ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Velocity Control of Trapped Ions for Transport Quantum Logic Gates

Master's Thesis of

Robin Oswald

Supervisors

Ludwig de Clercq
Trapped Ion Quantum Information Group

Amin Rezaeizadeh
Automatic Control Laboratory

Supervising Professors

Prof. Jonathan Home
Trapped Ion Quantum Information Group

Prof. Roy Smith
Automatic Control Laboratory

November 2015

Abstract

While quantum information processing with trapped ions has already been demonstrated at high fidelities with a few ions, scaling up to larger numbers is a formidable challenge. For doing so, the quantum CCD architecture proposes interconnecting many small ion traps and transporting ions between processing and storage regions [Wineland 98, Kielpinski 02]. More recently, transport quantum logic gates have been proposed to reduce the demands on the optical control [D. Leibfried 07]. Transport is therefore a key technique associated with scaling.

In this work, we implement adiabatic ion transport, paying special attention to keeping the transport velocity constant over time. We present a method based on quadratic programming to calculate feedforward control inputs for transport, taking common hardware constraints into account. We also identify and eliminate a flaw in our electronics used to implement transport. We then evaluate the use of iterative learning control to further improve the time-varying feedforward control inputs driving transport. With it, we have reduced velocity fluctuations from 0.1 m/s to 0.01 m/s at an overall transport speed of 2.8 m/s, paving the way for future experiments.

We also present a new method to infer the time-dependent Hamiltonian of a two level system from straightforward experimental measurements. We use it to study the Hamiltonian arising during a transport gate with $^{40}Ca^{+}$, inferring both the ions velocity and the laser beam intensity it experiences over time.

Acknowledgements

Working on this project has been a pleasure. I would like to thank everyone from the Trapped Ion Quantum Information group for helping me to overcome various obstacles with useful advice, practical help and encouragement. I also strongly appreciate the many lighter moments, be it the SOLA, the Boltzmann cup or just a game of kicker.

In particular, I would like to thank:

Prof. Jonathan Home for creating a fruitful environment for doing research. In particular, his involvement in all projects is remarkable, as is his easy approachability.

Ludwig de Clercq for supervising me from the physics side and for his great patience in explaining all sorts of things to me. I have very much enjoyed collaborating and alternately figuring things out with you.

Prof. Roy Smith and Amin Rezaeizadeh for joining this interdisciplinary effort and supervising me from the control engineering side. Being able to discuss various control problems was very helpful.

Prof. Manfred Morari for his excellent lecture on model predictive control that proved very useful, and for his enthusiastic encouragement early on.

Matteo Marinelli for repeatedly helping me perform the experiments until late at night, and for swiftly opening many apéros.

David Nadlinger, also for helping me with the experiments, and for countless improvements to the control software.

Vlad Negnevitsky for his help in the lab, even on weekends. And for helping me with electronics, as well as constantly supporting everybody else in the group.

Lastly, I would again like to thank everybody from the TIQI group - past and present - for building all the various pieces of the experimental apparatus. I was very fortunate to start with a working setup. Having also been around at the time we moved to the new and empty labs, I can only imagine how much work it took to get to the current state.

Contents

Chapter 1

Introduction

The past few decades have seen many advances in the experimental control of various physical systems at the quantum level, sparking the hope of building devices using the laws of quantum mechanics to their advantage. Among them, quantum computers are arguably the most famous. They promise to efficiently simulate quantum systems, which is known to be intractable with classical computers [Feynman 82].

Trapped ions are a promising system currently being investigated for building a quantum computer. All the required individual building blocks have already been demonstrated [Home 09]. Now, the main challenge lies in scaling. To that end, the quantum CCD architecture proposes the use of a large number of connected ion traps [Wineland 98, Kielpinski 02]. It features separate regions for storage and processing of information encoded in the electronic states of trapped ions, requiring transport between regions. Later, transport quantum logic gates were proposed to reduce the demands on the optical control [D. Leibfried 07]. Instead of performing logic gates by pulsing a laser beam on a static ion, the ion is transported through a static laser beam. Due to a Doppler shift, the laser frequency experienced by the ion depends on its velocity, as does the time it spends in the beam. Control of the velocity therefore becomes crucial, as we realized upon implementing transport gates [de Clercq 15c].

In this work, we implement adiabatic transport of single $^{40}Ca^{+}$ ions in practice, paying special attention to keeping the velocity constant over time. We transport only along a single axial direction, neglecting micromotion. For transport across more complicated geometries, the reader is referred to [Blakestad 09, Wright 13]. Also, we do not investigate motional heating during transport, as is done e.g. in [Rowe 02, Blakestad 10, Bowler 15].

