{\xi_i(t)}{2}} \dot{\varphi}_{\text{eff},i}(t) \left(\hat{c}_i^\dagger - \hat{c}_i \right) + \frac{i \dot{\xi}_i(t)}{4 \xi_i(t)} \left(\hat{c}_i^\dagger \hat{c}_i^\dagger - \hat{c}_i \hat{c}_i \right), \quad (3.121)$$

is another driving term which allows us to model flux-tunable transmons non adiabatically. The individual terms in Eq. (3.121) are proportional to the time derivative $\dot{\varphi}_i(t)$ of the external flux $\varphi_i(t)$ which one can use to control the system, see Eqs. (3.103) and (3.104). In the following we refer to the Hamiltonian Eq. (3.117) as the adiabatic effective Hamiltonian model when the term in Eq. (3.121) is neglected. Similarly, if we include the term we refer to the Hamiltonian Eq. (3.117) as the non-adiabatic effective Hamiltonian model. Additionally, we set $\dot{\varphi}_i(t) = 0$ for all fixed-frequency transmons i .

The fifth term describes the interactions between the different subsystems and reads

$$\begin{aligned} \hat{W}_{\text{Int.}}(t) = & \sum_{(i,i') \in I \times I'} g_{i,i'}^{(c,c)}(t) \left(\hat{c}_i^\dagger + \hat{c}_i \right) \otimes \left(\hat{c}_{i'}^\dagger + \hat{c}_{i'} \right) \\ & + \sum_{(j,j') \in J \times J'} g_{j,j'}^{(a,a)}(t) \left(\hat{a}_j^\dagger + \hat{a}_j \right) \otimes \left(\hat{a}_{j'}^\dagger + \hat{a}_{j'} \right) \\ & + \sum_{(j,i) \in J \times I} g_{j,i}^{(a,c)}(t) \left(\hat{a}_j^\dagger + \hat{a}_j \right) \otimes \left(\hat{c}_i^\dagger + \hat{c}_i \right), \end{aligned} \quad (3.122)$$

where $g_{i,j}^{(\cdot,\cdot)}(t)$ is a real-valued function which describes a time-dependent interaction strength. Here too, see Sec. 3.5, we model dipole-dipole interactions. In principle, we can choose the function $g_{i,j}^{(\cdot,\cdot)}(t)$ as we see fit. However, in this thesis, we restrict ourselves to the following three cases. The first case is defined as

$$g_{j,j'}^{(a,a)}(t) = G_{j,j'}, \quad (3.123)$$

where $G_{j,j'}$ is a constant real-valued parameter which determines the order of magnitude of the interaction strength. We use this constant function to

model the coupling between two resonator elements j and j' . The second case is defined as

$$g_{j,i}^{(a,c)}(t) = G_{j,i} \sqrt[4]{\frac{\tilde{E}_{J,i}(t)}{8\tilde{E}_{C,i}}}, \quad (3.124)$$

where $\tilde{E}_{J,i}(t)$ denotes a Josephson energy and $\tilde{E}_{C,i}$ refers to a capacitive energy. Here we model the coupling between a transmon i and a resonator j . The third case is defined as

$$g_{i,i'}^{(c,c)}(t) = G_{i,i'} \sqrt[4]{\frac{\tilde{E}_{J,i}(t)}{8\tilde{E}_{C,i}}} \sqrt[4]{\frac{\tilde{E}_{J,i'}(t)}{8\tilde{E}_{C,i'}}}. \quad (3.125)$$

Here we model the interactions between two transmons i and i' . The function $\tilde{E}_{J,i}(t)$ and the parameter $\tilde{E}_{C,i}$ are chosen as follows. Assuming we model a fixed-frequency transmon i with Josephson energy $E_{J,i}$ and capacitive energy $E_{C,i}$, we then set $\tilde{E}_{J,i}(t) = E_{J,i}$ and $\tilde{E}_{C,i} = E_{C,i}$. If we model a flux-tunable transmon i with effective Josephson energy $E_{J,\text{eff}}(t)$ and capacitive energy $E_{C,i}$, we set $\tilde{E}_{J,i}(t) = E_{J,\text{eff}}(t)$ and $\tilde{E}_{C,i} = E_{C,i}$. This effective interaction strength model is motivated by the work of Ref. [Koc+07].

3.7 Summary and conclusions

In this chapter we discussed the fundamental mathematical relations we use to describe electromagnetic systems in this thesis, see Sec. 3.1. Furthermore, we reviewed the circuit quantisation formalism. This formalism allows us to determine Hamiltonians for different electromagnetic systems, see Sec. 3.2. Additionally, we applied the formalism and derived circuit Hamiltonians for LC resonators as well as fixed-frequency and flux-tunable transmons, see Sec. 3.3 and 3.4. In Sec. 3.3, we also introduced a two-level system (TLS) model. Similarly, in Sec. 3.4, we also derived several effective Hamiltonians for fixed-frequency and flux-tunable transmons. These models can potentially mimic the dynamic behaviour of the associated circuit Hamiltonian models. Finally, see Sec. 3.5 and 3.6, we defined two generic models for non-ideal gate-based transmon quantum computers, i.e. these models are supposed to describe certain aspects of superconducting prototype gate-based quantum computers (PGQCs), see also Sec. 2.7 and 2.6. In Sec. 3.5 we define a generic circuit Hamiltonian model which describes a set of interacting fixed-frequency and flux-tunable transmons as well as LC resonators and TLSs. Here we model the interactions as dipole-dipole interactions. Similarly, in Sec. 3.6 we define a generic effective Hamiltonian which describes a

set of interacting fixed-frequency transmons, flux-tunable transmons and LC resonators. Here too, we model the interactions as dipole-dipole interactions.

The Hamiltonians we derive and/or define are a result of the assumptions we make, see Secs. 3.1-3.6. Consequently, the time evolution of a NIGQC modelled with the TDSE and a Hamiltonian is also a direct product of these assumptions. However, we can never be absolutely certain that the assumptions we make are sufficient or actually correct, when it comes to modelling a physical scenario. Consequently, we should always be sceptical about the models we use to describe the physical scenario of interest.

Real-time simulation algorithms for time-dependent Hamiltonians

The aim of this chapter is to provide a description of the simulation algorithms we employ to solve the time-dependent Schrödinger equation (TDSE) for the Hamiltonians Eqs. (3.114) and (3.117) as well as the extend simulation software we use to obtain most results in this thesis.

This chapter is structured as follows. In Sec. 4.1, we review the problems we face once we try to solve the TDSE numerically. Next, in Sec. 4.2 we provide the reader with a general discussion of the so-called product-formula algorithm, this is the algorithm we mainly use to solve the TDSE in this work. In Sec. 4.3 and 4.4 we discuss two second-order time-evolution operators which approximately solve the TDSE for the Hamiltonians Eqs. (3.114) and (3.117). Next, in Sec. 4.5, we discuss why simulations of the circuit Hamiltonian model in two alternative computational bases were abandoned. In Sec. 4.6, we discuss the update rules we use to implement the time-evolution operators we discuss in Sec. 4.3 and 4.4. In the end, in Sec. 4.7, we provide a high-level overview of the complete simulation software.

Since we solve the TDSE numerically, we always work with finite-dimensional Hilbert spaces. Therefore, in this chapter we always implicitly assume that operators are linear maps which act on a finite-dimensional Hilbert space. Furthermore, we use $\hbar = 1$ throughout this chapter.

4.1 Solving the time-dependent Schrödinger equation numerically

Unfortunately, we usually find that analytical solutions of the TDSE

$$i\partial_t |\Psi(t)\rangle = \hat{H}(t) |\Psi(t)\rangle, \quad (4.1)$$

where $\hat{H}(t)$ denotes an arbitrary model Hamiltonian which acts on the Hilbert space $\mathcal{H} \subseteq \mathbb{C}^{\dim}$, are rare and can usually only be obtained for simple model

Hamiltonians. Here we assume that no approximations are being made to simplify $\hat{H}(t)$. In particular, in most cases the problem becomes more difficult once we consider a composition of different subsystems which interact with one another. In this thesis, we are interested in exactly this case.

The formal solution of the TDSE reads

$$\hat{\mathcal{U}}(t, t_0) = \mathcal{T} \exp \left(-i \int_{t_0}^t \hat{H}(t') dt' \right), \quad (4.2)$$

where \mathcal{T} denotes the time-ordering symbol. Numerical solutions of the TDSE require a discretisation of the time domain, i.e. $t \rightarrow t_j = \tau j + t_0$. If we assume that $\hat{H}(t)$ is piecewise constant between the two time steps t_j and t_{j+1} , for a small enough time step difference τ , we might write

$$\hat{\mathcal{U}}(t, t_0) = \prod_{j=0}^{J-1} e^{-i\tau \hat{H}(t_j + \frac{\tau}{2})}. \quad (4.3)$$

This assumption enables us to reduce the problem to the implementation of a sequence of unitary operators

$$\hat{\mathcal{U}}(t_{j+1}, t_j) = e^{-i\tau \hat{H}(t_j + \frac{\tau}{2})}, \quad (4.4)$$

such that

$$|\Psi(t_{j+1})\rangle = \hat{\mathcal{U}}(t_{j+1}, t_j) |\Psi(t_j)\rangle. \quad (4.5)$$

There exist different algorithms which allow us to implement $\hat{\mathcal{U}}$ in a computer program. One popular approach is named full diagonalisation. If we assume that \hat{H} is represented by some matrix, in some unspecified basis, for some time step, we might use the QR algorithm, see Refs. [GV96; Fra61; Fra62], to diagonalise this matrix. The QR algorithm and others are implemented in various numerical libraries, e.g. the Linear Algebra Package (LAPACK), see Ref. [And+] or Intel's Math Kernel Library (MKL), see Ref. [Cor]. If we supply the library subroutines with a matrix representation H of \hat{H} , we can potentially obtain matrices for the basis transformation W and the diagonal matrix λ such that $H = W\lambda W^\dagger$. Assuming we can repeatedly diagonalise $\hat{H}(t_j + \frac{\tau}{2})$ for every time step t_j , we can solve our problem by implementing the time-evolution operator

$$\hat{\mathcal{U}}(t_{j+1}, t_j) = \hat{W} \left(t_j + \frac{\tau}{2} \right) e^{-i\tau \hat{\lambda}(t_j + \frac{\tau}{2})} \hat{W}^\dagger \left(t_j + \frac{\tau}{2} \right). \quad (4.6)$$

The advantage of this method is that it is conceptually very simple and a great deal of complexity is hidden inside the library subroutines. The

disadvantage of this method is that the computational work, in terms of time $\mathcal{O}^{\text{Time}}(\text{dim}^3)$ and space $\mathcal{O}^{\text{Space}}(\text{dim}^2)$ complexity, see Sec. 2.4, grows with the third and second power in terms of the dimensionality of the system, see Ref. [GV96]. Since the dimensionality itself grows exponentially with the number of subsystems added to the quantum-theoretical model, see Sec. 2.1, we find that this approach and other approaches whose time and space complexity show a non-linear scaling in terms of dimensionality, are not applicable to the problem at hand. Consequently, we would prefer to implement \hat{U} with an algorithm whose time and space complexity scale linearly in terms of dimensionality. Obviously, this does not solve the exponential scaling problem which is an intrinsic part of the quantum-theoretical formalism. However, assuming we can find an algorithm which exhibits sufficient concurrency, i.e. independent computational tasks which can be parallelised, we might be able to remedy this issue by using more computational resources, see Ref. [HW10].

Methods which might allow us to solve the problem at hand in a reasonable time and with reasonable memory requirements are the Chebyshev polynomial algorithm, a Lanczos type algorithm or the product-formula algorithm. Reference [RM06] provides an overview of all methods mentioned. The product-formula algorithm can often be implemented by only consuming computational resources which grow linearly with the dimension of the Hilbert space. Furthermore, usually we are able to create sufficient concurrency by making use of the freedoms the algorithm provides us with. In this thesis, we use the product-formula algorithm to solve the TDSE for the Hamiltonian models defined in Sec. 3.5 and 3.6.

4.2 The product-formula algorithm

In this section, we discuss the product-formula algorithm, see Refs. [DD83; Rae87], in general terms. Therefore, we assume that the problem at hand can be described by an arbitrary model Hamiltonian \hat{H} , i.e. we assume that this operator is Hermitian and the Hilbert space in question is $\mathcal{H}^{\text{dim}} \subseteq \mathbb{C}^{\text{dim}}$. Since we intend to implement $\hat{U}(t_{j+1}, t_j)$ given by Eq. (4.4) for the discretised time steps t_j , we can assume that this operator is time independent, i.e. this operator represents the piecewise constant Hamiltonian operator between the time steps t_{j+1} and t_j .

In the beginning, we assume that there exists an additive decomposition

$$\hat{H} = \sum_{m=0}^{M-1} \hat{A}_m, \quad (4.7)$$

of the model Hamiltonians. Here $M \in \mathbb{N}^0$ and the operators \hat{A}_m are assumed to be Hermitian. Note that we always implicitly assume that operators act on finite-dimensional Hilbert spaces. Consequently, we find that the different \hat{A}_m are bounded operators. These assumptions are sufficient to make use of the so-called generalised (Trotter) decomposition formula

$$e^{\hat{H}} = \lim_{n \rightarrow \infty} \left(\prod_{m=0}^{M-1} e^{\frac{\hat{A}_m}{n}} \right)^n, \quad (4.8)$$

see Refs. [Suz85; SMK77]. We have in mind to systematically apply this formula to the time-evolution operator $\hat{\mathcal{U}}$, for finite n and a given operator decomposition \hat{A}_m . In this way, we can shift the problem to the implementation of a product of operators $e^{-ia\tau\hat{A}_m}$, where a is some real-valued constant. Since the different \hat{A}_m are Hermitian, we find that the resulting time evolution is unitary, i.e. norm preserving. This is the general idea of the product-formula algorithm.

In practice, this raises two questions. First, in order to obtain efficient simulation code, how should we choose the operator decomposition \hat{A}_m ? This question can only be answered properly once we have fixed the simulation basis and the model Hamiltonian, i.e. this question is very problem specific. We discuss this issue in the following sections. The second question which we have to address is, how valid is the approximation once we consider a finite n , i.e. what is the numerical precision of this approximation. This question can be answered for the local error if we construct the approximations in a specific way.

For the time-evolution operator

$$\hat{\mathcal{U}} = e^{-i\tau\hat{H}}, \quad (4.9)$$

the first-order approximant reads

$$\hat{\mathcal{U}}_1(\tau) = \prod_{m=0}^{M-1} e^{-i\tau\hat{A}_m}. \quad (4.10)$$

We can use the first-order approximation $\hat{\mathcal{U}}_1$ to construct certain higher-order approximations, see Ref. [RM06]. The second-order approximation can be expressed as

$$\hat{\mathcal{U}}_2(\tau) = \hat{\mathcal{U}}_1^\dagger\left(-\frac{\tau}{2}\right)\hat{\mathcal{U}}_1\left(\frac{\tau}{2}\right). \quad (4.11)$$

Similarly, the fourth-order approximation can be expressed in terms of the second-order approximation

$$\hat{\mathcal{U}}_4(\tau) = \hat{\mathcal{U}}_2(a\tau)\hat{\mathcal{U}}_2(a\tau)\hat{\mathcal{U}}_2((1-4a)\tau)\hat{\mathcal{U}}_2(a\tau)\hat{\mathcal{U}}_2(a\tau), \quad (4.12)$$

where $a = 1/(4 - 4^{\frac{1}{3}})$. For the first-, second- and fourth-order approximation $p \in \{1, 2, 4\}$ there exists a common bound

$$\left\| \hat{\mathcal{U}}(\tau) - \hat{\mathcal{U}}_p(\tau) \right\| \leq b_p \tau^{p+1}, \quad (4.13)$$

for the local error, see Ref. [Rae87]. Here the constants b_p are real valued and

$$\left\| \hat{X} \right\| = \sup_{\|\psi\|_2=1} \left(\left\| \hat{X} |\psi\rangle \right\|_2 \right), \quad (4.14)$$

denotes the operator norm for the operator \hat{X} which is induced by the Euclidean norm $\|\psi\|_2 = \sqrt{\langle \psi | \psi \rangle}$.

Constructing a simulation algorithm in this way has several benefits. First, since we require a Hermitian operator decomposition \hat{A}_m , we find that the time evolution is norm preserving, i.e. unconditionally stable. Consequently, we find that the approximation does not affect this general property of the model. Second, we find that some higher-order approximations can be implemented recursively. Third, the local error can be controlled by choosing an appropriate τ in combination with a particular higher-order approximation. Last but not least, since the \hat{A}_m are chosen by ourselves, we are often able to implement the operators $e^{-i\tau\hat{A}_m}$ with computational resources that only scale linearly with the dimension of the system. These features make the product-formula algorithm a suitable candidate for simulating the real-time dynamics of a quantum system.

4.3 The time-evolution operator for the effective Hamiltonian model

In Sec. 4.2, we discussed the product-formula algorithm. This algorithm provides us with the recipe to determine a unitary operator $\hat{\mathcal{U}}$ that approximately solves the TDSE for a given Hamiltonian operator \hat{H} . In this section, we use this recipe to determine the operator $\hat{\mathcal{U}}^{\text{HB}}$ for the Hamiltonian Eq. (3.117). For the remainder of this chapter we use the variable t to denote the discretised time steps $t_j + \tau/2$. This is done for simplicity and notational convenience. In the following, we use the states

$$|\mathbf{z}_J, \mathbf{z}_I\rangle = \bigotimes_{j \in J} |\psi^{(z_j)}\rangle \bigotimes_{i \in I} |\psi^{(z_i)}\rangle, \quad (4.15)$$

as basis states for the state vector $|\Psi(t)\rangle$, see Eq. (4.1). Here $\mathbf{z}_J \in \mathcal{I}_{n_J}^{[J]}$, $\mathbf{z}_I \in \mathcal{I}_{n_I}^{[I]}$ and $\mathcal{I}_n^N = \{0, 1, \dots, n-1\}^N$ denotes a set of N-tuples, where the

tuple entries can take on values from 0 to $n - 1$. Furthermore, the state vectors $|\psi^{(z_{i/j})}\rangle$ denote the basis states of the harmonic oscillator given by Eqs. (3.33) and (3.97). We use the indices i and j to differentiate between transmons and resonators, respectively. The states in Eq. (4.15) form a basis

$$\mathcal{J}'_{\mathbf{n}} = \{|\mathbf{z}\rangle \in \mathcal{H}'_{\text{Sim.}} | \exists! \mathbf{z} \in \mathcal{I}'_{\mathbf{n}} : |\mathbf{z}\rangle = |\mathbf{z}_J, \mathbf{z}_I\rangle\}, \quad (4.16)$$

for the Hilbert space

$$\mathcal{H}'_{\text{Sim.}} = \bigotimes_{j \in J} \mathcal{H}_{\text{Res.}, j} \bigotimes_{i \in I} \mathcal{H}_{\text{Tra.}, i}, \quad (4.17)$$

and we find

$$\dim(\mathcal{H}'_{\text{Sim.}}) = n_J^{|J|} n_I^{|I|}, \quad (4.18)$$

for the dimensionality of the space we use to perform the simulations. The index set $\mathcal{I}'_{\mathbf{n}}$ is defined as

$$\mathcal{I}'_{\mathbf{n}} = \mathcal{I}_{n_J}^{|J|} \times \mathcal{I}_{n_I}^{|I|}, \quad (4.19)$$

where \times denotes the Cartesian product and $\mathbf{n} = (n_J, n_I)$. Here n_J and n_I denote the number of basis states we use to model the resonators and transmons, respectively.

One can show, see Sec. 2.1 and Ref. [NC11], that the product states given by Eq. (4.15) form a basis of the Hilbert space $\mathcal{H}'_{\text{Sim.}}$. We name this basis the bare harmonic basis or harmonic basis for short. Note that the notation we use for the simulation basis overloads the notational framework we use in this thesis. In Chap. 1, we used the states $|\mathbf{z}\rangle \in \mathcal{C}^N$ as the basis states of the state vector for the IGQC. In this section, we use $|\mathbf{z}\rangle \in \mathcal{J}'_{\mathbf{n}}$ as the simulation basis states of the NIGQC. Both states are elements of the same mathematical structure, namely a finite-dimensional Hilbert space. However, we usually find that the dimensionality is different. Furthermore, in Sec. 4.4, we use the states $|\mathbf{z}\rangle$ as basis states of another Hilbert space. For the remainder of this thesis, we always make clear which basis is meant if we refer to the states $|\mathbf{z}\rangle$.

As discussed in Sec. 4.2, the first step in finding $\hat{\mathcal{U}}^{\text{HB}}$ is choosing a Hermitian decomposition of the model Hamiltonian \hat{H} . Our choice for this decomposition is

$$\hat{H}(t) = \hat{H}_0(t) + \hat{V}_0(t) + \hat{V}_1(t) + \hat{V}_2(t). \quad (4.20)$$

The first term

$$\hat{H}_0(t) = \hat{H}_{\text{Tra.}, \Sigma}(t) + \hat{H}_{\text{Res.}, \Sigma}, \quad (4.21)$$

is defined as the sum of terms which describe non-interacting transmon and resonator elements. The second term

$$\hat{V}_0(t) = \sum_{i \in I} q_i(t) i \left(\hat{c}_i^\dagger - \hat{c}_i \right), \quad (4.22)$$

where $q_i(t) = -\sqrt{\xi_i(t)}\dot{\varphi}_{\text{eff},i}(t)/\sqrt{2}$ for all $i \in I$, is given by the sum of the first summand of the non-adiabatic driving term in Eq. (3.102). Similarly, the third term

$$\hat{V}_1(t) = \sum_{i \in I} s_i(t) i \left(\hat{c}_i^\dagger \hat{c}_i^\dagger - \hat{c}_i \hat{c}_i \right), \quad (4.23)$$

where $s_i(t) = \dot{\xi}_i(t)/(4\xi_i(t))$ for all $i \in I$, consists of the sum of the second summand in Eq. (3.102). The fourth term

$$\hat{V}_2(t) = \hat{D}_{\text{Dri},\Sigma}(t) + \hat{W}_{\text{Int.}}(t), \quad (4.24)$$

is defined as the sum of the charge drive term $\hat{D}_{\text{Dri},\Sigma}(t)$ and the dipole-dipole interaction term $\hat{W}_{\text{Int.}}(t)$. Note that this decomposition is motivated by the work presented in Ref. [Wil20a].

In the following, we express the different decomposition terms as

$$\hat{H}_0(t) = \sum_{i \in I} \hat{\lambda}_{\text{Tra},i}(t) + \sum_{j \in J} \hat{\lambda}_{\text{Res},j}, \quad (4.25a)$$

$$\hat{V}_0(t) = \hat{Q} \hat{\lambda}_Q(t) \hat{Q}^\dagger, \quad (4.25b)$$

$$\hat{V}_1(t) = \hat{S} \hat{\lambda}_S(t) \hat{S}^\dagger, \quad (4.25c)$$

$$\hat{V}_2(t) = \hat{T} \hat{\lambda}_T(t) \hat{T}^\dagger. \quad (4.25d)$$

The operator $\hat{H}_0(t)$ is diagonal in the basis $\mathcal{J}_{\mathbf{n}}$. The operators $\hat{\lambda}_{\text{Tra},i}(t)$ and $\hat{\lambda}_{\text{Res},j}$ simply contain the eigenvalues of the transmon and resonator terms, see Eqs. (3.110) and (3.34), respectively. Furthermore, the operators $\hat{V}_0(t)$, $\hat{V}_1(t)$ and $\hat{V}_2(t)$ in Eq. (4.25)(b-d) are defined in terms of the basis transformations

$$\hat{Q} = \bigotimes_{i \in I} \hat{Q}_{c_i}, \quad (4.26a)$$

$$\hat{S} = \bigotimes_{i \in I} \hat{S}_{c_i}, \quad (4.26b)$$

$$\hat{T} = \bigotimes_{j \in J} \hat{T}_{a_j} \bigotimes_{i \in I} \hat{T}_{c_i}, \quad (4.26c)$$

and the operators $\hat{\lambda}_Q(t)$, $\hat{\lambda}_S(t)$ and $\hat{\lambda}_T(t)$ are diagonal in the transformed bases $\{|\mathbf{w}_{Q/S/T}\rangle\}$, which can be obtained by making use of \hat{Q} , \hat{S} and \hat{T} . Note that $\hat{\lambda}_Q(t)$, $\hat{\lambda}_S(t)$ and $\hat{\lambda}_T(t)$ are usually non-diagonal in the basis $\mathcal{J}'_{\mathbf{n}}$. The different basis transformations in Eq. (4.26)(a-c) are themselves defined in terms of basis transformations \hat{Q}_{c_i} , \hat{S}_{c_i} , \hat{T}_{c_i} and \hat{T}_{a_j} which act on the individual transmon and resonator subspaces

$$\hat{Q}_{c_i} \hat{\lambda}_{Q_{c_i}} \hat{Q}_{c_i}^\dagger = i \left(\hat{c}_i^\dagger - \hat{c}_i \right), \quad (4.27a)$$

$$\hat{S}_{c_i} \hat{\lambda}_{S_{c_i}} \hat{S}_{c_i}^\dagger = i \left(\hat{c}_i^\dagger \hat{c}_i^\dagger - \hat{c}_i \hat{c}_i \right), \quad (4.27b)$$

$$\hat{T}_{c_i} \hat{\lambda}_{T_{c_i}} \hat{T}_{c_i}^\dagger = \left(\hat{c}_i^\dagger + \hat{c}_i \right), \quad (4.27c)$$

$$\hat{T}_{a_j} \hat{\lambda}_{T_{a_j}} \hat{T}_{a_j}^\dagger = \left(\hat{a}_j + \hat{a}_j^\dagger \right). \quad (4.27d)$$

We emphasise that if we represent the different operators \hat{Q}_{c_i} , \hat{S}_{c_i} , \hat{T}_{c_i} and \hat{T}_{a_j} numerically as matrices, the diagonalisation algorithm yields the same result for all i and j . Consequently, if we implement the algorithm and/or the corresponding time-evolution operator in a program, we only have to make use of a very small amount of memory to store the three matrices. This is one of the reasons why we express the problem in this way. Another reason for choosing this decomposition follows right away.

In the transformed bases $\{|\mathbf{w}_{Q/S/T}\rangle\}$, see basis transformations \hat{Q} , \hat{S} and \hat{T} , we can express the different diagonal operators as

$$\hat{\lambda}_Q(t) = \sum_{j \in J} q_j(t) \hat{\lambda}_{Q_{c_i}}, \quad (4.28)$$

$$\hat{\lambda}_S(t) = \sum_{j \in J} s_j(t) \hat{\lambda}_{S_{c_i}}, \quad (4.29)$$

$$\begin{aligned} \hat{\lambda}_T(t) &= \sum_{i \in I} \Omega_i(t) \hat{\lambda}_{T_{c_i}} \\ &+ \sum_{(i,i') \in I \times I'} g_{i,i'}^{(c,c)}(t) \left(\hat{\lambda}_{T_{c_i}} \otimes \hat{\lambda}_{T_{c_{i'}}} \right) \\ &+ \sum_{(j,j') \in J \times J'} g_{j,j'}^{(a,a)}(t) \left(\hat{\lambda}_{T_{a_j}} \otimes \hat{\lambda}_{T_{a_{j'}}} \right) \\ &+ \sum_{(j,i) \in J \times I} g_{j,i}^{(a,c)}(t) \left(\hat{\lambda}_{T_{a_j}} \otimes \hat{\lambda}_{T_{c_i}} \right). \end{aligned} \quad (4.30)$$

Therefore, we can lump the time dependencies into the diagonal operators completely.

The resulting second-order time-evolution operator for the Hermitian decomposition given by Eq. (4.20) reads

$$\begin{aligned} \hat{\mathcal{U}}^{\text{HB}} &= e^{-\frac{\tau}{2} \hat{H}_0(t)} \hat{S} e^{-\frac{\tau}{2} \hat{\lambda}_S(t)} \hat{S}^\dagger \hat{Q} e^{-\frac{\tau}{2} \hat{\lambda}_Q(t)} \hat{Q}^\dagger \hat{T} e^{-i\tau \hat{\lambda}_T(t)} \hat{T}^\dagger \\ &\hat{Q} e^{-\frac{\tau}{2} \hat{\lambda}_Q(t)} \hat{Q}^\dagger \hat{S} e^{-\frac{\tau}{2} \hat{\lambda}_S(t)} \hat{S}^\dagger e^{-\frac{\tau}{2} \hat{H}_0(t)}. \end{aligned} \quad (4.31)$$

In order to reduce the number of basis transformations which need to be implemented in a computer program, we introduce the operators $\hat{R} = \hat{S}^\dagger \hat{Q}$

and $\hat{U} = \hat{Q}^\dagger \hat{T}$ such that

$$\begin{aligned} \hat{\mathcal{U}}^{\text{HB}} = & e^{-\frac{\tau}{2}\hat{H}_0(t)} \hat{S} e^{-\frac{\tau}{2}\hat{\lambda}_S(t)} \hat{R} e^{-\frac{\tau}{2}\hat{\lambda}_Q(t)} \hat{U} e^{-i\tau\hat{\lambda}_T(t)} \hat{U}^\dagger \\ & e^{-\frac{\tau}{2}\hat{\lambda}_Q(t)} \hat{R}^\dagger e^{-\frac{\tau}{2}\hat{\lambda}_S(t)} \hat{S}^\dagger e^{-\frac{\tau}{2}\hat{H}_0(t)}. \end{aligned} \quad (4.32)$$

For the adiabatic effective Hamiltonian Eq. (3.117) we can derive the much simpler time-evolution operator

$$\hat{\mathcal{U}}_{\text{adia.}}^{\text{HB}} = e^{-\frac{\tau}{2}\hat{H}_0(t)} \hat{T} e^{-\tau\hat{\lambda}_T(t)} \hat{T}^\dagger e^{-\frac{\tau}{2}\hat{H}_0(t)}, \quad (4.33)$$

where we assume $\hat{V}_0(t) = 0$ and $\hat{V}_1(t) = 0$ for all times t or $\dot{\varphi}_i(t) \rightarrow 0$ for all i , see Eqs. (3.103) and (3.104).

Consequently, we find that we have to implement two types of diagonal operators and one type of non-diagonal operators for every iteration step t . The algorithmic update rules that allow us to implement these operators are subject of Sec. 4.6.

4.4 The time-evolution operator for the circuit Hamiltonian model

In Sec. 4.2, we discuss how to solve the TDSE numerically by implementing a unitary time-evolution operator $\hat{\mathcal{U}}$, for an arbitrary model Hamiltonian \hat{H} . In Sec. 4.3, we make use of this recipe and derive a time-evolution operator for the effective Hamiltonian Eq. (3.117). In this section, we do the same for Hamiltonian Eq. (3.114). Note that this model Hamiltonian is more complex than the one we discussed in Sec. 4.3. In the following, we restrict the discussion to the case where each type of subsystem, i.e. transmons, resonators and TLSs, is modelled with a fixed number of basis states. The simulation code allows us more freedom, i.e. we can choose two different values for the number of basis states we use to model the subsystem in question. A detailed discussion of this part of the simulation code goes beyond the scope of this section. Here, we intend to understand the structure of the second-order time-evolution operator as clearly as possible. Consequently, we omit some details regarding the actual simulation code in this section.

In the following, we use the product states

$$|\mathbf{z}_L, \mathbf{z}_K, \mathbf{z}_J, \mathbf{z}_I\rangle = \bigotimes_{l \in L} |\psi^{(z_l)}\rangle \bigotimes_{k \in K} |\psi^{(z_k)}\rangle \bigotimes_{j \in J} |\phi^{(z_j)}\rangle \bigotimes_{i \in I} |\phi^{(z_i)}\rangle, \quad (4.34)$$

as basis states for the state $|\Psi(t)\rangle$, see Eq. (4.1). Here $\mathbf{z}_L \in \mathcal{I}_{n_L}^{[L]}$, $\mathbf{z}_K \in \mathcal{I}_{n_K}^{[K]}$, $\mathbf{z}_J \in \mathcal{I}_{n_J}^{[J]}$, $\mathbf{z}_I \in \mathcal{I}_{n_I}^{[I]}$. Consequently, n_L , n_K , n_J and n_I denote the

numbers of basis states we use to model the dynamics of the system. We discuss the systems associated with the harmonic basis states $|\psi^{(z_{l/k})}\rangle$ and the transmon basis states $|\phi^{(z_{i/j})}\rangle$ in Sec. 3.3 and 3.4, respectively. Note that we differentiate between the fixed-frequency and flux-tunable transmon states by means of the indices i and j . The simulation basis can be expressed as

$$\mathcal{J}_{\mathbf{n}} = \{|\mathbf{z}\rangle \in \mathcal{H}_{\text{Sim.}} | \exists! \mathbf{z} \in \mathcal{I}_{\mathbf{n}}: |\mathbf{z}\rangle = |\mathbf{z}_L, \mathbf{z}_K, \mathbf{z}_J, \mathbf{z}_I\rangle\}, \quad (4.35)$$

where

$$\mathcal{H}_{\text{Sim.}} = \bigotimes_{l \in L} \mathcal{H}_{\text{TLS}, l} \otimes \bigotimes_{k \in K} \mathcal{H}_{\text{Res.}, k} \otimes \bigotimes_{j \in J} \mathcal{H}_{\text{Tun.}, j} \otimes \bigotimes_{i \in I} \mathcal{H}_{\text{Fix.}, i}, \quad (4.36)$$

denotes a Hilbert space with dimensionality

$$\dim(\mathcal{H}_{\text{Sim.}}) = n_L^{|L|} n_K^{|K|} n_J^{|J|} n_I^{|I|}, \quad (4.37)$$

and

$$\mathcal{I}_{\mathbf{n}} = \mathcal{I}_{n_L}^{|L|} \times \mathcal{I}_{n_K}^{|K|} \times \mathcal{I}_{n_J}^{|J|} \times \mathcal{I}_{n_I}^{|I|}, \quad (4.38)$$

is another index set, which is specified by the four tuple $\mathbf{n} = (n_K, n_L, n_J, n_I)$. Here, we denote $\mathcal{J}_{\mathbf{n}}$ as the transmon bare basis or transmon basis for short. As discussed in Sec. 2.1 and Ref. [NC11], one can show that the product states given by Eq. (4.34) form a basis of the space $\mathcal{H}_{\text{Sim.}}$. As mentioned before, in this thesis, see Chap. 2 and Sec. 4.3, we use the ket vectors $|\mathbf{z}\rangle$ to refer to basis states which belong to different bases. This has actually an advantage once we compare different models with one another, see Chap. 5.

Since all simulations in this thesis are performed by imposing the irrotational constraint $\beta_j = 1/2$, see Ref. [YSK19] and Eq. (3.60), on all flux-tunable transmons $j \in J$, we discuss the second-order time-evolution operator for this special case only. Furthermore, the notation we use for the circuit Hamiltonian in Eq. (3.114) is already quite cumbersome and this choice simplifies the notation to some extent. We begin the discussion of the second-order time-evolution operator $\hat{\mathcal{U}}^{\text{TB}}$ by recasting Hamiltonian Eq. (3.114) into the form

$$\hat{H}(t) = \hat{H}_0 + \hat{W}_0(t) + \hat{W}_1(t) + \hat{W}_2(t), \quad (4.39)$$

where the individual terms are defined as

$$\hat{H}_0 = \hat{H}_{\text{Fix.}, \Sigma}(0) + \hat{H}_{\text{Tun.}, \Sigma}(0) + \hat{H}_{\text{Res.}, \Sigma} + \hat{H}_{\text{TLS}, \Sigma}, \quad (4.40a)$$

$$\hat{W}_0(t) = \sum_{j \in J} (E_{J_{l,j}} + E_{J_{r,j}}) \left(\cos\left(\frac{\varphi_{0,j}}{2}\right) - \cos\left(\frac{\varphi_j(t)}{2}\right) \right) \cos(\hat{\varphi}_j), \quad (4.40b)$$

$$\hat{W}_1(t) = \sum_{j \in J} (E_{J_{l,j}} - E_{J_{r,j}}) \left(\sin\left(\frac{\varphi_{0,j}}{2}\right) - \sin\left(\frac{\varphi_j(t)}{2}\right) \right) \sin(\hat{\varphi}_j), \quad (4.40c)$$

$$\hat{W}_2(t) = \sum_{i \in I} -2E_{C,i}n_{g,i}(t)\hat{n}_i + \sum_{j \in J} -2E_{C,j}n_{g,j}(t)\hat{n}_j + \hat{V}_{\text{Int}}. \quad (4.40d)$$

The terms $\hat{H}_{\text{Fix},\Sigma}(0)$ and $\hat{H}_{\text{Tun},\Sigma}(0)$ are the fixed-frequency and flux-tunable transmon terms at time $t = 0$, see Eq. (3.114). These terms are specified by the charge-offset values $n_{0,i/j} = n_{g,i/j}(0)$ and the flux-offset values $\varphi_{0,j} = \varphi_j(0)$. Note that we neglect all terms which only contribute a global phase factor to the time evolution.

We use the form given by Eq. (4.39) as a product-formula decomposition, see Sec. 4.2. In the transmon basis, we express these terms as

$$\hat{H}_0 = \sum_{i \in I} \hat{\lambda}_{\text{Fix},i} + \sum_{j \in J} \hat{\lambda}_{\text{Tun},j} + \sum_{k \in K} \hat{\lambda}_{\text{Res},k} + \sum_{l \in L} \hat{\lambda}_{\text{TLS},l}, \quad (4.41a)$$

$$\hat{W}_0(t) = \hat{Q}\hat{\lambda}_Q(t)\hat{Q}^\dagger, \quad (4.41b)$$

$$\hat{W}_1(t) = \hat{S}\hat{\lambda}_S(t)\hat{S}^\dagger, \quad (4.41c)$$

$$\hat{W}_2(t) = \hat{T}\hat{\lambda}_T(t)\hat{T}^\dagger. \quad (4.41d)$$

The operators $\hat{\lambda}_{\text{Fix},i}$, $\hat{\lambda}_{\text{Tun},j}$, $\hat{\lambda}_{\text{Res},k}$ and $\hat{\lambda}_{\text{TLS},l}$ are diagonal in the transmon basis \mathcal{J}_n . Furthermore, the operators \hat{Q} , \hat{S} and \hat{T} denote basis transformations

$$\hat{Q} = \bigotimes_{j \in J} \hat{Q}_{\varphi_j}, \quad (4.42a)$$

$$\hat{S} = \bigotimes_{j \in J} \hat{S}_{\varphi_j}, \quad (4.42b)$$

$$\hat{T} = \bigotimes_{l \in L} \hat{T}_{b_l} \bigotimes_{k \in K} \hat{T}_{a_k} \bigotimes_{j \in J} \hat{T}_{n_j} \bigotimes_{i \in I} \hat{T}_{n_i}, \quad (4.42c)$$

which are defined in terms of another set of basis transformations

$$\hat{Q}_{\varphi_j} \hat{\lambda}_{Q_{\varphi_j}} \hat{Q}_{\varphi_j}^\dagger = \cos \hat{\varphi}_j, \quad (4.43a)$$

$$\hat{S}_{\varphi_j} \hat{\lambda}_{S_{\varphi_j}} \hat{S}_{\varphi_j}^\dagger = \sin \hat{\varphi}_j, \quad (4.43b)$$

$$\hat{T}_{n_{i/j}} \hat{\lambda}_{T_{n_{i/j}}} \hat{T}_{n_{i/j}}^\dagger = \hat{n}_{i/j}, \quad (4.43c)$$

$$\hat{T}_{a_k} \hat{\lambda}_{T_{a_k}} \hat{T}_{a_k}^\dagger = (\hat{a}_k + \hat{a}_k^\dagger), \quad (4.43d)$$

$$\hat{T}_{b_l} \hat{\lambda}_{T_{b_l}} \hat{T}_{b_l}^\dagger = (\hat{b}_l + \hat{b}_l^\dagger), \quad (4.43e)$$

which act on the subspaces of the individual subsystems. These transformations allow us to express the operators $\hat{\lambda}_Q(t)$, $\hat{\lambda}_S(t)$ and $\hat{\lambda}_T(t)$ as diagonal operators

$$\hat{\lambda}_Q(t) = \sum_{j \in J} (E_{J_{i,j}} + E_{J_{r,j}}) \left(\cos\left(\frac{\varphi_{0,j}}{2}\right) - \cos\left(\frac{\varphi_j(t)}{2}\right) \right) \hat{\lambda}_{Q_{\varphi_j}}, \quad (4.44a)$$

$$\hat{\lambda}_S(t) = \sum_{j \in J} (E_{J_{l,j}} - E_{J_{r,j}}) \left(\sin\left(\frac{\varphi_{0,j}}{2}\right) - \sin\left(\frac{\varphi_j(t)}{2}\right) \right) \hat{\lambda}_{S_{\varphi_j}}, \quad (4.44b)$$

$$\begin{aligned} \hat{\lambda}_T(t) = & \sum_{i \in I} -2E_{C_i} n_{g,i}(t) \hat{\lambda}_{T_{n_i}} + \sum_{j \in J} -2E_{C_j} n_{g,j}(t) \hat{\lambda}_{T_{n_j}} \\ & + \sum_{n \in \{0, \dots, 9\}} \sum_{(i,j) \in I_n \times J_n} G_{i,j}^{(n)} \left(\hat{\lambda}_{T_{v_{n,i}}} \otimes \hat{\lambda}_{T_{v_{n,j}}} \right), \end{aligned} \quad (4.44c)$$

in their respective basis, see the labels Q , S and T . Here, we implicitly defined a function which maps the different interaction operators $\hat{v}_{n,i}$, see Eq. (3.116), to the correct operators $\hat{\lambda}_{T_{v_{n,i}}}$. This formulation of the problem allows us to separate the time dependencies from the operators, in the individual summation terms.

We intend to implement a high-performance simulation code. Therefore we restrict the number of allowed trajectories of $\hat{H}(t)$. To this end, we require that in an arbitrary time interval $t \in [t_0, T]$ either the flux variables $\varphi_j(t)$ or the charge variables $n_{g,i/j}(t)$ vary but not both. This assumption allows us to derive the time-evolution operator $\hat{\mathcal{U}}^{\text{TB}}(t)$ by considering two special cases. We label these cases with Roman numerals I and II.

If we consider time intervals $t \in [t_0, T]$, where $\varphi_j(t) = \varphi_{0,j}$ for all $j \in J$, we find $\hat{W}_0(t) = 0$ and $\hat{W}_1(t) = 0$. Consequently, the Hamiltonian Eq. (3.114) simplifies considerably. The corresponding second-order time-evolution operator reads

$$\hat{\mathcal{U}}^{\text{I}}(t) = e^{-i\frac{\tau}{2}\hat{H}_0} \hat{T} e^{-i\tau\hat{\lambda}_T(t)} \hat{T}^\dagger e^{-i\frac{\tau}{2}\hat{H}_0}. \quad (4.45)$$

The operator we implement here is structurally equivalent to the one discussed in Ref. [Wil20a].

If we consider time intervals $t \in [t_0, T]$, where $\varphi_j(t) \neq \varphi(0)$ for some $j \in J$ and $n_{g,i}(t) = 0$ for all $i \in I$ and $n_{g,j}(t) = 0$ for all $j \in J$, we find that the second-order time-evolution operator can be expressed as

$$\hat{\mathcal{U}}^{\text{II}}(t) = \hat{Q} e^{-i\frac{\tau}{2}\hat{\lambda}_Q(t)} \hat{Q}^\dagger \hat{S} e^{-i\frac{\tau}{2}\hat{\lambda}_S(t)} \hat{S}^\dagger \hat{\mathcal{U}}^{\text{I}}(t_0) \hat{S} e^{-i\frac{\tau}{2}\hat{\lambda}_S(t)} \hat{S}^\dagger \hat{Q} e^{-i\frac{\tau}{2}\hat{\lambda}_Q(t)} \hat{Q}^\dagger. \quad (4.46)$$

We can simplify $\hat{\mathcal{U}}^{\text{II}}(t)$, in the sense that we aim to reduce the number of operators which have to be implemented in the computer program. To this end, we define an additional unitary operator $\hat{R} = \hat{Q}^\dagger \hat{S}$ such that the operator $\hat{\mathcal{U}}^{\text{II}}(t)$ reads

$$\hat{\mathcal{U}}^{\text{II}}(t) = \hat{Q} e^{-i\frac{\tau}{2}\hat{\lambda}_Q(t)} \hat{R} e^{-i\frac{\tau}{2}\hat{\lambda}_S(t)} \hat{S}^\dagger \hat{\mathcal{U}}^{\text{I}}(t_0) \hat{S} e^{-i\frac{\tau}{2}\hat{\lambda}_S(t)} \hat{R}^\dagger e^{-i\frac{\tau}{2}\hat{\lambda}_Q(t)} \hat{Q}^\dagger, \quad (4.47)$$

where we made use of the fact that $\hat{R}^\dagger = \hat{S}^\dagger \hat{Q}$. Consequently, we can express

the operator $\hat{\mathcal{U}}^{\text{TB}}(t)$ on the interval $t \in [t_0, T]$ as

$$\hat{\mathcal{U}}^{\text{TB}}(t) = \begin{cases} \hat{\mathcal{U}}^{\text{I}}(t) & \text{if Case I} \\ \hat{\mathcal{U}}^{\text{II}}(t) & \text{if Case II.} \end{cases} \quad (4.48)$$

An implementation of this operator in a computer program allows us to further reduce the number of operators which have to be applied to the state vector $|\Psi(t)\rangle$ given by Eq. (4.1). We can see the simplification of the operator structure once we consider the product of two operators $\hat{\mathcal{U}}^{\text{II}}(t)$, for two different time steps $t'' > t'$. The corresponding product reads

$$\begin{aligned} \hat{\mathcal{U}}^{\text{II}}(t'')\hat{\mathcal{U}}^{\text{II}}(t') &= \hat{Q}e^{-i\frac{\tau}{2}\hat{\lambda}_Q(t'')} \hat{R}e^{-i\frac{\tau}{2}\hat{\lambda}_S(t'')} \hat{S}^\dagger \hat{\mathcal{U}}^{\text{I}}(t'') \hat{S}e^{-i\frac{\tau}{2}\hat{\lambda}_S(t'')} \hat{R}^\dagger e^{-i\frac{\tau}{2}\hat{\lambda}_Q(t'')} \\ &\quad e^{-i\frac{\tau}{2}\hat{\lambda}_Q(t')} \hat{R}e^{-i\frac{\tau}{2}\hat{\lambda}_S(t')} \hat{S}^\dagger \hat{\mathcal{U}}^{\text{I}}(t') \hat{S}e^{-i\frac{\tau}{2}\hat{\lambda}_S(t')} \hat{R}^\dagger e^{-i\frac{\tau}{2}\hat{\lambda}_Q(t')} \hat{Q}^\dagger, \end{aligned} \quad (4.49)$$

as one can see, in this case we can avoid implementing the two operators \hat{Q}^\dagger and \hat{Q} due to the fact that the product of both yields the identity operator. However, if we implement the time-evolution operator in this way, we have to take into account that between two time steps t'', t' we cannot sample from $|\Psi(t)\rangle$ in the transmon basis, i.e. in this case we have to apply the transformations \hat{Q}^\dagger and \hat{Q} once again. If we simulate complete quantum circuits, we rarely sample from $|\Psi(t)\rangle$. In such cases this approach is preferable.

4.5 Simulations of the circuit Hamiltonian model in alternative bases

In the previous section we discussed a product-formula time-evolution operator $\hat{\mathcal{U}}^{\text{TB}}$ for a specific Hermitian decomposition of the circuit Hamiltonian given by Eq. (3.114). Here we use the transmon basis $\{|\phi^{(z)}(t=0)\rangle\}_{z \in \mathbb{N}^0}$, see Eqs. (4.34) and (3.84), as the computational basis to simulate the state vector $|\Psi(t)\rangle$ in Eq. (4.1). Note that in this context the computational basis is simply the basis we use to represent the abstract state vector $|\Psi(t)\rangle$ in a computer program, i.e. we should not confuse this computational basis with the one of the IGQC, see Chap. 2. The first question which comes to mind is: why do we use this basis to solve the TDSE? For the work in this thesis we implemented several product-formula time-evolution operators which make use of different computational bases. In this section we briefly summarise the results of this work.

If we use the charge basis $\{|n\rangle\}_{n \in \mathbb{Z}}$, see Eq. (3.77), as the computational basis, we find (data not shown) that a quasi approximation-free solution of

the TDSE can be obtained by simulating flux-tunable transmons with up to fifty one charge basis states. This means adding an additional flux-tunable transmon to the model increases the state vector size by a factor of fifty one. This number was obtained by performing the simulations presented in Chap. 5 with the charge basis simulator. Note that the corresponding simulations in the transmon basis require at most sixteen basis states for the flux-tunable transmon. Therefore, using the charge basis simulator for large-scale simulations of flux-tunable transmons is not feasible. The core simulation algorithm was devised by the author of Ref. [Wil16]. The author of this thesis simply modified the algorithm presented in Ref. [Wil16] such that the changes in the circuit Hamiltonian given by Eq. (3.60) are accounted for, see also Ref. [YSK19].

If we use the instantaneous basis $\{|\phi^{(z)}(t)\rangle\}_{z \in \mathbb{N}^0}$, see Eqs. (4.34) and (3.84), to model the different flux-tunable transmons in the system, i.e. the fixed-frequency transmons and resonators are still modelled with the basis states $\{|\phi^{(z)}(t=0)\rangle\}_{z \in \mathbb{N}^0}$ and $\{|\psi^{(z)}\rangle\}_{z \in \mathbb{N}^0}$, we find that we need to modify the circuit Hamiltonian in Eq. (3.114) such that the TDSE stays form invariant. For simplicity, we consider a single flux-tunable transmon whose dynamics is modelled with the circuit Hamiltonian in Eq. (3.61). The state vector which describes the time-evolution of the system can be expressed as

$$|\Psi^*(t)\rangle = \mathcal{Y}(t) |\Psi(t)\rangle, \quad (4.50)$$

where $\mathcal{Y}(t)$ is a unitary transformation which maps the basis states $|\phi^{(z)}(0)\rangle$ at time $t = 0$ to the states $|\phi^{(z)}(t)\rangle$ at time t . We require the TDSE to retain its original form, see Ref. [Wei15]. Therefore, we transform the Hamiltonian as

$$\hat{H}^*(t) = \mathcal{Y}(t) \hat{H}(t) \mathcal{Y}^\dagger(t) - i\mathcal{Y}(t) \partial_t \mathcal{Y}^\dagger(t). \quad (4.51)$$

In order to proceed, we have to determine an analytical expressions for the inner product $\langle \phi^{(z')}(t) | \partial_t | \phi^{(z)}(t) \rangle$ such that the term

$$\hat{\mathcal{D}}(t) = -i\mathcal{Y}(t) \partial_t \mathcal{Y}^\dagger(t), \quad (4.52)$$

can be implemented. However, to the best knowledge of the author, this expression is not known to the research community for the eigenfunctions in Eq. (3.85), see also Ref. [Cot02]. Note that the Josephson energy and the flux variables in Eq. (3.85) need to be substituted if Hamiltonian Eq. (3.61) is considered, see Sec. 3.4 for more details. Therefore, this approach was abandoned too. Furthermore, all attempts to implement the operator in Eq. (4.52) numerically failed.

4.6 Implementation of the second-order time-evolution operator

In Sec. 4.3 and 4.4, we discuss two second-order time-evolution operators $\hat{\mathcal{U}}^{\text{HB}}$ and $\hat{\mathcal{U}}^{\text{TB}}$ which allow us to solve the TDSE for Hamiltonians Eqs. (3.114) and (3.117) numerically, i.e. we can obtain an approximate solution. In this section, we discuss how to implement the operator $\hat{\mathcal{U}}^{\text{TB}}$ with respect to the state vector

$$|\Psi(t)\rangle = \sum_{\mathbf{z} \in \mathcal{I}_n} c_{\mathbf{z}}(t) |\mathbf{z}\rangle. \quad (4.53)$$

Since we can use the same algorithms to implement the time-evolution operators $\hat{\mathcal{U}}^{\text{HB}}$ and $\hat{\mathcal{U}}^{\text{TB}}$ given by Eqs. (4.32) and (4.48), respectively, we only discuss the implementation of $\hat{\mathcal{U}}^{\text{TB}}$ in detail. Transferring the results of the following discussion to the case of $\hat{\mathcal{U}}^{\text{HB}}$ should come at ease.

Broadly speaking, the time-evolution matrix \mathcal{U}^{TB} contains two types of matrices, namely diagonal and non-diagonal ones. Note that the diagonal matrices are not all diagonal in the same basis. In fact, this is the reason why we find the non-diagonal matrices in \mathcal{U}^{TB} , i.e. these matrices transform the basis for every term in the product-formula decomposition.

Since both types of matrices are defined in terms of tensor products, see Eq. (4.42) and the sums of tensor products, see Eq. (4.44) and the product-formula decomposition allows us to treat the terms H_0 , $W_0(t)$ and $W_1(t)$ and $W_2(t)$ individually, we find that we can evaluate the action of all operators either by considering their action with respect to the individual basis states $|\mathbf{z}\rangle$ or by considering a small set $\{|\mathbf{z}\rangle\}$ of interdependent basis states. In the following, we derive a set of update rules which allow us to implement these operators.

We begin with the update rules for the diagonal matrices. Here we can further differentiate. The approximant \mathcal{U}^{TB} contains diagonal matrices which exclusively act on individual subspaces, see H_0 , $W_0(t)$ and $W_1(t)$ as well as diagonal matrices which act on two distinct subspaces, see $W_2(t)$, i.e. the matrices which describe the interaction between two subsystems.

Without loss of generality, we consider a diagonal matrix $\lambda_{\text{Tun.},j}$ which is defined with respect to the subspace $\mathcal{H}_{\text{Tun.},j}$ of some flux-tunable transmon. The action of this matrix reads

$$e^{-i\tau\lambda_{\text{Tun.},j}} |\mathbf{z}\rangle = e^{-i\tau\lambda_{\text{Tun.}}^{(z_j)}} |\mathbf{z}\rangle, \quad (4.54)$$

where $z_j \in \{0, \dots, n_J - 1\}$ is the j th tuple entry of $\mathbf{z}_J = (z_{|J|-1}, \dots, z_0)$ in $|\mathbf{z}\rangle$, see Eq. (4.34). Consequently, $\lambda_{\text{Tun.}}^{(z_j)}$ is the eigenvalue which corresponds to

the eigenvector $|\phi^{(z_j)}\rangle$. In the end, we simply have to implement the update rule

$$c'_{\mathbf{z}} = c_{\mathbf{z}} e^{-i\tau \lambda_{\text{Tun.}}^{(z_j)}}, \quad (4.55)$$

where $c'_{\mathbf{z}}$ denotes the updated state vector coefficient.

Similarly, without loss of generality, we can consider a diagonal matrix product of the form

$$\lambda_{\text{Res.},k} \otimes \lambda_{\text{Tun.},j}, \quad (4.56)$$

where the matrix $\lambda_{\text{Res.},k}$ acts on the subspace $\mathcal{H}_{\text{Res.},k}$. The matrix $\lambda_{\text{Tun.},j}$ is defined as before. The action of the tensor product can be expressed as

$$e^{-i\tau \lambda_{\text{Res.},k} \otimes \lambda_{\text{Tun.},j}} |\mathbf{z}\rangle = e^{-i\tau \lambda_{\text{Res.}}^{(z_k)} \lambda_{\text{Tun.}}^{(z_j)}} |\mathbf{z}\rangle, \quad (4.57)$$

where $z_k \in \{0, \dots, n_K - 1\}$ is the k th tuple entry of $\mathbf{z}_K = (z_{|K|-1}, \dots, z_0)$ in $|\mathbf{z}\rangle$ and $\lambda_{\text{Res.}}^{(z_k)}$ is the eigenvalue which corresponds to the eigenvector $|\psi^{(z_k)}\rangle$. The update rule for this case reads

$$c'_{\mathbf{z}} = c_{\mathbf{z}} e^{-i\tau \lambda_{\text{Res.}}^{(z_k)} \lambda_{\text{Tun.}}^{(z_j)}}. \quad (4.58)$$

In principle, we can extend the procedure to products of more than two diagonal matrices. The argumentation is straightforward.

If we implement the matrix products in \mathcal{U}^{TB} , we actually work with the transformed basis states $|\mathbf{w}_Q\rangle$, $|\mathbf{w}_S\rangle$ and $|\mathbf{w}_T\rangle$, which are related to the basis states $|\mathbf{z}\rangle$ by means of the transformations Q , S and T , see Eq. (4.42). However, in order to provide the reader with a discussion which makes use of the new basis states, we would need to introduce new notation. We circumvent this inconvenience by providing a discussion which considers matrix products that are diagonal in the basis $\mathcal{J}_{\mathbf{n}}$. The arguments we make can be transferred to any basis, assuming that the matrices in question are diagonal in the corresponding bases.

So far, we only discussed how to implement diagonal matrices which exclusively act on specific subspaces or products of diagonal matrices which are diagonal in one shared basis. However, the approximant \mathcal{U}^{TB} contains the sums in Eq. (4.44) of those types of matrices. Nonetheless, this does not lead to further complications since the individual summation terms commute. Consequently, we can replace the exponential functions that contain the sums by products of exponential functions, which contain the individual summation terms. This means we can implement the individual terms with the update rules we have discussed so far.

The non-diagonal basis transformations Q , S and T all have the same form, i.e. we can express them as tensor products of matrices which act on

the different subspaces. Therefore, the tensor algebra calculus, see Sec. 2.1, allows us to update the state vector by applying the matrices which act on the different subspaces. We simply make repeated use of the identity given by Eq. (2.11). In the end, we only have to find one additional update rule for a full implementation of \mathcal{U}^{TB} .

As before, without loss of generality, we consider the matrix T_{n_j} which acts on the subspace $\mathcal{H}_{\text{Tun},j}$. Evaluating this matrix with respect to the basis vector $|\mathbf{z}^{(z'_j)}\rangle$ and the index z'_j yields

$$T_{n_j} |\mathbf{z}^{(z'_j)}\rangle = \sum_{z_j \in \{0, \dots, n_J - 1\}} T_{n_j}^{(z_j, z'_j)} |\mathbf{z}^{(z_j)}\rangle. \quad (4.59)$$

Here, we introduced the label z_j to the notation. This index simply highlights the corresponding tuple entry in \mathbf{z}_J . We might call the label j the target index. If we apply T_{n_j} to the state vector in Eq. (4.53), we can make use of Eq. (4.59) and rearrange the summation terms to find an update rule.

Assuming we fix all the remaining tuple entries in the basis state $|\mathbf{z}^{(z'_j)}\rangle$, i.e. all tuple entries, which are not z'_j , we find that the matrix T_{n_j} relates all basis states $|\mathbf{z}^{(z'_j)}\rangle$, where $z'_j \in \{0, \dots, n_J - 1\}$. The corresponding update rule reads

$$c'_{\mathbf{z}^{(z'_j)}} = \sum_{z_j \in \{0, \dots, n_J - 1\}} T_{n_j}^{(z'_j, z_j)} c_{\mathbf{z}^{(z_j)}}. \quad (4.60)$$

Consequently, if we model the subspace $\mathcal{H}_{\text{Tun},j}$ with n_J basis states, we find that n_J different state vector coefficients $c_{\mathbf{z}}$ are related by Eq. (4.60). This allows us to partition the full state vector into groups of size n_J . We can then update these groups independently, i.e. the groups which are built by fixing the tuple entries that are not the target tuple entry z'_j . Note that this reasoning can be applied to all subspaces $\mathcal{H}_{\text{TLS},l}$, $\mathcal{H}_{\text{Res},k}$, $\mathcal{H}_{\text{Tun},j}$ and $\mathcal{H}_{\text{Fix},i}$. This update rule can be implemented in many different ways and the space of possible implementations is quite large. For example, two important aspects of the particular implementation instance are the way we address the different state vector coefficients $c_{\mathbf{z}}$ and the programming paradigm itself.

A detailed discussion of all the subroutines we use to perform the simulations and the extended simulation code is beyond the scope of this chapter. However, in Sec. 4.7 we provide the reader with an overview of the simulation software.

4.7 An overview of the simulation software

In this section, we provide a high-level overview of the simulation software. So far, we discussed some of the components or modules of this simulation

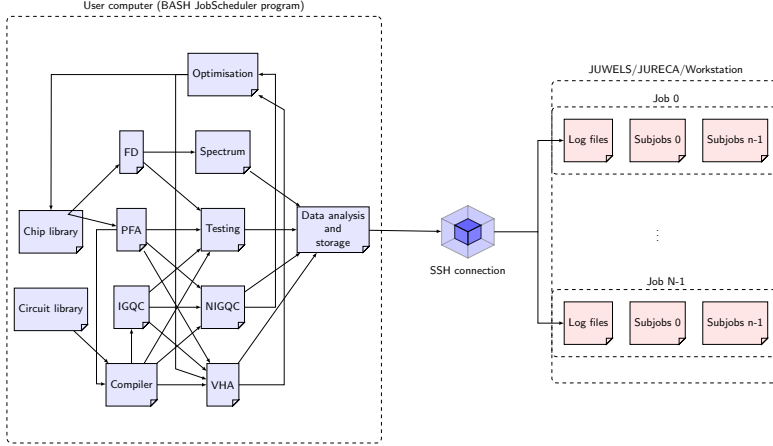


Figure 4.1: High-level sketch of the simulation software and the surrounding compute infrastructure. We use the blue rectangles to lump C++ source code into modules. The dashed lines surrounding the modules indicate the borders of the C++ simulation software but also the domain of a BASH program called JobScheduler. We use the program JobScheduler in combination with the MAKE build automation tool to control the compilation, execution, logging and data storage on various computer systems, including the supercomputers JUWELS and JURECA, see Refs. [Jül19; Jül18]. The modules which are named by means of abbreviations are full diagonalisation (FD), product-formula algorithm (PFA), ideal gate-based quantum computer (IGQC), non-ideal gate-based quantum computer (NIGQC) and variational quantum hybrid algorithm (VHA). The remaining module names are self explanatory. The functionality of all modules is explained in the main text.

software. For example, in Sec. 2.4 we discussed how to simulate the IGQC and in Secs. 4.1-4.6 we discussed how to solve the TDSE with full diagonalisation and the product-formula algorithm. The algorithms can be used to implement a numerical NIGQC model. Note that in the following we capitalise almost all of the computer science terminology.

Figure 4.1 shows a high-level sketch of the simulation software. Here we lump different parts of the simulation code into software modules, see blue rectangles and show the surrounding compute infrastructure, see dashed lines, red rectangles and other symbols. The dashed rectangle on the right side of Fig. 4.1 shows the boundaries of the source code. We use C++ as the basic programming language. In addition, we use Open Multi Processing (OpenMP), Message Passing Interface (MPI) and Compute Unified Device Architecture (CUDA) to parallelise the C++ simulation code.

Since we execute tasks on different computer systems like the supercomputers JUWELS and JURECA, see Refs. [Jül19; Jül18], we use a homemade BASH program named JobScheduler to control compilation, execution, logging and data storage. If the program JobScheduler executes a job, the following steps are performed. First, we update the source code by means of an SSH connection. Second, we execute a MAKE program to recompile some parts of the source code if changes have been made. Third, we create a directory structure which contains the result data, log files, SLURM job scripts and executables. Fourth, if the task is executed on a supercomputer system, we write a SLURM job script for every task or subjob. Fifth, we move executables and SLURM job scripts to the correct directories. In the end, if we use a supercomputer system, we submit the SLURM job scripts to the program SLURM and the program JobScheduler terminates. Note that every step in this sequence relies on the successful execution of the preceding step. In the following we briefly summarise the functionality of each module and sometimes discuss the relation between different modules.

The module named chip library, see first column on the left side, contains all NIGQC chip parameters. Here we manage the parameters which specify the Hamiltonians Eqs. (3.114) and (3.117) and the pulse parameters which determine the values of the time dependencies that can be found in the model Hamiltonians. All these parameters specify a NIGQC model, see Chap. 5-6.

The module named circuit library, see first column on the left side, has a similar functionality as the module chip library. Here we manage all the quantum circuits we use to perform simulations.

The module named FD, see second column on the left side, allows us to perform full diagonalisation of the Hamiltonians Eqs. (3.114) and (3.117). Here, we use the software packages LAPACK and MKL, see Refs. [And+; Cor], to compute the eigenvalues and eigenvectors for a given Hamiltonian.

The module is split into two parts. The first part consists of a class which organises all the data and the second part allows us to simulate the time evolution of a system by implementing Eq. (4.3) for various time steps.

The module named PFA, see second column on the left side, allows us to solve the TDSE with various product-formula algorithms. This module is also split into two parts, i.e. a part with classes which organise the simulation data and a part with subroutines which update the state vector data structure by means of the update rules we discuss in Sec. 4.6. We can determine an approximate solution of the TDSE for the Hamiltonian Eq. (3.117) in the harmonic basis, see Sec. 4.3 and the Hamiltonian Eq. (3.114) in the transmon basis, see Sec. 4.4. Furthermore, the user can choose between serial execution, parallel execution with OpenMP and parallel execution on a GPU, where we use CUDA to parallelise the source code.

The module named IGQC, see second column on the left side, enables us to simulate the IGQC. Here we update the state vector of a N -qubit system for a given quantum circuit, see Chap. 2. We can perform the simulations with the Feynman or the Schrödinger algorithm, see Sec. 2.4. The user has the choice between serial execution, parallel execution with OpenMP and parallel execution with OpenMP and MPI.

The module named Compiler, see second column on the left side, takes a quantum circuit and optionally a PFA class, with information about the chip architecture and the pulse parameters, as an input and creates control data structures. We can use the control data structures in combination with the PFA and IGQC modules to simulate complete quantum circuits, see modules NIGQC and VHAs. The data processing steps for the PFA module are quite complex and the computational effort grows with the number of qubits and the quantum circuit length. We discuss some of the details of these computations in Chap. 6. The corresponding data processing steps for the IGQC module are relatively simple and there is quasi no overhead in time and space.

The module named Optimisation, see third column on the left side, is a module to perform various types of optimisation. Here we use the library NLOpt, see Ref. [Joh], to solve the optimisation problems we encounter. The library provides us with more than a dozen optimisation algorithms.

The module named Spectrum, see third column on the left side, enables us to compute the spectrum of a system for different model parameters in parallel. Here we use the packages LAPACK and MKL, see Refs. [And+; Cor], to obtain the eigenvalues for a given Hamiltonian. We parallelised this procedure with MPI. This means every MPI process computes the eigenvalues for one subset of all model parameters. Furthermore, we can make use of the fact that the packages LAPACK and MKL are parallelised with OpenMP.

The module named Testing, see third column on the left side, allows us to systematically test critical parts of the software. We automatically test the modules FD, PFA, IGQC and Compiler every time after changes have been made to the corresponding source code files.

The module named NIGQC, see third column on the left side, enables us to compute various gate-error measures for a given quantum circuit. This means we can study how gate errors dynamically evolve over time, i.e. for consecutive gates in the quantum circuit. In Sec. 6.3 we discuss all gate-error measures we can compute with this module. We use MPI to parallelise this part of the simulation code.

The module named VHA, see third column on the left side, enables us to simulate variational hybrid algorithms with different optimisation algorithms and quantum computer models, i.e. IGQC or NIGQC. The user can specify the cost Hamiltonian given by Eq. (2.38) in combination with a suitable optimisation algorithm, see module Optimisation. This module takes a quantum circuit as an input. However, the quantum circuit is first preprocessed by the Compiler module. We use MPI to parallelise the simulation code. This means the user can run several problem instances in parallel.

The module named Data analysis and storage, see fourth column on the left side, manages data analysis and storage, as the name suggests. In almost all cases we can store data in binary or text format. The module also enables us to automatically store data in LATEX tables, see Appendix A and Appendix B. Here a subroutine generates the corresponding LATEX source code.

4.8 Summary, conclusions and outlook

In this chapter we reviewed the fundamental computational problems we face when modelling time-evolution processes with the time-dependent Schrödinger equation (TDSE), see Sec. 4.1. We provided a general discussion of the product-formula algorithm which can potentially remedy some of computational issues we face, see Sec. 4.2. Furthermore, in Sec. 4.3 we derived a product-formula time-evolution operator which allows us to approximately solve the TDSE for the effective Hamiltonian Eq. (3.117). Similarly, in Sec. 4.4 we derived a product-formula time-evolution operator which allows us to approximately solve the TDSE for the circuit Hamiltonian Eq. (3.114). Additionally, we reported on attempts to model the time evolution of the circuit Hamiltonian Eq. (3.114) in the charge basis and the instantaneous flux-tunable transmon basis, see Sec. 4.5. We also discussed the algorithmic update rules which allow us to implement the time-evolution of the state

vector for the effective and the circuit Hamiltonian in a computer program, see Sec. 4.6. Finally, we provided an overview of the simulation software we use to obtain the results in this thesis, see Sec. 4.7.

In order to obtain the results in this thesis, we implemented several time-evolution operators which approximately solve the TDSE for the circuit Hamiltonian Eq. (3.114). However, we were not able to model various systems, see Chap. 5, with less than sixteen basis states per flux-tunable transmon. Therefore, in practice it is not possible to model more than eight interacting flux-tunable transmons with the circuit model. Here we work under the constraints that we have to simulate the original circuit Hamiltonian (3.114) and that the computer hardware can store up to 2^{32} state vector coefficients $c_{\mathbf{z}}$.

For future work, it might be interesting to look into the question whether or not there exists a computational basis which allows us to simulate more than eight interacting flux-tunable transmons. Here one might consider the time-dependent harmonic basis, see Eq. (3.97) and Ref. [Wil20b]. Alternatively, one might solve the analytical problem discussed in the latter part of Sec. 4.5.

Analysis of effective models for flux-tunable transmon systems

Many studies with a focus on flux-tunable transmons are based on effective models and not the associated lumped-element circuit Hamiltonian models introduced in Chap. 3. Unfortunately, usually it is not known to what extent the predictions made by the effective and the circuit model deviate when a control pulse is applied, i.e. when the Hamiltonians are time-dependent. In this chapter, we compare the time evolution of state vectors which are obtained by solving the TDSE for both the effective and the circuit Hamiltonians, for microwave and unimodal control pulses. Here, we consider single-qubit (X) and two-qubit (ISWAP and CZ) gate type transitions. Note that we consider these transitions only in terms of the probability amplitudes and all transitions are activated by means of an external flux. Furthermore, in order to obtain a quasi approximation-free solution of the TDSE, we increase the number of basis states up to the point where the probability amplitudes have converged up to the third decimal.

The work in this chapter is structured as follows. In Sec. 5.1 we discuss the three different systems we study and the control pulse we use to activate the different transitions. Here, we also discuss the approximations which give rise to different effective models. Next, in Sec. 5.2 we study the pulse response of a single flux-tunable transmon for microwave and unimodal pulse forms. Here, we consider two effective models and the associated circuit model. In Sec. 5.3 and 5.4 we investigate various single-qubit (X) and two-qubit (ISWAP and CZ) transitions for a first two-qubit NIGQC model, see Fig. 5.1(a). Here we model the different transitions with a microwave pulse for various effective models and the associated circuit model. Similarly, in Sec. 5.5 and 5.6, we study excitations of the coupler element and two-qubit (ISWAP and CZ) transitions for a second two-qubit NIGQC model, see Fig. 5.1(b). Here we activate the different transitions with a microwave and a unimodal pulse for various effective models and the associated circuit model. Note that we use $\hbar = 1$ throughout this chapter.

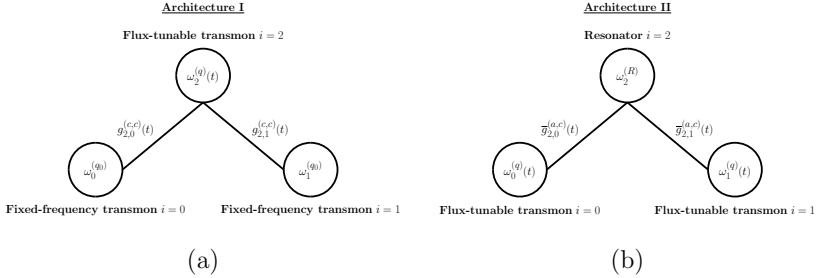


Figure 5.1: Sketches of two different device architectures I(a) and II(b) to realise NIGQCs, see Sec. 2.7. Panel(a) shows architecture I. Here we couple two fixed-frequency transmons characterised by the qubit frequencies $\omega_0^{(q_0)}$ and $\omega_1^{(q_0)}$ to a flux-tunable transmon characterised by the tunable frequency $\omega_2^{(g)}(t)$. We use the device parameters listed in Table 5.1 and Table 5.3 for our simulations of this system. Panel(b) shows architecture II. Here we couple two flux-tunable transmons characterised by the qubit frequencies $\omega_0^{(g)}(t)$ and $\omega_1^{(g)}(t)$ to a resonator characterised by the resonance frequency $\omega_2^{(R)}$. We use the device parameters listed in Table 5.2 and Table 5.4 for our simulations of this system. The dynamics is determined by solving the TDSE for the circuit Hamiltonian Eq. (3.114) and the effective Hamiltonian Eq. (3.117).

5.1 System specification and simulation parameters

Figures 5.1(a-b) show sketches of two different transmon systems or device architectures which potentially allow us to realise NIGQCs, see Sec. 2.7.

Architecture I consists of two fixed-frequency transmons $\omega_0^{(q_0)}$ and $\omega_1^{(q_0)}$ coupled to a flux-tunable transmon $\omega_2^{(g)}(t)$. The flux-tunable transmon functions as a coupler element. We use the device parameters listed in Table 5.1 and Table 5.3 for our simulations of this system. This architecture is subject of several experimental and theoretical studies, see Refs. [McK+16; Rot+17; Gan+20; Gu+21; Bla+21].

Architecture II consists of two flux-tunable transmons $\omega_0^{(g)}(t)$ and $\omega_1^{(g)}(t)$ coupled to a coupling resonator $\omega_2^{(R)}$. We use the device parameters listed in Table 5.2 and Table 5.4 for our simulations of this architecture. This device type is discussed in Refs. [Rol+19; Kri+20; Lac+20; Bla+21].

The device parameters in Tables 5.1-5.4 are used to fully specify the

Table 5.1: Device parameters for the circuit Hamiltonian Eq. (3.114). We use these parameters to model a device of type architecture I. The first column shows the subsystem indices i . The second and third columns show the transmon qubit frequencies $\omega_i^{(Q_0)}$ and anharmonicities $\alpha_i^{(Q_0)}$, respectively. The fourth column shows the capacitive energies E_{C_i} . The fifth and sixth columns show the left $E_{J_{l,i}}$ and right $E_{J_{r,i}}$ Josephson energies. The seventh column shows the flux-offset values $\varphi_{0,i}$. The eighth column shows the interaction strength constants $G_{2,i}^{(1)}$. Here we model dipole-dipole interactions between fixed-frequency and flux-tunable transmons. We show all parameters except the ones in the first and seventh column in units of GHz. These parameters are motivated by experiments discussed in Ref. [Gan+20].

i	$\omega_i^{(Q_0)}/2\pi$	$\alpha_i^{(Q_0)}/2\pi$	$E_{C_i}/2\pi$	$E_{J_{l,i}}/2\pi$	$E_{J_{r,i}}/2\pi$	$\varphi_{0,i}/2\pi$	$G_{2,i}^{(1)}/2\pi$
0	5.100	-0.310	1.079	13.456	n/a	n/a	0.085
1	6.200	-0.285	1.027	20.371	n/a	n/a	0.085
2	8.100	-0.235	0.880	17.897	21.486	0.15	n/a

Table 5.2: Device parameters for the circuit Hamiltonian Eq. (3.114). We use these parameters to model a device of type architecture II. The first column shows the subsystem indices i . The second and third columns show the transmon qubit frequencies $\omega_i^{(Q_0)}$ and anharmonicities $\alpha_i^{(Q_0)}$, respectively. The fourth column shows the capacitive energies E_{C_i} . The fifth and sixth columns show the left $E_{J_{l,i}}$ and right $E_{J_{r,i}}$ Josephson energies. The seventh column shows the flux-offset values $\varphi_{0,i}$. The eighth column shows the interaction strength constants $G_{2,i}^{(4)}$. Here we model dipole-dipole interactions between flux-tunable transmons and resonators. We show all parameters except the ones in the first and eighth column in units of GHz. These device parameters are motivated by experiments discussed in Ref. [Lac+20].

i	$\omega_i^{(R)}/2\pi$	$\omega_i^{(Q_0)}/2\pi$	$\alpha_i^{(Q_0)}/2\pi$	$E_{C_i}/2\pi$	$E_{J_{l,i}}/2\pi$	$E_{J_{r,i}}/2\pi$	$\varphi_{0,i}/2\pi$	$G_{2,i}^{(4)}/2\pi$
0	n/a	4.200	-0.320	1.068	3.140	9.419	0	0.300
1	n/a	5.200	-0.295	1.036	4.817	9.633	0	0.300
2	45.000	n/a	n/a	n/a	n/a	n/a	n/a	n/a

circuit Hamiltonian Eq. (3.114) and the effective Hamiltonian Eq. (3.117). These are the Hamiltonians we use to model the NIGQCs. One can use both architectures to implement ISWAP and CZ gate transitions with an external flux $\varphi(t)$, i.e. these transitions can potentially be used to implement the corresponding gate matrices we discuss in Chap. 1.

Table 5.3: Device parameters for the adiabatic effective Hamiltonian Eq. (3.117) where the interaction strength is constant. We use these parameters to model a device of type architecture I, see also Table 5.1 for the corresponding circuit Hamiltonian device parameters. The first column shows the subsystem indices i . The second and third columns show the transmon qubit frequencies $\omega_i^{(q_0)}$ and anharmonicities $\alpha_i^{(q_0)}$, respectively. The fourth column shows the flux-offset values $\varphi_{0,i}$. The fifth column shows the effective interaction strength $g_{2,i}^{(c,c)}(\varphi_{0,2})$ at the operating point $\varphi_{0,2}$. Here we model dipole-dipole interactions between fixed-frequency and flux-tunable transmons. All device parameters except the ones in the first and fourth columns are given in units of GHz.

i	$\omega_i^{(q_0)}/2\pi$	$\alpha_i^{(q_0)}/2\pi$	$\varphi_{0,i}/2\pi$	$g_{2,i}^{(c,c)}(\varphi_{0,2})/2\pi$
0	5.100	-0.310	0	0.146
1	6.200	-0.285	0	0.164
2	8.100	-0.235	0.15	n/a

Table 5.4: Device parameters for the adiabatic effective Hamiltonian Eq. (3.117) where the interaction strength is constant. We use these parameters to model a device of architecture II, see also Table 5.2 for the corresponding circuit Hamiltonian device parameters. The first column shows the subsystem indices i . The second column shows the resonator frequencies $\omega_i^{(R)}$. The third and fourth columns show the transmon qubit frequencies $\omega_i^{(q_0)}$ and anharmonicities $\alpha_i^{(q_0)}$, respectively. The fifth column shows the flux-offset values $\varphi_{0,i}$. The sixth column shows the effective interaction strength $g_{2,i}^{(a,c)}(\varphi_{0,i})$ at the operating points $\varphi_{0,i}$. Here we model dipole-dipole interactions between flux-tunable transmons and resonators. All device parameters except the ones in the first and fifth columns are given in units of GHz.

i	$\omega_i^{(R)}/2\pi$	$\omega_i^{(q_0)}/2\pi$	$\alpha_i^{(q_0)}/2\pi$	$\varphi_{0,i}/2\pi$	$g_{2,i}^{(a,c)}(\varphi_{0,i})/2\pi$
0	n/a	4.200	-0.320	0	0.307
1	n/a	5.200	-0.295	0	0.344
2	45.000	n/a	n/a	n/a	n/a

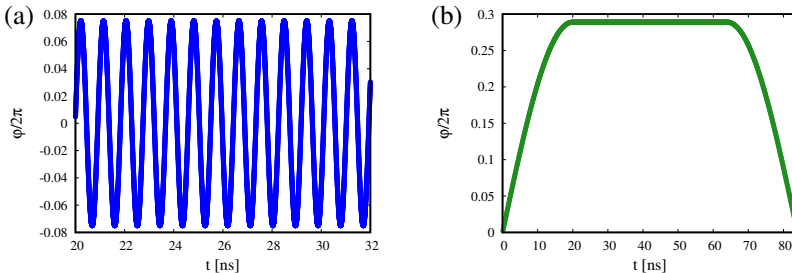


Figure 5.2: Fluxes $\varphi/2\pi$ given by Eq. (5.1) as a functions of time t for two sets of pulse parameters. Panel(a) shows a microwave pulse. Here we use the parameters $\omega^{(D)}/2\pi = 1.089$ GHz, $\delta/2\pi = 0.075$, $T_{r/f} = 13$ ns and $T_d = 205.4$ ns to obtain the results. The parameters are also listed in Table 5.6 row six. Panel(b) shows a unimodal pulse. Here we use the parameters $\omega^{(D)}/2\pi = 0$ GHz, $\delta/2\pi = 0.297$, $T_{r/f} = 20.0$ ns and $T_d = 84.0$ ns to acquire the results. The parameters are also listed in Table 5.6 row nine.

For our simulations we use the external flux (control pulse)

$$\varphi(t) = \varphi_0 + \delta e(t) \cos(\omega^{(D)}t), \quad (5.1)$$

where the real-valued pulse parameters φ_0 , δ and $\omega^{(D)}$ denote the flux offset, the pulse amplitude and the drive frequency, respectively. Note that the external flux $\varphi(t)$ and consequently the flux offset φ_0 and the pulse amplitude δ are without units. Furthermore, the Hamiltonian Eq. (3.61) is 2π periodic. Therefore, in this chapter we give the external flux $\varphi(t)$ and the parameters φ_0 , δ always in units of 2π . The real-valued envelope function $e(t)$ is defined as

$$e(t) = \begin{cases} \sin(\lambda t) & \text{if } 0 \leq t < T_{r/f} \\ 1 & \text{if } T_{r/f} \leq t \leq \Delta T \\ \sin(\frac{\pi}{2} + \lambda(t - \Delta T)) & \text{if } \Delta T < t \leq T_d, \end{cases} \quad (5.2)$$

where the real-valued pulse parameters $T_{r/f}$ and T_d denote the rise and fall time and the pulse duration, respectively. Furthermore, $\lambda = \pi/2T_{r/f}$ and $\Delta T = T_d - T_{r/f}$.

Figures 5.2(a-b) show the external flux $\varphi(t)/2\pi$ as functions of time t for two different sets of pulse parameters. In panel(a) we model a microwave pulse with the pulse amplitude $\delta/2\pi = 0.075$, the drive frequency $\omega^{(D)}/2\pi = 1.089$ GHz, the rise and fall time $T_{r/f} = 13$ ns and the pulse duration $T_d = 205.4$ ns. This microwave pulse can be used to activate ISWAP transitions

in architecture I if we drive the tunable coupler, see Fig. 5.1(a). In panel(b) we model a unimodal pulse with the pulse amplitude $\delta/2\pi = 0.297$, the drive frequency $\omega^{(D)}/2\pi = 0$ GHz, the rise and fall time $T_{r/f} = 20$ ns and the pulse duration $T_d = 84$ ns. This unimodal pulse can be used to activate ISWAP transitions in architecture II if we drive the flux-tunable transmon $\omega_1^{(q)}(t)$, see Fig. 5.1(b).

Table 5.5 contains a summary of the pulse parameters we use to model transitions with the circuit Hamiltonian Eq. (3.114). Similarly, in Table 5.6 we summarise the pulse parameters we use to model various transitions with the adiabatic effective Hamiltonian Eq. (3.117).

The overall objective of this chapter is to analyse how well the circuit and the effective Hamiltonian models agree with one another if we model architectures I and II, see Fig. 5.1(a-b). To this end, we implemented two simulation codes. One allows us to simulate the effective Hamiltonian model and the other allows to do the same with the circuit Hamiltonian model, see Chap. 4. The simulation code for the former model is equipped with the option to turn various approximations, i.e. simplifications, on and off.

To the best knowledge of the author, we often find that flux-tunable transmons are modelled adiabatically, see Refs. [McK+16; Rot+17; Gu+21; Yan+18; Bak+22]. This means that the non-adiabatic driving term

$$\hat{\mathcal{D}}_{\text{Dri.}}(t) = -i\sqrt{\frac{\xi(t)}{2}}\dot{\varphi}_{\text{eff.}}(t)(\hat{c}^\dagger - \hat{c}) + \frac{i}{4}\frac{\dot{\xi}(t)}{\xi(t)}(\hat{c}^\dagger\hat{c}^\dagger - \hat{c}\hat{c}), \quad (5.3)$$

first introduced in Eq. (3.102) is neglected completely, see also Eq. (3.117). We can control this assumption in the simulation code.

Furthermore, we can find instances where the interaction strength between flux-tunable transmons and other circuit elements is modelled as time independent, see Refs. [Gan+20; McK+16; Rot+17; Gu+21]. We model the interaction strength between a flux-tunable transmon i and a fixed-frequency transmon j with the function

$$g_{i,j}^{(c,c)}(t) = G_{i,j}\sqrt[4]{\frac{\tilde{E}_{J,i}(t)}{8\tilde{E}_{C,i}}}\sqrt[4]{\frac{E_{J_j}}{8E_{C_j}}}, \quad (5.4)$$

where $G_{i,j}$ is a constant real-valued parameter, $\tilde{E}_{J,j}(t)$ and E_{J_i} denote Josephson energies and $\tilde{E}_{C,j}$ and E_{C_i} refer to capacitive energies. Similarly, we model the interaction strength between a resonator element i and a flux-tunable transmon j with the function

$$g_{i,j}^{(a,c)}(t) = G_{i,j}\sqrt[4]{\frac{\tilde{E}_{J,j}(t)}{8\tilde{E}_{C,j}}}. \quad (5.5)$$

Table 5.5: Summary of the pulse parameters we use to perform simulations with the circuit Hamiltonian model, see Eq. (3.114) and Sec. 3.5. Here, we use a pulse of the form Eq. (5.1) to drive different systems. The first column contains table references. The corresponding tables contain the device parameters which specify the Hamiltonian. The second column shows figure references which refer the reader to the simulation results. The third column shows different gate types. We can potentially use the corresponding pulses to model these gates, assuming we can arrange the phases of the state vectors accordingly. If we cannot assign a gate to the transition, we denote the gate as not applicable (n/a). The fourth column shows the two states which exchange population due to the driving terms. The fifth column contains the drive frequencies $\omega^{(D)}$ in units of GHz. The sixth column shows the pulse amplitudes δ . The seventh and eighth columns show the rise and fall time $T_{r/f}$ and the pulse duration T_d in units of ns, respectively. The ninth column shows the minimum number of transmon basis states n_J we recommend for modelling the dynamics of the system in the transmon basis.

Parameters	Figure	Gate	States z	$\omega^{(D)}/2\pi$	$\delta/2\pi$	$T_{r/f}$	T_d	n_J
Table 5.1	Fig. 5.4(a)	X	$\{(0), (1)\}$	7.636	0.001	100.0	200.0	3
Table 5.1	Fig. 5.7(a)	X	$\{(0, 0, 0), (0, 1, 0)\}$	6.183	0.045	22.5	45.0	3
Table 5.1	Fig. 5.7(b)	X	$\{(0, 0, 0), (0, 0, 1)\}$	5.092	0.085	25.0	50.0	3
Table 5.1	Fig. 5.8(a-d)	ISWAP	$\{(0, 1, 0), (0, 0, 1)\}$	1.089	0.075	13.0	209.40	6
Table 5.1	Fig. 5.9(a-d)	CZ	$\{(0, 1, 1), (0, 2, 0)\}$	0.809	0.085	13.0	297.55	8
Table 5.2	Fig. 5.14(a-c)	n/a	$\{(0, 0, 0), (1, 0, 0)\}$	45.00	0.020	150.0	300.0	4
Table 5.2	Fig. 5.14(a-c)	n/a	$\{(0, 0, 0), (2, 0, 0)\}$	45.00	0.020	150.0	300.0	4
Table 5.2	Fig. 5.15(a-d)	ISWAP	$\{(0, 1, 0), (0, 0, 1)\}$	0	0.289	20.0	100.0	14
Table 5.2	Fig. 5.16(a-d)	CZ	$\{(0, 1, 1), (0, 0, 2)\}$	0	0.334	20.0	125.0	16

This interaction strength model is taken from Ref. [Koc+07] and explained in more detail in Sec. 3.6. We can control the time dependence in the simulation code. Additionally, we can also control the accuracy of the spectrum of a single flux-tunable transmon, see Sec. 5.2 for more details.