The remainder of this thesis is structured as follows:

Chapter 2 introduces the experimental apparatus and sketches the physics behind ion trapping as well as the laser-ion interactions relevant for this work. It also briefly outlines the laws of quantum mechanics and discusses their implications for the control of quantum systems.

Next, **Chapters 3, 4, 5 and 6** form the core of this investigation. Their logical progression in analogy to a feedback control loop is visualized in Figure 6.1.

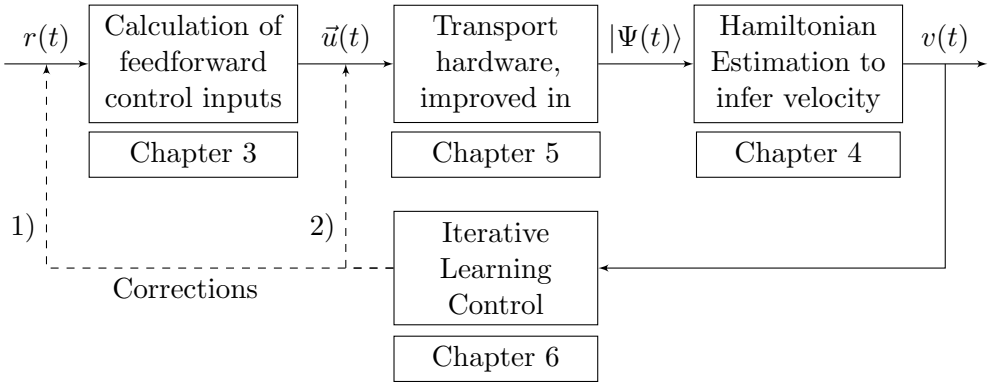


Figure 1.1: Outline of the core of this thesis.

Chapter 3 presents a method based on quadratic programming in order to calculate the feedforward control inputs used to implement adiabatic transport, taking into account various practical constraints of the control electronics.

Chapter 4 introduces a method to infer the velocity $v(t)$ of an ion during transport from the time-evolution of its internal electronic quantum state $|\Psi(t)\rangle$. It presents a simple experimental method and associated analysis techniques to infer the time-dependent Hamiltonian governing the dynamics of the internal quantum state. The velocity can then be read off from the inferred Hamiltonian.

Chapter 5 identifies a subtle flaw in the control electronics and documents modifications made to the hardware in order to resolve it.

Chapter 6 introduces iterative learning control (ILC) to further reduce velocity fluctuations. This is done by iteratively adding small corrections to the feedforward control inputs after measuring the velocity profile $v(t)$. In a first approach, ILC is used to make small corrections to the reference $r(t)$. In a second approach, the feedforward control inputs $\vec{u}(t)$ are modified directly. ILC thus closes the loop.

Chapter 7 presents the main conclusions and outlines directions for future investigations.

Chapter 2

Background Material

This chapter gives an overview of the physics required to understand this thesis, introduces the experimental apparatus, and outlines some of the challenges regarding control. Rather than giving a comprehensive introduction to the overall field, the aim is to present a brief overview of the knowledge required to make this thesis understandable in a self-contained way. Topics that are only relevant for specific chapters will be presented therein and are omitted here. For more details, the reader can consult the many references.

2.1 Ion trapping

In this work, we use a linear Paul trap with segmented dc electrodes to trap a single Calcium ion, specifically $^{40}Ca^{+}$. Figure 2.1 shows a drawing of the trap. It features a linear axis which is usually referred to as the trap axis. Because transport is done along this axis, we also call it the transport axis and often denote it as the z-axis. The plane perpendicular to it is spanned by the x-axis and y-axis, which are commonly also called the radials.
Using a combination of static ("dc") and time-varying ("rf") electric fields one can confine ions along all three axes. The rf electrodes provide confinement along the radials with oscillating electric fields. The segmented dc electrodes are used to form a confining potential well along the trap axis. We can write the overall potential $\Phi(\mathbf{r}, t)$ as a combination of the potentials due to the dc and rf electrodes

$$\Phi(\mathbf{r}, t) = \Phi_{\mathrm{dc}}(\mathbf{r}) + \Phi_{\mathrm{rf}}(\mathbf{r}) \cos(\Omega_{\mathrm{rf}} t). \tag{2.1}$$

The equations of motion for a particle of mass m and charge q under the influence of such a potential are then given by Newton's second law

$$\mathbf{F} = m\ddot{\mathbf{r}}(t) = -q\vec{\nabla}_{\mathbf{r}}\Phi(\mathbf{r}, t). \tag{2.2}$$

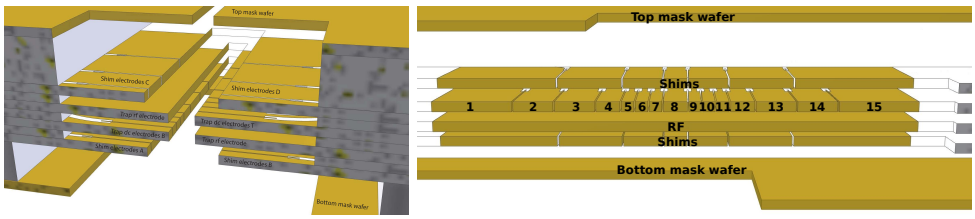


Figure 2.1: Schematic of the trap, cut transverse to the trap axis on the left side, and along the trap axis on the right side. The figures show the arrangement of the segmented shim and dc electrodes, as well as the rf electrodes. Adapted from [Kienzler 15] and [de Clercq 15c].