Although, we are mainly interested in architecture I and II, it seems plausible to first investigate how well the effective models in Eqs. (3.109) and (3.110) for a single flux-tunable transmon can cover the dynamics of the corresponding circuit Hamiltonian Eq. (3.61). This question is the subject of the next section.

Table 5.6: Summary of pulse parameters we use to perform simulations with the adiabatic effective Hamiltonian model, see Eq. (3.117) and Sec. 3.6. Here, we use a pulse of the form Eq. (5.1) to drive different systems. The first column contains table references. The corresponding tables contain the device parameters which specify the Hamiltonian. The second column shows figure references which refer the reader to the simulation results. The third column shows different gate types. We can potentially use the corresponding pulses to model these gates, assuming we can arrange the phases of the state vectors accordingly. The fourth column shows the two states which exchange population due to the driving terms. The fifth column contains the drive frequencies $\omega^{(D)}$ in units of GHz. The sixth column shows the pulse amplitudes δ . The seventh and eighth columns show the rise and fall time $T_{r/f}$ and the pulse duration T_d in units of ns, respectively. If we cannot model the transition with the adiabatic effective Hamiltonian, we denote the parameters and gates as not applicable (n/a). The ninth column shows the different cases we model with the effective model Hamiltonian. In case A we model the system with a time-independent interaction strength and a non-adjusted spectrum. In case B we model the system with a time-dependent interaction strength and a non-adjusted spectrum. In case C we model the system with a time-dependent interaction strength and an adjusted spectrum.

Parameters	Figure	Gate	States z	$\omega^{(D)}/2\pi$	$\delta/2\pi$	$T_{r/f}$	T_d	Case
Table 5.3	n/a	n/a	$\{(0), (1)\}$	n/a	n/a	n/a	n/a	n/a
Table 5.3	n/a	n/a	$\{(0, 0, 0), (0, 1, 0)\}$	n/a	n/a	n/a	n/a	n/a
Table 5.3	n/a	n/a	$\{(0, 0, 0), (0, 0, 1)\}$	n/a	n/a	n/a	n/a	n/a
Table 5.3	Fig. 5.11(a)	ISWAP	$\{(0, 1, 0), (0, 0, 1)\}$	1.088	0.075	13.0	139.6	A
Table 5.3	Fig. 5.11(b)	ISWAP	$\{(0, 1, 0), (0, 0, 1)\}$	1.089	0.075	13.0	205.4	B
Table 5.3	Fig. 5.11(c)	CZ	$\{(0, 1, 1), (0, 2, 0)\}$	0.807	0.085	13.0	196.5	A
Table 5.3	Fig. 5.11(d)	CZ	$\{(0, 1, 1), (0, 2, 0)\}$	0.807	0.085	13.0	272.0	B
Table 5.4	n/a	n/a	$\{(0, 0, 0), (1, 0, 0)\}$	n/a	n/a	n/a	n/a	n/a
Table 5.4	n/a	n/a	$\{(0, 0, 0), (2, 0, 0)\}$	n/a	n/a	n/a	n/a	n/a
Table 5.4	Fig. 5.18(a)	ISWAP	$\{(0, 1, 0), (0, 0, 1)\}$	0	0.297	20.0	84.0	A
Table 5.4	Fig. 5.18(b)	ISWAP	$\{(0, 1, 0), (0, 0, 1)\}$	0	0.289	20.0	96.0	C
Table 5.4	Fig. 5.18(c)	CZ	$\{(0, 1, 1), (0, 0, 2)\}$	0	0.343	20.0	105.0	A
Table 5.4	Fig. 5.18(d)	CZ	$\{(0, 1, 1), (0, 0, 2)\}$	0	0.334	20.0	121.0	C

5.2 Simulations of a single flux-tunable transmon

In this section, we compare how the effective Hamiltonians Eq. (3.109) and Eq. (3.110) and the circuit Hamiltonian Eq. (3.61) react to different types of pulses $\varphi(t)$, i.e. external fluxes. All these models are used to describe isolated flux-tunable transmons. We study how the different systems, i.e. models, react to a resonant microwave pulse, see Fig. 5.2(a) and a unimodal pulse, see Fig. 5.2(b). If not stated otherwise, we use the system parameters listed in Table 5.1 row $i = 2$, the pulse given by Eq. (5.1) and three basis states to perform the simulations.

The effective Hamiltonian Eq. (3.110) is so simple that we can solve the TDSE for an arbitrary pulse $\varphi(t)$. If we assume that the initial state of the system is given by

$$|\Psi(t_0)\rangle = \sum_{z \in \mathbb{N}^0} c_z(t_0) |\psi^{(z)}(t_0)\rangle, \quad (5.6)$$

where $c_z(t) \in \mathbb{C}$ for all times t and $|\psi^{(z)}(t)\rangle$ denotes the instantaneous harmonic basis states, we find that the state vector at time t is given by the expression

$$|\Psi(t)\rangle = \sum_{z \in \mathbb{N}^0} e^{-i \int_{t_0}^t E^{(z)}(t') dt'} c_z(t_0) |\psi^{(z)}(t)\rangle. \quad (5.7)$$

Consequently, the probability amplitudes $|\langle z | \Psi(t) \rangle|^2$ do not change over time, no matter how we modulate the energies $E^{(z)}(t)$.

We now discuss how the Hamiltonians Eq. (3.109) and Eq. (3.61) react to a microwave pulse, see Fig. 5.2(a). Figures 5.3(a-b) show the ground-state probabilities $p^{(0)}$ as functions of the pulse duration T_d and the drive frequency $\omega^{(D)}$. Here, we use a pulse with the amplitude $\delta/2\pi = 0.001$ and the rise and fall time $T_{r/f} = T_d/2$ to obtain the results. In panel(a) we solve the TDSE for the circuit Hamiltonian Eq. (3.61). Similarly, in panel(b) we solve the TDSE for the effective Hamiltonian Eq. (3.109). Furthermore, we center the two chevron patterns in Figs. 5.3(a-b) around the transition frequencies $\omega^{(0)} = 7.636$ GHz(a) and $\omega^{(0)} = 7.643$ GHz(b). As one can see, the results in Figs. 5.3(a-b) show a qualitative and quantitative similar behaviour if centered around the corresponding transition frequency.

Figures 5.4(a-b) show the time evolutions of the probabilities $p^{(0)}(t)$ (a) and $p^{(1)}(t)$ (b) as functions of the time t for the effective and the circuit model. Panels(a-b) show the time evolution for the frequencies which cut through the chevron patterns in Figs. 5.3(a-b), i.e. we use the frequencies

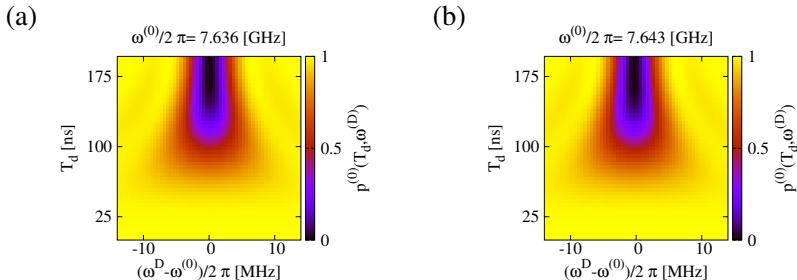


Figure 5.3: Ground-state probabilities $p^{(0)}$ as functions of the pulse duration T_d and the drive frequency $\omega^{(D)}$. We use the system parameters listed in Table 5.1, row $i = 2$ and the pulse in Eq. (5.1) with $T_{r/f} = T_d/2$ and $\delta/2\pi = 0.001$ to obtain the results. Here, we model a microwave pulse, see Fig. 5.2(a). In panel(a) we solve the TDSE for the circuit Hamiltonian in Eq. (3.61). Similarly, in panel(b) solve the TDSE for the effective Hamiltonian in Eq. (3.109). At time $t = 0$ we initialise the systems in the corresponding ground states. We center the two chevron patterns around the transition frequencies $\omega^{(0)} = 7.636$ GHz(a) and $\omega^{(0)} = 7.643$ GHz(b).

$\omega^{(0)} = 7.636$ GHz(a) and $\omega^{(0)} = 7.643$ GHz(b) as drive frequencies. All the other simulation parameters are the same as for Figs. 5.3(a-b).

In Figs. 5.4(a-b) we can observe a good qualitative and quantitative agreement between the time evolutions of the different probabilities for the circuit and the effective model.

Overall, for the microwave pulses we use to model transitions of the form $z = 0 \rightarrow z = 1$, the only distinct difference between the two models is the shift in the transition frequency. However, this shift is something we should expect. The fourth-order cosine expansion of the Hamiltonian Eq. (3.61) leads to an effective model with a spectrum which is not exact, i.e. the spectra of both Hamiltonians do not match exactly.

We performed additional simulations (data not shown), where we include higher-order terms up to the 60th order in the cosine expansion of the Hamiltonian Eq. (3.61). Here we find that the results for the chevron pattern in Fig. 5.4(b) stay the same as long as we centre the results for the ground-state probability around the corresponding transition frequency $\omega^{(0)}$. If we add enough terms to the cosine expansion, the results converge, i.e. the transition frequency converges and no additional shifts at the order of MHz are noticeable anymore.

Also, we should emphasise that the adiabatic effective Hamiltonian Eq. (3.110)

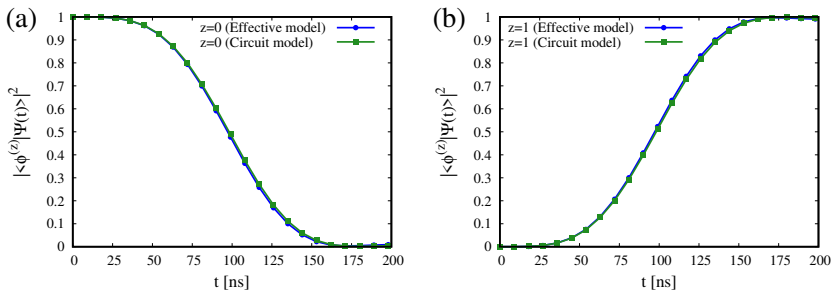


Figure 5.4: Ground-state $p^{(0)}(t)$ (a) and first-excited-state $p^{(1)}(t)$ (b) probabilities as functions of time t for the effective and the circuit model. We use the system parameters listed in Table 5.1, row $i = 2$ and the pulse in Eq. (5.1) with $T_{r/f} = T_d/2$ and $\delta/2\pi = 0.001$ to obtain the results. Here, we model a microwave pulse, see Fig. 5.2(a). Panels(a-b) show the results obtained with the circuit Hamiltonian Eq. (3.61) and the drive frequency $\omega^{(D)}/2\pi = 7.636$ GHz as well as the results obtained with the effective Hamiltonian Eq. (3.109) and the drive frequency $\omega^{(D)}/2\pi = 7.643$ GHz. This means we use the frequencies which cut through the chevron patterns in Figs. 5.3(a-b) as drive frequencies. At time $t = 0$ we initialise the systems in the corresponding ground states.

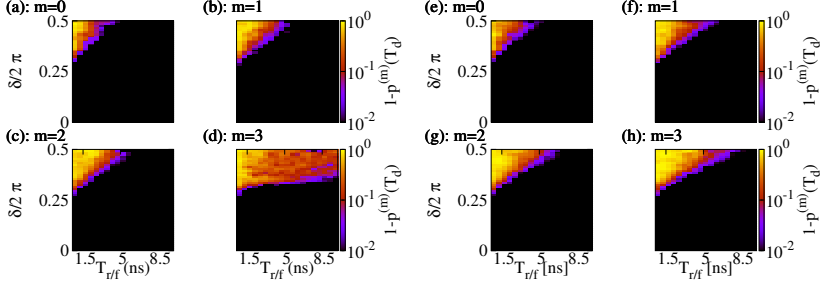


Figure 5.5: Probabilities $1 - p^{(z)}$ for $z \in \{0, 1, 2, 3\}$ as functions of the rise and fall time $T_{r/f}$ and the pulse amplitude δ . We use the system parameters listed in Table 5.1, row $i = 2$ and the pulse in Eq. (5.1) with $T_d = 50$ ns and $\omega^{(D)} = 0$ GHz to obtain the results. Here we model a unimodal pulse, see Fig. 5.2(b). In panels(a-d) we solve the TDSE for the circuit Hamiltonian Eq. (3.61). Similarly, in panels(e-h) we solve the TDSE for the effective Hamiltonian Eq. (3.109). At time $t = 0$ we initialise the systems in the eigenstates $|z\rangle$ for $z \in \{0, 1, 2, 3\}$.

cannot, by definition, be used to model the $z = 0 \rightarrow z = 1$ transitions with a microwave pulse.

We now discuss how the Hamiltonians Eqs. (3.61) and (3.109) react to a unimodal pulse, see Fig. 5.2(b). Figures 5.5(a-h) show the probabilities $1 - p^{(z)}$ at time T_d for $z \in \{0, 1, 2, 3\}$ as functions of the rise and fall time $T_{r/f}$ and the pulse amplitude δ . We use the duration $T_d = 50$ ns and the drive frequency $\omega^{(D)} = 0$ GHz to model the pulse $\varphi(t)$. In panels(a-d) we solve the TDSE for the circuit Hamiltonian Eq. (3.61). Similarly, in panels(e-h) we solve the TDSE for the non-adiabatic effective Hamiltonian Eq. (3.109). At time $t = 0$ we initialise the systems in the corresponding eigenstates $|z\rangle$ for $z \in \{0, 1, 2, 3\}$. Furthermore, we use twenty basis states to obtain the results presented in Figs. 5.5(a-h).

The simulations test whether or not the systems have left the adiabatic regime for a given set of pulse parameters. The bright areas correspond to regions where the systems do not fully remain in their corresponding initial eigenstates and therefore have left the adiabatic regime. As expected, we can observe that this is the case for $T_{r/f} \rightarrow 0$ and $\delta/2\pi \rightarrow 0.5$.

We also find that the circuit Hamiltonian results in panels(a-c) and the effective Hamiltonian results in panels(e-g) for the states $z \in \{0, 1, 2\}$ agree qualitatively. However, the results in panel(d) and panel(h) for $z = 3$ show qualitative and quantitative differences.

Here too, we should emphasise that the adiabatic effective Hamiltonian Eq. (3.110) cannot, by definition, be used to model any of the transitions discussed above.

In Figs. 5.3(a-b) we can observe that the transition frequencies $\omega^{(0)}$ for the effective and the circuit model deviate. Therefore, we want to investigate how well the analytical expressions for the energies $E^{(z)}(\varphi(t))$ actually approximate the numerically exact values $E_{\text{exact.}}^{(z)}(\varphi(t))$. Since the circuit Hamiltonian Eq. (3.61) has two symmetry points, we only have to consider the flux variable on the interval $\varphi/2\pi \in [0, 0.5]$.

If we follow the derivation of the Hamiltonian Eq. (3.109) presented in Sec. 3.4, we find that the energies of the flux-tunable transmon are approximated by the expression

$$(E^{(z)}(\varphi) - E^{(0)}(\varphi)) = \left(z\omega^{(q)}(\varphi) + \frac{\alpha^{(q_0)}}{2}z(z-1) \right), \quad (5.8)$$

where

$$\omega^{(q)}(\varphi) = \omega^{(q_0)} \sqrt[4]{\cos\left(\frac{\varphi}{2}\right)^2 + d^2 \sin\left(\frac{\varphi}{2}\right)^2} + \alpha^{(q_0)}. \quad (5.9)$$

Here $\omega^{(q_0)} = \sqrt{E_C E_\Sigma}$ and $\alpha^{(q_0)} = -E_C/4$ are expressed in terms of the capacitive and Josephson energies which characterise a flux-tunable transmon, see Sec. 3.4. We denote this type of parametrisation as approximation I.

In the literature one can find instances, see for example Refs. [McK+16; Rot+17; Gan+20; Gu+21], where the tunable coupler frequency $\omega^{(q)}(\varphi)$ in Eq. (5.8) is modelled differently. Here one uses

$$\omega^{(q)}(\varphi) = \omega^{(q_0)} \sqrt[4]{\cos\left(\frac{\varphi}{2}\right)^2 + d^2 \sin\left(\frac{\varphi}{2}\right)^2}, \quad (5.10)$$

with $\omega^{(q_0)} = \text{const.}$, $\alpha^{(q_0)} = \text{const.}$ and $d = \text{const.}$ to parameterise the energies of the effective flux-tunable transmon or an experimental flux-tunable transmon. This second parametrisation will be denoted as approximation II.

Additionally, the literature also provides us with more complex expressions for the tunable frequency and the anharmonicity in Eq. (5.8). One such example is given by the authors of Ref. [Did+18]. The corresponding expression for the tunable frequency reads

$$\omega(\varphi) = \sqrt{2E_C E_{J_{\text{eff}}}(\varphi)} - \frac{E_C}{4} \sum_{n=0}^{24} a_n \Xi(\varphi)^n. \quad (5.11)$$

Similarly, the tunable anharmonicity is modelled with the expression

$$\alpha(\varphi) = -\frac{E_C}{4} \sum_{n=0}^{24} b_n \Xi(\varphi)^n. \quad (5.12)$$

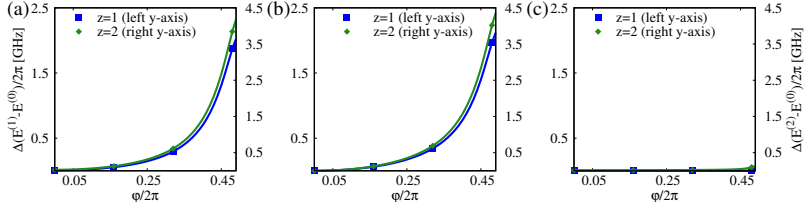


Figure 5.6: Deviations $\Delta(E^{(z)} - E^{(0)})$ given by Eq. (5.14) between the numerically exact eigenvalues of the circuit Hamiltonian Eq. (3.61) and three approximations I(a), II(b) and III(c) for the eigenvalues as functions of the flux $\varphi/2\pi$. The left y-axis, square markers and blue lines, shows the deviations for $z = 1$. Similarly, the right y-axis, diamond markers and green lines, shows the deviations for $z = 2$. We use the energy parameters listed in Table 5.1 row $i = 2$ to obtain the results. The approximations I-III are discussed in the main text, see Eqs. (5.8)–(5.12).

Here a_n and b_n are real-valued coefficients and the function $\Xi(\varphi)$ can be expressed as

$$\Xi(\varphi) = \sqrt{\frac{E_C}{2E_{J_{\text{eff}}}(\varphi)}}. \quad (5.13)$$

We emphasise that both expression are taken from Ref. [Did+18]. This type of parametrisation is denoted as approximation III.

We measure the deviations between the numerically exact values and the approximations I-III with the expression

$$\Delta(E^{(z)}(\varphi) - E^{(0)}(\varphi)) = \left| \left(E_{\text{exact.}}^{(z)}(\varphi) - E_{\text{exact.}}^{(0)}(\varphi) \right) - (E^{(z)}(\varphi) - E^{(0)}(\varphi)) \right|, \quad (5.14)$$

where $z \in \{1, 2\}$. Figures 5.6(a-c) show the deviations $\Delta(E^{(z)}(\varphi) - E^{(0)}(\varphi))$ as functions of the flux φ for the approximations I(a), II(b) and III(c). The left y-axis, square markers and blue lines, shows the deviations for $z = 1$. Similarly, the right y-axis, diamond markers and green lines, shows the deviations for $z = 2$. We use the circuit Hamiltonian Eq. (3.61) and $N_c = 50$ charge basis states to obtain the numerically exact values.

Figure 5.6(a) shows the deviations for approximation I. As one can see, the deviations start to grow slowly and then rapidly increase. In the end, at $\varphi/2\pi = 0.5$ the first energy gap deviates by about 2 GHz and the second

energy gap by about 4 GHz. Figure 5.6(b), for approximation II, shows similar qualitative and quantitative features as Fig. 5.6(a). Figure 5.6(c) shows barely any deviations on the GHz energy scale. Only if $\varphi/2\pi \rightarrow 0.5$, we see some deviations for the second energy gap. The deviations for the first energy gap are at most 4 MHz, for the case presented here.

We also computed the deviations for the flux-tunable transmon parameters listed in Table 5.2 row $i = 0$ and $i = 1$ (data not shown). Here we find similar qualitative features as in Fig. 5.6(a-c). However, the numerical values for the deviations can vary by a couple of GHz for different parameters E_C , $E_{J,l}$ and $E_{J,r}$.

The deviations in terms of the Hamiltonian spectra we can see in Fig. 5.6(a-c) can potentially affect the modelling of non-adiabatic quantum gates, see Ref. [DiC+19; Fox+20], in small NIGQC models. Here we use the pulse amplitude δ to tune the energies of two states into resonance. Furthermore, so far we only discussed the case for a single, isolated flux-tunable transmon. It might be the case that the corresponding errors, in terms of the spectra, for an interacting multi-transmon system are more complex and larger. In such systems, energy levels tend to repel each other such that the resulting behaviour with respect to an external parameter, like the external flux or the interaction strength, leads to complex energy level structures, see for example Ref. [Ber+22].

Additionally, the results in Figs. 5.3(a-b) and Figs. 5.5(a-h) are obtained for the operating point $\varphi_0/2\pi = 0.15$. This means, we should expect that the previously discussed deviations vary with the operating point we use for the simulations.

5.3 Simulations of suppressed transitions in the adiabatic effective two-qubit model: architecture I

In Sec. 5.2, we discussed the case of a single flux-tunable transmon. Here, we use effective and circuit Hamiltonian models to study how such systems react to different external fluxes $\varphi(t)$, see also Eq. (5.1) and Fig. 5.2(a-b). We find that the adiabatic effective Hamiltonian Eq. (3.110) does not allow us, by definition, to model any transitions at all. However, the authors of Refs. [McK+16; Rot+17; Gan+20] show that if one uses the adiabatic effective Hamiltonian Eq. (3.110) to model flux-tunable transmons in architecture I, see Fig. 5.1(a), one can at least model ISWAP and CZ two-qubit transitions. In this section, we investigate whether or not certain single-qubit type

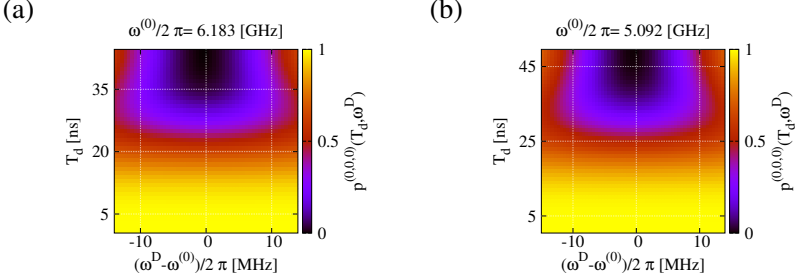


Figure 5.7: Probabilities $p^{\mathbf{z}}$ for $\mathbf{z} = (0, 0, 0)$ as functions of the pulse duration T_d and the drive frequency $\omega^{(D)}$. We use the circuit Hamiltonian Eq. (3.114), the initial state $|0, 0, 0\rangle$, the device parameters listed in Table 5.1 and the control pulse given by Eq. (5.1) with the rise and fall time $T_{r/f} = T_d/2$ to obtain the results. In panel(a) we model the transition $\mathbf{z} = (0, 0, 0) \rightarrow \mathbf{z} = (0, 1, 0)$ and we use the pulse amplitude $\delta/2\pi = 0.045$. In panel(b) we model the transition $\mathbf{z} = (0, 0, 0) \rightarrow \mathbf{z} = (0, 0, 1)$ and we use the pulse amplitude $\delta/2\pi = 0.085$.

transitions are also absent.

As already suggested, we study a system of type architecture I, see Fig. 5.1(a), which contains two fixed-frequency transmons and these are both coupled to a single flux-tunable transmon. Here we use the positive discrete indices z_0 , see Table 5.1 row $i = 0$ and z_1 , see Table 5.1 row $i = 1$, to address the fixed-frequency transmons and the index z_2 , see Table 5.1 row $i = 2$, to address the flux-tunable transmon. To this end, we define the basis state tuple $\mathbf{z} = (z_2, z_1, z_0)$.

Figures 5.7(a-b) show the ground-state probabilities $p^{(0,0,0)}$ as functions of the pulse duration T_d and the drive frequency $\omega^{(D)}$. We use the circuit Hamiltonian Eq. (3.114), the device parameters listed in Table 5.1, the pulse given by Eq. (5.1) with $T_{r/f} = T_d/2$ and three basis states for every transmon to obtain the results. In Fig. 5.7(a), we can observe how the system reacts to pulses with different drive frequencies $\omega^{(D)}$ and the constant pulse amplitude $\delta/2\pi = 0.045$. Here, we model the transition $\mathbf{z} = (0, 0, 0) \rightarrow \mathbf{z} = (0, 1, 0)$. Similarly, in Fig. 5.7(b), we see how the system reacts to pulses with different drive frequencies $\omega^{(D)}$ and the constant pulse amplitude $\delta/2\pi = 0.085$. Here, we model the transition $\mathbf{z} = (0, 0, 0) \rightarrow \mathbf{z} = (0, 0, 1)$. In Figs. 5.7(a-b) we can observe how the system leaves the bare ground state $\mathbf{z} = (0, 0, 0)$ for various pulse parameters.

We find that the corresponding effective model, where the flux-tunable

Table 5.7: Table with grid search parameters and the results of the corresponding searches in terms of the values for $\epsilon = 1 - \min_{(\omega^{(D)}, \delta, t) \in \mathcal{G}} (p^{(0,0,0)}(\omega^{(D)}, \delta, t))$, where $\mathcal{G} \subseteq \mathbb{R}^3$. Here $p^{(0,0,0)}(\omega^{(D)}, \delta, t) = |\langle 0, 0, 0 | \Psi(\omega^{(D)}, \delta, t) \rangle|^2$ denotes the probability for finding the system in the bare ground state when a pulse of the form Eq. (5.1) is applied to the system and $|\Psi(0)\rangle = |0, 0, 0\rangle$. We use the effective Hamiltonian Eq. (3.117) and the parameters listed in Table 5.3 to model a two-qubit system of type architecture I. The first column contains letters which allow us to enumerate the different searches we perform. The second column shows frequency intervals for the drive frequency $\omega^{(D)}$ in units of GHz. The third column shows the search interval for the pulse amplitude δ . The fourth column shows the search interval for the pulse duration T_d in ns. The fifth column shows the results for the different searches in terms of ϵ . We use the step parameters $\Delta\omega^{(D)}/2\pi = 10^{-5}$ GHz, $\Delta\delta/2\pi = 10^{-3}$ and $\Delta t = 0.2$ ns to obtain the results.

Case	$\omega^{(D)}/2\pi$	$\delta/2\pi$	T_d	ϵ
A	[4.90, 5.30]	[0.000, 0.110]	[0, 300]	0.001
B	[6.00, 6.40]	[0.000, 0.110]	[0, 300]	0.001
C	[0.00, 0.00]	[0.000, 0.000]	[0, 300]	0.001

transmon is modelled adiabatically, does not allow us to qualitatively and/or quantitatively reproduce the results we show in Figs. 5.7(a-b). For these simulations we use the adiabatic effective Hamiltonian Eq. (3.117), the device parameters listed in Table 5.1 and Table 5.3, the pulse given by Eq. (5.1) and four basis states for every transmon.

For this reason, we search for the missing transitions $\mathbf{z} = (0, 0, 0) \rightarrow \mathbf{z} = (0, 0, 1)$ and $\mathbf{z} = (0, 0, 0) \rightarrow \mathbf{z} = (0, 1, 0)$ in a more systematic manner. We define the expression

$$\epsilon = 1 - \min_{(\omega^{(D)}, \delta, t) \in \mathcal{G}} (p^{(0,0,0)}(\omega^{(D)}, \delta, t)), \quad (5.15)$$

where $\mathcal{G} \subseteq \mathbb{R}^3$ and $p^{(0,0,0)}(\omega^{(D)}, \delta, t) = |\langle 0, 0, 0 | \Psi(\omega^{(D)}, \delta, t) \rangle|^2$ and evaluate it for a suitable parameter grid \mathcal{G} . If we initialise the system in the bare ground state $|\Psi(0)\rangle = |0, 0, 0\rangle$ and the resulting bare ground state probability $p^{(0,0,0)}(\omega^{(D)}, \delta, t)$ substantially deviates from one for at least one tuple $(\omega^{(D)}, \delta, t) \in \mathbb{R}^3$, we would find $\epsilon \rightarrow 1$. Therefore, we define search grids \mathcal{G} which potentially contain the desired transitions.

Table 5.7 shows the results for three different cases or search parameter grids \mathcal{G} . We define the search grids \mathcal{G} by means of the step parameters $\Delta\omega^{(D)}/2\pi = 10^{-5}$ GHz, $\Delta\delta/2\pi = 10^{-3}$ and $\Delta t = 0.2$ ns and search intervals

for the drive frequency $\omega^{(D)}$, the pulse amplitude δ and the pulse duration T_d . We use the effective Hamiltonian Eq. (3.117), the device parameters listed in Table 5.3 and the pulse given by Eq. (5.1) to obtain the results. Note that there exist several ways to modify the effective Hamiltonian model. We can choose between two different sets of functions to model the tunable transmon qubit frequency and anharmonicity and we can perform simulations with static and dynamic interaction strengths $g^{(c,c)}(t)$. Furthermore, we can mix the different cases. We performed simulations for all four scenarios and we find that the numerical results are identical, up to the third decimal in terms of ϵ . Therefore, we only present the results for one case.

The first row shows the case where we search for the transition $\mathbf{z} = (0, 0, 0) \rightarrow \mathbf{z} = (0, 0, 1)$. Here we find the value $\epsilon = 0.001$. The second row shows the case where we search for the transition $\mathbf{z} = (0, 0, 0) \rightarrow \mathbf{z} = (0, 1, 0)$. Here too, we find the value 0.001 for ϵ . The third row shows the case for the free time evolution and again we find $\epsilon = 0.001$. Consequently, for the search grids we define in Table 5.7, we find that the bare ground state probability $p^{(0,0,0)}(\omega^D, \delta, t)$ reacts to various pulses, see row one and two, in the same way it reacts to the absence of a pulse, see third row. Note that the chevron patterns in Fig. 5.7(a-b) are several MHz wide. Therefore, we might conclude that these transitions are absent from the model specified above. Obviously, we cannot make this claim with absolute certainty. It is possible, that there exist other pulse parameters which might allow us to model these transitions. However, the results in Table 5.7 indicate the absence of the transitions we find in the circuit Hamiltonian model.

So far we only considered the transitions $\mathbf{z} = (0, 0, 0) \rightarrow \mathbf{z} = (0, 0, 1)$ and $\mathbf{z} = (0, 0, 0) \rightarrow \mathbf{z} = (0, 1, 0)$. However, we find that more such transitions are absent in the effective Hamiltonian model, i.e. we can activate these transitions in the circuit Hamiltonian model but not in the effective Hamiltonian model. For example, similar to the results we discuss in Sec. 5.2, we are able to excite the flux-tunable transmon itself $\mathbf{z} = (0, 0, 0) \rightarrow \mathbf{z} = (1, 0, 0)$ by applying a pulse with a drive frequency $\omega^{(D)}$ near 7.6 GHz.

Furthermore, the results in Sec. 5.2 suggest that the non-adiabatic driving term in Eq. (5.3) can possibly be used to model the suppressed transitions. Therefore, we included the driving term into the model and performed the corresponding simulations, see Figs. 5.7(a-b), again. We find (data not shown) that the transitions $\mathbf{z} = (0, 0, 0) \rightarrow \mathbf{z} = (0, 0, 1)$ and $\mathbf{z} = (0, 0, 0) \rightarrow \mathbf{z} = (0, 1, 0)$ and others can be modelled by including the non-adiabatic driving term in Eq. (5.3).

5.4 Simulations of unsuppressed transitions in the adiabatic effective two-qubit model: architecture I

In Sec. 5.2 we study a single flux-tunable transmon, see Table 5.1 row $i = 2$ and we find that various transitions between the states of this system can be modelled with the circuit Hamiltonian Eq. (3.61) but not with the adiabatic effective Hamiltonian Eq. (3.110). Similarly, in Sec. 5.3, we study a two-qubit system of type architecture I, see Fig. 5.1(a), consisting of two fixed-frequency transmons and a single flux-tunable transmon. Here, we identify transitions which can be modelled with the circuit Hamiltonian model. However, it seems to be the case that these transitions are absent in the associated adiabatic effective model, where the flux-tunable transmon is modelled by means of Eq. (3.110). In this section, we study the same two-qubit system, but here we focus on two different transitions which can be described with both Hamiltonian models.

The system we model is of type architecture I, see Fig. 5.1(a). We use the positive discrete indices z_0 and z_1 to address the two fixed-frequency transmons, see Table 5.1 row $i = 0$ and $i = 1$, respectively. These are the transmons which are usually used as qubits in this particular architecture, see for example Refs. [McK+16; Rot+17; Gan+20]. Similarly, we use the index z_2 to address the flux-tunable transmon, see Table 5.1 row $i = 2$, which acts as a coupler element, i.e. this subsystem is supposed to convey interactions between the two fixed-frequency transmons we use as qubits. Finally, we address the different bare basis states $|\mathbf{z}\rangle$ of the system, see also Sec. 4.3 and 4.4, by means of the tuples $\mathbf{z} = (z_2, z_1, z_0)$.

In the following, we study two different transitions: the $\mathbf{z} = (0, 0, 1) \rightarrow \mathbf{z} = (0, 1, 0)$ transition is sometimes used to implement ISWAP gates and the $\mathbf{z} = (0, 1, 1) \rightarrow \mathbf{z} = (0, 2, 0)$ transition is often used to implement CZ gates. References [McK+16; Rot+17; Rot20; Gan+20] discuss systems of type architecture I in this context.

Figures 5.8(a-d) show the probabilities $p^{\mathbf{z}}(t)$ for $\mathbf{z} = (0, 0, 1)$ and $\mathbf{z} = (0, 1, 0)$ as functions of time t . Here, $p^{\mathbf{z}}(t) = |\langle \mathbf{z} | \Psi(t) \rangle|^2$ and $|\Psi(0)\rangle = |0, 0, 1\rangle$. We use the circuit Hamiltonian Eq. (3.114), the device parameters listed in Table 5.1 and the pulse given by Eq. (5.1) to describe the dynamics of the system. The pulse is modelled with the drive frequency $\omega^{(D)}/2\pi = 1.089$ GHz, the pulse amplitude $\delta/2\pi = 0.075$, the rise and fall time $T_{r/f} = 13.00$ ns and the pulse duration $T_d = 209.40$ ns. We model the transition with different numbers of basis states n_J . We use $n_J = 3$ in Fig. 5.8(a), $n_J = 4$ in

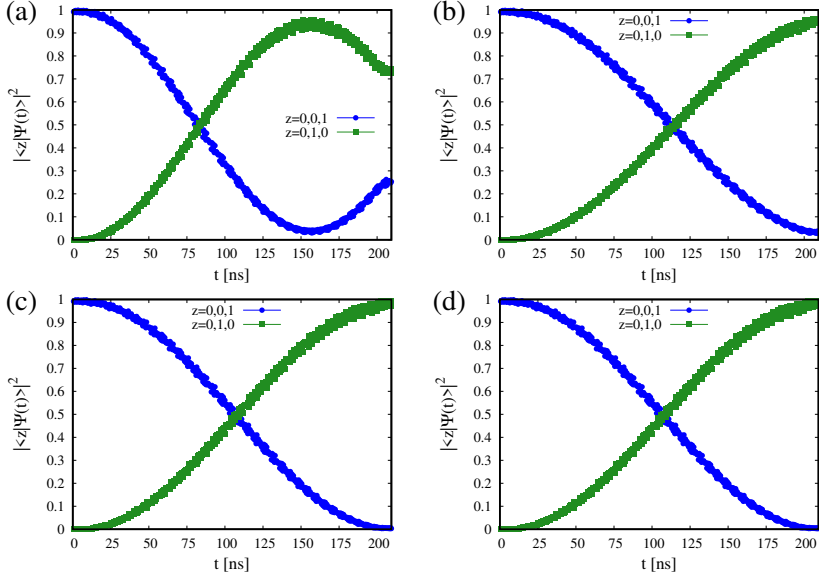


Figure 5.8: Probabilities $p^z(t) = |\langle \mathbf{z} | \Psi(t) \rangle|^2$ as a functions of time t for $\mathbf{z} = (0, 0, 1)$ and $\mathbf{z} = (0, 1, 0)$. Here, $|\Psi(0)\rangle = |0, 0, 1\rangle$. We use the circuit Hamiltonian Eq. (3.114), the device parameters listed in Table 5.1 and the pulse given by Eq. (5.1) to generate the results. The pulse parameters are the drive frequency $\omega^{(D)}/2\pi = 1.089$ GHz, the amplitude $\delta/2\pi = 0.075$, the rise and fall $T_{r/f} = 13.00$ ns and the pulse duration $T_d = 209.40$ ns. Furthermore, n_J denotes the number of flux-tunable transmon basis states and we use $n_J = 3$ in panel(a), $n_J = 4$ in panel(b), $n_J = 6$ in panel(c) and $n_J = 15$ in panel(d) to model the dynamics of the system. All pulse parameters are listed in Table 5.5.

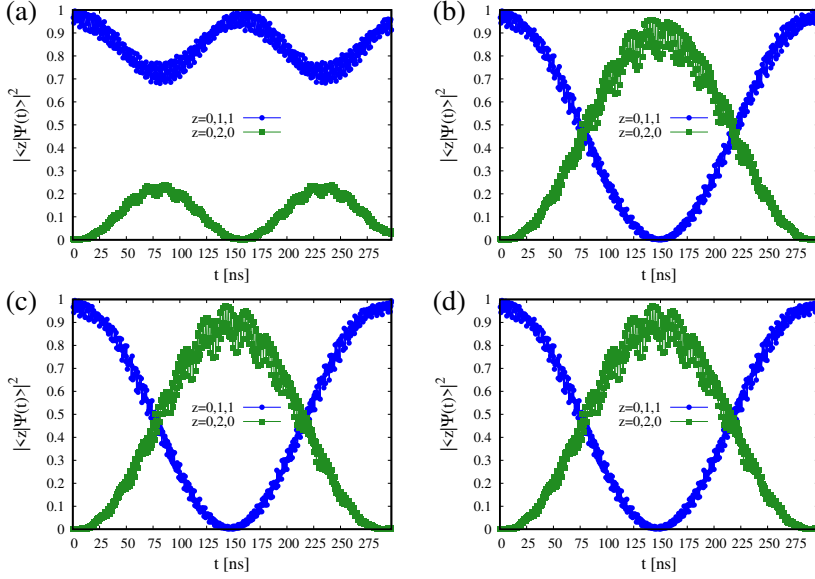


Figure 5.9: Probabilities $p^z(t) = |\langle \mathbf{z} | \Psi(t) \rangle|^2$ as a functions of time t for $\mathbf{z} = (0, 1, 1)$ and $\mathbf{z} = (0, 2, 0)$. Here, $|\Psi(0)\rangle = |0, 1, 1\rangle$. We use the circuit Hamiltonian Eq. (3.114), the device parameters listed in Table 5.1 and the pulse given by Eq. (5.1) to obtain the results. The pulse parameters are the drive frequency $\omega^{(D)}/2\pi = 0.809$ GHz, the amplitude $\delta/2\pi = 0.085$, the rise and fall time $T_{r/f} = 13.00$ ns and the pulse duration $T_d = 297.55$ ns. Furthermore, n_J denotes the number of flux-tunable transmon basis states and we use $n_J = 3$ in panel(a), $n_J = 4$ in panel(b), $n_J = 8$ in panel(c) and $n_J = 15$ in panel(d) to model the dynamics of the system. All pulse parameters are listed in Table 5.5.

Fig. 5.8(b), $n_J = 6$ in Fig. 5.8(c) and $n_J = 15$ Fig. 5.8(d).

As can be seen, Figs. 5.8(a-d) show an exchange of population between the states $\mathbf{z} = (0, 0, 1)$ and $\mathbf{z} = (0, 1, 0)$. However, we observe that the time evolution of the probabilities changes with the number of basis states n_J we use to model the dynamics of the system. Figures 5.8(a-b) show that at time T_d there is no full swap between the states $\mathbf{z} = (0, 0, 1)$ and $\mathbf{z} = (0, 1, 0)$. In Fig. 5.8(c) the solution of the TDSE has converged up to the third decimal. Figure 5.8(d) functions as a graphical reference solution, i.e. we cannot find any noticeable differences between Fig. 5.8(c) and Fig. 5.8(d). In this case it suffices to describe the dynamics of the system with $n_J = 6$ basis states. Furthermore, the functions $p^{\mathbf{z}}(t)$ in Figs. 5.8(a-d) exhibit small oscillations at the order of nanoseconds. The oscillations seem to increase with the amount of population which resides in the corresponding state \mathbf{z} .

Figures 5.9(a-d) show the probabilities $p^{\mathbf{z}}(t) = |\langle \mathbf{z} | \Psi(t) \rangle|^2$ for $\mathbf{z} = (0, 1, 1)$ and $\mathbf{z} = (0, 2, 0)$ as functions of time t . The initial state of the system is $|\Psi(0)\rangle = |0, 1, 1\rangle$. We use the same Hamiltonian, device parameters and pulse as in Figs. 5.8(a-d). The pulse is modelled with the drive frequency $\omega^{(D)}/2\pi = 0.809$ GHz, the pulse amplitude $\delta/2\pi = 0.085$, the rise and fall time $T_{r/f} = 13.00$ ns and the pulse duration $T_d = 297.55$ ns. Furthermore, we use $n_J = 3$ in Fig. 5.9(a), $n_J = 4$ in Fig. 5.9(b), $n_J = 8$ in Fig. 5.9(c) and $n_J = 15$ Fig. 5.9(d).

Figures 5.9(a-d) exhibit similar features as Figs. 5.8(a-d). First, we observe that we need more than three or four basis states to model the dynamics of the system. In this case we have to use at least $n_J = 8$ basis states. Second, we see that the functions $p^{\mathbf{z}}(t)$ oscillate at the order of nanoseconds. We also observe that the oscillations in Figs. 5.9(c-d) at around 150 ns seem to be stronger than everything we observe in Figs. 5.8(c-d).

We now turn our attention to the effective Hamiltonian model. As previously discussed, see Sec. 5.2 and 5.3, we might use different functional dependencies to model the tunable energies and interaction strengths. For example, in Refs. [McK+16; Rot+17; Rot20; Gan+20], the authors use models where the interaction strengths are assumed to be constant. Furthermore, here the energies are modelled with $\omega^{(q)}(\varphi)$ given by Eq. (5.10) and $\alpha^{(q\alpha)}(\varphi) = \text{const.}$ The simulation software allows us to switch between different model options or rather assumptions and in the following we intend to make use of this functionality. This means we perform simulations of all four modelling scenarios: simulations with and without an adjusted spectrum as well as simulations with and without time-dependent interaction strengths. Here we use the adiabatic effective Hamiltonian Eq. (3.117), the pulse given by Eq. (5.1) and four basis states for every transmon to model the two-qubit system with the effective model. Furthermore, we either use the device parameters listed in

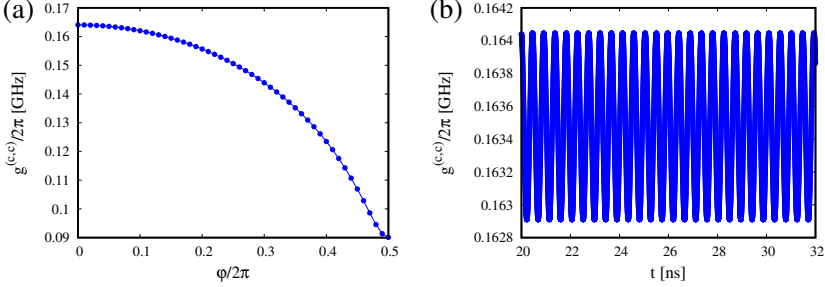


Figure 5.10: Effective interaction strength $g^{(c,c)}$ between a fixed-frequency and a flux-tunable transmon as a function of the external flux φ in panel(a) and as a function of time t in panel(b). We use Eq. (5.4) and the energies listed in Table 5.1, row $i = 1$ and $i = 2$, to obtain the numerical values for $g^{(c,c)}$. In panel(a) we compute the effective interaction strength $g^{(c,c)}(\varphi)$ for the interval $\varphi/2\pi \in [0, 0.5]$. In panel(b) we use the control pulse $\varphi(t)$ given by Eq. (5.1) with the flux-offset value $\varphi_0/2\pi = 0.15$, the drive frequency $\omega^{(D)}/2\pi = 1.089$ GHz, the pulse amplitude $\delta/2\pi = 0.075$, the rise and fall time $T_{r/f} = 13.0$ ns and the pulse duration $T_d = 205.4$ ns to obtain the time evolution of the effective interaction strength.

Table 5.1 or the parameters listed in Table 5.3 to model the dynamics. The parameters in Table 5.1 are necessary once we use Eq. (5.11) and Eq. (5.12) to model the tunable energies of the system or once we use Eq. (5.4) to model the time-dependent interaction strengths. Otherwise, we can use the parameters in Table 5.3.

For the device parameters and the control pulse we use, we find (data not shown) that the adjusted spectrum barely affects the simulation results. Therefore, we only present the results for cases where $\omega^{(q)}(\varphi)$ given by Eq. (5.10) and $\alpha^{(q_0)}(\varphi) = \text{const.}$ are used to model the tunable energies of the flux-tunable transmon z_2 . Still, we present the results for both interaction strength cases.

Figure 5.10(a) shows the interaction strength $g^{(c,c)}$ as a function of the flux variable φ . Similarly, in Fig. 5.10(b) we show the interaction strength $g^{(c,c)}$ as a function of time t . We use Eq. (5.4) and the energy parameters listed in Table 5.1, row $i = 1$ and $i = 2$ to obtain $g^{(c,c)}$ for various flux values φ . The time evolution in Fig. 5.10(b) is modelled with the pulse $\varphi(t)$ given by Eq. (5.1). Here we use the flux-offset value $\varphi_0/2\pi = 0.15$, the drive frequency $\omega^{(D)}/2\pi = 1.089$ GHz, the pulse amplitude $\delta/2\pi = 0.075$, the rise and fall time $T_{r/f} = 13.0$ ns and the pulse duration $T_d = 205.4$ ns.

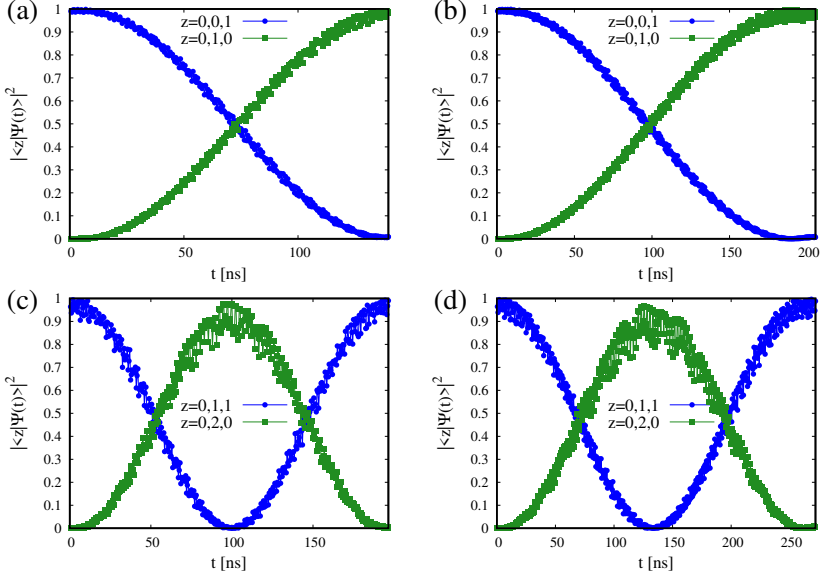


Figure 5.11: Probabilities $p^z(t) = |\langle \mathbf{z} | \Psi(t) \rangle|^2$ as functions of time t . In panels(a,b) we model the transitions $\mathbf{z} = (0, 0, 1) \rightarrow \mathbf{z} = (0, 1, 0)$. In panels(c,d) we model the transitions $\mathbf{z} = (0, 1, 1) \rightarrow \mathbf{z} = (0, 2, 0)$. We use the effective Hamiltonian Eq. (3.117), the device parameters listed in Table 5.3 and the pulse given by Eq. (5.1) with the rise and fall time $T_d = 13$ ns to obtain the results. The remaining, non-specified pulse parameters are the drive frequency $\omega^{(D)}$, the amplitude δ and the pulse duration T_d . Most of these parameters change for every panel we show. We use $\omega^{(D)}/2\pi = 1.088$ GHz, $\delta/2\pi = 0.075$ and $T_d = 139.6$ ns in panel(a), $\omega^{(D)}/2\pi = 1.089$ GHz, $\delta/2\pi = 0.075$ and $T_d = 205.4$ ns in panel(b), $\omega^{(D)}/2\pi = 0.807$ GHz, $\delta/2\pi = 0.085$ and $T_d = 196.5$ ns in panel(c) as well as $\omega^{(D)}/2\pi = 0.807$ GHz, $\delta/2\pi = 0.085$ and $T_d = 272.0$ ns in panel(d). The first-order series expansion is used to model the tunable qubit frequency for all panels. In panels(a,c) we use $g^{(c,c)}(t) = \text{const.}$ to model a static effective interaction strength. Similarly, in panels(b,d) we use $g^{(c,c)}(t)$ given by Eq. (5.4) to model a dynamic effective interaction strength, see also Fig. 5.10(b). All pulse parameters and cases are listed in Table 5.6.

As can be seen in Fig. 5.10(a), the interaction strength varies by about 75 MHz over the range $\varphi/2\pi \in [0, 0.5]$. Furthermore, in Fig. 5.10(b) we observe a variation of the interaction strength which is at the order of 1 MHz. The control pulse $\varphi(t)$ we use to model the time evolution of $g^{(c,c)}$ is also used to model the transition $\mathbf{z} = (0, 0, 1) \rightarrow \mathbf{z} = (0, 1, 0)$ with the effective model, see Table 5.6 row six. The results in Fig. 5.10(b) suggest that the assumption of a static interaction strength, see Refs. [McK+16; Rot+17; Rot20; Gan+20], might be justified. In the following, we show two counterexamples.

Figures 5.11(a-d) show the probabilities $p^{\mathbf{z}}(t) = |\langle \mathbf{z} | \Psi(t) \rangle|^2$ as functions of time t . In Figs. 5.11(a-b) we model the $\mathbf{z} = (0, 0, 1) \rightarrow \mathbf{z} = (0, 1, 0)$ transition with, see Fig. 5.11(b) and without, see Fig. 5.11(a), a time-dependent interaction strength. Similarly, in Figs. 5.11(c-d) we model the $\mathbf{z} = (0, 1, 1) \rightarrow \mathbf{z} = (0, 2, 0)$ transition with, see Fig. 5.11(d) and without, see Fig. 5.11(c), a time-dependent interaction strength.

We determined the pulse parameters for Figs. 5.11(a-d) as follows. First, we fix the pulse amplitude δ and rise and fall time $T_{r/f}$. Here, we use the parameters we found for the circuit Hamiltonian model, see Table 5.5. Second, we perform a drive frequency $\omega^{(D)}$ spectroscopy and determined the frequency at the center of the chevron pattern, see also Fig. 5.7(a-b). This means we use the frequency at the center of the chevron pattern as the drive frequency. At last, we determine the pulse duration T_d for this set of parameters. The pulse parameters are the rise and fall time $T_{r/f} = 13.0$ ns in Figs. 5.11(a-d), $\omega^{(D)}/2\pi = 1.088$ GHz, $\delta/2\pi = 0.075$ and $T_d = 139.6$ ns in Fig. 5.11(a), $\omega^{(D)}/2\pi = 1.089$ GHz, $\delta/2\pi = 0.075$ and $T_d = 205.4$ ns in Fig. 5.11(b), $\omega^{(D)}/2\pi = 0.807$ GHz, $\delta/2\pi = 0.085$ and $T_d = 196.5$ ns in Fig. 5.11(c) as well as $\omega^{(D)}/2\pi = 0.807$ GHz, $\delta/2\pi = 0.085$ and $T_d = 272.0$ ns in Fig. 5.11(d). A summary of these parameters is provided in Table 5.6, row five to eight.

We first discuss the results for the $\mathbf{z} = (0, 0, 1) \rightarrow \mathbf{z} = (0, 1, 0)$ transition. As one can see, the probabilities in Figs. 5.11(a-b) show a qualitatively similar behaviour. However, we can observe a quantitative difference between both results, namely that the pulse duration T_d in Fig. 5.11(b) is roughly 65.8 ns longer. Note that in Fig. 5.11(b) we model the system with a time-dependent interaction strength. Furthermore, the circuit Hamiltonian model predicts a gate duration T_d of 209.4 ns. Consequently, there is a 4 ns discrepancy between the circuit Hamiltonian model and the effective model with a time-dependent interaction strength and a 69.8 ns difference between the circuit Hamiltonian model and the effective model without a time-dependent interaction strength.

Now we turn our attention to the $\mathbf{z} = (0, 1, 1) \rightarrow \mathbf{z} = (0, 2, 0)$ transition. Figures 5.11(c-d) exhibit similar features as Figs. 5.11(a-b). First, we

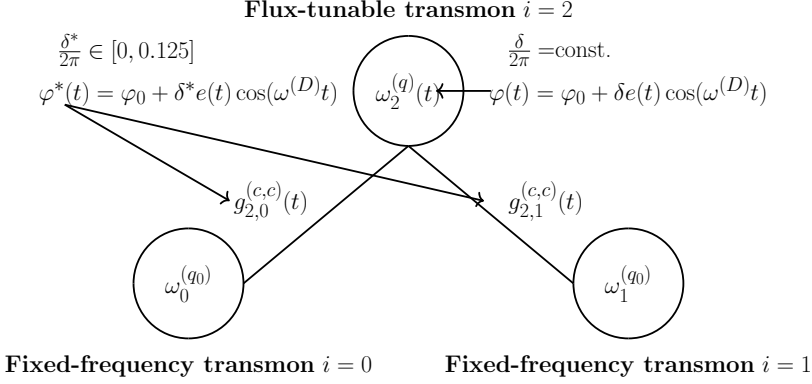


Figure 5.12: Sketch of the simulation scenario we use to obtain the results shown in Figs. 5.13(a-b). The sketch shows architecture I, see Fig. 5.1(a), with different time dependencies $\varphi^*(t)$ and $\varphi(t)$ given by Eq. (5.1) for the control pulse or external flux. We model the time-dependent interaction strengths $g^{(c,c)}(t)$ given by Eq. (5.4) with the pulse $\varphi^*(t)$ and the tunable frequency $\omega^{(q)}(t)$ with the pulse $\varphi(t)$. The only difference between these two pulses is that we model the pulse $\varphi^*(t)$ with the amplitudes $\delta^*/2\pi \in [0, 0.125]$ and the pulse $\varphi(t)$ with the amplitude $\delta/2\pi = \text{const.}$ This allows us to observe the transitioning behaviour from a model with a static interaction strength $\delta^* = 0$ to a model with a dynamic interaction strength $\delta^* = \delta$. We show the results for this scenario because the combined effects of changing the coupler amplitude δ and the interaction strength amplitude δ^* together mixes up two different mechanisms which affect the ISWAP and CZ transitions. Furthermore, in Fig. 5.13(a) we use the same amplitude $\delta/2\pi = 0.075$ we use to model the ISWAP transitions in Fig. 5.11(a-b). Similarly, in Fig. 5.13(b) we use the same amplitude $\delta/2\pi = 0.085$ we use to model the CZ transitions in Fig. 5.11(c-d).

find that Figs. 5.11(c-d) show a similar qualitative behaviour in terms of the time evolution of the probabilities $p^z(t)$. Second, we see a shift in the pulse duration T_d between both effective models. The pulse duration difference between both effective models is 75.5 ns. For this case, the circuit Hamiltonian predicts a gate duration of $T_d = 297.55$ ns. Therefore, we find a deviation of 25.55 ns between the effective model with time-dependent interaction strength and the circuit model and a difference of 101.05 ns between the effective model without a time-dependent interaction strength and the circuit model.

Next, we investigate how the time-dependent interaction strength $g^{(c,c)}(t)$ affects the gate duration T_d for the ISWAP and CZ transitions in more detail. To this end, we implement the simulation scenario presented in Fig. 5.12. The results for this scenario are displayed in Figs. 5.13(a-b). The sketch in Fig. 5.12 shows architecture I with two different control pulses $\varphi(t)$ and $\varphi^*(t)$. Both pulses only deviate in terms of the amplitudes, i.e. we model the pulse $\varphi^*(t)$ with the amplitudes $\delta^* \in [0, 0.125]$ and the pulse $\varphi(t)$ with the amplitude $\delta = \text{const.}$. The pulse $\varphi^*(t)$ is used to control the time-dependent interaction strengths $g^{(c,c)}(t)$ given by Eq. (5.4). We use the other pulse $\varphi(t)$ to control the tunable frequency $\omega^q(t)$ of the coupler element. In order to obtain the results in Figs. 5.13(a-b), we use the amplitudes $\delta/2\pi = 0.075$ (a) and $\delta/2\pi = 0.085$ (b). This means, if the pulse amplitude is set to zero $\delta^* = 0$, we model the cases we present in Figs. 5.11(a,c). Furthermore, if we set both pulse amplitudes equal $\delta^* = \delta$, we model the cases we present in Figs. 5.11(b,d). All pulses are modelled with the rise and fall time $T_{r/f} = 13$ ns and the pulse duration $T_d = 300$ ns. This simulation scenario allows us to separate the two mechanisms which affect the ISWAP and CZ transitions. On the one hand, we have the time-dependent coupler which can be used to activate the ISWAP and CZ transitions in the adiabatic effective model without a time-dependent interaction strength. On the other hand, we have the time-dependent interaction strength which seems to affect the time evolution of the system.

Figures 5.13(a-b) show the probabilities $p^{\mathbf{z}}(t) = |\langle \mathbf{z} | \Psi(t) \rangle|^2$ for $\mathbf{z} = (0, 0, 1)$ (a) and $\mathbf{z} = (0, 1, 1)$ (b) as functions of time t . In panel(a) we model $\mathbf{z} = (0, 0, 1) \rightarrow \mathbf{z} = (0, 1, 0)$ ISWAP transitions with the drive frequencies $\omega^{(D)}/2\pi = 1.088$ GHz, blue lines and unfilled markers and $\omega^{(D)}/2\pi = 1.089$ GHz, green lines and filled markers. In panel(b) we model $\mathbf{z} = (0, 1, 1) \rightarrow \mathbf{z} = (0, 2, 0)$ CZ transitions with the drive frequencies $\omega^{(D)}/2\pi = 0.807$ GHz, blue lines and unfilled markers and $\omega^{(D)}/2\pi = 0.808$ GHz, green lines and filled markers.

In Fig. 5.13(a), we see that for the pulse amplitudes $\delta^*/2\pi \in [0, 0.01]$ there is barely a change in the first minima of function $p^{(0,0,1)}(t)$. However, for the pulse amplitudes $\delta^*/2\pi \in (0.01, 0.125]$ we can observe increases of roughly 25 ns for the different amplitudes. In Fig. 5.13(b), we find similar features for the function $p^{(0,1,1)}(t)$.

The results in Figs. 5.11(a-d) and Figs. 5.13(a-b) raise the question, why do the oscillations of the interaction strength $g^{(c,c)}(t)$ affect the gate duration T_d so strongly? Despite performing additional simulations, we were not able to find a conclusive answer to this question. Consequently, we have to leave this question for future research.

There remains one other obvious open question in this section, namely

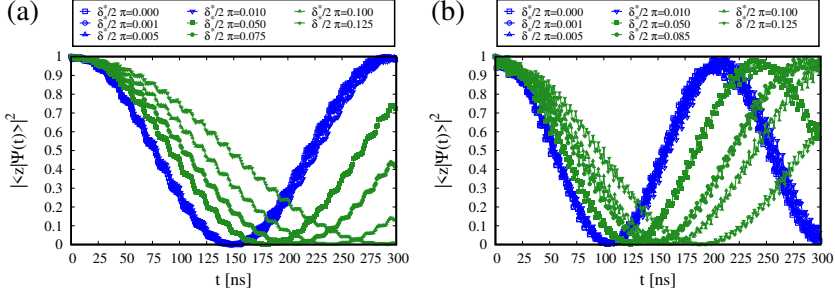


Figure 5.13: Probabilities $p^{\mathbf{z}}(t) = |\langle \mathbf{z} | \Psi(t) \rangle|^2$ for $\mathbf{z} = (0, 0, 1)$ (a) and $\mathbf{z} = (0, 1, 1)$ (b) as functions of time t . Here, we model the transition from a model with a static interaction strength to a model with a dynamic interaction strength as illustrated in Fig. 5.12. This means we model the time-dependent interaction strength given by Eq. (5.4) with the pulse $\varphi^*(t)$ in Eq. (5.1) and the pulse amplitudes $\delta^*/2\pi \in [0, 0.125]$. The tunable frequency of the coupler is modelled with the pulse $\varphi(t)$ in Eq. (5.1). However, here we use different constant pulse amplitudes $\delta/2\pi = 0.075$ (a) and $\delta/2\pi = 0.085$ (b). The remaining pulse parameters are the same for both pulses. The rise and fall time $T_{r/f} = 13$ ns and the pulse duration $T_d = 300$ ns are the same for all pulses. In panel(a) we use the drive frequencies $\omega^{(D)}/2\pi = 1.088$, blue lines and unfilled markers and $\omega^{(D)}/2\pi = 1.089$, green lines and filled markers. Similarly, in panel(b) we use the drive frequencies $\omega^{(D)}/2\pi = 0.807$, blue lines and unfilled markers and $\omega^{(D)}/2\pi = 0.808$, green lines and filled markers. If $\delta^* = 0$, we model the system with a static interaction strength. Therefore, we model the scenarios in Figs. 5.11(a,c). If $\delta^* = \delta$, we model the scenarios in Figs. 5.11(b,d).

does the non-adiabatic driving term in Eq. (5.3) affect the ISWAP and CZ transitions we model in this section. Here, we talk about the driving term that we implicitly investigated in Sec. 5.2. Broadly speaking, we can answer this question as follows: we find (data not shown) that the non-adiabatic driving term in Eq. (5.3) barely affects the ISWAP and CZ transitions we model in this section. If we want to be precise, we have to say that the probabilities at time T_d are at most affected by the third decimal and the overall qualitative transitioning behaviour does not seem to be affected at all. Note that we only verified this for the relevant pulse parameters in Table 5.6. It is possible that larger drive frequencies $\omega^{(D)}$ and pulse amplitudes δ can cause larger deviations between the adiabatic and non-adiabatic effective models. Furthermore, if the drive frequency $\omega^{(D)}$ comes energetically near various other transitions, see Sec. 5.3, we can expect to see additional deviations between the adiabatic and non-adiabatic effective Hamiltonian.

In summary, we find that for short time scales around 250 ns, the adiabatic effective model with time-dependent interaction strength and the circuit Hamiltonian model predict similar outcomes in terms of the probabilities $p^{\mathbf{z}}(t) = |\langle \mathbf{z} | \Psi(t) \rangle|^2$ which result from the state vector $|\Psi(t)\rangle$ for the corresponding model. Additionally, the deviations between the effective model with static interaction strength and the circuit Hamiltonian model for the same scenarios seem to be too strong to be neglected.

5.5 Simulations of suppressed transitions in the adiabatic effective two-qubit model: architecture II

In Sec. 5.3 and 5.4, we studied the relation between the circuit and effective Hamiltonian models for a two-qubit system of type architecture I. In this section, we do the same, but we consider another two-qubit system, namely a system of type architecture II, see Fig. 5.1(b).

The system consists of two flux-tunable transmons which are coupled to a single resonator element. We use the positive discrete indices z_0 and z_1 to index the basis states of the first flux-tunable transmon, see Table 5.2 row $i = 0$ and the second flux-tunable transmon, see Table 5.2 row $i = 1$, respectively. The coupler basis states are indexed by means of the positive discrete index z_2 , see Table 5.2 row $i = 2$. The individual transmon and harmonic oscillator basis states are discussed in Sec. 3.3 and 3.4, respectively. We address the bare basis states $|\mathbf{z}\rangle = |z_2, z_1, z_0\rangle$ of the two-qubit system by means of the tuples $\mathbf{z} = (z_2, z_1, z_0)$. The ground $z_{0/1} = 0$ and first-excited

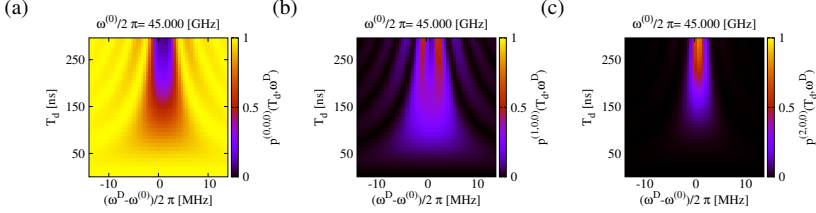


Figure 5.14: Probabilities $p^{\mathbf{z}}$ for $\mathbf{z} = (0, 0, 0)$ (a), $\mathbf{z} = (1, 0, 0)$ (b) and $\mathbf{z} = (2, 0, 0)$ (c) as functions of the pulse duration T_d and the drive frequency $\omega^{(D)}$. We use the circuit Hamiltonian Eq. (3.114), the initial state $|0, 0, 0\rangle$, the device parameters listed in Table 5.2 and the control pulse given by Eq. (5.1) with the rise and fall time $T_{r/f} = T_d/2$ and the pulse amplitude $\delta/2\pi = 0.020$ to obtain the results. The results are centered around the eigenfrequency of the coupling resonator in Fig. 5.1(b).

$z_{0/1} = 1$ transmon states are used as qubit states. This choice is motivated by the work in Refs. [Rol+19; Bla+21; Kri+20].

The simulations in this section are motivated by three observations. First, while performing simulations with the circuit Hamiltonian Eq. (3.114) to obtain the results described in Sec. 5.6, we noticed (data now shown) that the coupler states can be excited considerably. Note that the results in Sec. 5.6 are obtained with a unimodal pulse, see Fig. 5.2(b). Excitations of the coupling resonator usually occurred for pulses with fast rising (falling) pulse flanks. Second, the authors of Ref. [Rol+19] neglect the coupler states by assumption. Note that the authors of Ref. [Rol+19] investigate a system that consists of two flux-tunable transmons coupled via a coupling resonator. Here, a so-called bimodal pulse is used to drive the system. Third, in Sec. 5.4 we found that certain transitions between the states of the system illustrated in Fig. 5.1(a) are seemingly suppressed in the adiabatic effective model for architecture I. In this section we study transitions which are seemingly suppressed in the adiabatic effective model for architecture II.

Figures 5.14(a-c) show the probabilities $p^{\mathbf{z}}$ for $\mathbf{z} = (0, 0, 0)$ (a), $\mathbf{z} = (1, 0, 0)$ (b) and $\mathbf{z} = (2, 0, 0)$ (c) as functions of the pulse duration T_d and the drive frequency $\omega^{(D)}$. Here, we use the circuit Hamiltonian Eq. (3.114), the device parameters listed in Table 5.2 and the control pulse given by Eq. (5.1) to obtain the results. The control pulse parameters are $\delta/2\pi = 0.020$ and $T_{r/f} = T_d/2$, see Fig. 5.2(a). The system is initialised in the ground state $|0, 0, 0\rangle$. Furthermore, we use $n_J = 4$ and $n_K = 4$ basis states to model the dynamics of the system.

We can clearly observe transitions from the ground state $\mathbf{z} = (0, 0, 0)$ to the first-excited $\mathbf{z} = (1, 0, 0)$ and second-excited $\mathbf{z} = (2, 0, 0)$ states of the coupling resonator. The results for the third-excited state $\mathbf{z} = (3, 0, 0)$ are omitted for simplicity. Note that the energy gap between the ground state and the first-excited coupler state are roughly the same as the energy gap between the first-excited coupler state and the second-excited coupler state.

Additionally, we performed analogous simulations with the adiabatic and non-adiabatic effective Hamiltonian Eq. (3.117). We find (data not shown) that the non-adiabatic effective Hamiltonian Eq. (3.117) allows us to model the transitions displayed in Figs. 5.14(a-c). Here, we modelled the non-adiabatic effective Hamiltonian Eq. (3.117) with and without a time-dependent interaction strength. Furthermore, we adjusted the spectrum as discussed in Sec. 5.2. The transitions in Fig. 5.14(a-c) can be activated with all four effective models. However, the adiabatic effective Hamiltonian Eq. (3.117) does not react to the control pulse in the same manner, i.e. we cannot find any excitations of the ground state. Here again, we simulated all four non-adiabatic effective models. These results are in agreement with the findings in Sec. 5.2 and 5.3. Note that in this case we did not look more closely for the missing transitions as in Sec. 5.3.

The results presented in this section suggest that modelling flux-tunable transmons as adiabatic anharmonic oscillators can suppress potential excitations of the coupling resonator. Obviously, this is only the case if we consider scenarios where an external flux drive is present.

5.6 Simulations of unsuppressed transitions in the adiabatic effective two-qubit model: architecture II

In Sec. 5.5 we studied transitions which are suppressed in the adiabatic effective model for architecture II, see Fig. 5.1(b). In this section, we consider transitions which are unsuppressed in the adiabatic effective model for architecture II. Note that we introduced the notation we used to address the states of the system at the beginning of Sec. 5.5.

In the following, we investigate two different transitions, namely the $\mathbf{z} = (0, 0, 1) \rightarrow \mathbf{z} = (0, 1, 0)$ and $\mathbf{z} = (0, 1, 1) \rightarrow \mathbf{z} = (0, 0, 2)$ transitions. The former (latter) transitions can potentially be used to implement ISWAP (CZ) gates. As in Sec. 5.4, we first discuss the results of the circuit Hamiltonian model and then study the results of the effective models. We use the Hamiltonian Eq. (3.114), the device parameters listed in Table 5.2 and the

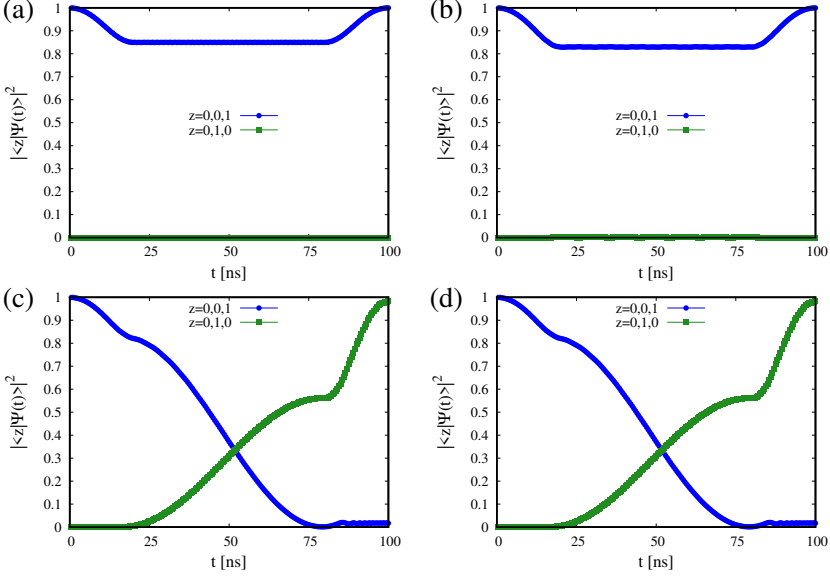


Figure 5.15: Probabilities $p^{\mathbf{z}}(t) = |\langle \mathbf{z} | \Psi(t) \rangle|^2$ as functions of time t for $\mathbf{z} = (0, 0, 1)$ and $\mathbf{z} = (0, 1, 0)$. We use the circuit Hamiltonian Eq. (3.114), the device parameters listed in Table 5.2 and the pulse given by Eq. (5.1) to obtain the results. The pulse parameters are the drive frequency $\omega^{(D)}/2\pi = 0$ GHz, the amplitude $\delta/2\pi = 0.289$, the rise and fall time $T_{r/f} = 20.0$ ns and the pulse duration $T_d = 100.0$ ns. Furthermore, n_J denotes the number of flux-tunable transmon basis states and we use $n_J = 3$ in panel(a), $n_J = 4$ in panel(b), $n_J = 14$ in panel(c) and $n_J = 25$ in panel(d). The resonator is modelled with $n_K = 4$ harmonic basis states. All pulse parameters are listed in Table 5.5.

pulse $\varphi(t)$ given by Eq. (5.1) to model the dynamics of the two-qubit system with the circuit model.

Figures 5.15(a-d) show the probabilities $p^{\mathbf{z}}(t)$ for $\mathbf{z} = (0, 0, 1)$ and $\mathbf{z} = (0, 1, 0)$ as functions of time t . Here $p^{\mathbf{z}}(t) = |\langle \mathbf{z} | \Psi(t) \rangle|^2$ and the initial state of the system is $|\Psi(0)\rangle = |0, 0, 1\rangle$. We use the drive frequency $\omega^{(D)}/2\pi = 0$ GHz, the amplitude $\delta/2\pi = 0.289$, the rise and fall time $T_{r/f} = 20.0$ ns and the pulse duration $T_d = 100.0$ ns to model the control pulse $\varphi(t)$. Furthermore, we simulate the system with different numbers of basis states n_J to model the flux-tunable transmons in the system. We use $n_J = 3$ in Fig. 5.15(a),

$n_J = 4$ in Fig. 5.15(b), $n_J = 14$ in Fig. 5.15(c) and $n_J = 25$ in Fig. 5.15(d). The resonator is modelled with $n_K = 4$ basis states only.

On the one hand, we observe that the time evolutions of the probabilities $p^{\mathbf{z}}(t)$ in Figs. 5.15(a-b) and Figs. 5.15(c-d) exhibit qualitative and quantitative deviations. On the other hand, we see that the time evolutions of the probabilities $p^{\mathbf{z}}(t)$ in Figs. 5.15(a) and Figs. 5.15(b) as well as Figs. 5.15(c) and Figs. 5.15(d) are similar. Additionally, in Figs. 5.15(a-b) we can observe that the states $\mathbf{z} = (0, 0, 1)$ and $\mathbf{z} = (0, 1, 0)$ do not exchange any population. Instead, we see that after the application of the pulse, the system mainly returns to its initial state. Only if we increase the number of basis states to $n_J = 14$, see Fig. 5.15(c), we find that the probabilities $p^{\mathbf{z}}(t)$ have converged to the third decimal. Here we use the solution for $n_J = 25$ as a graphical reference solution, i.e. there are no noticeable differences between the results in Figs. 5.15(c-d).

Figures 5.16(a-d) show the probabilities $p^{\mathbf{z}}(t) = |\langle \mathbf{z} | \Psi(t) \rangle|^2$ for $\mathbf{z} = (0, 1, 1)$ and $\mathbf{z} = (0, 0, 2)$ as functions of time t . The initial state of the system is $|\Psi(0)\rangle = |0, 1, 1\rangle$. Here we use the drive frequency $\omega^{(D)} = 0$ GHz, the amplitude $\delta/2\pi = 0.3335$, the rise and fall time $T_{r/f} = 20.0$ ns and the pulse duration $T_d = 125.0$ ns to model the pulse $\varphi(t)$. Furthermore, the pulse duration T_d is chosen such that the population exchange between the states $\mathbf{z} = (0, 1, 1)$ and $\mathbf{z} = (0, 0, 2)$ is not optimal, see Figs. 5.16(c-d) at time T_d . This is done for the following reason. The system has the tendency to return back to its initial state if we model the system with not enough basis states n_J , see Figs. 5.16(a-b). Consequently, it would be difficult to detect changes in the probabilities $p^{\mathbf{z}}(t)$ at time T_d if we calibrate the system such that $p^{\mathbf{z}}(T_d) \rightarrow 1$ for $\mathbf{z} = (0, 1, 1)$.

Figures 5.16(a-d) show similar features as Fig. 5.15(a-d). The results for $n_J = 3$ and $n_J = 4$, see Figs. 5.16(a-b), exhibit qualitative and quantitative differences and we have to increase the number of basis states to $n_J = 16$ until the probabilities $p^{\mathbf{z}}(T_d)$ converge to the desired decimal.

We now focus on the corresponding effective Hamiltonian models. For these simulations, we use the adiabatic effective Hamiltonian Eq. (3.117), the pulse $\varphi(t)$ given by Eq. (5.1) and four basis states for every transmon and resonator in the system. Here, as in Sec. 5.4, we have several options in specifying the effective Hamiltonian model. First, we can either use a time-dependent or a time-independent interaction strength to model the coupling between the flux-tunable transmons and the resonator. Second, we can choose between two sets of functions, see Sec. 5.2, to model the tunable energies of the transmons. The effective model we define, determines the device parameters we need to perform the simulations. If we model the system with a time-dependent interaction strength and/or the tunable energies with the

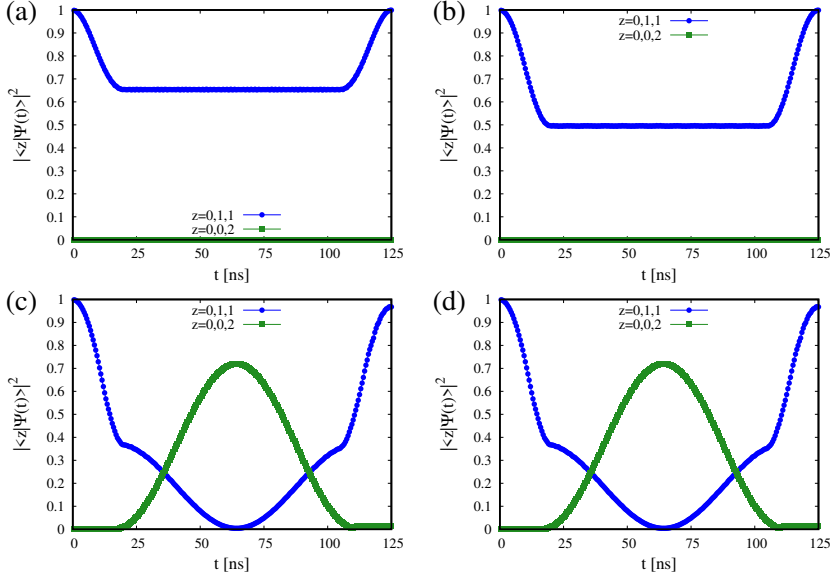


Figure 5.16: Probabilities $p^{\mathbf{z}}(t) = |\langle \mathbf{z} | \Psi(t) \rangle|^2$ as functions of time t for $\mathbf{z} = (0, 1, 1)$ and $\mathbf{z} = (0, 0, 2)$. We use the circuit Hamiltonian Eq. (3.114), the device parameters listed in Table 5.2 and the pulse given by Eq. (5.1) to obtain the results. The pulse parameters are the drive frequency $\omega^{(D)}/2\pi = 0$ GHz, the amplitude $\delta/2\pi = 0.3335$, the rise and fall time $T_{r/f} = 20.0$ ns and the pulse duration $T_d = 125.00$ ns. Furthermore, n_J denotes the number of flux-tunable transmon basis states and we use $n_J = 3$ in panel(a), $n_J = 4$ in panel(b), $n_J = 16$ in panel(c) and $n_J = 25$ in panel(d). The resonator is modelled with $n_K = 4$ harmonic basis states. All pulse parameters are listed in Table 5.5.

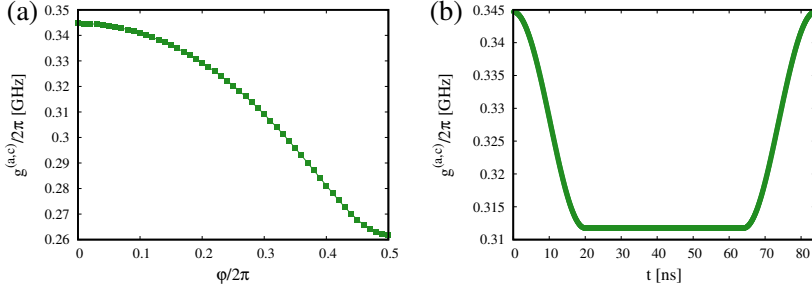


Figure 5.17: Effective interaction strength $g^{(a,c)}$ between a resonator and a flux-tunable transmon as a function of the external flux φ in panel(a) and as a function of time t in panel(b). We use Eq. (5.5) and the energies listed in Table 5.2 row $i = 1$ to obtain the numerical values for $g^{(a,c)}$. In panel(a) we compute the effective interaction strength $g^{(a,c)}(\varphi)$ for the interval $\varphi/2\pi \in [0, 0.5]$. In panel(b) we use the control pulse $\varphi(t)$ given by Eq. (5.1) with the flux-offset value $\varphi_0/2\pi = 0$, the drive frequency $\omega^{(D)}/2\pi = 0$ GHz, the pulse amplitude $\delta/2\pi = 0.297$, the rise and fall time $T_{r/f} = 20.0$ ns and the pulse duration $T_d = 84.0$ ns to obtain the effective interaction strength $g^{(a,c)}(\varphi(t))$ as a function of time t .

functions Eqs. (5.11) and (5.12), we have to use the device parameters listed in Table 5.2. The last remaining case can be modelled with the parameters listed in Table 5.4.

Figure 5.17(a) shows the effective interaction strength $g^{(a,c)}$ as a function of the flux φ . Figure 5.17(b) shows the effective interaction strength $g^{(a,c)}$ for a flux pulse $\varphi(t)$ given by Eq. (5.1) as a function of time t . We use the energies listed in Table 5.2, row $i = 1$, the drive frequency $\omega^{(D)} = 0$ GHz, the amplitude $\delta/2\pi = 0.297$, the rise and fall time $T_{r/f} = 20.0$ ns and the pulse duration $T_d = 84.0$ ns to generate the data for Figs. 5.17(a-b). Note that the pulse parameters are only necessary to generate the data for Fig. 5.17(b).

In Fig. 5.17(a) we can observe that the interaction strength $g^{(a,c)}$ varies by about 85 MHz over the interval $\varphi/2\pi \in [0, 0.5]$. If we use the pulse $\varphi(t)$ to compute the interaction strength $g^{(a,c)}$, specified above, we find that $g^{(a,c)}$ drops by about 35 MHz for a period of roughly 45 ns, afterwards $g^{(a,c)}$ returns back to its initial value. Note that in Sec. 5.6 we found that oscillations of about 1 MHz can shift the pulse duration T_d up to about 65 ns.

Figures 5.18(a-d) show the probabilities $p^{\mathbf{z}}(t) = |\langle \mathbf{z} | \Psi(t) \rangle|^2$ as functions of time t . In Figs. 5.18(a-b) we model $\mathbf{z} = (0, 0, 1) \rightarrow \mathbf{z} = (0, 1, 0)$ ISWAP transitions with, see Fig. 5.18(b) and without, see Fig. 5.18(a), a

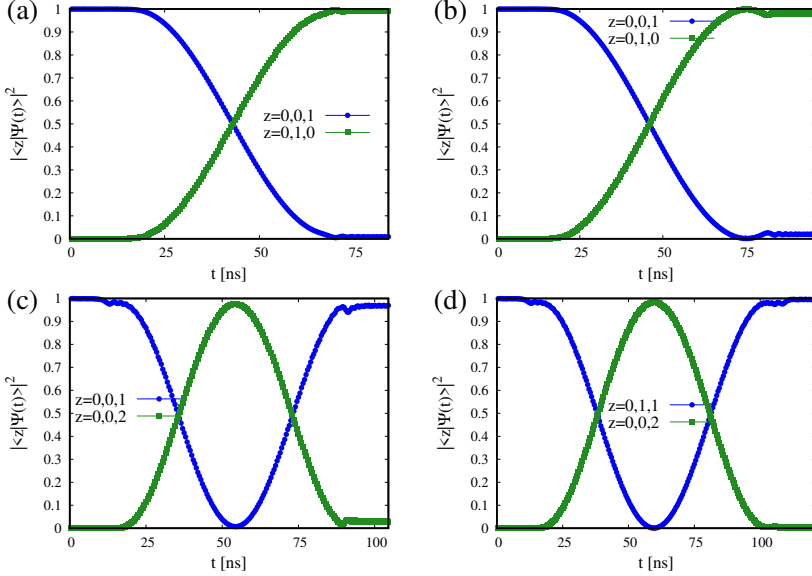


Figure 5.18: Probabilities $p^{\mathbf{z}}(t) = |\langle \mathbf{z} | \Psi(t) \rangle|^2$ as functions of time t . In panels(a,b) we model the transitions $\mathbf{z} = (0,0,1) \rightarrow \mathbf{z} = (0,1,0)$. In panels(c,d) we model the transitions $\mathbf{z} = (0,1,1) \rightarrow \mathbf{z} = (0,0,2)$. We use the effective Hamiltonian Eq. (3.117), the device parameters listed in Table 5.4 and the pulse given by Eq. (5.1) with the drive frequency $\omega^{(D)}/2\pi = 0$ GHz and the rise and fall time $T_d = 20$ ns to obtain the results. The remaining, non-specified pulse parameters are the amplitude δ and the pulse duration T_d . Both parameters are different for every panel. We use $\delta/2\pi = 0.297$ and $T_d = 84.0$ ns in panel(a), $\delta/2\pi = 0.289$ and $T_d = 96.0$ ns in panel(b), $\delta/2\pi = 0.343$ and $T_d = 105.0$ ns in panel(c) as well as $\delta/2\pi = 0.334$ and $T_d = 121.0$ ns in panel(d). In panels(a,c) we use the first-order series expansion to model the tunable qubit frequency. Here, we also use $g^{(a,c)}(t) = \text{const.}$ to model a static effective interaction strength. Similarly, in panels(b,d) we use the higher-order series expansion to model the energies of the flux-tunable transmons and $g^{(a,c)}(t)$ given by Eq. (5.5) to model a dynamic effective interaction strength, see Fig. 5.17(b). All pulse parameters and cases are listed in Table 5.6.

time-dependent interaction strength. Similarly, in Figs. 5.18(c-d) we model $\mathbf{z} = (0, 1, 1) \rightarrow \mathbf{z} = (0, 0, 2)$ CZ transitions with, see Fig. 5.18(d) and without, see Fig. 5.18(c), a time-dependent interaction strength.

We determined the pulse parameters as follows. First, we fix the rise and fall time $T_{r/f}$. Here, we use the rise and fall time $T_{r/f} = 20$ ns we found for the corresponding circuit Hamiltonian transitions. Next, we perform a spectroscopy simulation for the pulse amplitude δ . Here we use the amplitude in the center of the corresponding chevron pattern. At last, we adjust the pulse duration such that the corresponding operation can be realised with the two previously specified pulse parameters. Furthermore, we use the drive frequency $\omega^{(D)} = 0$ GHz for all pulses.

The results in Fig. 5.18(a) are obtained with the pulse amplitude $\delta/2\pi = 0.297$ GHz and the pulse duration $T_d = 84.0$ ns. Here we use the tunable frequency given by Eq. (5.10) and a constant anharmonicity to model the spectrum of the flux-tunable transmon. If we compare these results with the one of the corresponding circuit Hamiltonian simulations, see Table 5.5, we find that the pulse amplitudes deviate by $\Delta\delta/2\pi = 0.008$. Furthermore, if we compare the pulse durations T_d for both cases, we find a difference of 16 ns.

The results in Fig. 5.18(b) are obtained with the pulse amplitude $\delta/2\pi = 0.289$ GHz and the pulse duration $T_d = 96.0$ ns. Here we use the tunable frequency given by Eq. (5.11) and the tunable anharmonicity in Eq. (5.12) to model the spectrum of the flux-tunable transmon. Consequently, if we use the adjusted spectrum to model the flux-tunable transmon, the deviations between the effective and the circuit model practically disappear. Furthermore, the difference in terms of the pulse duration has reduced to 4 ns. Interestingly, even though the nominal changes in interaction strength $g^{(a,c)}$ in Fig. 5.18(b) are substantially larger than for the case $g^{(c,c)}$ in Fig. 5.10(b) in Sec. 5.4, we find much smaller deviations between the two effective models, i.e. the one with and the one without a time-dependent interaction strength.

The results in Fig. 5.18(c-d) are obtained with the same simulation settings as the results in Fig. 5.18(a-b). Here we find similar features as before. The large differences in terms of the pulse amplitude disappear if we adjust the spectrum of the flux-tunable transmon in the effective model. Furthermore, the deviations in terms of the pulse duration T_d are reduced to 4 ns if we model the effective model with a time-dependent interaction strength. It seems peculiar that we often find a difference of 4 ns for the pulse duration T_d if we compare the effective and the circuit model. However, to the best knowledge of the author, this finding is just a coincidence.