Solving equation (2.2) for a general setting is quite complicated. In our operating regime, the solutions assume the form [Sasura 02]

$$x(t) \propto \left[1 + \frac{q_x}{2}\cos(\Omega_{\mathrm{rf}} t)\right] \cos(\omega_x t + \varphi_x) \tag{2.3}$$

$$y(t) \propto \left[1 + \frac{q_y}{2}\cos(\Omega_{\mathrm{rf}} t)\right] \cos(\omega_y t + \varphi_y) \tag{2.4}$$

$$z(t) \propto \cos(\omega_z t + \varphi_z) \tag{2.5}$$

where ω_i stands for the trap frequencies and q_x and q_y depend on the exact operating conditions. The motion at angular frequency Ω_{rf} is called *micromotion*. Experimentally, we can minimize its effect with the help of another set of electrodes, usually referred to as the shim electrodes. In this work, we thus neglect the effect of micromotion and assume the ion to be well confined in the radial direction. We will thus mainly focus on the potential along the trap axis. Finally, we can simplify the overall potential using the pseudo-potential approximation by averaging the forces over time. We then obtain

$$U(\mathbf{r}) = \frac{1}{2}\frac{m w_z^2 z^2}{q} + \frac{1}{2}\frac{m w_r^2 (x^2 + y^2)}{q}. \tag{2.6}$$

We thus approximately have three decoupled harmonic oscillators. Under typical operating conditions, we have radial angular frequencies ω_x and ω_y between $2\pi \times 3$ MHz and $2\pi \times 4$ MHz, and a rf frequency Ω_{rf} around $2\pi \times 115$ MHz.

The ion trap is placed in an ultra-high vacuum chamber in order to minimize the loss of ions due to collisions with the background gas. Loading an ion is accomplished by heating a slab of Calcium and ionizing the neutral, evaporated Calcium atoms by a two-photon photoionization process using lasers with wavelengths of 375 nm and 423 nm. Afterwards, the ion is cooled close to the motional ground state using laser-cooling techniques.

2.2 Laser-ion interactions

2.2.1 Calcium as a qubit

A single trapped ion can approximately be described as a two level system together with three decoupled quantum mechanical harmonic oscillators corresponding to the motional degrees of freedom. In this work, we focus on the pseudo-spin system encoded in the electronic states of $^{40}Ca^{+}$, ignoring the motional degrees of freedom. We thus avoid driving any transitions between different motional states, the so-called sidebands (blue or red) and drive only carrier transitions.