In Sec. 5.4, we investigate whether or not the non-adiabatic driving term in Eq. (5.3) affects the ISWAP and CZ transitions if we simulate architecture

I with the effective model. Here we find that the transitions are barely affected. Similarly, we added the driving term in Eq. (5.3) to the adiabatic effective model for the flux-tunable transmons in architecture II and repeated the simulations again. We find (data not shown) that the ISWAP and CZ transitions for the relevant pulse parameters listed in Table 5.6 are barely affected. This means that the probabilities p^z at time T_d are at most affected by the third decimal. Furthermore, the overall transitioning behaviour does not seem to be affected at all. As before, we emphasise that different pulse parameters can potentially cause larger deviations.

5.7 Summary, conclusions and outlook

In this chapter we investigated how different approximations, which result in different effective Hamiltonians, affect the transitions from one state to another. To this end, we studied the probability amplitudes which are associated with the different states. Furthermore, we compared the results of the effective Hamiltonian models to the associated circuit Hamiltonian models. Note that here we only considered transitions which can be activated by means of an external flux.

In Sec. 5.2 we investigated a single flux-tunable transmon. Here we compared how the non-adiabatic effective Hamiltonian Eq. (3.109), the adiabatic effective Hamiltonian Eq. (3.110) and the circuit Hamiltonian Eq. (3.61) react to various microwave pulses, see Fig. 5.2(a) and unimodal pulses, see Fig. 5.2(b).

Obviously, the adiabatic effective Hamiltonian Eq. (3.110) cannot, by definition, be used to model any transitions. Note that the adiabatic model for the flux-tunable transmon is often used to describe flux-tunable transmons in multi-qubit systems. Conversely, we found that the non-adiabatic effective Hamiltonian Eq. (3.109) and the circuit Hamiltonian Eq. (3.61) can be used to model various resonant and non-adiabatic transitions. Additionally, for the case of resonant transitions, driven by a microwave pulse, we showed an example where the predictions for both models match qualitatively and quantitatively with minor deviations in one of the pulse parameters. For the case of non-adiabatic transitions, driven by a unimodal pulse, we showed an example where the predictions made by both models match qualitatively for most instances. Furthermore, we showed that the spectrum of the flux-tunable transmon in question, modelled with the tunable frequencies given by Eqs. (5.9) and (5.10) can deviate up to several GHz from the numerically exact spectrum. In conclusion, the non-adiabatic effective Hamiltonian Eq. (3.110) can cover at least some of the aspects of the associated circuit

Hamiltonian Eq. (3.61).

In Sec. 5.3 and 5.4 we considered a two-qubit NIGQC model, see Sec. 2.7, which we denote as architecture I, see Fig. 5.1(a). This device consists of two fixed-frequency transmons whose ground and first-excited states are supposed to function as qubit states and a flux-tunable transmon which works as a time-dependent coupler. The transitions in this device architecture are usually activated by means of microwave pulses, see Fig. 5.2(a).

We showed evidence that the adiabatic effective Hamiltonian Eq. (3.117) is not able to describe certain single-qubit (X) type transitions with a flux drive, see for example Table 5.6 rows three and four. However, we also found that the non-adiabatic effective Hamiltonian Eq. (3.117) can be used to model exactly these transitions. Note that the definitions of the adiabatic and non-adiabatic effective Hamiltonians only differ by the non-adiabatic driving term in Eq. (5.3). These results are discussed in Sec. 5.3. Consequently, when studying the single-qubit (X) type transitions with an effective model for architecture I, one should consider using the driving term in Eq. (5.3) to do so.

Furthermore, for architecture I we found that modelling the interaction strength of the adiabatic effective Hamiltonian Eq. (3.117) with the time-dependent function given by Eq. (5.4) leads to large shifts in the pulse duration of up to 75 ns for ISWAP and CZ gate transitions, see Figs. 5.11(a-d), Figs. 5.13(a-b) and Table 5.6. Note that the time dependence in Eq. (5.4) induces oscillations of about 1 MHz in the interaction strength, see Fig. 5.10(b). Therefore, a time-dependent interaction strength of the form Eq. (5.4) can have a non-negligible impact on the pulse duration for ISWAP and CZ transitions if we model these transitions with the effective model.

In Sec. 5.4, we compared the results of the effective models with and without time-dependent interaction strength to the results of the associated circuit model. If we model architecture I with a time-independent interaction strength and the adiabatic effective Hamiltonian Eq. (3.117), we find deviations of up to 100 ns for the pulse duration with regard to the associated circuit Hamiltonian Eq. (3.114). If we model architecture I with a time-dependent interaction strength and the adiabatic effective Hamiltonian Eq. (3.117), we find deviations of up to 25 ns for the pulse duration with regard to the associated circuit Hamiltonian Eq. (3.114).

Additionally, we found that the non-adiabatic driving term in Eq. (5.3) barely affects the ISWAP and CZ transitions for the relevant pulse parameters in Table 5.6. Note that different pulse parameters can potentially lead to more severe deviations.

In Sec. 5.5 and 5.6 we studied a two-qubit NIGQC model which we denote as architecture II, see Fig. 5.1(b). This device consists of two flux-tunable

transmons whose ground and first-excited states are supposed to function as qubit states and a resonator element which works as a time-independent coupler. The transitions in this device architecture are usually activated by means of a unimodal pulses, see Fig. 5.2(b).

We found that we can excite the coupling resonator with a microwave control pulse if we use the circuit Hamiltonian Eq. (3.114) and the non-adiabatic effective Hamiltonian Eq. (3.117). However, as in Sec. 5.3 we found that the adiabatic effective Hamiltonian Eq. (3.117) does not allow us to model the targeted transitions. These results are discussed in Sec. 5.5. Consequently, when studying excitations of the coupler states with an effective model for architecture II, one should consider using the driving term in Eq. (5.3) to do so.

We also found that modelling the interaction strength of the adiabatic effective Hamiltonian Eq. (3.117) with the time-dependent function given by Eq. (5.5) leads to modest shifts in the pulse durations of about 15 ns for ISWAP and CZ gate transitions, see Figs. 5.18(a-d) and Table 5.6. Note that the time dependence in Eq. (5.5) induces a square pulse like reduction of roughly 35 MHz of the interaction strength, see Fig. 5.17(b).

For architecture II, we also compared the results of the effective models with and without time-dependent interaction strength to the results of the associated circuit model. If we use the time-dependent model given by Eq. (5.4) for the interaction strength, we find deviations of up to 4 ns for the pulse duration with regard to the circuit model. If we use the time-independent model for the interaction strength, we find deviations of up to 20 ns for the pulse duration with regard to the circuit model.

Additionally, we found that if we use the adiabatic effective Hamiltonian Eq. (3.117) to model ISWAP and CZ type transitions, we have to use the tunable frequency in Eq. (5.11) and the tunable anharmonicity in Eq. (5.12) to model the spectra of flux-tunable transmons, assuming we want the pulse amplitudes for the effective and circuit model to match. The findings with regard to the spectra in Sec. 5.2 make these results seem plausible.

For architecture II, we also found that the non-adiabatic driving term in Eq. (5.3) barely affects the ISWAP and CZ transitions for the relevant pulse parameters in Table 5.6. Here too, note that different pulse parameters can potentially lead to more severe deviations. These results are discussed in Sec. 5.6.

In conclusion, as long as we cannot prove that two time-evolution operators, see Eq. (4.2), for two different models are the same or close for the moments in time which are of interest to the physical scenario we investigate, we should not assume that the two different models make similar predictions for the same physical scenario.

So far, we only considered how different approximations, i.e. simplifications, affect the time evolution of selected probabilities or probability amplitudes. In Chap. 6, we investigate how different approximations change the gate-error signatures for complete quantum circuits, see Ref. [Wil+17].

On the fragility of gate-error metrics in simulations of flux-tunable transmon quantum computers

Building a gate-based quantum computer in the real world requires a tremendous amount of precision in terms of control over a quantum system. Therefore, the need to somehow quantify deviations between the states of the ideal gate-based quantum computer (IGQC), see Secs. 2.1-2.5 and prototype gate-based quantum computers (PGQCs), see Sec. 2.6, arises. Additionally, many theoretical studies, see for example Refs. [Gu+21; Rot+17; Yan+18; McK+16; Wit+21], focus on deviations between the states of the IGQC and non-ideal gate-based quantum computers (NIGQCs), see Sec. 2.7.

In this chapter we study how susceptible gate-error quantifiers, that aim to quantify these deviations, are to the assumptions (approximations) which make up the underlying model. Here, we consider NIGQC models which describe certain aspects of the time evolution of PGQCs. A review of the literature suggests that PGQCs are susceptible to many time-dependent factors, see Sec. 2.6. Consequently, it seems to be worth knowing to what extent the gate-error quantifiers are affected by changes in the model once we describe the time evolution of a system as a real-time process. In Chap. 5 we discussed two device architectures, see Figs. 5.1(a-b). A detailed discussion of only one device architecture already takes up a large part of this thesis. Therefore, in this chapter we focus on one device architecture only. The author of this thesis decided to present the results for architecture II, which seemed to be more understandable.

This chapter is structured as follows. In Secs. 6.1-6.3 we specify the model. First, in Sec. 6.1, we introduce the device architecture and the device parameters. Next, in Sec. 6.2, we introduce the control pulses we use to realise the elementary gate sets. Then, in Sec. 6.3, we introduce the gate-error quantifiers we use to assess to what extent the state of a NIGQC and the state of the IGQC deviate from one another for a given sequence of

gates. The main results of this chapter are discussed in Secs. 6.4-6.8. First, in Sec. 6.4, we study the spectrum of a four-qubit NIGQC and its relevance for the implementation of two-qubit gates. Next, in Sec. 6.5, we discuss the gate-error metrics for the elementary gate sets we use to implement all gate sequences with our NIGQCs. Then, in Sec. 6.6, we investigate how modelling the dynamics of a NIGQC with more and less basis states affects the gate-error quantifiers we compute. Afterwards, in Sec. 6.7, we study how changes in the control pulse parameters affect the gate-error quantifiers we compute. Next, in Sec. 6.8, we look at how modelling flux-tunable transmons as adiabatic and non-adiabatic anharmonic oscillators affects the gate-error quantifiers. Finally, in Sec. 6.9, we summarise and discuss the findings. Here, we also present a series of conclusions drawn from our findings. Note that we use $\hbar = 1$ throughout this chapter.

6.1 System specification and simulation parameters

In this section we introduce the device architecture and the model parameters we use to obtain the results in this chapter. The device parameters, control pulses and some technical details relevant to the implementation of different gates were provided by the Quantum Device Lab which is affiliated with the Eidgenössische Technische Hochschule Zurich. Note that Refs. [Lac+20; And+20; Kri+20] discuss a series of experiments with PGQCs characterised by similar device parameters.

Figures 6.1(a-c) display illustrations of the three different NIGQCs we model in this chapter. The $\omega_i^{(R)}$ parameters denote resonator frequencies. Similarly, the $\omega_i^{(Q)}$ parameters denote transmon qubit park frequencies. These are the frequencies at which the transmon qubits reside if no external fluxes are applied to the system. Furthermore, the $G_{i,j}^{(4)}$ parameters denote the interaction strength constants, see Eq. (3.115)(e). We assume that interactions are of the dipole-dipole type, see Sec. 3.5 and 3.6. In this chapter we model two-qubit ($N=2$), three-qubit ($N=3$) and four-qubit ($N=4$) NIGQCs, see Figs. 6.1(a-c), respectively. Here $N \in \mathbb{N}$ denotes the number of qubits.

The device parameters for the Hamiltonians in Eqs. (3.114) and (3.117) are listed in Table 6.1. We were provided with the measured qubit frequencies, anharmonicities, flux offset values and asymmetry factors, see Sec. 2.6, Sec. 3.4 and Ref. [Nag19]. Furthermore, the resonator frequencies are given. We use the relations

$$\omega_i^{(Q)} = (E_i^{(1)} - E_i^{(0)}), \quad (6.1a)$$

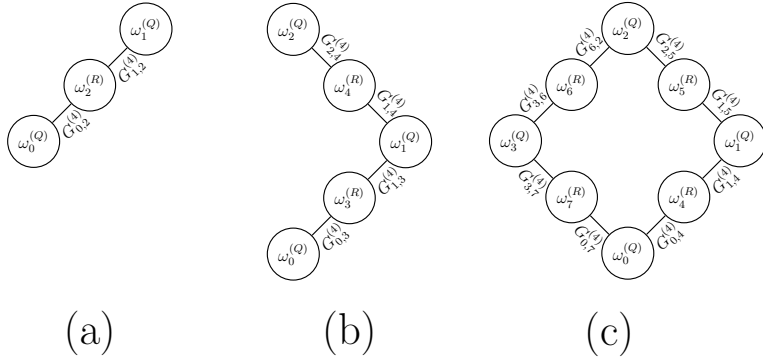


Figure 6.1: Illustrations of a two-qubit(a), three-qubit(b) and four-qubit(c) NIGQC. The different NIGQCs consist of LC resonators, or resonators for short, with resonance frequencies $\omega_i^{(R)}$ and flux-tunable transmon qubits, or transmon qubits for short, with park frequencies $\omega_i^{(Q)}$, see Sec. 3.3 and 3.4, respectively. Here, i denotes the discrete index we use to address the different subsystems. The parameters $G_{i,j}^{(4)}$ denote interaction strength constants, see Eq. (3.115)(e). The interactions between the resonators and transmon qubits are modelled as dipole-dipole interactions. We use the model Hamiltonians Eqs. (3.114) and (3.117) to describe the dynamics of the different systems. The devices parameters are listed in Table 6.1.

Table 6.1: Device parameters for the two-qubit, three-qubit and four-qubit NIGQCs illustrated in Figs. 6.1(a-c). The first column contains the discrete indices i we use to index the different subsystems. The second column contains the resonator frequencies $\omega_i^{(R)}$. The third and fourth column contain the transmon qubit frequencies $\omega_i^{(Q)}$ and anharmonicities $\alpha_i^{(Q)}$. The fifth column contains the capacitive energies E_{C_i} for the transmon qubits. The sixth and seventh column contain the left $E_{J_{i,l}}$ and right $E_{J_{i,r}}$ Josephson energies. The eighth column contains the flux offset values $\varphi_{0,i} = \varphi_j(t)$ at time $t = 0$. These values define operating points for the different transmon qubits. All units, except for the flux offset, are in GHz. We do not list the asymmetry factors $d_i = 0.5$ and the interaction strength constants $G_{i,j}^{(4)} = 0.300$ because they are constant for all $i, j \in \mathbb{N}^0$. Note that we use $\hbar = 1$ throughout this chapter.

i	$\omega_i^{(R)}/2\pi$	$\omega_i^{(Q)}/2\pi$	$\alpha_i^{(Q)}/2\pi$	$E_{C_i}/2\pi$	$E_{J_{i,l}}/2\pi$	$E_{J_{i,r}}/2\pi$	$\varphi_{0,i}/2\pi$
0	n/a	4.200	-0.320	1.068	2.355	7.064	0
1	n/a	5.200	-0.295	1.037	3.612	10.837	0
2	n/a	5.700	-0.285	1.017	4.374	13.122	0
3	n/a	4.960	-0.300	1.045	3.281	9.843	0
4 – 7	45.000	n/a	n/a	n/a	n/a	n/a	n/a

$$\alpha_i^{(Q)} = (E_i^{(2)} - E_i^{(0)}) - 2\omega_i^{(Q)}, \quad (6.1b)$$

$$d_i = \frac{(E_{J_{r,i}} - E_{J_{l,i}})}{(E_{J_{r,i}} + E_{J_{l,i}})}, \quad (6.1c)$$

to fit the given experimental parameters to the lowest three energy levels of the circuit Hamiltonian Eq. (3.60). This procedure yields the capacitive and Josephson energies E_{C_i} , $E_{J_{r,i}}$ and $E_{J_{l,i}}$ as a result. The model parameters used deviate at most tens of MHz from the ones which were provided. Note that we use the constant asymmetry factor $d_i = 0.5$ for all transmon qubits as well as the constant frequency $\omega_i^{(R)} = 45.0$ GHz for all coupling resonators. Furthermore, we use a fixed value of 300 MHz for the interaction strength constants $G_{i,j}^{(4)}$. This value for the interaction strength constants roughly reproduces the gate durations for the two-qubit CZ gates which were observed in experiments.

For almost all simulations of the circuit Hamiltonian Eq. (3.114), we use sixteen bare basis states for all transmon qubits which experience a flux drive. Furthermore, the transmon qubits which do not experience a flux drive and the coupling resonators are modelled with four basis states only.

For the simulations of the effective Hamiltonian Eq. (3.117), we use four bare basis states for all transmon qubits and coupling resonators. Also, we use a time-dependent interaction strength to model the dipole-dipole interactions. Additionally, we use the higher-order series expansions Eqs. (5.11) and (5.12) to model the flux-tunable frequencies and anharmonicities, see Chap. 5 for more details.

6.2 Control pulses and gate implementation

In this section we introduce the control pulses we use to implement different gates with our NIGQCs.

Figures 6.2(a-c) show the external charge $n(t)$ (a) and external flux $\varphi(t)$ (b-c) as functions of time t . The pulse in Fig. 6.2(a) is a microwave pulse (MP). The pulse in Fig. 6.2(c) is usually referred to as a unimodal pulse (UMP). Similarly, we denote the pulse in Fig. 6.2(c) as a bimodal pulse (BMP). Note that this pulse is also sometimes referred to as net-zero flux pulse, see Ref. [Rol+19].

We use the MP

$$n(t) = aG(t, \sigma, T_d) \cos(\omega^{(D)}t - \phi) + b\dot{G}(t, \sigma, T_d) \sin(\omega^{(D)}t - \phi), \quad (6.2)$$

to implement single-qubit $R^{(x)}(\pi/2)$ rotations, see Eq. (2.19)(a), by making use of the charge-dependent linear driving terms in Eq. (3.115)(b) and

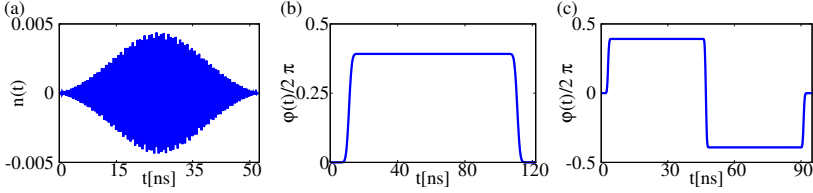


Figure 6.2: External charge $n(t)$ (a) and external flux $\varphi(t)$ (b-c) as functions of time t . In panel(a) we show a microwave pulse (MP). In panels(b-c) we display a unimodal pulse (UMP) and a bimodal pulse (BMP), respectively.

Eq. (3.120). Here a denotes the pulse amplitude, $\omega^{(D)}$ is the drive frequency and b is the amplitude of the DRAG component of the pulse, see Ref. [Mot+09]. Furthermore, the real-valued function $G(t, \sigma, T_d)$ denotes a Gaussian envelope that is centered around half of the pulse duration T_d . The parameter σ determines the shape of the envelope.

The variable ϕ denotes the phase of the MP. We use this phase to implement so-called virtual $R^{(z)}(\phi)$ rotations, see Eq. (2.19)(c) or virtual Z gates for short, see Ref. [McK+17]. The simulation code is used to compute the phases ϕ for the MPs in Eq. (6.2) has to be specifically designed for the two-qubit gates we implement on our NIGQCs. This algorithm preprocesses every program, i.e. every sequence of gates and computes the correct pulse phases ϕ for the different MPs in the pulse sequence. Since we only implement two-qubit CZ gates on our NIGQCs, we find that the preprocessing is rather simple. Note that the CZ gate commutes with the single-qubit z-axis rotation. We omit a detailed discussion of the preprocessing algorithm which is part of the compiler module in Fig. 4.1 to shorten this chapter.

We use the UMP

$$\varphi(t) = \frac{\delta}{2} \left(\operatorname{erf} \left(\frac{t}{\sqrt{2}\sigma} \right) - \operatorname{erf} \left(\frac{(t - T_p)}{\sqrt{2}\sigma} \right) \right), \quad (6.3)$$

and BMP

$$\varphi(t) = \frac{\delta}{2} \left(\operatorname{erf} \left(\frac{t}{\sqrt{2}\sigma} \right) - 2\operatorname{erf} \left(\frac{(t - T_p/2)}{\sqrt{2}\sigma} \right) + \operatorname{erf} \left(\frac{(t - T_p)}{\sqrt{2}\sigma} \right) \right). \quad (6.4)$$

to implement two-qubit CZ gates by making use of the flux-dependent driving terms in Eq. (3.115)(b) as well as Eqs. (3.118) and (3.121). Here, the parameter δ denotes the pulse amplitude, σ refers to a parameter that allows us to control how fast the pulse flanks rise (fall) and T_p denotes the pulse time. We use another parameter T_d , the pulse duration, to add some buffer

time to the control pulses. The pulse duration T_d is not part of Eqs. (6.3) and (6.4) but only part of the simulation software. Note that the Hamiltonian Eq. (3.61) is 2π periodic. Therefore, in this chapter we give the external flux $\varphi(t)$ and the pulse amplitude δ always in units of 2π .

Every flux pulse is followed by single-qubit z-axis rotations $R_i^{(z)}(\phi_i)$ applied to every transmon qubit in the system, see Ref. [Bla+21, Section VII B 2]. In the end, all z-axis rotations are executed as virtual Z gates by means of the MPs as discussed above. Finally, at the end of the program, i.e. the pulse sequence, we only once transform the frame of reference. This transformation can be expressed as $c_{\mathbf{z}} \mapsto e^{i\phi_{\mathbf{z}}} c_{\mathbf{z}}$ for $\mathbf{z} \in \{0, 1\}^N$, where the different phase factors $\phi_{\mathbf{z}}$ are computed by the preprocessing algorithm, see also Ref. [Wil20a, Section 3.3.2].

6.3 Computation of gate-error quantifiers

In this section we introduce the gate-error quantifiers used to assess the deviations between the states of the IGQC and various NIGQC models for given sequences of gates.

In Sec. 2.1 we described the computations performed with the IGQC as unitary mappings of the form

$$|\psi\rangle \mapsto \hat{U} |\psi\rangle, \quad (6.5)$$

where $\hat{U} \in \mathbb{U}(\mathcal{H}^{2^N})$ is a unitary operator defined on the 2^N -dimensional Hilbert space \mathcal{H}^{2^N} . We can use a simple gate-error quantifier like the statistical distance

$$\mu_{\text{SD}}(\mathbf{p}, \tilde{\mathbf{p}}) = \frac{1}{2} \sum_{\mathbf{z} \in \{0,1\}^N} \|p_{\mathbf{z}} - \tilde{p}_{\mathbf{z}}\|_1, \quad (6.6)$$

where $\mathbf{p} = (p_{\mathbf{z}})_{\mathbf{z} \in \{0,1\}^N}$ and $\tilde{\mathbf{p}} = (\tilde{p}_{\mathbf{z}})_{\mathbf{z} \in \{0,1\}^N}$ denote discrete probability distributions, to quantify how well a computation \hat{U} was realised with a PGQC or NIGQC. Here, $\|\cdot\|_1$ denotes the absolute value, \mathbf{p} is defined in terms of the probability amplitudes $p_{\mathbf{z}} = \|c_{\mathbf{z}}\|_1^2$ of the IGQC for a given initial state $|\psi\rangle$ and $\tilde{\mathbf{p}}$ denotes the reference distribution we compare against. We can define $\tilde{\mathbf{p}}$ to be the relative frequencies obtained in a PGQC experiment or the probability amplitudes obtained from the truncated state vector of an NIGQC. The statistical distance is also known as the total variation distance, see Refs. [SWS15; San16; NC11]. Note that if we use the statistical distances to quantify gate errors, we completely ignore the phase of the system. Furthermore, the statistical distance only takes into account a single

initial state. Hence, often theoretically more sophisticated gate-error quantifiers are considered. However, in order to provide a proper definition of these quantifiers, we have to extend the mathematical framework we use.

So far we have considered computations as mappings between two state vectors. However, there exists an alternative formalism to describe the state of a quantum system, the density operator formalism. Note that this formalism is used widely in physics, e.g. in statistical physics see Ref. [Bal91; Bal07]. The density operator formalism also relies on four postulates and/or assumptions which set the stage for the complete mathematical framework.

If we consider an N -qubit IGQC, the first postulate states that the state of a quantum system is completely described by a density operator $\hat{\rho} \in \mathbb{P}(\mathcal{H}^{2^N})$. The set of density operators $\mathbb{P}(\mathcal{H}^{2^N}) \subseteq \mathbb{L}(\mathcal{H}^{2^N})$ contains operators $\hat{\rho}$ which satisfy the three conditions: $\text{Tr}(\hat{\rho}) = 1$, $\hat{\rho} = \hat{\rho}^\dagger$ and for all $|\psi\rangle \in \mathcal{H}^{2^N}$ the inequality $\langle\psi|\hat{\rho}|\psi\rangle \geq 0$ holds true. Operators which satisfy the last condition are also known as positive (semi-definite) operators and one often writes $\hat{\rho} \geq 0$ as a shorthand for $\forall |\psi\rangle \in \mathcal{H}^{2^N}: \langle\psi|\hat{\rho}|\psi\rangle \geq 0$. The second postulate states that the evolution of a closed system is described by a unitary transformation of the form

$$\hat{\rho} \mapsto \hat{U}\hat{\rho}\hat{U}^\dagger. \quad (6.7)$$

As for the state vector formalism, the third postulate provides us with relations which allow us to determine the event probabilities

$$p(i) = \text{Tr} \left(\hat{E}_i^\dagger \hat{E}_i \hat{\rho} \right), \quad (6.8)$$

where the set of operators $\{\hat{E}_i\} \subseteq \mathbb{L}(\mathcal{H}^{2^N})$ allows us to distinguish between different discrete events $i \in I \subseteq \mathbb{N}^0$, i.e. measurement outcomes. This set has to satisfy the relation

$$\sum_{i \in I} \hat{E}_i^\dagger \hat{E}_i = \hat{I}. \quad (6.9)$$

Furthermore, the density operator

$$\hat{\rho}' = \frac{\hat{E}_i \hat{\rho} \hat{E}_i^\dagger}{p(i)}, \quad (6.10)$$

describes the state after the measurement if we find that event i has taken place. The fourth postulate states that composed systems are described by a tensor product structure. Here we assume that if the individual systems $n \in \mathbb{N}$ are prepared in the states $\hat{\rho}_n$, then the state of the composite system is given by

$$\hat{\rho} = \bigotimes_{n=0}^{N-1} \hat{\rho}_n. \quad (6.11)$$

The density operator formalism is a convenient theoretical tool which allows us to describe certain physical scenarios with much more ease, compared to the state vector formalism. For example, we might describe situations where the initial state of the system $|\psi\rangle \in \mathcal{H}^{2^N}$ is not known definitively but only with a certain probability. Also, density operators often simplify the analysis of individual subsystems which are part of a larger system, in the sense of postulate four. We do not make much use of the formalism in this thesis. Therefore, we cut the discussion of this subject short. A much more detailed introduction can be found in Refs. [NC11; Wat18]. In fact, the only reason why we have to consider density operators and the corresponding mathematical framework is that some gate-error quantifiers are defined in terms of so-called quantum operations.

In Sec. 2.1 we introduce operators, i.e. linear functions which map one vector to another vector. Since we find that the set $\mathbb{L}(\mathcal{H}^{2^N})$ together with the addition of operators, scalar multiplication and the field of complex numbers \mathbb{C} forms a vector space of its own, we can simply define linear functions which map operators to other operators and retain some of the old terminology. We denote these functions as superoperators $\tilde{\mathcal{E}}$. Superoperators, like every other function, can possess features like symmetries. Additionally, we can restrict the domain and range of superoperators such that we map only certain types of operators to others. In this thesis we only consider superoperators which map density operators to other density operators. The superoperators we are interested in are usually denoted as quantum operations and we impose the following constraints on them. First, we require them to be Hermiticity preserving, which means $\tilde{\mathcal{E}}(\hat{\rho})^\dagger = \tilde{\mathcal{E}}(\hat{\rho}^\dagger)$ for all $\hat{\rho} \in \mathbb{P}(\mathcal{H}^{2^N})$. Second, we would like to work with so-called completely positive superoperators. A completely positive superoperator $\tilde{\mathcal{E}}$ maps all positive $\hat{\rho} \in \mathcal{P}(\mathcal{H}^D \otimes \mathcal{H}^{2^N})$, where $D \in \mathbb{N}$, to positive operators by means of the map $\tilde{I} \otimes \tilde{\mathcal{E}}$, where \tilde{I} is the identity operator associated with the Hilbert space \mathcal{H}^D . The statements made in the last sentence are not as accurate as required. However, since we do not make much use of quantum operations in this thesis, we abstain from extending the mathematical framework any further. A much more extensive discussion of the subject is provided by Refs. [Wat18; NC11]. Additionally, if we find that a quantum operation $\tilde{\mathcal{E}}$ is also trace preserving $\text{Tr}(\tilde{\mathcal{E}}(\hat{\rho})) = \text{Tr}(\hat{\rho})$, we denote this quantum operation as quantum or error channel.

With the mathematical framework discussed, we can define the quantum operation

$$\tilde{\mathcal{E}}_U(\hat{\rho}) = \hat{U}\hat{\rho}\hat{U}^\dagger, \quad (6.12)$$

which describes the ideal target operation. Furthermore, we can define the

quantum operation

$$\check{\mathcal{E}}_M(\hat{\rho}) = \hat{M}\hat{\rho}\hat{M}^\dagger, \quad (6.13)$$

which describes the actual operation performed with the NIGQCs we model in this chapter. Here, the operator \hat{M} is defined as

$$\hat{M} = \hat{P}\hat{\mathcal{U}}(t, t_0)\hat{P}, \quad (6.14)$$

where $\hat{\mathcal{U}}(t, t_0)$ denotes the time-evolution operator

$$\hat{\mathcal{U}}(t, t_0) = \mathcal{T} \exp \left(-i \int_{t_0}^t \hat{H}(t') dt' \right), \quad (6.15)$$

at time t and \hat{P} denotes a projection onto the computational subspace. Note that the definitions in Eqs. (6.12) and (6.13) are taken from the author of Ref. [Wil20a, Section 6.1].

We can use the quantum operations in Eqs. (6.12) and (6.13) to define the average fidelity as

$$\mu_{\text{Favg}} = \int \langle \psi | \check{\mathcal{E}}_M \check{\mathcal{E}}_U^{-1} (|\psi\rangle\langle\psi|) |\psi\rangle d|\psi\rangle, \quad (6.16)$$

where the integral is taken over all normalised states $|\psi\rangle \in \mathcal{H}^{2^N} : \langle\psi|\psi\rangle = 1$. If we define the auxiliary operator

$$\hat{V} = \hat{U}\hat{M}^\dagger, \quad (6.17)$$

we can express the average fidelity as

$$\mu_{\text{Favg}} = \frac{\left\| \text{Tr}(\hat{V}) \right\|_1^2 + \text{Tr}(\hat{M}\hat{M}^\dagger)}{D(D+1)}, \quad (6.18)$$

where $D = 2^N$. A derivation of Eq. (6.18) can be found in Ref. [Jin+21, Section 7]. Furthermore, we define the average infidelity

$$\mu_{\text{IFavg}} = 1 - \mu_{\text{Favg}}. \quad (6.19)$$

and a leakage measure

$$\mu_{\text{Leak}} = 1 - \left(\frac{\text{Tr}(\hat{M}\hat{M}^\dagger)}{D} \right), \quad (6.20)$$

whose definition is motivated by the average fidelity in Eq. (6.18) and the fact that $\text{Tr}(\hat{M}\hat{M}^\dagger)$ can be expressed solely in terms of the probability amplitudes

such that the relation $0 \leq \text{Tr}(\hat{M}\hat{M}^\dagger) \leq D$ holds true. Consequently, we find that $\mu_{\text{Leak}} \in [0, 1]$. The last gate-error quantifier we define is the diamond distance

$$\mu_\diamond = \frac{1}{2} \left\| \check{\mathcal{E}}_{\text{M}} \check{\mathcal{E}}_{\text{U}}^{-1} - \hat{I} \right\|_\diamond, \quad (6.21)$$

where the diamond norm

$$\|\check{\mathcal{A}}\|_\diamond = \sup_{\mathcal{H}^{\tilde{D}}} \left\{ \sup_{\hat{\rho} \in \mathcal{P}(\mathcal{H}^{\tilde{D}} \otimes \mathcal{H}^D)} \{ \|\check{I} \otimes \check{\mathcal{A}}(\hat{\rho})\|_{\text{Tr}} \} \right\}, \quad (6.22)$$

for an arbitrary superoperator $\check{\mathcal{A}}$ associated with the Hilbert space \mathcal{H}^D can be defined, see Ref. [San16, Section 2.1.1], in terms of a supremum over the finite-dimensional Hilbert spaces $\mathcal{H}^{\tilde{D}}$ and the trace norm

$$\|X\|_{\text{Tr}} = \text{Tr} \left(\sqrt{\hat{X}^\dagger \hat{X}} \right). \quad (6.23)$$

We can express the diamond distance as an infimum

$$\mu_\diamond = \frac{1}{2} \inf_{Q \in \text{GL}_4(\mathbb{C})} \left\{ \left\| \left(\hat{V}, \hat{I} \right) \hat{Q}^{-\dagger} \hat{Q}^{-1} \left(\hat{V}, \hat{I} \right)^T \right\|_2^{\frac{1}{2}} \right. \\ \left. \left\| \left(\hat{V}, -\hat{I} \right) \hat{Q}^\dagger \hat{Q}^1 \left(\hat{V}, -\hat{I} \right)^T \right\|_2^{\frac{1}{2}} \right\}, \quad (6.24)$$

over the set of complex, invertible two by two matrices, see Ref. [JKP09] or as a supremum

$$\mu_\diamond = \frac{1}{2} \sup_{|\psi\rangle \in \mathcal{H}^{2N}} \left\{ \left\| \left(\hat{V}^\dagger \otimes \hat{I} \right) |\psi\rangle\langle\psi| \left(\hat{V}^\dagger \otimes \hat{I} \right)^\dagger - |\psi\rangle\langle\psi| \right\|_{\text{Tr}} \right\}, \quad (6.25)$$

over the set of all state vectors $|\psi\rangle \in \mathcal{H}^{2N}$ of the IGQC, see Ref. [Wat18]. To the best knowledge of the author, a closed form expression for the diamond distance μ_\diamond is not known to the research community. Therefore, we use the relation

$$\mu_\diamond^{(\text{inf})} \leq \mu_\diamond \leq \mu_\diamond^{(\text{sup})}, \quad (6.26)$$

and the algorithms discussed in Ref. [Wil20a, Section 6.1.2] to obtain the value of μ_\diamond up to the fourth decimal.

Figure 6.3 shows an illustration of the computations we perform to obtain the gate-error quantifiers average infidelity $\mu_{\text{IF}_{\text{avg}}}$ given by Eq. (6.19), the leakage measure μ_{Leak} in Eq. (6.20) and the diamond distance μ_\diamond given by Eqs. (6.24) and (6.25). Here the gate-error quantifier μ_x denotes an arbitrary

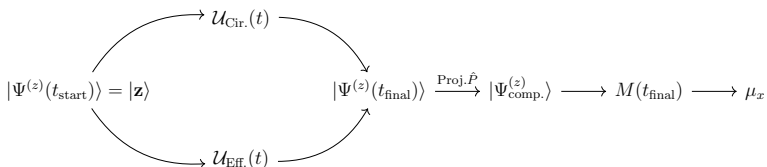


Figure 6.3: Illustration of the computations we perform to obtain a gate-error quantifier μ_x . Here, x denotes an arbitrary gate-error quantifier label. First, we simulate the time evolution $|\Psi^{(z)}(t_{\text{final}})\rangle$ for all 2^N initial states $|\Psi^{(z)}(t_{\text{start}})\rangle = |\mathbf{z}\rangle$ of a N -qubit NIGQC with the algorithms discussed in Chap. 4. Next, we apply the projection matrix P to obtain the computational states $|\Psi_{\text{comp.}}^{(z)}\rangle = P|\Psi^{(z)}(t_{\text{final}})\rangle$. Then, we construct the matrix M given by Eq. (6.27), i.e. we store the different computational states as the columns of the matrix M . Note that we use the programming paradigm MPI to parallelise the simulations of the 2^N time evolutions. Furthermore, all simulation are performed on the supercomputer JUWELS, see Ref. [Jül19]. Finally, we can use the matrix U of some target operation and the matrix M to compute $V = UM^\dagger$ and consequently μ_x by means of the expressions Eq. (6.19), Eq. (6.20) and Eqs. (6.24) and (6.25).

gate-error quantifier with the label x . First, we simulate the time evolution of the system $|\Psi^{(z)}(t_{\text{final}})\rangle$ for all 2^N computational basis states $|\Psi^{(z)}(t_{\text{start}})\rangle = |\mathbf{z}\rangle$ of an N -qubit NIGQC with the algorithms discussed in Chap. 4. Then, we compute the computational states $|\Psi_{\text{comp.}}^{(z)}\rangle = P|\Psi^{(z)}(t_{\text{final}})\rangle$ by applying the projection matrix P . Afterwards, we compute the matrix

$$M = \sum_{\mathbf{z} \in \{0,1\}^N} |\Psi_{\text{comp.}}^{(z)}\rangle \langle \mathbf{z}|, \quad (6.27)$$

where $|\mathbf{z}\rangle$ denotes one of the 2^N Cartesian unit vectors in the computer program. This means we store the computational states $|\Psi_{\text{comp.}}^{(z)}\rangle$ as the columns of the matrix M . In the end, we compute the matrix $V = UM^\dagger$ for the target operation U . Note that we parallelise the computations of the 2^N time evolutions with the programming paradigm MPI. Furthermore, the computations are performed on the supercomputer JUWELS, see Ref. [Jül19].

6.4 Spectrum of the four-qubit NIGQC obtained with the circuit Hamiltonian

The main subject of this section is the spectrum of the four-qubit NIGQC, illustrated in Fig. 6.4(c), and its importance for the realisation of two-qubit CZ gates. Hence, we begin this section with a general discussion of the gate implementation mechanism and then turn our attention to the spectrum itself.

In order to realise a CZ gate, we tune a target computational state like $|0, 0, 1, 1\rangle$ into resonance with a non-computational state like $|0, 0, 0, 2\rangle$. Then, we wait some time until the population between the two states has swapped back and forth. Hopefully, in the end, we find that the target computational state has gained an additional phase of $e^{i\pi}$ with respect to the remaining computational states of the NIGQC. Furthermore, since the remaining computational states have presumably acquired only a so-called dynamic phase, see Ref. [Section 6.6][Wei15], we apply z-axis rotations $R_i^{(z)}(\phi_i)$ to the different transmon qubits, see Ref. [Bla+21, Section VII B]. In this context, the word “tune” means that we tune the energies of the instantaneous basis states. Therefore, in our discussion of the gate implementation mechanism we refer to the instantaneous basis states of the system.

The flux control pulses we use to implement the CZ gates have pulse durations T_d between 80 and 125 ns and the time it takes for the population to swap back and forth lies roughly between 75 and 100 ns. The swap time

corresponds to the plateau time of a control pulse, see for example Figs. 6.2(b-c). We can control the plateau or swap time by changing the interaction strength constant G accordingly. For the simulations in this chapter, we use a constant interaction strength of 300 MHz for all dipole-dipole interactions. This constant roughly reproduces the gate times of the experiments. The flux control pulse amplitude δ , corresponds to the plateau height of a pulse. The pulse amplitude is determined by the condition that the states involved in the population exchange should have approximately the same energies. Furthermore, both pulse flanks play a crucial role in the gate implementation. The role of the pulse flanks is discussed together with the spectrum of the four-qubit NIGQC later in this section.

The authors of Ref. [Bla+21, Section VII B 2] distinguish between so-called adiabatic, see Ref. [DiC+19], and non-adiabatic gates, see Ref. [DiC+10]. However, in quantum theory the words adiabatic and non-adiabatic are usually only used in the context of the adiabatic approximation, see Ref. [Wei15, Section 6.6]. Clearly, since two instantaneous basis states exchange population, we find that the gate implementation mechanism described above is not an adiabatic process. Therefore, in the literature, the words adiabatic and non-adiabatic are used differently in this context. To the best knowledge of the author, in this particular context, gates implemented with shorter pulses are referred to as non-adiabatic and gates implemented with longer pulses are referred to as adiabatic gates. Additionally and also to the best knowledge of the author, in most theoretical studies, transmon qubits are modelled as adiabatic anharmonic oscillators or as adiabatic two-level systems, see for example Refs. [Rol+19; Fox+20; Yan+18]. Furthermore, we have to take into account an additional complication, namely that assigning labels to the energies $E^{(z)}(\varphi)$ and eigenstates $|\phi^{(z)}(\varphi)\rangle$ of a continuous set of Hermitian matrices $\{H(\varphi)\}$ is a non-trivial problem in itself, see Refs. [Hun27; NW93; Uhl20; SK20]. Here, $z \in \mathbb{N}^0$ and $\varphi/2\pi \in [0, 1]$. Also, once we apply flux control pulses like the once displayed in Figs. 6.2(b-c), we cannot exclude the possibility that additional transitions between the states of the system occur, i.e. more transitions than the ones needed to implement the CZ gates.

Since we would like to provide clarity on the issues mentioned above, we avoid using the nomenclature used in Ref. [Bla+21, Section VII B 2], while discussing the spectrum of the four-qubit NIGQC.

Figures 6.4(a-c) show the lowest fourteen energy levels of the four-qubit NIGQC illustrated in Fig. 6.1(c) as functions of the external flux offset φ . In panels(a-c) we drive the second ($i = 1$), third ($i = 2$) and fourth ($i = 3$) transmon qubit, respectively. The results are obtained with the device parameters listed in Table 6.1, the circuit Hamiltonian Eq. (3.114) and a standard diagonalisation algorithm, see Refs. [Cor; And+]. All transmon

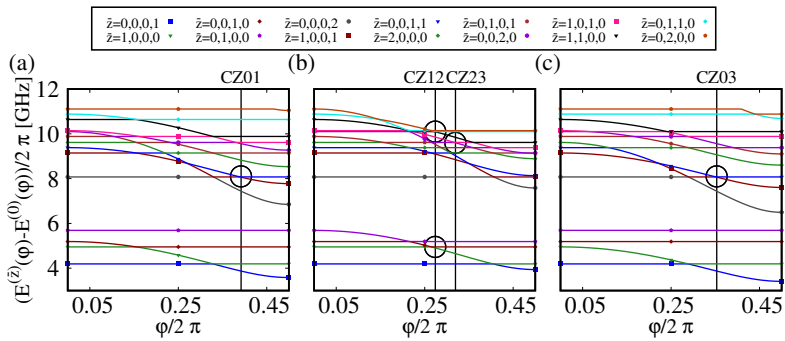


Figure 6.4: Energy levels as functions of the external flux φ . The lowest fourteen energy levels of the four-qubit NIGQC illustrated in Fig. 6.1(c) are computed. In panels(a,b,c) we tune the external flux φ of the second ($i = 1$), third ($i = 2$) and fourth ($i = 3$) transmon qubit, respectively. We use the device parameters listed in Table 6.1, the circuit Hamiltonian Eq. (3.114) and a standard diagonalisation algorithm, see Refs. [Cor; And+], to obtain the results. The energies $E^{(\bar{z})}(\varphi)$ of the interacting Hamiltonian Eq. (3.114) are labelled according to the sorted energies $E^{(\mathbf{z})}(\varphi)$ of the non-interacting Hamiltonian Eq. (3.114) at the operating points $\varphi_{0,i} = 0$.

qubits are modelled with three basis states. Similarly, all coupling resonators are modelled with two basis states only. Note that the circuit Hamiltonian in Eq. (3.61) has two symmetry points $\varphi/2\pi = 0.5$ and $\varphi/2\pi = 1$.

The energy levels $E^{(\bar{z})}(\varphi)$ of the interacting Hamiltonian with $G \neq 0$ are labeled according to the sorted energy levels $E^{(z)}(\varphi)$ of the non-interacting Hamiltonian with $G = 0$ at the flux offset value $\varphi = 0$. For clarity and better visibility, we add three markers to each energy level. Furthermore, we mark points of particular importance by means of black vertical lines and circles. The vertical lines indicate the pulse amplitudes $\varphi = \delta$ used to implement the CZ gates, see Table A.6. The circles indicate the energy level repulsions (ELRs) used to implement the CZ gates as well as ELRs that can lead to problems for the implementation of two-qubit CZ gates, see $\varphi/2\pi = 0.275$ in Fig. 6.4(b) for the CZ_{1,2} gate.

The operating point for all transmon qubits is $\varphi = 0$. This means, we always start from this point and tune the external flux to the point $\varphi = \delta$, see Figs. 6.4(a-c). On our way to the target ELR, i.e. the ELR that we use to activate population exchange between a target computational state and a non-computational state as described above, we have to pass through many unused ELRs with sufficient speed. Unused ELRs are the ones we do not use to activate population exchange between the target computational state and the non-computational state. If we do not pass through the unused ELRs sufficiently fast, we can observe (data not shown) population exchange between the two states whose energy levels repel each other. If we pass through the unused ELRs too fast, we can observe (data not shown) a variety of other transitions between the states of the system. For example, we can potentially observe that the coupler leaves its ground state. We can make these observations either by studying the matrix in Eq. (6.27) during the optimisation of the control pulse parameters or by explicitly studying the probabilities during the time evolution of the system. The algorithm we use to optimise the control pulse parameters seems to search for a balance between these two mechanisms. In order to find this balance, the optimisation algorithm mainly optimises the parameters σ , T_p and T_d . The pulse amplitude δ is usually only fine tuned to achieve nearly perfect population exchange and alignment of the phases. Note that we have to take into account all 2^N computational states and not only the two states which exchange population.

With this in mind, in Fig. 6.4(b), we can point out a problem that cannot be solved by optimising the control pulse parameters. If we consider the flux offset value $\varphi/2\pi = 0.275$ which is used to implement the CZ_{1,2} gate by means of the ELR marked in the energy band between 8 and 12 GHz, we can see a second ELR in the energy band between 4 and 6 GHz that is close to the flux offset value $\varphi/2\pi = 0.275$. This second ELR leads to

unwanted transitions between different computational states of the NIGQC. Some authors, see Ref. [And+20, Supplementary Material Section 1], suggest to remedy this issue by additionally driving non-interacting transmon qubits. Other authors, see Ref. [Aru+19, Supplementary Material Section VI C 1], suggest to solve this problem by redefining the target two-qubit gate such that the time evolution of the system fits more naturally to the target gate.

Furthermore, if we want to grasp the whole difficulty of the problem, we also have to take into account that driving different transmon qubits results in different energy levels, see Figs. 6.4(a-c). Consequently, if we begin to design a device or system, we are already confronted with an optimisation problem of exponential size. We have to avoid ELRs between all 2^N computational basis states of the NIGQC and at the same time take into account that driving different transmon qubits leads to different energy levels. To the best knowledge of the author, the largest PGQC in existence, based on the device architecture discussed in this chapter, has seventeen transmon qubits, see Ref. [Kri+22].

So far in this section we only considered two-qubit gates. However, single-qubit gates suffer from similar problems as two-qubit gates. The MP control pulses we use to implement the $R^{(x)}(\pi/2)$ rotations are not monochromatic, see Fig. 6.2(a). Therefore, adding a single transmon qubit to the system reduces the number of available frequencies that NIGQC computational basis states can be separated by. Since an IGQC with N qubits can execute single-qubit gates on every qubit, we potentially have to take into account the frequency bandwidths of all N MPs and the energy levels of all 2^N computational basis states. Consequently, we are confronted with another optimisation problem of exponential size.

6.5 Gate metrics for the elementary gate set

In this section we discuss the results of the control pulse optimisation for the single-qubit $R^{(x)}(\pi/2)$ rotations and the CZ gates. These gates form the elementary gate sets of our NIGQCs. This means we express all gates in a sequence of gates, i.e. a program, in terms of the elementary gate set and the virtual $R^{(z)}(\phi)$ rotations as discussed in Sec. 2.3. A useful table with gate decompositions can be found in Ref. [Wil20a, Appendix B]. Note that the results for the effective model presented in this section are obtained with the adiabatic effective Hamiltonian Eq. (3.117). Furthermore, the dipole-dipole interactions are time dependent and we use the series expansions in Eqs. (5.11) and (5.12) to model the flux-tunable frequencies and anharmonicities in Eq. (3.118), see Chap. 5 for more details.

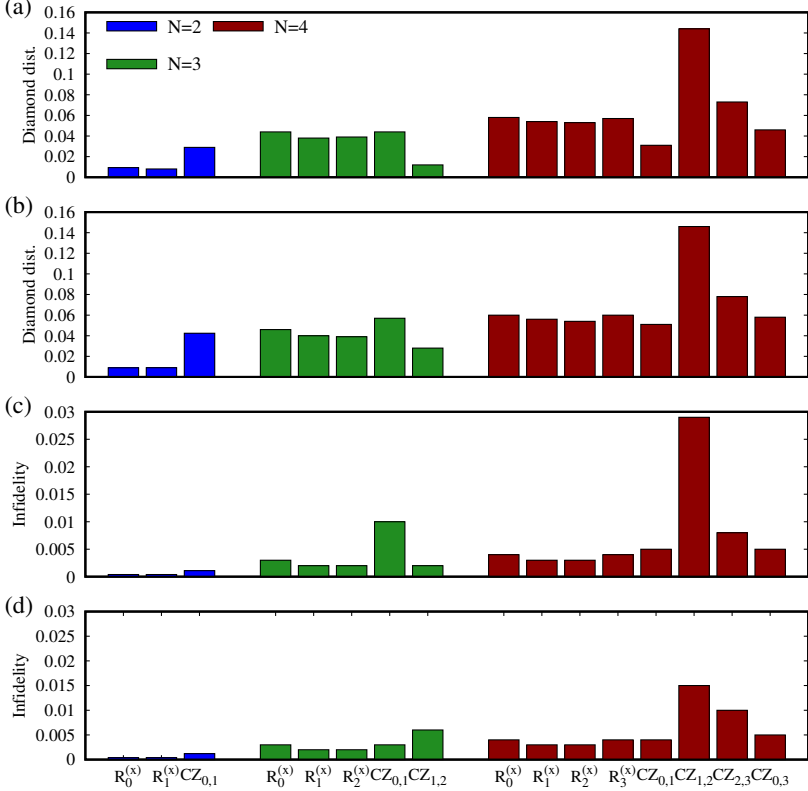


Figure 6.5: Gate-error metrics for single-qubit $R^{(x)}(\pi/2)$ and two-qubit CZ gates for the two-qubit (in blue), three-qubit (in green) and four-qubit (in red) NIGQCs illustrated in Figs. 6.1(a-c), respectively. In panels(a-b) we show the diamond distance μ_\diamond given by Eqs. (6.24) and (6.25) for the different NIGQCs obtained with the circuit Hamiltonian Eq. (3.114)(a) and the effective Hamiltonian Eq. (3.117)(b). Similarly, in panels(c-d) we show the average infidelity $\mu_{\text{IF}_{\text{avg}}}$ given by Eq. (6.19) for the different NIGQCs obtained with the circuit Hamiltonian Eq. (3.114)(c) and the effective Hamiltonian Eq. (3.117)(d). We use the device parameters listed in Table 6.1 and the control pulse parameters listed in Tables A.1-A.12 to obtain the results. Note that the gate-error metrics are also listed in Tables B.1-B.6.

Figures 6.5(a-d) show the diamond distance μ_\diamond (a,b) given by Eqs. (6.24) and (6.25) and the average infidelity $\mu_{\text{IF}_{\text{avg}}}$ (c,d) given by Eq. (6.19) for the two-qubit (in blue), three-qubit (in green) and four-qubit (in red) NIGQCs illustrated in Figs. 6.1(a-c), respectively. We use the optimisation algorithms provided by the NLOpt library, see Ref. [Joh], the device parameters listed in Table 6.1, the control pulse parameters listed in Tables A.1-A.6 (Tables A.7-A.12) and the circuit Hamiltonian Eq. (3.114) (effective Hamiltonian Eq. (3.117)) to obtain the results displayed in panels(a,c) (panels(b,d)). Note that the results are also listed in Tables B.1-B.6.

The lowest value for the diamond distance μ_\diamond (the average infidelity $\mu_{\text{IF}_{\text{avg}}}$) obtained with the circuit Hamiltonian Eq. (3.114) is 0.0080 (0.0004). These gate-error metrics were obtained with the control pulse parameters for the $R_1^{(x)}(\pi/2)$ gate. However, when we optimise the control pulse parameters for the $R_1^{(x)}(\pi/2)$ gate by simulating the circuit Hamiltonian Eq. (3.114) with the device parameters of the four-qubit NIGQC, the best value we can obtain is 0.058 (0.004) for the diamond distance μ_\diamond (the average infidelity $\mu_{\text{IF}_{\text{avg}}}$).

Similarly, the lowest value for the diamond distance μ_\diamond (the average infidelity $\mu_{\text{IF}_{\text{avg}}}$) obtained with the effective Hamiltonian Eq. (3.117) is 0.0089 (0.0004). These gate-error metrics were obtained with the effective Hamiltonian Eq. (3.114) and the control pulse parameters for the $R_0^{(x)}(\pi/2)$ gate. However, when we optimise the control pulse parameters for the $R_0^{(x)}(\pi/2)$ gate by simulating the effective Hamiltonian Eq. (3.117) with the device parameters of the four-qubit NIGQC, the best value we can obtain is 0.060 (0.004) for the diamond distance μ_\diamond (the average infidelity $\mu_{\text{IF}_{\text{avg}}}$).

Therefore, adding additional circuit elements to the system does not allow us to reproduce the good results for the case of the two-qubit NIGQCs.

The largest value for the diamond distance μ_\diamond (the average infidelity $\mu_{\text{IF}_{\text{avg}}}$) obtained with the circuit Hamiltonian Eq. (3.114) is 0.144 (0.029). Similarly, the largest value for the diamond distance μ_\diamond (the average infidelity $\mu_{\text{IF}_{\text{avg}}}$) obtained with the effective Hamiltonian Eq. (3.117) is 0.146 (0.015). All these gate-error metrics were obtained with the control pulse parameters for the CZ_{1,2} gates by simulating the circuit Hamiltonian Eq. (3.114) (effective Hamiltonian Eq. (3.117)) for the four-qubit NIGQC illustrated in Fig. 6.1(c).

We may explain these results by studying the energy levels in Fig. 6.4(b). The pulse amplitude we use to activate the CZ_{1,2} gate transitions is $\delta/2\pi = 0.275$. This value for the pulse amplitude activates not only the desired transitions which can be associated with the ELR, see black circle, in the energy band between 8 and 12 GHz but also transitions between computational states in the energy band between 4 and 6 GHz, see black circle. Ultimately,

the optimisation algorithm cannot find a value for the pulse amplitude that only activates the transitions of the upper ELR. Note that in order to reproduce this problem with the effective Hamiltonian Eq. (3.117), we have to use the higher-order series expansions in Eqs. (5.11) and (5.12) to model the flux-tunable frequency and anharmonicity of the transmon qubits, i.e. a first-order expansion does not lead to similar results (data not shown).

Furthermore, Figs. 6.5(a-d) also show a tendency to larger gate-error metrics for NIGQCs with more transmon qubits. Note that both models show this tendency. The largest system consists of eight circuit elements and the smallest system is made up of three circuit elements.

While obtaining the control pulse parameters for the two-qubit gates for NIGQCs of increasing size, we noticed increasing difficulty in navigating through the various ELRs in Fig. 6.4(a-c). The task is to pass through most ELRs sufficiently fast (slow) such that undesirable transitions do not occur. However, the increasing complexity of the energy levels makes this a difficult challenge. On the one hand, we can observe (data not shown) that using control pulses with fast rising (falling) pulse flanks can potentially cause all sorts of undesirable transitions. For example, for such pulses we could observe that the coupler leaves its ground state, see also the discussion in Sec. 5.5. On the other hand, we can observe (data not shown) that using control pulses with slow rising (falling) pulse flanks leads to transitions between states caused by the ELRs we have to pass through before we reach the one ELR we use to implement the actually desired transitions. The single-qubit $R^{(x)}(\pi/2)$ rotations suffer from a similar problem in the frequency space. The MPs we use to implement the $R^{(x)}(\pi/2)$ rotations are not monochromatic, this can potentially cause transitions between different states. Both these issues may explain the trend to larger gate-error metrics for NIGQCs of increasing size. Note that we discuss these issues also in Sec. 6.4.

Another explanation for the tendency to larger gate-error metrics might be the increasing difficulty of the problem passed to the optimisation algorithm. Optimising the control pulse parameters of an N -qubit NIGQC amounts to influencing the time evolution of the system such that the matrix M in Eq. (6.27) nearly perfectly aligns with the matrix U of the target gate. This means perfectly aligning 2^{2N+1} double-precision numbers without knowing how the system responds to a change in one or more of the control pulse parameters. Note that this task has to be carried out for all gates in the elementary gate set. To the best knowledge of the author, there exists no optimisation algorithm that can solve this task with guarantee of success.

Finally, we find (data not shown) that using the control pulse parameters optimised for the circuit (effective) model does not allow us to obtain comparable results with the effective (circuit) model. If we compute the gate-error

metrics for one model with the control pulse parameters optimised for the other model, we obtain values close to one for the diamond distance and the average infidelity. There are two reasons which can potentially explain this finding. First, in Chap. 5, we found that the circuit Hamiltonian Eq. (3.114) and the effective Hamiltonian Eq. (3.117) predict similar outcomes if we define the effective model accordingly. However, there still remain deviations sufficiently large to cause the mismatch between both models. Second, if we intend to reproduce the results presented in Figs. 6.5(a-d), we have to fine-tune control pulse parameters like the drive frequency $\omega^{(D)}$ ($\hbar\omega^{(D)}$) for the MP and the pulse amplitude δ ($\hbar\omega^{(q)}(\delta)$) for the UMP up to the sixth decimal (a couple of KHz) within one model. Furthermore, even within one model these parameters are highly susceptible to changes in the device parameters. For example, if we simulate the circuit and the effective model, we find that changing the interaction strength constant G from 300 to 301 MHz forces us to repeat the control pulse parameter optimisation for all gates.

6.6 Influence of higher states on gate-error quantifiers obtained with the circuit Hamiltonian

In this section we discuss how higher states affect the computation of gate-error quantifiers obtained with the circuit Hamiltonian Eq. (3.114) and the device parameters listed in Table 6.1. Here we only consider the four-qubit NIGQC illustrated in Fig. 6.1(c) and the execution of $R_0^{(x)}(\pi/2)$ gates.

Table 6.2 shows the diamond distance μ_\diamond given by Eqs. (6.24) and (6.25), the average infidelity $\mu_{\text{IF}_{\text{avg}}}$ in Eq. (6.19), the leakage measure μ_{Leak} given by Eq. (6.20) and the statistical distance μ_{SD} in Eq. (6.6) for the $R_0^{(x)}(\pi/2)$ gate. The statistical distance is obtained for the initial state $|0, 0, 0, 0\rangle$. Furthermore, the results in the first (second) row are obtained with the device parameters listed in Table 6.1, the control pulse parameters listed in Table A.5 row one, the circuit Hamiltonian Eq. (3.114) and four (sixteen) basis states for every transmon qubit. Note that the coupling resonators are modelled with four basis states only.

Table 6.2: Gate-error quantifiers for the $R_0^{(x)}(\pi/2)$ gate executed on the four-qubit NIGQC illustrated in Fig. 6.1(c). The first column shows the target gate. The second column shows the number of basis states n_J we use to model the dynamics of the transmon qubits. Note that all coupling resonators are modelled with four basis states only. The third column shows the diamond distance μ_\diamond given by Eqs. (6.24) and (6.25). The fourth column shows the average infidelity $\mu_{\text{IF}_{\text{avg}}}$ given by Eq. (6.19). The fifth column shows the leakage measure μ_{Leak} given by Eq. (6.20). The sixth column shows the statistical distance μ_{SD} given by Eq. (6.6) for the initial state $|0, 0, 0, 0\rangle$. We use the device parameters listed in Table 6.1, the control pulse parameters listed in Table A.5 first row and the circuit Hamiltonian Eq. (3.114) to obtain the results.

Gate	States n_J	μ_\diamond	$\mu_{\text{IF}_{\text{avg}}}$	μ_{Leak}	μ_{SD}
$R_0^{(x)}(\pi/2)$	4	0.0505	0.0037	0.0024	0.0014
$R_0^{(x)}(\pi/2)$	16	0.0584	0.0040	0.0024	0.0014

We can see that if we use sixteen basis states, the diamond distance and average infidelity show an increase in the third decimal. However, the leakage measure and the statistical distance are unaffected up to the fourth decimal. This finding can potentially be explained by the fact that the leakage measure and the statistical distance can be expressed solely in terms of the squares of the state vector amplitudes. Therefore, both these quantifiers do not take into account the phase of the system.

Figures 6.6(a-d) show the diamond distance μ_\diamond (a) in Eqs. (6.24) and (6.25), the average infidelity $\mu_{\text{IF}_{\text{avg}}}$ (b) given by Eq. (6.19), the leakage measure μ_{Leak} (c) in Eq. (6.20) and the statistical distance μ_{SD} (d) given Eq. (6.6) for the computational state $|0, 0, 0, 0\rangle$ as functions of the number of $R_0^{(x)}(\pi/2)$ gates executed on the four-qubit NIGQC illustrated in Fig. 6.1(c). Here we use the same simulation model and the same simulation parameters as before, see Table 6.2. The blue solid (green dashed) line represents the results for the case where all transmon qubits are modelled with four (sixteen) basis states. As before, all coupling resonators are modelled with four basis states only.

We can see that, after the execution of a few $R_0^{(x)}(\pi/2)$ gates, the diamond distance and the average infidelity start to exhibit deviations if we model the time evolution of the system with four and sixteen basis states. The deviations grow up to about 10% after the execution of twenty $R_0^{(x)}(\pi/2)$ gates. As before, we can observe that the leakage measure and the statistical

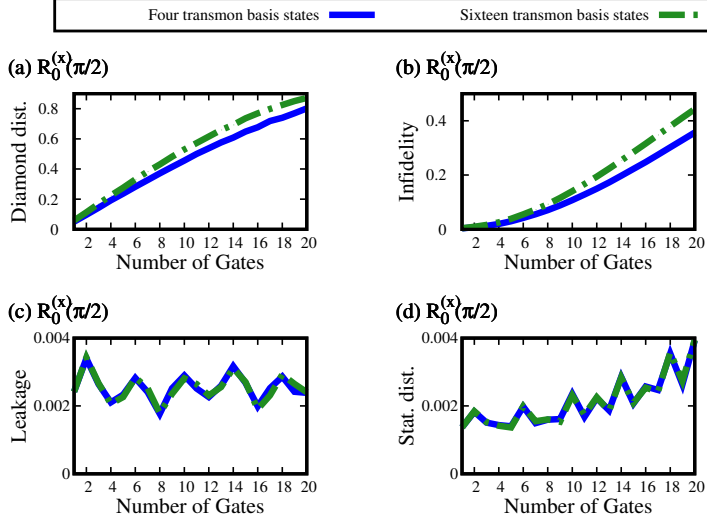


Figure 6.6: Gate-error quantifiers as functions of the number of $R_0^{(x)}(\pi/2)$ gates executed on the four-qubit NIGQC illustrated in Fig. 6.1(c). All resonators are modelled with four basis states and the transmon qubits are modelled with four (blue solid line) and sixteen (green dashed line) basis states n_J . Here we use the device parameters listed in Table 6.1, the control pulse parameters listed in Table A.5 first row and the circuit Hamiltonian Eq. (3.114) to obtain the results. In panels(a-d) we show the diamond distance μ_\diamond (a) given by Eqs. (6.24) and (6.25), the average infidelity $\mu_{\text{IF}_{\text{avg}}}$ (b) in Eq. (6.19), the leakage measure μ_{Leak} (c) given by Eq. (6.20) and the statistical distance μ_{SD} (d) in Eq. (6.6) for the initial state $|0, 0, 0, 0\rangle$.

distance are less affected. Here, we find that both quantifiers are roughly the same up to the fourth decimal. To the best knowledge of the author, in the literature, it seems to be common practice to use two or three basis states only to model the dynamics of NIGQCs and to compute gate errors, see for example Refs. [Gu+21; Wit+21; McK+16; Rot+17; Rol+19; Yan+18].

The results presented in this section lead the author of this thesis to advocate the view that gate-error metrics like the diamond distance and the average infidelity are only valid for the number of basis states they are obtained for. Consequently, changing the number of basis states leads to a new NIGQC model in the sense that the gate-error metrics obtained for the old and the new model are not guaranteed to be the same. Here we exclude the unlikely case that the truncated time-evolution operators, see Eqs. (6.27) and (6.15), are known to be the same for both models.

6.7 Influence of parameter changes on gate-error quantifiers obtained with the circuit Hamiltonian

In this section we discuss how changes in the control pulse parameters affect the computation of gate-error quantifiers obtained with the circuit Hamiltonian Eq. (3.114) and the device parameters listed in Table 6.1. To this end, we execute simple gate repetition programs that consist of twenty $\text{CNOT}_{0,1} = H_0 \text{CZ}_{0,1} H_0$ gates on the two-qubit, three-qubit and four-qubit NIGQCs illustrated in Fig. 6.1(a-c) and vary the control pulse amplitude $\delta = \delta_0 + \Delta\delta$ of the UMP that realises the $\text{CZ}_{0,1}$ gate. Here δ_0 denotes the optimised control pulses amplitude of the UMP and $\Delta\delta/2\pi \in \{0, 10^{-6}, 10^{-5}, 10^{-4}\}$ refers to the offset value we use to model the parameter change.

Table 6.3 shows the diamond distance μ_\diamond given by Eqs. (6.24) and (6.25), the average infidelity $\mu_{\text{IF}_{\text{avg}}}$ in Eq. (6.19), the leakage measure μ_{Leak} given by Eq. (6.20) and the statistical distance μ_{SD} in Eq. (6.6) for $\text{CNOT}_{0,1}$ gates executed on the two-qubit NIGQC illustrated in Fig. 6.1(a). The statistical distance is obtained for the initial state $|0, 0, 0, 0\rangle$. The results are obtained with the device parameters listed in Table 6.1, the control pulse parameters listed in Table A.1 and A.2 first rows and the circuit Hamiltonian Eq. (3.114). The different rows show the gate-error quantifiers obtained with slightly different $\Delta\delta$ control pulse amplitudes.

Table 6.3: Gate-error quantifiers for the $\text{CNOT}_{0,1}$ gate executed on the two-qubit NIGQC illustrated in Fig. 6.1(c). Here we use UMPs to implement the CZ gates. The first column shows the target gate. The second column shows pulse amplitude offset value $\Delta\delta$ we use to model parameter changes. The third column shows the diamond distance μ_\diamond given by Eqs. (6.24) and (6.25). The fourth column shows the average infidelity $\mu_{\text{IF}_{\text{avg}}}$ in Eq. (6.19). The fifth column shows the leakage measure μ_{Leak} given by Eq. (6.20). The sixth column shows the statistical distance μ_{SD} in Eq. (6.6) for the initial state $|0, 0, 0, 0\rangle$. We use the device parameters listed in Table 6.1, the control pulse parameters listed in Table A.1 and A.2 first rows and the circuit Hamiltonian Eq. (3.114) to obtain the results.

Gate	$\Delta\delta/2\pi$	μ_\diamond	$\mu_{\text{IF}_{\text{avg}}}$	μ_{Leak}	μ_{SD}
$\text{CNOT}_{0,1}$	0	0.0386	0.0018	0.0012	0.0013
$\text{CNOT}_{0,1}$	10^{-6}	0.0390	0.0018	0.0012	0.0013
$\text{CNOT}_{0,1}$	10^{-5}	0.0456	0.0022	0.0012	0.0013
$\text{CNOT}_{0,1}$	10^{-4}	0.1594	0.0156	0.0018	0.0038

As one can see, the diamond distance is affected by the third decimal if we add an offset value of $\Delta\delta/2\pi = 10^{-6}$ to the optimised pulse amplitude δ_0 . Furthermore, the remaining gate-error quantifiers stay the same up to the fourth decimal for this offset value. If we increase the offset value by a factor of ten to $\Delta\delta/2\pi = 10^{-5}$, we can observe that the diamond distance is affected by the second decimal and the average infidelity is affected by the third decimal. In this case too, the leakage measure and the statistical distance stay the same up to the fourth decimal. If we increase the offset value again such that $\Delta\delta/2\pi = 10^{-4}$, we find that all gate-error quantifiers are affected. The diamond distance, average infidelity, leakage measure and the statistical distance are affected by the first, second, fourth and third decimal, respectively.

Figures 6.7(a-f) show the diamond distance $\mu_\diamond(\text{a,c,e})$ in Eqs. (6.24) and (6.25) and the average infidelity $\mu_{\text{IF}_{\text{avg}}}(\text{b,d,f})$ given by Eq. (6.19) as functions of the number of executed gates on the two-qubit(a-b), the three-qubit(c-d) and the four-qubit(e-f) NIGQCs illustrated in Figs. 6.1(a-c), respectively. The program we execute consists of twenty $\text{CNOT}_{0,1}$ gates. Since we are interested in how the gate errors behave as a function of the number of executed gates, we do not optimise the circuit by removing identities. The results are obtained with the device parameters listed in Table 6.1, the control pulse parameters listed in Tables A.1-A.6 first rows and the circuit Hamiltonian Eq. (3.114). The blue solid (green dashed, red dashed, violet solid) line shows

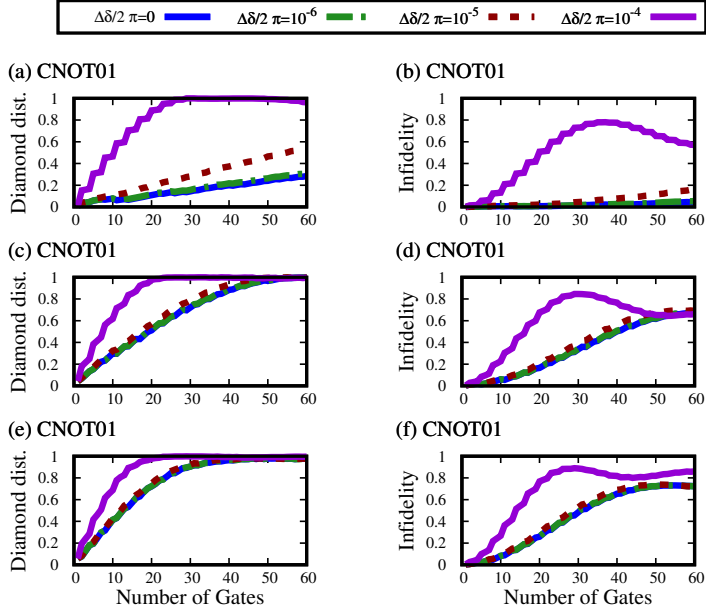


Figure 6.7: Gate-error metrics diamond distance $\mu_{\diamond}(a,c,e)$ given by Eqs. (6.24) and (6.25) and average infidelity $\mu_{\text{IF}_{\text{avg}}}(b,d,f)$ given by Eq. (6.19) as functions of the number of gates executed on the two-qubit(a-b), three-qubit(c-d) and four-qubit(e-f) NIGQCs illustrated in Figs. 6.1(a-c), respectively. The different NIGQCs execute the same program, namely a program with twenty $\text{CNOT}_{0,1}$ instructions. Note that all CNOT gates are implemented by means of two MPs and one UMP. The program is executed four times on each NIGQC. Here, we vary the pulse amplitude $\delta = \delta_0 + \Delta\delta$, where δ_0 denotes the optimised control pulse amplitude and $\Delta\delta$ denotes the offset value we use to model the parameter change. We use $\Delta\delta/2\pi \in \{0, 10^{-6}, 10^{-5}, 10^{-4}\}$ (blue solid line, green dashed line, red dashed line, violet solid line), the device parameter listed in Table 6.1, the control pulse parameters listed in Tables A.1-A.6 first rows and the circuit Hamiltonian Eq. (3.114) to obtain the results.

the results for the offset value $\Delta\delta = 0$ ($\Delta\delta = 10^{-6}$, $\Delta\delta = 10^{-5}$, $\Delta\delta = 10^{-4}$).

We can see that the offset value $\Delta\delta/2\pi = 10^{-6}$ only slightly affects the qualitative and quantitative behaviour of the gate-error trajectories displayed in Figs. 6.7(a-f). Furthermore, deviations usually only become visible at the end of the gate repetition programs. If we increase the offset value to $\Delta\delta/2\pi = 10^{-5}$, we find that the qualitative and quantitative behaviour of the gate-error trajectories becomes increasingly affected by the control pulse parameter change. The deviations seem to build up over time, i.e. the deviations seem to grow with the number of executed gates. If we use the offset value $\Delta\delta/2\pi = 10^{-4}$ to model the parameter change, we can observe some type of tipping behaviour in Figs. 6.7(a-f). The qualitative and quantitative behaviour of all gate-error trajectories show substantial non-linear change. Note that we can determine similar results (data not shown) if we model the parameter change by adding offset values $\Delta\omega^{(D)} \in \{0, 10^{-6}, 10^{-5}, 10^{-4}\}$ to the optimised drive frequency $\omega_0^{(D)}$ of the $R_0^{(x)}(\pi/2)$ pulse which allows us to realise the Hadamard gates H_0 .

The results discussed in this section show the fragile nature of the diamond distance and the average infidelity obtained with the circuit Hamiltonian Eq. (3.114) and the optimised control pulse parameters. The results rely on the assumption that we are able to control the flux-tunable qubit frequencies (the pulse amplitudes) at least up to a few KHz ($\Delta\delta/2\pi = 10^{-6}$). Furthermore, we should emphasise that we only discussed the case where one of the many control pulse parameters of a NIGQC was changed, namely the pulse amplitude of the UMP which realises the $CZ_{0,1}$ gate. This means we implicitly assumed that the remaining control pulse parameters are constant. Therefore, we should not be surprised by an increased susceptibility in cases where this assumption is not justified.

In order to emphasise the fragile nature of the gate-error metrics, we convert the offset value $\Delta\delta/2\pi = 10^{-6}$ for the external flux $\Delta\Phi_e = (\Phi_0\Delta\delta)/2\pi$ from Weber to Tesla and compare the associated magnetic field strength $\Delta|B|$ with the earth's magnetic field. Here, we consider an area of ten by ten micrometer and assume that the external flux is given by $\Phi_e = |B|A$, where $|B|$ is the magnetic field strength and A is the area of the surface we consider. The assumption about the area size is motivated by Ref. [RMC22]. The result $\Delta|B|$ for the offset value $\Delta\delta/2\pi = 10^{-6}$ is proportional to 10^{-11} Tesla. The earth's magnetic field is about 10^{-5} Tesla strong.

In our discussion so far, we ignored another feature of the gate-error trajectories displayed in Figs. 6.7(a-f). We can observe that the initial values for the gate-error metrics are poor predictors for the future behaviour of the gate error sequence which develops over time. Additionally, most gate-

error trajectories presented in this section certainly do not behave linearly. The results presented in this section confirm previous findings obtained for a different device architecture and therefore add additional evidence to the conclusion stated in Ref. [Wil+17].

In this thesis, we can provide an explanation for this finding. Even for very small gate-error metrics μ_x , where x denotes a label for an arbitrary gate-error metric, the time evolution $|\Psi(t)\rangle$ of the system is still governed by the TDSE and not only by the gate-error metric μ_x obtained for one particular moment in time t' , see Fig. 6.3. For the opposite to be true, we would need to show that a sufficiently small value for the gate-error metric μ_x imposes constraints on the time evolution of the system described by the state vector $|\Psi(t)\rangle$. Furthermore, since the time evolution of an arbitrary system, modelled with the Hamiltonian $\hat{H}(t)$, is by assumption generated by the time-evolution operator in Eq. (6.15) (the TDSE), we would need to prove that this statement is true for all Hamiltonians $\hat{H}(t)$. However, to the best knowledge of the author, there exists no such proof in the literature.

The results presented in this section lead the author of this thesis to advocate the view that gate-error metrics like the diamond distance and the average infidelity are at most measures of closeness at one particular point in time t' . How the gate errors in a NIGQC model develop over time is primarily determined by the TDSE and the model Hamiltonian.

6.8 Influence of the adiabatic approximation on gate-error quantifiers obtained with the effective Hamiltonian

In this section we discuss how a seemingly justified adiabatic approximation affects the computation of gate-error quantifiers obtained with the effective Hamiltonian Eq. (3.117) and the device parameters listed in Table 6.1. Since we are interested in the dynamic features of gate errors that emerge in NIGQC models, we execute simple gate repetition programs that consist of forty $\text{CZ}_{0,1}$ and forty $\text{CNOT}_{0,1} = \text{H}_0\text{CZ}_{0,1}\text{H}_0$ gates on the two-qubit, three-qubit and four-qubit NIGQCs illustrated in Fig. 6.1(a-c), respectively. We perform all simulations twice. During the first run (the adiabatic case), we model transmon qubits as adiabatic anharmonic oscillators, see Sec. 3.6 and 5.2. During the second run (the non-adiabatic case), we model transmon qubits as non-adiabatic anharmonic oscillators, see Sec. 3.6 and 5.2. The approximation we discuss in this section is often used in the literature, see for example Refs. [Fox+20; Gu+21; Rol+19; Rot+17; McK+16; Bak+22;

[Yan+18; Bla+21](#)].

In our simulations, we realise the $CZ_{0,1}$ gates with two types of control pulses, namely the UMP given by Eq. (6.3) and BMP given by Eq. (6.4), see Fig. 6.2(b-c), respectively. Since the BMP in Fig. 6.2(c) exhibits a fast falling pulse flank at around half of the pulse time T_p , we expect the deviations between the adiabatic and non-adiabatic case to be larger for the BMP. Furthermore, for both control pulses we find (data not shown) that the time derivative $\dot{\varphi}(t)$ of the external flux is practically zero for most of the pulse duration. Therefore, the deviations between both models should originate from the pulse flanks of the control pulses, see Fig. 6.2(b-c).

We first discuss the simulation results obtained with the UMP and the control pulse parameters listed in Tables A.7-A.8 first rows and then the simulation results acquired with the BMP and the control pulse parameters listed in Tables A.7-A.8 first and second rows.

Table 6.4 shows the diamond distance μ_\diamond given by Eqs. (6.24) and (6.25), the average infidelity $\mu_{\text{IF}_{\text{avg}}}$ in Eq. (6.19), the leakage measure μ_{Leak} given by Eq. (6.20) and the statistical distance μ_{SD} in Eq. (6.6) for $CZ_{0,1}$ gates executed on the two-qubit (first and second row), three-qubit (third and fourth row) and four-qubit (fifth and second sixth) NIGQCs illustrated in Fig. 6.1(a-c), respectively. The statistical distance is obtained for the computational state $|0, 0, 0, 0\rangle$. The odd (even) row numbers show the results obtained with the adiabatic (non-adiabatic) effective Hamiltonian Eq. (3.117) and the UMP. The simulation parameters we use are discussed at the beginning of this section.

Table 6.4: Gate-error quantifiers for the $\text{CZ}_{0,1}$ gate executed on the two-qubit, three-qubit and four-qubit NIGQCs illustrated in Figs. 6.1(a-c), respectively. Here we use the UMP to implement the CZ gates. The first column shows the target gate. The second column shows the type of effective Hamiltonian we use to describe the transmon qubits, see Eqs. (3.109) and (3.110). The third column shows figure references for the different NIGQCs we model. The fourth column shows the diamond distance μ_\diamond given by Eqs. (6.24) and (6.25). The fifth column shows the average infidelity $\mu_{\text{IF}_{\text{avg}}}$ in Eq. (6.19). The sixth column shows the leakage measure μ_{Leak} given by Eq. (6.20). The seventh column shows the statistical distance μ_{SD} in Eq. (6.6) for the initial state $|0, 0, 0, 0\rangle$. We use the device parameters listed in Table 6.1, the control pulse parameters listed in Table A.8, Table A.10 and Table A.12 first rows and the effective Hamiltonian Eq. (3.117) to obtain the results.

Gate	Adiabatic	System	μ_\diamond	$\mu_{\text{IF}_{\text{avg}}}$	μ_{Leak}	μ_{SD}
$\text{CZ}_{0,1}$	Yes	Figs. 6.1(a)	0.0424	0.0012	0.0005	0.0012
$\text{CZ}_{0,1}$	No	Figs. 6.1(a)	0.0425	0.0012	0.0005	0.0012
$\text{CZ}_{0,1}$	Yes	Figs. 6.1(b)	0.0569	0.0033	0.0017	0.0026
$\text{CZ}_{0,1}$	No	Figs. 6.1(b)	0.0574	0.0033	0.0017	0.0026
$\text{CZ}_{0,1}$	Yes	Figs. 6.1(c)	0.0514	0.0040	0.0028	0.0025
$\text{CZ}_{0,1}$	No	Figs. 6.1(c)	0.0509	0.0040	0.0028	0.0025

We can see (first and second rows) that the diamond distance for the two-qubit NIGQC modelled with the non-adiabatic effective Hamiltonian Eq. (3.117) is affected by the fourth decimal. Furthermore, the diamond distances for the three-qubit and four-qubit NIGQCs modelled with the non-adiabatic effective Hamiltonian Eq. (3.117) are affected by the third decimal. All other gate-error quantifiers are not affected up to the fourth decimal.

Figures 6.8(a-l) show the diamond distance $\mu_\diamond(\text{a,c,e,g,i,k})$ given by Eqs. (6.24) and (6.25) and the average infidelity $\mu_{\text{IF}_{\text{avg}}}(\text{b,d,f,h,j,l})$ in Eq. (6.19) as functions of the number of gates executed on the two-qubit(a-d), three-qubit(e-h) and four-qubit(i-l) NIGQCs illustrated in Figs. 6.1(a-c), respectively. The results for the CZ and CNOT gate repetition programs are displayed in panels(a,b,e,f,i,j) and panels(c,d,g,h,k,l), respectively. Here we use UMPs to realise the $\text{CZ}_{0,1}$ gates. The results obtained with the adiabatic (non-adiabatic) effective Hamiltonian Eq. (3.117) are presented as a solid (dashed) blue (green) line. The simulation parameters we use are discussed at the beginning of this section.

Overall, we can observe that the results for the adiabatic and non-adiabatic case show a good qualitative agreement. In panels(a,c,d,e,g,i) deviations be-

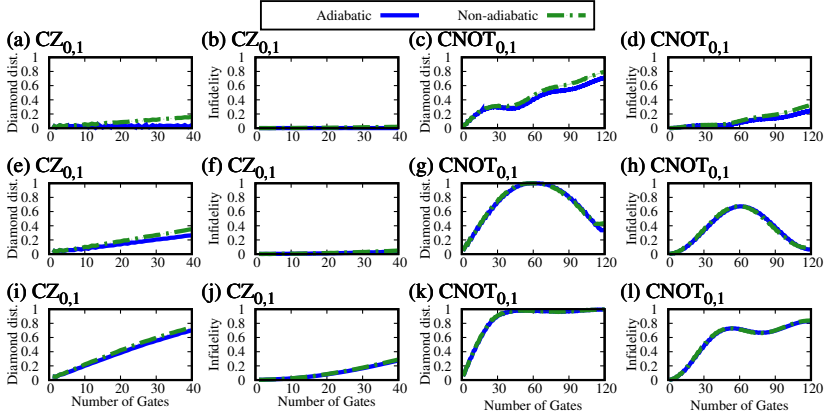


Figure 6.8: Gate-error metrics diamond distance $\mu_{\diamond}(\text{a,e,i,c,g,k})$ given by Eqs. (6.24) and (6.25) and average infidelity $\mu_{\text{IF}_{\text{avg}}}(\text{b,f,j,d,h,l})$ given by Eq. (6.19) as functions of the number of gates executed on the two-qubit(a-d), three-qubit(e-h) and four-qubit(i-l) NIGQCs illustrated in Figs. 6.1(a-c), respectively. The different NIGQCs execute the same programs, namely a program with forty $\text{CZ}_{0,1}$ (a,b,e,f,i,j) instructions and forty $\text{CNOT}_{0,1}$ (c,d,g,h,k,l) instructions. Note that all CNOT gates are implemented by means of two MPs and one UMP. The programs are executed two times on each NIGQC. During the first run (blue solid line) we model the transmon qubits as adiabatic anharmonic oscillators, see Eq. (3.110). During the second run (green dashed line) we model the transmon qubits as non-adiabatic anharmonic oscillators, see Eq. (3.109). We use the device parameter listed in Table 6.1, the control pulse parameters listed in Tables A.7-A.12 first rows and the effective Hamiltonian Eq. (3.117) to obtain the results.

tween both models seem to grow with the number of gates executed on the various NIGQCs. In panels(b,f,h,j,k,l) the deviations are barely noticeable.

We should probably emphasise that the UMPs we use in this section are rather long if we compare them with other instances which can be found in the literature, see for example Ref. [Fox+20]. Obviously, one cannot generalise the results presented in this section and conclude that modelling transmon qubits without the non-adiabatic drive term in Eq. (3.121) is justified. We now turn our attention to the results for BMPs.

Table 6.4 shows the diamond distance μ_\diamond in Eqs. (6.24) and (6.25), the average infidelity $\mu_{\text{IF}_{\text{avg}}}$ given by Eq. (6.19), the leakage measure μ_{Leak} in Eq. (6.20) and the statistical distance μ_{SD} given by Eq. (6.6) for $\text{CZ}_{0,1}$ gates executed on the two-qubit (first and second row), three-qubit (third and fourth row) and four-qubit (fifth and second sixth) NIGQCs illustrated in Fig. 6.1(a-c), respectively. The statistical distance is obtained for the computational state $|0, 0, 0, 0\rangle$. The odd (even) row numbers show the results obtained with the adiabatic (non-adiabatic) effective Hamiltonian Eq. (3.117) and the BMP. The simulation parameters we use are discussed at the beginning of this section.

Table 6.5: Gate-error quantifiers for the $\text{CZ}_{0,1}$ gate executed on the two-qubit, three-qubit and four-qubit NIGQCs illustrated in Figs. 6.1(a-c), respectively. Here we use the BMP to implement the CZ gates. The rows and columns show the same unit-less quantities as Table 6.4. We use the device parameters listed in Table 6.1, the control pulse parameters listed in Table A.8, Table A.10 and Table A.12 second rows and the effective Hamiltonian Eq. (3.117) to obtain the results.

Gate	Adiabatic	System	μ_\diamond	$\mu_{\text{IF}_{\text{avg}}}$	μ_{Leak}	μ_{SD}
$\text{CZ}_{0,1}$	Yes	Figs. 6.1(a)	0.0167	0.0006	0.0005	0.0004
$\text{CZ}_{0,1}$	No	Figs. 6.1(a)	0.0195	0.0007	0.0005	0.0004
$\text{CZ}_{0,1}$	Yes	Figs. 6.1(b)	0.0306	0.0042	0.0036	0.0020
$\text{CZ}_{0,1}$	No	Figs. 6.1(b)	0.0336	0.0042	0.0035	0.0019
$\text{CZ}_{0,1}$	Yes	Figs. 6.1(c)	0.0415	0.0043	0.0035	0.0024
$\text{CZ}_{0,1}$	No	Figs. 6.1(c)	0.0435	0.0044	0.0035	0.0024

If we consider the results for the two-qubit, three-qubit and four-qubit cases, we can see that the diamond distances are all affected by the third decimal. Furthermore, most average infidelities are affected by the fourth decimal. The leakage measure and the statistical distance are only affected in one case, i.e. for the three-qubit case.

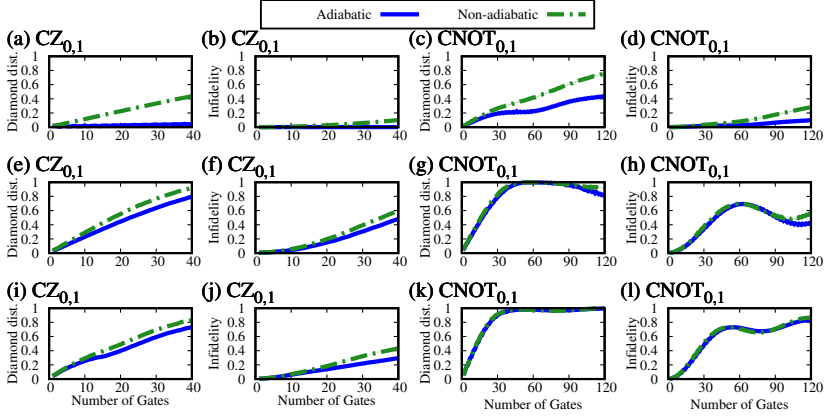


Figure 6.9: Gate-error metrics diamond distance $\mu_{\diamond}(a,e,i,c,g,k)$ given by Eqs. (6.24) and (6.25) and average infidelity $\mu_{\text{IF}_{\text{avg}}}(b,f,j,d,h,l)$ given by Eq. (6.19) as functions of the number of gates executed on the two-qubit(a-d), three-qubit(e-h) and four-qubit(i-l) NIGQCs illustrated in Figs. 6.1(a-c), respectively. The different NIGQCs execute the same programs, namely a program with forty $\text{CZ}_{0,1}$ (a,b,e,f,i,j) instructions and forty $\text{CNOT}_{0,1}$ (c,d,g,h,k,l) instructions. Note that all CNOT gates are implemented by means of two MPs and one BMP. The programs are executed two times on each NIGQC. During the first run (blue solid line) we model the transmon qubits as adiabatic anharmonic oscillators, see Eq. (3.110). During the second run (green dashed line) we model the transmon qubits as non-adiabatic anharmonic oscillators, see Eq. (3.109). We use the device parameter listed in Table 6.1, the control pulse parameters listed in Tables A.7-A.12 and the effective Hamiltonian Eq. (3.117) to obtain the results.

Figures 6.9(a-l) show the diamond distance $\mu_{\diamond}(\text{a,c,e,g,i,k})$ given by Eqs. (6.24) and (6.25) and the average infidelity $\mu_{\text{IF avg}}(\text{b,d,f,h,j,l})$ in Eq. (6.19) as functions of the number of gates executed on the two-qubit(a-d), three-qubit(e-h) and four-qubit(i-l) NIGQCs illustrated in Figs. 6.1(a-c), respectively. The results for the CZ and CNOT gate repetition programs are displayed in panels(a,b,e,f,i,j) and panels(c,d,g,h,k,l), respectively. Here we use BMPs to realise the $\text{CZ}_{0,1}$ gates. The results obtained with the adiabatic (non-adiabatic) effective Hamiltonian Eq. (3.117) are presented as a solid (dashed) blue (green) line. The simulation parameters we use are discussed at the beginning of this section.

As can be seen, the small numerical deviations we discussed in the context of Table 6.5 can over time substantially affect the quantitative and qualitative behaviour of the gate-error trajectories acquired with the adiabatic effective Hamiltonian Eq. (3.117). Additionally, the deviations between the adiabatic and non-adiabatic case are larger for the simulation scenario where we realise the $\text{CZ}_{0,1}$ gates with BMPs instead of the UMPs. Here again, we should emphasise that the UMPs and BMPs are very similar pulses, except for the fast falling pulse flank at around half of the pulse time T_p , see Fig. 6.2(b-c).

If we compare Figs. 6.8(a-l) and Figs. 6.9(a-l), we find that the qualitative and quantitative behaviour of the gate-error trajectories obtained with UMPs and BMPs and the corresponding independently optimised control pulse parameters is often rather different. Note that the remaining simulation parameters are identical for both cases. We can potentially explain this observation by considering the results discussed in Sec. 6.7. Here we found that the qualitative and quantitative behaviour of gate-error trajectories is often affected by small changes in the control pulse parameters. Consequently, since we optimised the control pulse parameters for the UMPs and BMPs independently, the deviations between Figs. 6.8(a-l) and Figs. 6.9(a-l) can also be caused by the different control pulse parameters we use to obtain the results. Obviously, this explanation is not conclusive.

In this section we discussed results located at the small border between the adiabatic and non-adiabatic effective models. We emphasise that the BMPs we use in this section have rather long pulse durations. For example, the authors of Ref. [Rol+19] investigate a device that consists of two transmon qubits coupled via a resonator. Here, the transmon qubits are modelled as adiabatic anharmonic oscillators and the BMPs which are used to realise CZ gates are about 40 ns long. The numerical results presented in Ref. [Rol+19] are likely to change once the transmon qubits are no longer modelled as adiabatic anharmonic oscillators. Furthermore, we emphasise that we only discussed the case where one of the transmon qubits was modelled non-adiabatically. Consequently, we may observe larger deviations for programs

where more than one non-adiabatically modelled transmon qubit experiences a flux drive.

The results discussed in this section are only concerned with a single seemingly justified and often applied approximation. However, it is usually the case that an effective Hamiltonian is the result of numerous approximations, see Chap. 5. Therefore, it seems impossible to decide whether or not an approximation (assumption) that can substantially affect the time-evolution of a system is justified or not. Obviously, here we consider scenarios where gate-error metrics like the diamond distance or the average infidelity are obtained by simulating the time-evolution of a system. As in one of the previous sections, we exclude the unlikely case that the truncated time-evolution operators, see Eqs. (6.27) and (6.15), are known to be the same for both models.

The data presented in this section lead the author of this thesis to advocate the view that every additional approximation or assumption leads to a new independent NIGQC model. Furthermore, the results presented in this section lead the author of this thesis to conjecture that the qualitative and quantitative behaviour of gate-error trajectories emerges due to a complex interplay of small deviations with respect to the target gates which occur over time. Note that the adiabatic and non-adiabatic effective Hamiltonian Eq. (3.117) only deviate for small time intervals, where $\dot{\varphi}(t) \neq 0$ and still we can observe how the gate-error trajectories for the different cases often diverge as time progresses. Certainly, gate errors in NIGQC models usually do not add up linearly.

6.9 Summary, Discussion and Conclusions

In this chapter we studied sequences of gate errors, in the form of gate-error quantifiers, that emerge if one repeatedly executes a gate on one of the two-qubit, three-qubit and four-qubit NIGQCs illustrated in Figs. 6.1(a-c), respectively. We call these sequences gate-error trajectories. The state of a NIGQC is by assumption completely determined by a time-dependent state vector $|\Psi(t)\rangle$ and the time-evolution of this state vector is, also by assumption, governed by the TDSE and a time-dependent model Hamiltonian $\hat{H}(t)$. We used the circuit Hamiltonian Eq. (3.114) and the effective Hamiltonian Eq. (3.117) to model the time-evolution of various NIGQCs. In Secs. 6.1-6.3 we specified the model. First, in Sec. 6.1, we introduced the device architecture and the device parameters, see Figs. 6.1(a-c) and Table 6.1. Next, in Sec. 6.2, we introduced the control pulses used to realise the different gates. The single-qubit $R^{(x)}(\pi/2)$ rotations are implemented with microwave pulses

(MPs), see Fig. 6.2(a). The two-qubit CZ gates are realised with unimodal pulses (UMPs), see Fig. 6.2(b) and bimodal pulses (BMPs), see Fig. 6.2(c). Then, in Sec. 6.3, we introduced the gate-error quantifiers used to assess to what extent the state of a NIGQC and the state of the IGQC deviate from one another for a given sequence of gates. Here, we discussed the diamond distance given by Eqs. (6.24) and (6.25), the average infidelity in Eq. (6.19), the leakage measure given by Eq. (6.20) and the statistical distance in Eq. (6.6). The main results in this chapter are presented in Secs. 6.4-6.8.

In Sec. 6.4 we discussed the energy levels of the four-qubit NIGQC illustrated in Figs. 6.1(c) as functions of the external flux φ . Here we used the circuit Hamiltonian Eq. (3.114) to obtain the results presented in Figs. 6.4(a-c). We also discussed the relevance of the spectrum for the two-qubit gates we implement and identified a problem that can potentially limit the scaling capabilities of the device architecture we studied in this chapter.

In Sec. 6.5 we discussed the gate-error metrics for the elementary gate sets, see Fig. 6.5, obtained with the optimised control pulse parameters, see Tables A.1-A.12. Here we used the circuit Hamiltonian Eq. (3.114) and the effective Hamiltonian Eq. (3.117) to obtain the results. We found that the gate-error metrics increase with the system size and discussed two potential, not mutually, exclusive causes for this trend.

In Sec. 6.6 we discussed how modelling the four-qubit NIGQC illustrated in Figs. 6.1(c) with four and sixteen basis states affects the gate-error quantifiers. Here we used the circuit Hamiltonian Eq. (3.114) and MP for the simulations of $R^{(x)}(\pi/2)$ single-qubit gates. The results are displayed in Table 6.2 and Figs. 6.6(a-d). We found that after one (twenty) $R^{(x)}(\pi/2)$ gates the diamond distance and the average infidelity for the two different numbers of basis states exhibit deviations at the order of 0.1% (10%). Furthermore, we found that the deviations for the leakage measure and the statistical distance are smaller by about a factor of one hundred. These results can potentially be explained by the fact that the leakage measure and the statistical distance can be expressed solely in terms of the squares of the state vector amplitudes. Note that gate-error quantifiers are often modelled with two or three basis states only, see for example Refs. [Gu+21; Wit+21; McK+16; Rot+17; Rol+19; Yan+18].

In Sec. 6.7 we discussed how modelling the two-qubit, three-qubit and four-qubit NIGQCs illustrated in Figs. 6.1(a-c) with slightly different control pulse parameters affects the gate-error quantifiers. Here we used the circuit Hamiltonian Eq. (3.114) for the simulations of $\text{CNOT}_{0,1} = H_0 \text{CZ}_{0,1} H_0$ gates. The results are displayed in Table 6.3 and Figs. 6.7(a-f). We used the pulse amplitude $\delta = \delta_0 + \Delta\delta$, where δ_0 refers to the optimised control pulse amplitude and $\Delta\delta/2\pi \in \{0, 10^{-6}, 10^{-5}, 10^{-4}\}$ denotes the offset value used

to model the parameter change, to realise the $\text{CZ}_{0,1}$ gates with UMPs. We found that all non-zero offset values affect the diamond distance to some extent and most non-zero offset values affect the average infidelity. The leakage measure and the statistical distance were less affected. Furthermore, we noticed some type of tipping behaviour for the offset value $\Delta\delta = 10^{-4}$, i.e., for this offset value, the qualitative and quantitative behaviour of the gate-error trajectories changed substantially. Note that we found this type of tipping behaviour for all three NIGQCs, see Fig. 6.1(a-c). We also converted the offset value $\Delta\delta/2\pi = 10^{-6}$ for the external flux $\Delta\Phi_e = (\Phi_0\Delta\delta)/2\pi$ from Weber to Tesla and found that gate-error metrics are sensitive to field strengths of about 10^{-11} Tesla. Here, we used an area size of ten by ten micrometer, see Ref. [RMC22]. In comparison, the earth's magnetic field is about 10^{-5} Tesla strong. Additionally, we found that the gate-error trajectories in Figs. 6.7(a-f) exhibit interesting non-linear behaviour.

In Sec. 6.8 we discussed how modelling the transmon qubits which constitute the two-qubit, three-qubit and four-qubit NIGQCs illustrated in Figs. 6.1(a-c) as adiabatic and non-adiabatic anharmonic oscillators, see Eqs. (3.110) and (3.109), affects the gate-error quantifiers. Here we used the effective Hamiltonian Eq. (3.117) for the simulations of $\text{CZ}_{0,1}$ and $\text{CNOT}_{0,1} = \text{H}_0\text{CZ}_{0,1}\text{H}_0$ gates with UMPs and BMPs. The results are displayed in Table 6.4 and 6.5 as well as Figs. 6.8(a-l) and Figs. 6.9(a-l). For the cases where we realised the $\text{CZ}_{0,1}$ gates with UMPs we found deviations between 0.001% (for the execution of a single $\text{CZ}_{0,1}$ gate, compare Table 6.4 first and second rows) and 20% (for the execution of forty $\text{CZ}_{0,1}$ gates, see Fig. 6.8(a)) for the diamond distance. For the remaining gate-error quantifiers we found smaller deviations. Additionally, for most cases the qualitative behaviour of the gate-error trajectories is barely affected by the approximation. The BMP seems to justify the approximation to a lesser extent, see Figs. 6.2(b-c) and in fact we found larger qualitative and quantitative deviations once we used BMPs to realise $\text{CZ}_{0,1}$ gates. Furthermore, we found that the gate-error trajectories for the CZ and CNOT gate repetition programs show interesting non-linear behaviour. Additionally, the qualitative behaviour of the gate-error trajectories that emerges by executing both programs can be substantially different. Therefore, the author of this thesis stated the conjecture that the qualitative and quantitative behaviour of gate-error trajectories emerges due to a complex interplay of small deviations with respect to the target gates which occur over time. Certainly, gate errors in the form of gate-error metrics, in NIGQC models usually do not add up linearly.

The results presented in this chapter, see Secs. 6.5-6.8, show that making even minor changes to a NIGQC model can affect the computation of gate-error metrics like the diamond distance and the average infidelity. On the

one hand, the mathematical relations used to obtain the gate-error metrics make the data we saw seem trivial, see Fig. 6.3. On the other hand, it is worth knowing to what extent the gate-error metrics are affected by changes in the model. Additionally, in Secs. 6.6-6.8 we focused on individual aspects of the model that can affect the computation of gate-error metrics. However, it is not too difficult to imagine what occurs once we start to combine the individual aspects by changing several aspects of the model at the same time, in such a case it seems highly unlikely that we are still able to establish a root cause for the varying gate-error metrics, i.e. the gate errors.

Therefore, the data discussed in this chapter leads the author of this thesis to conclude that almost all assumptions (approximations) we make with regard to the model can substantially affect the gate-error metrics. For this reason, the author of this thesis advocates the view that every additional approximation or assumption leads to a new independent NIGQC model. Here we exclude the unlikely case that the truncated time-evolution operators, see Eqs. (6.27) and (6.15), are known to be the same for the different models. The results presented in this chapter were selected to emphasise the small border between different NIGQC models. The author of this thesis is aware of various simulation scenarios where the deviations between different NIGQC models are of greater extent. However, the data presented was also selected to add evidence to the conjecture that the qualitative and quantitative behaviour of gate-error trajectories emerges due to a complex interplay of small deviations with respect to the target gates which occur over time.

In Secs. 6.6-6.8 we found that close values for the diamond distance and average infidelity for a target gate can result in substantially different qualitative behaviour of the gate-error metrics that emerge over time if we repeat the target gate several times. This leads the author of this thesis to conclude that the gate-error metrics alone cannot be used to predict the future behaviour of the sequence of gate-error metrics that emerges over time, let alone the performance of actual programs (algorithms). The data we discussed was obtained with the circuit Hamiltonian Eq. (3.114) and the effective Hamiltonian Eq. (3.117). Consequently, this is not a feature of one particular NIGQC model. Furthermore, the authors of Refs. [Wil+17; Wil20a] found similar evidence for a different device architecture modelled with a circuit Hamiltonian. Therefore, this is not a feature of one particular device architecture either. Hence, the author of this thesis advocates the view that gate-error metrics are at most a measure of closeness for a selected point in time. Certainly, gate-error metrics are not good predictors for future behaviour of a system. This finding is something we could have anticipated since the gate-error metrics have been derived in the context of the IGQC model. Changes of the state vector in this model are by assumption modelled as if they occur instan-

taneously. However, the future state of the system is always, by assumption, governed by the TDSE. Consequently, from a theoretical point of view there seems to be no reason to believe that gate-error metrics alone can be used to predict the time evolution of a system.

The discussion so far leads to the following line of reasoning. If we cannot even conclusively compare two NIGQC models and determine the root cause for the deviations between two models, how can we actually investigate gate errors in the form of gate-error metrics like the diamond distance and the average infidelity in PGQCs? Additionally, in Sec. 2.6 and the introduction to this chapter we already discussed several issues which affect the time evolution of PGQCs. Note that even the seemingly more complicated circuit Hamiltonian Eq. (3.114) which is based on the lumped-element approximation, see Sec. 3.1, does not take into account all of the issues discussed in the literature. Also, so far we did not take into account that measuring the state of the system is a non-trivial problem in itself, see Sec. 2.6. Adding all these factors to the uncertainties that the complex experimental setup, again see Sec. 2.6, already provides makes it seem impossible that a robust study of gate errors in the form of gate-error metrics like the diamond distance and the average infidelity is possible. All these issues lead the author of this thesis to advocate the view that the assessment of PGQCs should not be based on the seemingly fragile gate-error metrics considered in this chapter but on benchmark protocols like the ones discussed in Refs. [Wil+17; Mic+17].

Summary, concluding remarks and outlook

In Chap. 1 we provided an introduction to this thesis. Next, in Chap. 2, we reviewed the model of the ideal gate-based quantum computer (IGQC), see Secs. 2.1-2.5 and provided a distinction between the IGQC model, prototype gate-based quantum computers (PGQCs) and non-ideal gate-based quantum computers (NIGQCs), i.e. the time-dependent Hamiltonian models used to describe the time evolution of transmon systems in this thesis.

Then, in Chap. 3, we discussed the lumped-element approximation which provides the mathematical relations used to obtain circuit Hamiltonians, see Sec. 3.1. Furthermore, we reviewed the circuit quantisation formalism, see Sec. 3.2 and derived circuit Hamiltonians for LC resonators, fixed-frequency and flux-tunable transmons, see Sec. 3.3 and 3.4. Additionally, we provided clear and concise derivations of effective Hamiltonians for fixed-frequency and flux-tunable transmons, see Sec. 3.4. At the end of this chapter, we used these subsystems to define a many-particle circuit Hamiltonian (3.114) and an associated effective Hamiltonian Eq. (3.117), see Sec. 3.5 and 3.6 respectively. The interactions between the different subsystems are modelled as dipole-dipole interactions.

Afterwards, in Chap. 4, we reviewed the fundamental problems that we face if we numerically solve the TDSE for most time-dependent Hamiltonians, see Sec. 4.1. Furthermore, we provided a general discussion of the product-formula approach which can potentially be used to mitigate these problems, see Sec. 4.2 and discussed the product-formula algorithms used to solve the TDSE for the effective Hamiltonian Eq. (3.117) and the circuit Hamiltonian Eq. (3.114), see Sec. 4.3 and 4.4 respectively. Additionally, we reported on attempts to numerically solve the TDSE for the circuit Hamiltonian Eq. (3.114) in two alternative bases. At the end of this chapter, we discussed the update rules used to implement the product-formula algorithms in the simulation software and provided an overview of the simulation software itself, see Sec. 4.6 and 4.7 respectively.

Next, in Chap. 5, we studied how various approximations affect the time evolution of transmon systems modelled with the TDSE and the effective Hamiltonian Eq. (3.117). Additionally, we compared the time evolution of the circuit Hamiltonian Eq. (3.114) with the one of the effective Hamiltonian Eq. (3.117) for selected scenarios. Note that in this chapter we only considered the probability amplitudes relevant to the transitions which we study and transitions which can be activated by means of an external flux. We focused on three systems: a single flux-tunable transmon and two two-qubit device architectures, see Figs. 5.1(a-b). For the case of the single flux-tunable transmon, we found that the non-adiabatic effective Hamiltonian Eq. (3.109) covers at least some of the dynamic aspects of the associated circuit Hamiltonian Eq. (3.61), see Sec. 5.2. For device architecture I, see Fig. 5.1(a), we identified resonant transitions which are seemingly suppressed in the adiabatic effective model, see Sec. 5.3. Note that these transitions can be modelled with the non-adiabatic effective Hamiltonian Eq. (3.117) and the circuit Hamiltonian Eq. (3.114). Furthermore, we found that nominal small time-dependent oscillations, see Fig. 5.10(b), of the dipole-dipole interaction strength can substantially affect the ISWAP and CZ transitions modelled in Sec. 5.4. Additionally, we also found that the ISWAP and CZ transitions, i.e. the corresponding probability amplitudes, modelled in Sec. 5.4 are barely affected by the non-adiabatic driving term in Eq. (3.121). For device architecture II, see Fig. 5.1(b), we also identified resonant transitions which are seemingly suppressed in the adiabatic effective model, see Sec. 5.5. Again, these transitions can be modelled with the non-adiabatic effective Hamiltonian Eq. (3.117) and the circuit Hamiltonian Eq. (3.114). Additionally, we found that nominal large time-dependent reductions, see Fig. 5.17(b), of the dipole-dipole interaction strength affect the ISWAP and CZ transitions modelled in Sec. 5.6 to a lesser extent, cf. the results discussed in Sec. 5.4 and 5.6. Furthermore, we found that the ISWAP and CZ transitions, i.e. the corresponding probability amplitudes, modelled in Sec. 5.6 are barely influenced by the non-adiabatic driving term in Eq. (3.121).

Then, in Chap. 6, we investigated how susceptible certain gate-error quantifiers are to the approximations which make up the simulation model. Here, we considered the diamond distance, the average infidelity, a leakage measure and the statistical distance, see Sec. 6.3. Note that while the statistical distance and the leakage measure can be expressed solely in terms of probability amplitudes, the gate-error metrics diamond distance and average infidelity take into account the phases of the system. In this chapter we executed $R^{(x)}(\pi/2)$, CZ and CNOT gate repetition programs on the two-qubit, three-qubit and four-qubit NIGQCs illustrated in Figs. 6.1(a-c). Note that in this chapter we only considered device architecture II, cf. Chap. 5. We found that

several aspects of the simulation model clearly affect the gate-error metrics diamond distance and average infidelity, see Secs. 6.6-6.8. First, in Sec. 6.6 we considered how the number of basis states used to model the dynamics of the system can influence the gate-error quantifiers we compute, see Table 6.2 and Figs. 6.6(a-d). Then, in Sec. 6.7, we studied how changing the control pulse parameters can affect the gate-error quantifiers we compute, see Table 6.3 and Figs. 6.7(a-f). Afterwards, in Sec. 6.8 we considered how neglecting the non-adiabatic driving term in Eq. (3.121) can influence the gate-error quantifiers we compute, see Table 6.4 and 6.5, Figs. 6.8(a-l) and Figs. 6.9(a-l). The statistical distance and the leakage measure are affected to a lesser extent. Additionally, we found that gate errors in the form of gate-error metrics like the diamond distance and the average infidelity often do not behave linearly as a function of the number of gates executed on the different NIGQCs, see for example Figs. 6.8(a-l). In this regard, we should mention that, like the authors of Refs. [Wil+17; Wil20a], we found evidence that gate-error metrics like the diamond distance and the average infidelity are poor predictors of future gate errors which emerge over time. We also provided a clear and concise explanation for this finding, see Sec. 6.9. Furthermore, we identified potential problems for the scaling capabilities of the device architecture studied in Chap. 6, see Sec. 6.4 and 6.5.

The results presented in this work lead the author of this thesis to make the following concluding remarks. From the author's perspective the findings in Chap. 5 seem more than just plausible for the following reason. As long as we cannot prove that two time-evolution operators, see Eq. (4.2), for two different models are the same or close for the moments in time which are of interest to the physical scenario we investigate, we should not assume that the two different models make similar predictions for the same physical scenario. We already discussed this argument at the end of Sec. 3.4, 5.7 and 6.9. The results regarding the time dependence of the interaction strength in the effective model presented in Sec. 5.4 and 5.6 provide a striking example for a case where the intuitive argument that the nominal small oscillations in the dipole-dipole interaction strength, see Fig. 5.10(b), are negligible fails. Furthermore, these results become even more intriguing once we highlight the fact that the nominal much larger reduction in the time-dependent interaction strength shown in Fig. 5.17(b) affects the time evolution of a different system to a lesser extent. Also, dropping the driving term in Eq. (3.121) seems to result in the suppression of all sorts of transitions, see Sec. 5.3 and 5.5. Overall, the results in Chap. 5 show that modelling the time evolution of quantum systems is a difficult task since we cannot decide beforehand which dependencies are relevant to the problem at hand. Obviously, making intuitive decisions regarding the model can lead to unintended consequences.

Furthermore, in Chap. 6 we saw that theoretically sophisticated gate-error metrics like the diamond distance and the average infidelity are poor predictors for the future behaviour of the system. This finding seems trivial once we acknowledge the fact that the TDSE in Eq. (4.1) still governs the time evolution of the system and not only the gate-error metrics derived in the theoretical context of the static IGQC model. Additionally, since most approximations (assumptions) seem to clearly affect gate-error metrics, the author of this thesis advocates the view that every additional approximation (assumption) leads to a new independent NIGQC model. In other words, in almost all cases, we simply cannot easily estimate how modifying a model Hamiltonian affects the time evolution which is generated by the time-evolution operator in Eq. (4.2). Here, we exclude the unlikely case that the time-evolution operators, see Eq. (4.2), are known to be the same for the different models. The author of this thesis would like to make one final remark regarding the results presented in this thesis. Some authors emphasise the relevance of their research regarding PGQCs or NIGQC models by highlighting the aspects which seem relevant to the field of gate-based quantum computing, i.e. to the realisation of a fully functioning real-world gate-based quantum computer. The author of this thesis would like to highlight a different aspect of this type of work. The study of PGQCs and the associated NIGQC models allows us to learn more about the subtle issues that affect the time evolution of driven quantum systems.

The results presented in this work lead the author of this thesis to highlight the following future research opportunities. A review of the literature suggests, see Refs. [McK+16; Rot+17; Yan+18; Fox+20; Gu+21; Gan+20; McK+17; Did+18; Mot+09; Rol+19] and more, that it is more common to simplify NIGQC models which aim to describe superconducting PGQCs based on transmon qubits than to add more complexity to them. If we use real-time simulations to describe the dynamics of PGQCs, it seems more plausible to find a more balanced approach to the problem at hand. On the one hand, we would like to work with models which take into account all the relevant factors, see Sec. 2.6. On the other hand, we find that these models are usually not solvable, numerically and/or analytically. However, we are also aware of the fact that NIGQC models are rather fragile. Consequently, it seems plausible to strategically remove certain aspects of the model which prevent a solution but at the same time to add additional complexity in terms of variables like system size. However, we should have at least a rudimentary understanding of the approximations (assumptions) we make, see Chap. 5. For example, we could use the effective Hamiltonian Eq. (3.117) as the generator of the dynamics and add the read-out resonators and the measurement process to the simulation model, see Sec. 2.6. Since we use a

computer model to do so, we can turn on and off this aspect of the model and study how the measurement process itself affects the data acquisition. Note that this is only one aspect which could be integrated into the model. We should also take into account how susceptible gate-error metrics like the diamond distance and the average infidelity are. Consequently, it seems plausible to focus on modelling more robust quantities first, i.e. quantities which can be expressed in terms of probability amplitudes, see Chap. 6. Additionally, the author of this thesis was not able to find a satisfactory theoretical explanation for the time-dependent interaction strength effect discussed in Sec. 5.4 and 5.6. Hopefully other researchers are more successful in solving this intriguing problem. Note that parts of the simulation software used to obtain the results in this thesis are available as open source software, see Ref. [Lag20].

Appendices

A Tables with control pulse parameters for different NIGQC models

Table A.1: Control pulse parameters for single-qubit $R^{(x)}(\pi/2)$ gates implemented with the microwave pulse (MP) in Eq. (6.2). The first column shows the gate we model. The second column shows the pulse duration T_d in ns. The third column shows the drive frequency $\omega^{(D)}$ in GHz. The fourth column shows the unit-less pulse amplitude parameter a . The fifth column shows the nameless parameter σ in ns. The sixth column displays the DRAG amplitude b in ns. We use these parameters to model the two-qubit NIGQC illustrated in Fig. 6.1(a) with the circuit Hamiltonian Eq. (3.114).

Gate	T_d	$\omega^{(D)}/2\pi$	a	σ	b
$R_0^{(x)}(\pi/2)$	52.250	4.196	0.004	12.082	0.072
$R_1^{(x)}(\pi/2)$	52.950	5.195	0.005	10.000	0.070

Table A.2: Control pulse parameters for two-qubit CZ gates implemented with the unimodal pulse (UMP) in Eq. (6.3). The first column shows the gate we model. The second column shows the pulse type we use. The third column shows the pulse time T_p in ns. The fourth column shows the pulse duration T_d in ns. The fifth column shows the unit-less pulse amplitude δ . The sixth column displays the nameless parameter σ in ns. The parameter in column six allows us to control the pulse flanks. We use these parameters to model the two-qubit NIGQC illustrated in Fig. 6.1(a) with the circuit Hamiltonian Eq. (3.114).

Gate	Pulse	T_p	T_d	$\delta/2\pi$	σ
CZ _{0,1}	UMP	99.835	125.000	0.392	1.313

Table A.3: Control pulse parameters for single-qubit $R^{(x)}(\pi/2)$ gates implemented with the microwave pulse (MP) in Eq. (6.2). The units in this table are the same as in Table A.1. We use these parameters to model the three-qubit NIGQC illustrated in Fig. 6.1(b) with the circuit Hamiltonian Eq. (3.114).

Gate	T_d	$\omega^{(D)}/2\pi$	a	σ	b
$R_0^{(x)}(\pi/2)$	52.250	4.196	0.004	12.093	0.168
$R_1^{(x)}(\pi/2)$	52.950	5.190	0.004	9.997	0.067
$R_2^{(x)}(\pi/2)$	52.950	5.695	0.004	10.011	0.066

Table A.4: Control pulse parameters for two-qubit CZ gates implemented with the unimodal pulse (UMP) in Eq. (6.3). The units in this table are the same as in Table A.2. We use these parameters to model the three-qubit NIGQC illustrated in Fig. 6.1(b) with the circuit Hamiltonian Eq. (3.114).

Gate	Pulse	T_p	T_d	$\delta/2\pi$	σ
CZ _{0,1}	UMP	96.026	125.000	0.391	1.823
CZ _{1,2}	UMP	75.367	110.000	0.276	0.513

Table A.5: Control pulse parameters for single-qubit $R^{(x)}(\pi/2)$ gates implemented with the microwave pulse (MP) in Eq. (6.2). The units in this table are the same as in Table A.1. We use these parameters to model the four-qubit NIGQC illustrated in Fig. 6.1(c) with the circuit Hamiltonian Eq. (3.114).

Gate	T_d	$\omega^{(D)}/2\pi$	a	σ	b
$R_0^{(x)}(\pi/2)$	52.250	4.193	0.004	12.378	0.047
$R_1^{(x)}(\pi/2)$	52.950	5.190	0.004	10.255	0.063
$R_2^{(x)}(\pi/2)$	52.950	5.689	0.004	10.312	0.065
$R_3^{(x)}(\pi/2)$	52.950	4.951	0.005	10.191	0.012

Table A.6: Control pulse parameters for two-qubit CZ gates implemented with the unimodal pulse (UMP) in Eq. (6.3). The units in this table are the same as in Table A.2. We use these parameters to model the four-qubit NIGQC illustrated in Fig. 6.1(c) with the circuit Hamiltonian Eq. (3.114).

Gate	Pulse	T_p	T_d	$\delta/2\pi$	σ
CZ _{0,1}	UMP	100.241	125.000	0.392	1.283
CZ _{1,2}	UMP	68.046	90.000	0.275	0.182
CZ _{2,3}	UMP	80.500	94.000	0.320	0.500
CZ _{0,3}	UMP	97.708	116.000	0.353	1.458

Table A.7: Control pulse parameters for single-qubit $R^{(x)}(\pi/2)$ gates implemented with the microwave pulse (MP) in Eq. (6.2). The units in this table are the same as in Table A.1. We use these parameters to model the two-qubit NIGQC illustrated in Fig. 6.1(a) with the effective Hamiltonian Eq. (3.117).

Gate	T_d	$\omega^{(D)}/2\pi$	a	σ	b
$R_0^{(x)}(\pi/2)$	52.250	4.196	0.058	12.082	0.072
$R_1^{(x)}(\pi/2)$	52.950	5.195	0.065	10.000	0.070

Table A.8: Control pulse parameters for two-qubit CZ gates implemented with the unimodal pulse (UMP) and bimodal pulse (BMP) in Eqs. (6.3) and (6.4), respectively. The units in this table are the same as in Table A.2. We use these parameters to model the two-qubit NIGQC illustrated in Fig. 6.1(a) with the effective Hamiltonian Eq. (3.117).

Gate	Pulse	T_p	T_d	$\delta/2\pi$	σ
CZ _{0,1}	UMP	87.258	95.000	0.391	0.459
CZ _{0,1}	BMP	88.570	95.000	0.392	0.394

Table A.9: Control pulse parameters for single-qubit $R^{(x)}(\pi/2)$ gates implemented with the microwave pulse (MP) in Eq. (6.2). The units in this table are the same as in Table A.1. We use these parameters to model the three-qubit NIGQC illustrated in Fig. 6.1(b) with the effective Hamiltonian Eq. (3.117).

Gate	T_d	$\omega^{(D)}/2\pi$	a	σ	b
$R_0^{(x)}(\pi/2)$	52.250	4.196	0.058	12.082	0.072
$R_1^{(x)}(\pi/2)$	52.950	5.189	0.065	10.000	0.070
$R_2^{(x)}(\pi/2)$	52.950	5.694	0.066	9.990	0.032

Table A.10: Control pulse parameters for two-qubit CZ gates implemented with the unimodal pulse (UMP) and bimodal pulse (BMP) in Eqs. (6.3) and (6.4), respectively. The units in this table are the same as in Table A.2. We use these parameters to model the three-qubit NIGQC illustrated in Fig. 6.1(b) with the effective Hamiltonian Eq. (3.117).

Gate	Pulse	T_p	T_d	$\delta/2\pi$	σ
CZ _{0,1}	UMP	87.252	95.006	0.391	0.494
CZ _{0,1}	BMP	90.057	92.188	0.391	0.420
CZ _{1,2}	UMP	68.831	80.000	0.276	0.554

Table A.11: Control pulse parameters for single-qubit $R^{(x)}(\pi/2)$ gates implemented with the microwave pulse (MP) in Eq. (6.2). The units in this table are the same as in Table A.1. We use these parameters to model the four-qubit NIGQC illustrated in Fig. 6.1(c) with the effective Hamiltonian Eq. (3.117).

Gate	T_d	$\omega^{(D)}/2\pi$	a	σ	b
$R_0^{(x)}(\pi/2)$	52.250	4.191	0.058	12.082	0.072
$R_1^{(x)}(\pi/2)$	52.950	5.189	0.065	10.000	0.070
$R_2^{(x)}(\pi/2)$	52.950	5.688	0.066	9.990	0.032
$R_3^{(x)}(\pi/2)$	52.950	4.950	0.066	9.990	0.032

Table A.12: Control pulse parameters for two-qubit CZ gates implemented with the unimodal pulse (UMP) and bimodal pulse (BMP) in Eqs. (6.3) and (6.4), respectively. The units in this table are the same as in Table A.2. We use these parameters to model the four-qubit NIGQC illustrated in Fig. 6.1(c) with the effective Hamiltonian Eq. (3.117).

Gate	Pulse	T_p	T_d	$\delta/2\pi$	σ
CZ _{0,1}	UMP	87.254	95.013	0.391	0.453
CZ _{0,1}	BMP	89.925	98.114	0.392	0.400
CZ _{1,2}	UMP	67.802	115.238	0.275	0.338
CZ _{2,3}	UMP	71.620	98.197	0.320	0.543
CZ _{0,3}	UMP	92.616	124.768	0.353	1.877

B Tables with gate-error quantifiers for different NIGQC models

Table B.1: Gate-error quantifiers for $R^{(x)}(\pi/2)$ and CZ gates. The first column shows the target gate. The second column shows the control pulse we use to obtain the actual gate. The third column shows the unit-less diamond distance μ_\diamond given by Eqs. (6.24) and (6.25). The fourth column shows the unit-less average infidelity $\mu_{\text{IF}_{\text{avg}}}$ given by Eq. (6.19). The fifth column shows the leakage measure μ_{Leak} given by Eq. (6.20). We use the circuit Hamiltonian Eq. (3.114), the device parameters listed in Table 6.1 and the control pulse parameters listed in Table A.1 and A.2 to obtain the gate-error quantifiers in this table and to model the two-qubit NIGQC illustrated in Fig. 6.1(a).

Gate	Pulse	μ_\diamond	$\mu_{\text{IF}_{\text{avg}}}$	μ_{Leak}
$R_0^{(x)}(\pi/2)$	MP	0.0093	0.0004	0.0004
$R_1^{(x)}(\pi/2)$	MP	0.0080	0.0004	0.0004
$\text{CZ}_{0,1}$	UMP	0.0290	0.0011	0.0008

Table B.2: Gate-error quantifiers for $R^{(x)}(\pi/2)$ and CZ gates. The rows and columns show the same unit-less quantities as Table B.1. We use the circuit Hamiltonian Eq. (3.114), the device parameters listed in Table 6.1 and the control pulse parameters listed in Table A.3 and A.4 to obtain the gate-error quantifiers in this table and to model the three-qubit NIGQC illustrated in Fig. 6.1(b).

Gate	Pulse	μ_\diamond	$\mu_{\text{IF}_{\text{avg}}}$	μ_{Leak}
$R_0^{(x)}(\pi/2)$	MP	0.044	0.003	0.002
$R_1^{(x)}(\pi/2)$	MP	0.038	0.002	0.001
$R_2^{(x)}(\pi/2)$	MP	0.039	0.002	0.001
$\text{CZ}_{0,1}$	UMP	0.044	0.010	0.010
$\text{CZ}_{1,2}$	UMP	0.012	0.002	0.002

Table B.3: Gate-error quantifiers for $R^{(x)}(\pi/2)$ and CZ gates. The rows and columns show the same unit-less quantities as Table B.1. We use the circuit Hamiltonian Eq. (3.114), the device parameters listed in Table 6.1 and the control pulse parameters listed in Table A.5 and A.6 to obtain the gate-error quantifiers in this table and to model the four-qubit NIGQC illustrated in Fig. 6.1(c).

Gate	Pulse	μ_{\diamond}	$\mu_{\text{IF}_{\text{avg}}}$	μ_{Leak}
$R_0^{(x)}(\pi/2)$	MP	0.058	0.004	0.002
$R_1^{(x)}(\pi/2)$	MP	0.054	0.003	0.002
$R_2^{(x)}(\pi/2)$	MP	0.053	0.003	0.002
$R_3^{(x)}(\pi/2)$	MP	0.057	0.004	0.002
CZ _{0,1}	UMP	0.031	0.005	0.004
CZ _{1,2}	UMP	0.144	0.029	0.018
CZ _{2,3}	UMP	0.073	0.008	0.005
CZ _{0,3}	UMP	0.046	0.005	0.003

Table B.4: Gate-error quantifiers for $R^{(x)}(\pi/2)$ and CZ gates. The rows and columns show the same unit-less quantities as Table B.1. We use the effective Hamiltonian Eq. (3.117), the device parameters listed in Table 6.1 and the control pulse parameters listed in Table A.7 and A.8 to obtain the gate-error quantifiers in this table and to model the two-qubit NIGQC illustrated in Fig. 6.1(a).

Gate	Pulse	μ_{\diamond}	$\mu_{\text{IF}_{\text{avg}}}$	μ_{Leak}
$R_0^{(x)}(\pi/2)$	MP	0.0089	0.0004	0.0004
$R_1^{(x)}(\pi/2)$	MP	0.0090	0.0004	0.0004
CZ _{0,1}	UMP	0.0424	0.0012	0.0005
CZ _{0,1}	BMP	0.0167	0.0006	0.0005

Table B.5: Gate-error quantifiers for $R^{(x)}(\pi/2)$ and CZ gates. The rows and columns show the same unit-less quantities as Table B.1. We use the effective Hamiltonian Eq. (3.117), the device parameters listed in Table 6.1 and the control pulse parameters listed in Table A.9 and A.10 to obtain the gate-error quantifiers in this table and to model the three-qubit NIGQC illustrated in Fig. 6.1(b).

Gate	Pulse	μ_{\diamond}	$\mu_{\text{IF}_{\text{avg}}}$	μ_{Leak}
$R_0^{(x)}(\pi/2)$	MP	0.046	0.003	0.002
$R_1^{(x)}(\pi/2)$	MP	0.040	0.002	0.002
$R_2^{(x)}(\pi/2)$	MP	0.039	0.002	0.002
$\text{CZ}_{0,1}$	UMP	0.057	0.003	0.002
$\text{CZ}_{0,1}$	BMP	0.031	0.004	0.004
$\text{CZ}_{1,2}$	UMP	0.028	0.006	0.006

Table B.6: Gate-error quantifiers for $R^{(x)}(\pi/2)$ and CZ gates. The rows and columns show the same unit-less quantities as Table B.1. We use the effective Hamiltonian Eq. (3.117), the device parameters listed in Table 6.1 and the control pulse parameters listed in Table A.11 and A.12 to obtain the gate-error quantifiers in this table and to model the four-qubit NIGQC illustrated in Fig. 6.1(c).

Gate	Pulse	μ_{\diamond}	$\mu_{\text{IF}_{\text{avg}}}$	μ_{Leak}
$R_0^{(x)}(\pi/2)$	MP	0.060	0.004	0.003
$R_1^{(x)}(\pi/2)$	MP	0.056	0.003	0.002
$R_2^{(x)}(\pi/2)$	MP	0.054	0.003	0.002
$R_3^{(x)}(\pi/2)$	MP	0.060	0.004	0.002
$\text{CZ}_{0,1}$	UMP	0.051	0.004	0.003
$\text{CZ}_{0,1}$	BMP	0.041	0.004	0.004
$\text{CZ}_{1,2}$	UMP	0.146	0.015	0.005
$\text{CZ}_{2,3}$	UMP	0.078	0.010	0.007
$\text{CZ}_{0,3}$	UMP	0.058	0.005	0.003

List of Figures

2.1	Photograph of a chip which contains structures such that five superconducting qubits might be realised in an experiment. . .	18
2.2	Photographs of an experimental setup to operate a superconducting integrated circuit at around 10 mK.	20
3.1	Circuit diagram of a network containing a linear capacitor with capacitance C and a linear inductor with inductance L	33
3.2	Circuit diagram of a network containing a linear capacitor with capacitance C , a Josephson junction with Josephson energy E_J and a voltage source $V_g(t)$	36
3.3	Circuit diagram of a network containing linear capacitors with capacitances C_l (left) and C_r (right), two Josephson junctions with Josephson energies $E_{J,l}$ (left) and $E_{J,r}$ (right) as well as an external flux $\Phi_e(t)$ threading through the center loop. . . .	38
3.4	Circuit diagram of a network. The right branch contains linear capacitors with capacitances C_l (left) and C_r (right), two Josephson junctions with Josephson energies $E_{J,l}$ (left) and $E_{J,r}$ (right) as well as an external flux $\Phi_e(t)$ threading through the loop which contains the two Josephson junctions.	41
3.5	The lowest five energy levels of a fixed-frequency transmon as a function of the effective offset charge n_g	45
3.6	Overlaps $ \langle \phi^{(z)}(0) \phi^{(0)}(n_g) \rangle ^2$ (a) and $ \langle \phi^{(z)}(0) \phi^{(1)}(n_g) \rangle ^2$ (b) between the instantaneous eigenstates $ \phi^{(0/1)}(\varphi)\rangle$ and various measurement basis states $ \phi^{(z)}(0)\rangle$ as functions of the external charge n_g	46
3.7	The lowest five energy levels of a flux-tunable transmon as a function of the flux offset φ	49
3.8	Overlaps $ \langle \phi^{(z)}(0) \phi^{(0)}(\varphi) \rangle ^2$ (a) and $ \langle \phi^{(z)}(0) \phi^{(1)}(\varphi) \rangle ^2$ (b) between the instantaneous eigenstates $ \phi^{(0/1)}(\varphi)\rangle$ and various measurement basis states $ \phi^{(z)}(0)\rangle$ as functions of the external flux φ	49

4.1	High-level sketch of the simulation software and the surrounding compute infrastructure.	77
5.1	Sketches of two different device architectures I(a) and II(b) to realise NIGQCs, see Sec. 2.7.	83
5.2	Fluxes $\varphi/2\pi$ given by Eq. (5.1) as a functions of time t for two sets of pulse parameters.	86
5.3	Ground-state probabilities $p^{(0)}$ as functions of the pulse duration T_d and the drive frequency $\omega^{(D)}$	91
5.4	Ground-state $p^{(0)}(t)$ (a) and first-excited-state $p^{(1)}(t)$ (b) probabilities as functions of time t for the effective and the circuit model. We use the system parameters listed in Table 5.1, row $i = 2$ and the pulse in Eq. (5.1) with $T_{r/f} = T_d/2$ and $\delta/2\pi = 0.001$ to obtain the results.	92
5.5	Probabilities $1 - p^{(z)}$ for $z \in \{0, 1, 2, 3\}$ as functions of the rise and fall time $T_{r/f}$ and the pulse amplitude δ	93
5.6	Deviations $\Delta(E^{(z)} - E^{(0)})$ given by Eq. (5.14) between the numerically exact eigenvalues of the circuit Hamiltonian and three approximations I(a), II(b) and III(c) for the eigenvalues as functions of the flux $\varphi/2\pi$	95
5.7	Probabilities $p^{\mathbf{z}}$ for $\mathbf{z} = (0, 0, 0)$ as functions of the pulse duration T_d and the drive frequency $\omega^{(D)}$ (circuit model architecture I).	97
5.8	Probabilities $p^{\mathbf{z}}(t) = \langle \mathbf{z} \Psi(t) \rangle ^2$ as functions of time t for $\mathbf{z} = (0, 0, 1)$ and $\mathbf{z} = (0, 1, 0)$ (circuit model architecture I). . .	101
5.9	Probabilities $p^{\mathbf{z}}(t) = \langle \mathbf{z} \Psi(t) \rangle ^2$ as functions of time t for $\mathbf{z} = (0, 1, 1)$ and $\mathbf{z} = (0, 2, 0)$ (circuit model architecture I). . .	102
5.10	Effective interaction strength $g^{(c,e)}$ between a fixed-frequency and a flux-tunable transmon as a function of the external flux φ in panel(a) and as a function of time t in panel(b).	104
5.11	Probabilities $p^{\mathbf{z}}(t) = \langle \mathbf{z} \Psi(t) \rangle ^2$ as functions of time t (effective model architecture I).	105
5.12	Sketch of the simulation scenario we use to obtain the results shown in Figs. 5.13(a-b).	107
5.13	Probabilities $p^{\mathbf{z}}(t) = \langle \mathbf{z} \Psi(t) \rangle ^2$ for $\mathbf{z} = (0, 0, 1)$ (a) and $\mathbf{z} = (0, 1, 1)$ (b) as functions of time t (effective model architecture I).	109
5.14	Probabilities $p^{\mathbf{z}}$ for $\mathbf{z} = (0, 0, 0)$ (a), $\mathbf{z} = (1, 0, 0)$ (b) and $\mathbf{z} = (2, 0, 0)$ (c) as functions of the pulse duration T_d and the drive frequency $\omega^{(D)}$ (circuit model architecture II).	111

5.15	Probabilities $p^{\mathbf{z}}(t) = \langle \mathbf{z} \Psi(t) \rangle ^2$ as functions of time t for $\mathbf{z} = (0, 0, 1)$ and $\mathbf{z} = (0, 1, 0)$ (circuit model architecture II).	113
5.16	Probabilities $p^{\mathbf{z}}(t) = \langle \mathbf{z} \Psi(t) \rangle ^2$ as functions of time t for $\mathbf{z} = (0, 1, 1)$ and $\mathbf{z} = (0, 0, 2)$ (circuit model architecture II).	115
5.17	Effective interaction strength $g^{(a,c)}$ between a resonator and a flux-tunable transmon as a function of the external flux φ in panel(a) and as a function of time t in panel(b).	116
5.18	Probabilities $p^{\mathbf{z}}(t) = \langle \mathbf{z} \Psi(t) \rangle ^2$ as functions of time t . (effective model architecture II)	117
6.1	Illustrations of a two-qubit(a), three-qubit(b) and four-qubit(c) NIGQC.	125
6.2	External charge $n(t)$ (a) and external flux $\varphi(t)$ (b-c) as functions of time t .	128
6.3	Illustration of the computations we perform to obtain a gate-error quantifier μ_x , where x denotes an arbitrary gate-error quantifier label.	134
6.4	Energy levels as functions of the external flux φ .	137
6.5	Gate-error metrics for single-qubit $R^{(x)}(\pi/2)$ and two-qubit CZ gates for the two-qubit (in blue), three-qubit (in green) and four-qubit (in blue) NIGQCs illustrated in Figs. 6.1(a-c), respectively.	140
6.6	Gate-error quantifiers as functions of the number of $R_0^{(x)}(\pi/2)$ gates executed on a four-qubit NIGQC.	145
6.7	Gate-error metrics diamond distance $\mu_\diamond(a,c,e)$ in Eq. (6.24) and Eq. (6.25) and average infidelity $\mu_{\text{IF}_{\text{avg}}}(b,d,f)$ in Eq. (6.19) as functions of the number of gates executed on the two-qubit(a-b), three-qubit(c-d) and four-qubit(e-f) NIGQCs illustrated in Figs. 6.1(a-c), respectively (parameter change study).	148
6.8	Gate-error metrics diamond distance $\mu_\diamond(a,e,i,c,g,k)$ given by Eqs. (6.24) and (6.25) and average infidelity $\mu_{\text{IF}_{\text{avg}}}(b,f,j,d,h,l)$ given by Eq. (6.19) as functions of the number of gates executed on the two-qubit(a-d), three-qubit(e-h) and four-qubit(i-l) NIGQCs illustrated in Figs. 6.1(a-c), respectively (approximation study with UMP pulse).	153
6.9	Gate-error metrics diamond distance $\mu_\diamond(a,e,i,c,g,k)$ given by Eqs. (6.24) and (6.25) and average infidelity $\mu_{\text{IF}_{\text{avg}}}(b,f,j,d,h,l)$ given by Eq. (6.19) as functions of the number of gates executed on the two-qubit(a-d), three-qubit(e-h) and four-qubit(i-l) NIGQCs illustrated in Figs. 6.1(a-c), respectively (approximation study with BMP pulse).	155

List of Tables

3.1	Parameters for a fixed-frequency (row $i = 0$) and a flux-tunable transmon (row $i = 1$). The first column contains indices i for the different parameter sets. The second and third column show the qubit frequency $\omega^{(Q_0)}$ and anharmonicity $\alpha^{(Q_0)}$	44
5.1	Device parameters for the circuit Hamiltonian (Architecture I).	84
5.2	Device parameters for the circuit Hamiltonian (Architecture II).	84
5.3	Device parameters for the adiabatic effective Hamiltonian Eq. (3.117) where the interaction strength is constant.	85
5.4	Device parameters for the adiabatic effective Hamiltonian Eq. (3.117) where the interaction strength is constant.	85
5.5	Summary of the pulse parameters we use to perform simulations with the circuit Hamiltonian model.	88
5.6	Summary of pulse parameters we use to perform simulations with the adiabatic effective Hamiltonian model.	89
5.7	Table with grid search parameters and the results of the corresponding searches in terms of the values for ϵ	98
6.1	Device parameters for the two-qubit, three-qubit and four-qubit NIGQCs illustrated in Figs. 6.1(a-c).	126
6.2	Gate-error quantifiers for the $R_0^{(x)}(\pi/2)$ gate executed on the four-qubit NIGQC illustrated in Fig. 6.1(c).	144
6.3	Gate-error quantifiers for the $\text{CNOT}_{0,1}$ gate executed on the two-qubit NIGQC illustrated in Fig. 6.1(c).	147
6.4	Gate-error quantifiers for the $\text{CZ}_{0,1}$ gate executed on the two-qubit, three-qubit and four-qubit NIGQCs illustrated in Fig. 6.1.152	
6.5	Gate-error quantifiers for the $\text{CZ}_{0,1}$ gate executed on the two-qubit, three-qubit and four-qubit NIGQCs illustrated in Fig. 6.1.154	

A.1	Control pulse parameters for single-qubit $R^{(x)}(\pi/2)$ gates implemented with the microwave pulse (MP) in Eq. (6.2) (two-qubit circuit Hamiltonian NIGQC model)	i
A.2	Control pulse parameters for two-qubit CZ gates implemented with the unimodal pulse (UMP) in Eq. (6.3) (two-qubit circuit Hamiltonian NIGQC model).	ii
A.3	Control pulse parameters for single-qubit $R^{(x)}(\pi/2)$ gates implemented with the microwave pulse (MP) in Eq. (6.2) (three-qubit circuit Hamiltonian NIGQC model).	ii
A.4	Control pulse parameters for two-qubit CZ gates implemented with the unimodal pulse (UMP) in Eq. (6.3) (three-qubit circuit Hamiltonian NIGQC model).	ii
A.5	Control pulse parameters for single-qubit $R^{(x)}(\pi/2)$ gates implemented with the microwave pulse (MP) in Eq. (6.2) (four-qubit circuit Hamiltonian NIGQC model).	iii
A.6	Control pulse parameters for two-qubit CZ gates implemented with the unimodal pulse (UMP) in Eq. (6.3) (four-qubit circuit Hamiltonian NIGQC model).	iii
A.7	Control pulse parameters for single-qubit $R^{(x)}(\pi/2)$ gates implemented with the microwave pulse (MP) in Eq. (6.2) (two-qubit effective Hamiltonian NIGQC model).	iii
A.8	Control pulse parameters for two-qubit CZ gates implemented with the unimodal pulse (UMP) and bimodal pulse (BMP) in Eqs. (6.3) and (6.4), respectively (two-qubit effective Hamiltonian NIGQC model).	iv
A.9	Control pulse parameters for single-qubit $R^{(x)}(\pi/2)$ gates implemented with the microwave pulse (MP) in Eq. (6.2) (three-qubit effective Hamiltonian NIGQC model).	iv
A.10	Control pulse parameters for two-qubit CZ gates implemented with the unimodal pulse (UMP) and bimodal pulse (BMP) in Eqs. (6.3) and (6.4), respectively (three-qubit effective Hamiltonian NIGQC model).	iv
A.11	Control pulse parameters for single-qubit $R^{(x)}(\pi/2)$ gates implemented with the microwave pulse (MP) in Eq. (6.2) (four-qubit effective Hamiltonian NIGQC model).	v
A.12	Control pulse parameters for two-qubit CZ gates implemented with the unimodal pulse (UMP) and bimodal pulse (BMP) in Eqs. (6.3) and (6.4), respectively (four-qubit effective Hamiltonian NIGQC model).	v
B.1	Gate-error quantifiers for $R^{(x)}(\pi/2)$ and CZ gates (two-qubit circuit Hamiltonian NIGQC model).	vi

B.2	Gate-error quantifiers for $R^{(x)}(\pi/2)$ and CZ gates (three-qubit circuit Hamiltonian NIGQC model).	vi
B.3	Gate-error quantifiers for $R^{(x)}(\pi/2)$ and CZ gates (four-qubit circuit Hamiltonian NIGQC model).	vii
B.4	Gate-error quantifiers for $R^{(x)}(\pi/2)$ and CZ (two-qubit effective Hamiltonian NIGQC model).	vii
B.5	Gate-error quantifiers for $R^{(x)}(\pi/2)$ and CZ gates (three-qubit effective Hamiltonian NIGQC model).	viii
B.6	Gate-error quantifiers for $R^{(x)}(\pi/2)$ and CZ gates (four-qubit effective Hamiltonian NIGQC model).	viii

List of acronyms

BASH	Bourne again shell
BMP	Bimodal pulse
CUDA	Compute unified device architecture
ELR	Energy level repulsion
EMF	Electromotive forces
FD	Full diagonalisation
HB	Harmonic basis
IGQC	Ideal gate-based quantum computer
JURECA	Jülich research on exascale cluster architectures
JUWELS	Jülich wizard for European leadership science
LAPACK	Linear algebra package
MKL	Math kernel library
MP	Microwave pulse
MPI	Message passing interface
NIGQC	Non-ideal gate-based quantum computer
OpenMP	Open multi-processing
PFA	Product formula algorithm
PGQC	Prototype gate-based quantum computer
QFT	Quantum Fourier transformation

SLURM	Simple Linux utility for resource management
SSH	Secure shell
TB	Transmon basis
TDSE	Time-dependent Schrödinger equation
TLS	Two-level system
UMP	Unimodal pulse
VHA	Variational hybrid algorithm

List of publications

1. H. Lagemann, D. Willsch, M. Willsch, F. Jin, H. De Raedt, and K. Michielsen, "On the fragility of gate-error metrics in simulation models of flux-tunable transmon quantum computers", submitted to Phys. Rev. A. URL: [Link](#)
2. H. Lagemann, D. Willsch, M. Willsch, F. Jin, H. De Raedt, and K. Michielsen, "Numerical analysis of effective models for flux-tunable transmon systems", Phys. Rev. A 106, 022615 (2022). URL: [Link](#)
3. F. Jin, D. Willsch, M. Willsch, H. Lagemann, K. Michielsen and H. De Raedt, "Random State Technology", J. Phys. Soc. Jpn. 90, 012001 (2021). URL: [Link](#)
4. D. Willsch, H. Lagemann, M. Willsch, F. Jin, H. De Raedt and K. Michielsen, "Benchmarking Supercomputers with the Jülich Universal Quantum Computer Simulator", NIC Symposium 2020, Publication Series of the John von Neumann Institute for Computing (NIC) NIC Series 50, 255 - 264 (2020). URL: [Link](#)

Acknowledgments

First of all, I would like to thank Kristel Michielsen and Hans De Raedt for introducing me to the field of computational physics and for providing me with the interesting problem I worked on for the last three years.

Of course, I also thank the rest of the QIP group for having all these fun group meetings and lunch breaks. Special thanks goes to Dennis Willsch, Madita Willsch and Fengping Jin. Dennis always helped me, with seemingly endless patience, solve large and small problems. Furthermore, with his dissertation he laid the foundation for many of the results presented in this thesis. Madita is probably the best proofreader I know and clearly helped me improve my writing skills. Fengping is a fun office mate, but also a scientific force to be reckoned with!

Furthermore, I would like to thank David DiVincenzo for co-supervising this dissertation and for providing valuable feedback regarding subjects I was not overly familiar with at the beginning of my time as a PhD student.

I acknowledge support from the project OpenSuperQ (820363) of the EU Quantum Flagship. Thanks goes to all project partners but special thanks goes to Christopher Warren and Jorge Fernandez-Pendas from Chalmers University of Technology as well as Stefania Lazar from ETH Zurich for answering my numerous questions about the experiments they are working on. Furthermore, I acknowledge the Gauss Centre for Supercomputing e.V. for providing computing time on the GCS Supercomputer JUWELS [Jül19] at the Jülich Supercomputing Centre (JSC) and the project Jülich UNified Infrastructure for Quantum computing (JUNIQ) that has received funding from the German Federal Ministry of Education and Research (BMBF) and the Ministry of Culture and Science of the State of North Rhine-Westphalia.

I would also like to thank Klaus Morawetz for giving me a brutal but fruitful crash course in theoretical physics as well as Navina Kleemann and Binh Nguyen for making the years as undergraduate students more enjoyable and sometimes bearable.

Last but not least, I thank family and friends which have barely seen me for the last three years. Certainly, not everyone grows up in an environment

which supports critical thought and autonomy. Furthermore, not all five year old grandsons get a proper answer to the question: how does a nuclear power plant work, paired with a small introduction to nuclear physics.

Bibliography

Books

- [DiV13] David DiVincenzo, ed. *Quantum Information Processing*. Vol. 52. Schriften des Forschungszentrums Jülich. Reihe Schlüsseltechnologien / Key Technologies. Jülich: Forschungszentrum Jülich GmbH Zentralbibliothek, Verlag, 2013, getr. Zählung. URL: [Link](#) (cit. on pp. [17](#), [48](#)).
- [AB06] Sanjeev Arora et al. *Computational Complexity: A Modern Approach*. Cambridge University Press, 2006. URL: [Link](#) (cit. on pp. [12](#), [14](#)).
- [AM76] N. W. Ashcroft et al. *Solid State Physics*. Holt-Saunders, 1976 (cit. on p. [35](#)).
- [Bal12] Constantine A. Balanis. *Advanced Engineering Electromagnetics*. Wiley, 2012 (cit. on p. [28](#)).
- [Bal91] Roger Balian. *From Microphysics to Macrophysics*. Vol. 1. Springer, 1991. URL: [Link](#) (cit. on p. [130](#)).
- [Bal07] Roger Balian. *From Microphysics to Macrophysics*. Vol. 2. Springer, 2007. URL: [Link](#) (cit. on pp. [35](#), [130](#)).
- [CGD92] C. Cohen-Tannoudji et al. *Atom-Photon Interactions: Basic Processes and Applications*. New York: Wiley, 1992. URL: [Link](#) (cit. on pp. [47](#), [55](#)).
- [Cor+09] Thomas H. Cormen et al. *Introduction to Algorithms*. 3rd ed. The MIT Press, 2009. URL: [Link](#) (cit. on pp. [12–14](#)).
- [Esq98] Pablo Esquinazi, ed. *Tunneling Systems in Amorphous and Crystalline Solids*. 1st ed. Springer, 1998. URL: [Link](#) (cit. on p. [35](#)).
- [FCA60] Robert M. Fano et al. *Electromagnetic Fields, Energy and Forces*. John Wiley & Sons, 1960. URL: [Link](#) (cit. on p. [28](#)).

- [GV96] Gene H. Golub et al. *Matrix Computations*. Third. The Johns Hopkins University Press, 1996. URL: [Link](#) (cit. on pp. 61, 62).
- [GS11] Stephen J. Gustafson et al. *Mathematical Concepts of Quantum Mechanics*. Second. Springer, 2011. URL: [Link](#) (cit. on pp. 4, 6).
- [HW10] Georg Hager et al. *Introduction to High Performance Computing for Scientists and Engineers*. CRC Press, Inc., 2010 (cit. on p. 62).
- [HK71] Kenneth Hoffman et al. *Linear algebra*. Second. Prentice-Hall, Inc., Englewood Cliffs, N.J., 1971 (cit. on p. 7).
- [LB13] Daniel A. Lidar et al., eds. *Quantum Error Correction*. Cambridge University Press, 2013. URL: [Link](#) (cit. on p. 1).
- [Max10] James Clerk Maxwell. *A Treatise on Electricity and Magnetism*. Vol. 2. Cambridge Library Collection - Physical Sciences. Cambridge University Press, 2010 (cit. on p. 27).
- [NC11] Michael A. Nielsen et al. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2011 (cit. on pp. 1, 3, 4, 6, 8, 9, 11, 15, 65, 69, 129, 131).
- [Wat18] John Watrous. *The Theory of Quantum Information*. Cambridge University Press, 2018. URL: [Link](#) (cit. on pp. 6, 131, 133).
- [Wei15] Steven Weinberg. *Lectures on Quantum Mechanics*. 2nd ed. Cambridge University Press, 2015 (cit. on pp. 4, 6, 16, 22, 34, 50, 51, 53, 73, 135, 136).
- [Wen+58] G. Wendt et al. *Elektrische Felder und Wellen / Electric Fields and Waves*. Springer, 1958. URL: [Link](#) (cit. on pp. 27–29).
- [Zan13] Andrew Zangwill. *Modern Electrodynamics*. Cambridge University Press, 2013. URL: [Link](#) (cit. on pp. 27, 28, 55).

Electronic references

- [FGG14] Edward Farhi et al. *A Quantum Approximate Optimization Algorithm*. 2014 (cit. on p. 16).
- [MM14] John M. Martinis et al. *UCSB final report for the CSQ program: Review of decoherence and materials physics for superconducting qubits*. 2014 (cit. on p. 19).
- [McD12] Kirk T. McDonald. *Voltage Drop, Potential Difference and EMF*. 2012. URL: [Link](#) (cit. on p. 28).

- [Nag19] Mahdi Naghiloo. *Introduction to Experimental Quantum Measurement with Superconducting Qubits*. 2019. URL: [Link](#) (cit. on pp. 17, 19, 21, 124).
- [SK20] Usha Srinivasan et al. *A sorting algorithm for complex eigenvalues*. 2020. URL: [Link](#) (cit. on p. 136).
- [Uhl20] Frank Uhlig. *Coalescing Eigenvalues and Crossing Eigencurves of 1-Parameter Matrix Flows*. 2020. URL: [Link](#) (cit. on p. 136).

Theses

- [Cot02] Audrey Cottet. “Implementation of a quantum bit in a superconducting circuit”. PhD thesis. Université Paris 6, 2002. URL: [Link](#) (cit. on pp. 44, 45, 73).
- [Lag19] Hannes Lagemann. “Development and implementation of a gate-based quantum computer simulator for high-dimensional Hilbert spaces”. Masterarbeit, RWTH Aachen, 2019. Masterarbeit. RWTH Aachen, 2019, p. 70. URL: [Link](#) (cit. on pp. 3, 11, 13).
- [Rot20] Marco Roth. “Analysis of scalable coupling schemes for superconducting quantum computers”. Dissertation. RWTH Aachen University, 2020. URL: [Link](#) (cit. on pp. 100, 103, 106).
- [San16] Yuval Sanders. “Characterizing Errors in Quantum Information Processors”. Dissertation. University of Waterloo, 2016. URL: [Link](#) (cit. on pp. 129, 133).
- [Ulr17] Jascha Ulrich. “Large impedances and Majorana bound states in superconducting circuits”. Dissertation. RWTH Aachen University, 2017. URL: [Link](#) (cit. on p. 30).
- [Wil16] Dennis Willsch. “Simulation of quantum computer hardware based on superconducting circuits”. Master’s thesis. RWTH Aachen University, 2016. URL: [Link](#). [Link](#) (cit. on pp. 43, 73).
- [Wil20a] Dennis Willsch. “Supercomputer simulations of transmon quantum computers”. Dissertation. RWTH Aachen University, 2020. URL: [Link](#) (cit. on pp. 46, 47, 66, 71, 129, 132, 133, 139, 160, 164).
- [Wil20b] Madita Franziska Willsch. “Study of quantum annealing by simulating the time evolution of flux qubits”. Dissertation. RWTH Aachen University, 2020. URL: [Link](#) (cit. on pp. 51, 81).

Software

- [And+] E. Anderson et al. *LAPACK:Linear Algebra PACKage*. URL: [Link](#) (cit. on pp. [61](#), [78](#), [79](#), [136](#), [137](#)).
- [Cor] Intel Corporation. *MKL:Math Kernel Library*. URL: [Link](#) (cit. on pp. [61](#), [78](#), [79](#), [136](#), [137](#)).
- [Joh] Steven G. Johnson. *The NLOpt nonlinear-optimization package*. URL: [Link](#) (cit. on pp. [79](#), [141](#)).
- [Lag20] Hannes Lagemann. *JUSQUACE:JUelich Superconducting QUantum Computer Emulator*. 2020. URL: [Link](#) (cit. on p. [166](#)).

Other references

- [Ami09] M. H. S. Amin. “Consistency of the Adiabatic Theorem”. In: *Phys. Rev. Lett.* 102 (22 June 2009), p. 220401. URL: [Link](#) (cit. on p. [50](#)).
- [And+20] Christian Kraglund Andersen et al. “Repeated quantum error detection in a surface code”. In: *Nature Physics* 16.8 (Aug. 2020), pp. 875–880. URL: [Link](#) (cit. on pp. [21](#), [124](#), [139](#)).
- [Aru+19] Frank Arute et al. “Quantum Supremacy using a Programmable Superconducting Processor”. In: *Nature* 574 (2019), pp. 505–510. URL: [Link](#) (cit. on pp. [1](#), [17](#), [139](#)).
- [Bak+22] Aneirin J. Baker et al. “Single shot i-Toffoli gate in dispersively coupled superconducting qubits”. In: *Applied Physics Letters* 120.5 (2022), p. 054002. URL: [Link](#) (cit. on pp. [87](#), [150](#)).
- [Bar+95] Adriano Barenco et al. “Elementary gates for quantum computation”. In: *Phys. Rev. A* 52 (5 Nov. 1995), pp. 3457–3467. URL: [Link](#) (cit. on p. [11](#)).
- [Ben+20] Andreas Bengtsson et al. “Improved Success Probability with Greater Circuit Depth for the Quantum Approximate Optimization Algorithm”. In: *Phys. Rev. Applied* 14 (3 Sept. 2020), p. 034010. URL: [Link](#) (cit. on pp. [17](#), [21](#)).
- [Ber+22] Christoph Berke et al. “Transmon platform for quantum computing challenged by chaotic fluctuations”. In: *Nature Communications* 13.1 (May 2022), p. 2495. URL: [Link](#) (cit. on p. [96](#)).

- [Bla+21] Alexandre Blais et al. “Circuit quantum electrodynamics”. In: *Rev. Mod. Phys.* 93 (2 May 2021), p. 025005. URL: [Link](#) (cit. on pp. 22, 47, 50, 56, 83, 111, 129, 135, 136, 151).
- [Bur+22] Daniel Burgarth et al. “One bound to rule them all: from Adiabatic to Zeno”. In: *Quantum* 6 (June 2022), p. 737. URL: [Link](#) (cit. on p. 53).
- [BKD04] Guido Burkard et al. “Multilevel quantum description of decoherence in superconducting qubits”. In: *Phys. Rev. B* 69 (6 Feb. 2004), p. 064503. URL: [Link](#) (cit. on pp. 29, 31).
- [Bur+19] Jonathan J. Burnett et al. “Decoherence benchmarking of superconducting qubits”. In: *npj Quantum Information* 5.1 (June 2019), p. 54. URL: [Link](#) (cit. on pp. 19, 22, 35).
- [DN06] Christopher M. Dawson et al. “The Solovay-Kitaev Algorithm”. In: *Quantum Info. Comput.* 6 (2006), pp. 81–95. URL: [Link](#) (cit. on p. 11).
- [DD83] Hans De Raedt et al. “Applications of the generalized Trotter formula”. In: *Phys. Rev. A* 28 (6 Dec. 1983), pp. 3575–3580. URL: [Link](#) (cit. on p. 62).
- [Dev97] M. H. Devoret. “Quantum fluctuations in electrical circuits”. In: *Fluctuations quantiques: Les Houches, session LXIII, 27 Juin - 28 Juillet 1995 = Quantum fluctuations*. Ed. by Serge Reynaud et al. Elsevier, 1997, pp. 351–386. URL: [Link](#) (cit. on p. 29).
- [DiC+10] L. DiCarlo et al. “Preparation and measurement of three-qubit entanglement in a superconducting circuit”. In: *Nature* 467.7315 (Sept. 2010), pp. 574–578. URL: [Link](#) (cit. on p. 136).
- [DiC+19] L. DiCarlo et al. “Demonstration of two-qubit algorithms with a superconducting quantum processor”. In: *Nature* 460 (7252 July 2019), p. 120502. URL: [Link](#) (cit. on pp. 96, 136).
- [Did+18] Nicolas Didier et al. “Analytical modeling of parametrically modulated transmon qubits”. In: *Phys. Rev. A* 97 (2 Feb. 2018), p. 022330. URL: [Link](#) (cit. on pp. 50, 94, 95, 165).
- [Dir25] P.A.M. Dirac. “The fundamental equations of quantum mechanics”. In: *Proceedings of the Royal Society A* 109 (752 Dec. 1925), pp. 642–653. URL: [Link](#) (cit. on pp. 22, 32).
- [Fen49] W. Fenchel. “On Conjugate Convex Functions”. In: *Canadian Journal of Mathematics* 1.1 (1949), pp. 73–77 (cit. on p. 32).

- [Fox+20] B. Foxen et al. “Demonstrating a Continuous Set of Two-Qubit Gates for Near-Term Quantum Algorithms”. In: *Phys. Rev. Lett.* 125 (12 Sept. 2020), p. 120504. URL: [Link](#) (cit. on pp. 96, 136, 150, 154, 165).
- [Fra61] J. G. F. Francis. “The QR Transformation A Unitary Analogue to the LR Transformation—Part 1”. In: *The Computer Journal* 4.3 (Jan. 1961), pp. 265–271. URL: [Link](#) (cit. on p. 61).
- [Fra62] J. G. F. Francis. “The QR Transformation—Part 2”. In: *The Computer Journal* 4.4 (Jan. 1962), pp. 332–345. URL: [Link](#) (cit. on p. 61).
- [Gan+20] M. Ganzhorn et al. “Benchmarking the noise sensitivity of different parametric two-qubit gates in a single superconducting quantum computing platform”. In: *Phys. Rev. Research* 2 (3 Sept. 2020), p. 033447. URL: [Link](#) (cit. on pp. 44, 52, 83, 84, 87, 94, 96, 100, 103, 106, 165).
- [GKS76] Vittorio Gorini et al. “Completely positive dynamical semigroups of N-level systems”. In: *Journal of Mathematical Physics* 17.5 (1976), pp. 821–825. URL: [Link](#) (cit. on p. 23).
- [Gu+21] Xiu Gu et al. “Fast Multiqubit Gates through Simultaneous Two-Qubit Gates”. In: *PRX Quantum* 2 (4 Dec. 2021), p. 040348. URL: [Link](#) (cit. on pp. 1, 47, 50, 52, 83, 87, 94, 123, 146, 150, 158, 165).
- [Hei+18] Johannes Heinsoo et al. “Rapid High-fidelity Multiplexed Read-out of Superconducting Qubits”. In: *Phys. Rev. Applied* 10 (3 Sept. 2018), p. 034040. URL: [Link](#) (cit. on p. 21).
- [Hun27] F. Hund. “Zur Deutung der Molekelspektren. I”. In: *Zeitschrift für Physik* 40.10 (Oct. 1927), pp. 742–764. URL: [Link](#) (cit. on p. 136).
- [HD90] J. Huyghebaert et al. “Product formula methods for time-dependent Schrödinger problems”. In: *J. Phys. A: Math. Gen.* 23.24 (Dec. 1990), pp. 5777–5793. URL: [Link](#) (cit. on p. 2).
- [Jin+21] Fengping Jin et al. “Random State Technology”. In: *Journal of the Physical Society of Japan* 90.1 (2021), p. 012001. URL: [Link](#) (cit. on p. 132).
- [JKP09] Nathaniel Johnston et al. “Computing Stabilized Norms for Quantum Operations via the Theory of Completely Bounded Maps”. In: *Quantum Information and Computation* 9 (Dec. 2009). URL: [Link](#) (cit. on p. 133).

- [Jül18] Jülich Supercomputing Centre. “JURECA: Modular supercomputer at Jülich Supercomputing Centre”. In: *Journal of large-scale research facilities* 4 (2018), A132. URL: [Link](#) (cit. on pp. 77, 78).
- [Jül19] Jülich Supercomputing Centre. “JUWELS: Modular Tier-0/1 Supercomputer at the Jülich Supercomputing Centre”. In: *Journal of large-scale research facilities* 5 (2019), A135. URL: [Link](#) (cit. on pp. 77, 78, 134, 135, xviii).
- [Koc+07] Jens Koch et al. “Charge-insensitive qubit design derived from the Cooper pair box”. In: *Phys. Rev. A* 76 (4 Oct. 2007), p. 042319. URL: [Link](#) (cit. on pp. 36, 40, 45, 47, 58, 88).
- [Kri+19] S. Krinner et al. “Engineering cryogenic setups for 100-qubit scale superconducting circuit systems”. In: *EPJ Quantum Technology* 6.1 (May 2019), p. 2. URL: [Link](#) (cit. on pp. 19, 20).
- [Kri+20] S. Krinner et al. “Benchmarking Coherent Errors in Controlled-Phase Gates due to Spectator Qubits”. In: *Phys. Rev. Applied* 14 (2 Aug. 2020), p. 024042. URL: [Link](#) (cit. on pp. 17, 83, 111, 124).
- [Kri+22] Sebastian Krinner et al. “Realizing repeated quantum error correction in a distance-three surface code”. In: *Nature* 605.7911 (May 2022), pp. 669–674. URL: [Link](#) (cit. on pp. 21, 139).
- [Lac+20] Nathan Lacroix et al. “Improving the Performance of Deep Quantum Optimization Algorithms with Continuous Gate Sets”. In: *PRX Quantum* 1 (2 Oct. 2020), p. 110304. URL: [Link](#) (cit. on pp. 1, 44, 83, 84, 124).
- [Lin76] G. Lindblad. “On the generators of quantum dynamical semi-groups”. In: *Communications in Mathematical Physics* 48.2 (June 1976), pp. 119–130. URL: [Link](#) (cit. on p. 23).
- [McE+22] Matt McEwen et al. “Resolving catastrophic error bursts from cosmic rays in large arrays of superconducting qubits”. In: *Nature Physics* 18.1 (Jan. 2022), pp. 107–111. URL: [Link](#) (cit. on p. 19).
- [McK+16] David C. McKay et al. “Universal Gate for Fixed-Frequency Qubits via a Tunable Bus”. In: *Phys. Rev. Applied* 6 (6 Dec. 2016), p. 064007. URL: [Link](#) (cit. on pp. 47, 50, 52, 83, 87, 94, 96, 100, 103, 106, 123, 146, 150, 158, 165).

- [McK+17] David C. McKay et al. “Efficient Z gates for quantum computing”. In: *Phys. Rev. A* 96 (2 Aug. 2017), p. 022330. URL: [Link](#) (cit. on pp. [128](#), [165](#)).
- [Mic+17] Kristel Michielsen et al. “Benchmarking gate-based quantum computers”. In: *Computer Physics Communications* 220 (2017), pp. 44–55. URL: [Link](#) (cit. on p. [161](#)).
- [Mot+09] F. Motzoi et al. “Simple Pulses for Elimination of Leakage in Weakly Nonlinear Qubits”. In: *Phys. Rev. Lett.* 103 (11 Sept. 2009), p. 110501. URL: [Link](#) (cit. on pp. [128](#), [165](#)).
- [MCL19] Clemens Müller et al. “Towards understanding two-level-systems in amorphous solids: insights from quantum circuits”. In: *Reports on Progress in Physics* 82.12 (Oct. 2019), p. 124501. URL: [Link](#) (cit. on p. [35](#)).
- [Mur21] Conal E. Murray. “Material matters in superconducting qubits”. In: *Materials Science and Engineering: R: Reports* 146 (2021), p. 100646. URL: [Link](#) (cit. on p. [19](#)).
- [Pre18] John Preskill. “Quantum Computing in the NISQ era and beyond”. In: *Quantum* 2 (2018), p. 79. URL: [Link](#) (cit. on p. [1](#)).
- [Rae87] Hans De Raedt. “Product formula algorithms for solving the time dependent Schroedinger equation”. In: *Computer Physics Reports* 7 (1987), pp. 1–72. URL: [Link](#) (cit. on pp. [2](#), [62](#), [64](#)).
- [RM06] Hans De Raedt et al. “Computational Methods for Simulating Quantum Computers”. In: *Handbook of Theoretical and Computational Nanotechnology*. Ed. by Michael Rieth et al. American Scientific Publishers, 2006 (cit. on pp. [62](#), [63](#)).
- [Rae+19] Hans De Raedt et al. “Massively parallel quantum computer simulator, eleven years later”. In: *Computer Physics Communications* 237 (2019), pp. 47–61. URL: [Link](#) (cit. on p. [12](#)).
- [RD22] R.-P. Riwar et al. “Circuit quantization with time-dependent magnetic fields for realistic geometries”. In: *npj Quantum Information* 8.1 (Mar. 2022), p. 36. URL: [Link](#) (cit. on pp. [31](#), [38](#)).
- [Rol+19] M. A. Rol et al. “Fast, High-Fidelity Conditional-Phase Gate Exploiting Leakage Interference in Weakly Anharmonic Superconducting Qubits”. In: *Phys. Rev. Lett.* 123 (12 Sept. 2019), p. 120502. URL: [Link](#) (cit. on pp. [1](#), [19](#), [50](#), [83](#), [111](#), [127](#), [136](#), [146](#), [150](#), [156](#), [158](#), [165](#)).

- [Rot+17] Marco Roth et al. “Analysis of a parametrically driven exchange-type gate and a two-photon excitation gate between superconducting qubits”. In: *Phys. Rev. A* 96 (6 Dec. 2017), p. 062323. URL: [Link](#) (cit. on pp. 1, 17, 47, 50, 52, 83, 87, 94, 96, 100, 103, 106, 123, 146, 150, 158, 165).
- [RMC22] Thomas Roth et al. “The Transmon Qubit for Electromagnetics Engineers: An Introduction.” In: *IEEE Antennas and Propagation Magazine* (2022), pp. 2–14 (cit. on pp. 149, 159).
- [SWS15] Yuval R Sanders et al. “Bounding quantum gate error rate based on reported average fidelity”. In: *New Journal of Physics* 18.1 (Dec. 2015), p. 012002. URL: [Link](#) (cit. on p. 129).
- [Sho94] Peter W. Shor. “Algorithms for quantum computation: discrete logarithms and factoring”. In: *Proceedings 35th Annual Symposium on Foundations of Computer Science*. Nov. 1994, pp. 124–134 (cit. on p. 15).
- [Sho97] Peter W. Shor. “Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer”. In: *SIAM J. Comput.* 26.5 (1997), pp. 1484–1509. URL: [Link](#) (cit. on p. 15).
- [Suz85] Masuo Suzuki. “Decomposition formulas of exponential operators and Lie exponentials with some applications to quantum mechanics and statistical physics”. In: *Journal of Mathematical Physics* 26.4 (1985), pp. 601–612. URL: [Link](#) (cit. on p. 63).
- [SMK77] Masuo Suzuki et al. “Monte Carlo Simulation of Quantum Spin Systems. I”. In: *Progress of Theoretical Physics* 58.5 (Nov. 1977), pp. 1377–1387. URL: [Link](#) (cit. on p. 63).
- [Tur37] Alan M. Turing. “On Computable Numbers, with an Application to the Entscheidungsproblem”. In: *Proceedings of the London Mathematical Society* 2.42 (1937), pp. 230–265. URL: [Link](#) (cit. on p. 1).
- [UH16] Jascha Ulrich et al. “Dual approach to circuit quantization using loop charges”. In: *Phys. Rev. B* 94 (9 Sept. 2016), p. 094505. URL: [Link](#) (cit. on pp. 29, 30).
- [VD17] Uri Vool et al. “Introduction to Quantum Electromagnetic Circuits”. In: *International Journal of Circuit Theory and Applications* 45 (July 2017). URL: [Link](#) (cit. on p. 29).

- [Wer+21] M. Werninghaus et al. “Leakage reduction in fast superconducting qubit gates via optimal control”. In: *npj Quantum Information* 7.1 (Jan. 2021), p. 14. URL: [Link](#) (cit. on p. 19).
- [Wil+17] Dennis Willsch et al. “Gate-error analysis in simulations of quantum computers with transmon qubits”. In: *Physical Review A* 96 (2017), pp. 062302-1–062302-11. URL: [Link](#) (cit. on pp. 46, 122, 150, 160, 161, 164).
- [Wil+20a] Dennis Willsch et al. “Benchmarking Supercomputers with the Jülich Universal Quantum Computer Simulator”. In: *NIC Symposium 2020*. Vol. 50. Publication Series of the John von Neumann Institute for Computing (NIC) NIC Series. Jülich: Forschungszentrum Jülich GmbH Zentralbibliothek, Verlag, Feb. 2020, pp. 255–264. URL: [Link](#) (cit. on p. 11).
- [Wil+20b] Madita Willsch et al. “Real-time simulation of flux qubits used for quantum annealing”. In: *Phys. Rev. A* 101 (1 Jan. 2020), p. 012327. URL: [Link](#) (cit. on pp. 35, 56).
- [Wit+21] Nicolas Wittler et al. “Integrated Tool Set for Control, Calibration, and Characterization of Quantum Devices Applied to Superconducting Qubits”. In: *Phys. Rev. Applied* 15 (3 Mar. 2021), p. 034080. URL: [Link](#) (cit. on pp. 123, 146, 158).
- [Yan+18] Fei Yan et al. “Tunable Coupling Scheme for Implementing High-Fidelity Two-Qubit Gates”. In: *Phys. Rev. Applied* 10 (5 Nov. 2018), p. 054062. URL: [Link](#) (cit. on pp. 50, 52, 87, 123, 136, 146, 151, 158, 165).
- [YSK19] Xinyuan You et al. “Circuit quantization in the presence of time-dependent external flux”. In: *Phys. Rev. B* 99 (17 May 2019), p. 174512. URL: [Link](#) (cit. on pp. 31, 38, 40, 69, 73).
- [YD84] Bernard Yurke et al. “Quantum network theory”. In: *Phys. Rev. A* 29 (3 Mar. 1984), pp. 1419–1437. URL: [Link](#) (cit. on p. 29).

Eidesstattliche Erklärung

Ich, Hannes Alfred Lagemann, erkläre hiermit, dass diese Dissertation und die darin dargelegten Inhalte die eigenen sind und selbstständig, als Ergebnis der eigenen originären Forschung, generiert wurden.

Hiermit erkläre ich an Eides statt

1. Diese Arbeit wurde vollständig oder größtenteils in der Phase als Doktorand dieser Fakultät und Universität angefertigt;
2. Sofern irgendein Bestandteil dieser Dissertation zuvor für einen akademischen Abschluss oder eine andere Qualifikation an dieser oder einer anderen Institution verwendet wurde, wurde dies klar angezeigt;
3. Wenn immer andere eigene- oder Veröffentlichungen Dritter herangezogen wurden, wurden diese klar benannt;
4. Wenn aus anderen eigenen- oder Veröffentlichungen Dritter zitiert wurde, wurde stets die Quelle hierfür angegeben. Diese Dissertation ist vollständig meine eigene Arbeit, mit der Ausnahme solcher Zitate;
5. Alle wesentlichen Quellen von Unterstützung wurden benannt;
6. Wenn immer ein Teil dieser Dissertation auf der Zusammenarbeit mit anderen basiert, wurde von mir klar gekennzeichnet, was von anderen und was von mir selbst erarbeitet wurde;
7. Teile dieser Arbeit wurden zuvor veröffentlicht und zwar in: H. Lagemann, D. Willsch, M. Willsch, F. Jin, H. De Raedt, and K. Michielsen, "Numerical analysis of effective models for flux-tunable transmon systems", Phys. Rev. A 106, 022615 (2022).

Datum und Unterschrift

xxx

Band / Volume 43

Algorithms for massively parallel generic *hp*-adaptive finite element methods

M. Fehling (2020), vii, 78 pp

ISBN: 978-3-95806-486-7

URN: urn:nbn:de:0001-2020071402

Band / Volume 44

The method of fundamental solutions for computing interior transmission eigenvalues

L. Pieronek (2020), 115 pp

ISBN: 978-3-95806-504-8

Band / Volume 45

Supercomputer simulations of transmon quantum computers

D. Willsch (2020), IX, 237 pp

ISBN: 978-3-95806-505-5

Band / Volume 46

The Influence of Individual Characteristics on Crowd Dynamics

P. Georg (2021), xiv, 212 pp

ISBN: 978-3-95806-561-1

Band / Volume 47

Structural plasticity as a connectivity generation and optimization algorithm in neural networks

S. Diaz Pier (2021), 167 pp

ISBN: 978-3-95806-577-2

Band / Volume 48

Porting applications to a Modular Supercomputer

Experiences from the DEEP-EST project

A. Kreuzer, E. Suarez, N. Eicker, Th. Lippert (Eds.) (2021), 209 pp

ISBN: 978-3-95806-590-1

Band / Volume 49

Operational Navigation of Agents and Self-organization Phenomena in Velocity-based Models for Pedestrian Dynamics

Q. Xu (2022), xii, 112 pp

ISBN: 978-3-95806-620-5

Band / Volume 50

Utilizing Inertial Sensors as an Extension of a Camera Tracking System for Gathering Movement Data in Dense Crowds

J. Schumann (2022), xii, 155 pp

ISBN: 978-3-95806-624-3

Band / Volume 51

**Final report of the DeepRain project
Abschlußbericht des DeepRain Projektes**

(2022), ca. 70 pp

ISBN: 978-3-95806-675-5

Band / Volume 52

JSC Guest Student Programme Proceedings 2021

I. Kabadshow (Ed.) (2023), ii, 82 pp

ISBN: 978-3-95806-684-7

Band / Volume 53

Applications of variational methods for quantum computers

M. S. Jattana (2023), vii, 160 pp

ISBN: 978-3-95806-700-4

Band / Volume 54

**Crowd Management at Train Stations in Case of
Large-Scale Emergency Events**

A. L. Braun (2023), vii, 120 pp

ISBN: 978-3-95806-706-6

Band / Volume 55

**Gradient-Free Optimization of Artificial and Biological Networks
using Learning to Learn**

A. Yeğenoğlu (2023), II, 136 pp

ISBN: 978-3-95806-719-6

Band / Volume 56

**Real-time simulations of transmon systems with
time-dependent Hamiltonian models**

H. A. Lagemann (2023), iii, 166, XXX pp

ISBN: 978-3-95806-720-2

IAS Series
Band / Volume 56
ISBN 978-3-95806-720-2