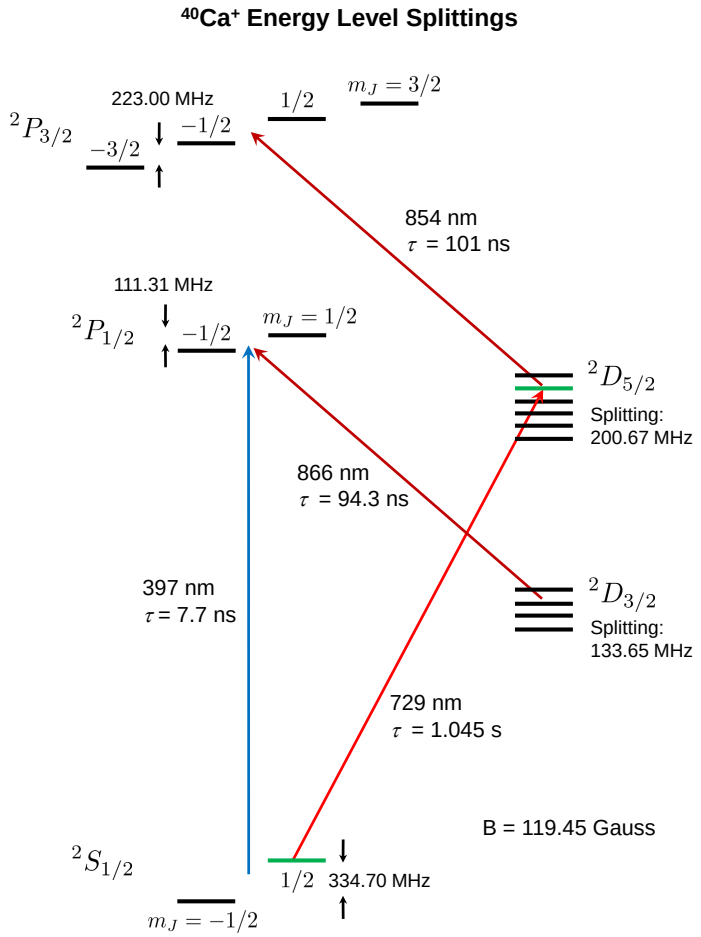


Figure 2.2: Level scheme of $^{40}Ca^{+}$ in a magnetic field of 119.45 G, together with various transitions and their natural lifetimes. Adapted from [Lo 15].

The level scheme of $^{40}Ca^{+}$ under the influence of a magnetic field of 119.45 G

is shown in Figure 2.2. We use the quadrupole transition between the levels $|S_{1/2}, m_J = 1/2\rangle$ and $|D_{5/2}, m_J = 3/2\rangle$ as an optical qubit, defining our two level system. The transition has a long natural lifetime of approximately one second and can be addressed using a laser with a wavelength of 729 nm. Additional lasers with wavelengths of 397 nm, 854 nm and 866 nm are used for state initialization to $|S_{1/2}, m_J = 1/2\rangle$ via optical pumping and state readout using fluorescence detection of the dipole transition corresponding to 397 nm between the $P_{1/2}$ and $S_{1/2}$ states. The lasers are also used to implement various laser cooling techniques, such as Doppler cooling, sideband cooling, and electromagnetically induced transparency cooling. Details can be found in [Kienzler 15, Lo 15].

2.2.2 Bloch sphere

The state of a two level system is described by its wave function $|\Psi(t)\rangle$. It consists of a complex superposition of the two energy eigenstates $|0\rangle$ and $|1\rangle$. We can thus write

$$|\Psi(t)\rangle = \alpha(t)\,|0\rangle + \beta(t)\,|1\rangle \tag{2.7}$$

where $\alpha(t), \beta(t) \in \mathbb{C}$ and $|\alpha(t)|^2 + |\beta(t)|^2 = 1$. Multiplying the wave function $|\Psi(t)\rangle$ by an overall phase has no physical significance, as it leaves all possible measurements unchanged. Therefore, one often fixes the overall phase, and describes the state of a two level system using just two parameters $\theta(t)$ and $\phi(t)$. This results in

$$|\Psi(t)\rangle = \cos\left(\frac{\theta(t)}{2}\right)|0\rangle + e^{i\phi(t)}\sin\left(\frac{\theta(t)}{2}\right)|1\rangle \tag{2.8}$$
$$\text{with } 0 \leq \theta(t) \leq \pi \text{ and } 0 \leq \phi(t) < 2\pi$$

The two parameters $\theta(t)$ and $\phi(t)$ can be interpreted as spherical coordinates, leading to the picture of the so-called Bloch sphere, illustrated in Figure 2.3. The state $|\Psi(t)\rangle$ can thus be seen as a vector on the unit sphere.

2.2.3 Rabi oscillations

A two level system interacting with a strong classical electric field (e.g. from a laser), oscillating at a frequency ω close to the natural frequency $\omega_0 = \Delta E/\hbar$ of the transition between the two energy levels shows oscillatory behavior. Specifically, the populations in the states vary sinusoidally over time.

Resonant interaction

Let us first analyze the simplest case when the driving electric field has a constant amplitude and is exactly on resonance with the transition, e.g. $\omega = \omega_0$. We can then write the Hamiltonian as

$$\hat{H}_{\text{res}} = -\frac{\hbar}{2}\Omega\hat{\sigma}_x \tag{2.9}$$

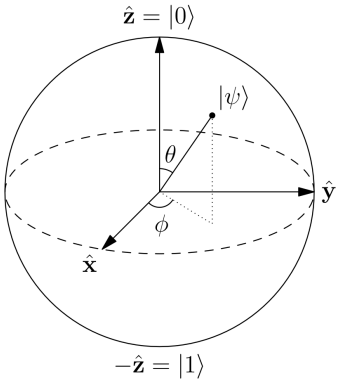


Figure 2.3: Visualization of the Bloch sphere. Note that the state $|0\rangle$ is at the top. Adapted from Wikipedia.

where $\hat{\sigma}_x$ is a Pauli matrix, and Ω is called the Rabi frequency and is proportional to the amplitude of the electric field. Because the Hamiltonian is constant over time, we can easily solve for the time evolution of the system using the unitary propagator U:

$$|\Psi(t)\rangle = U(t;0)\,|\Psi(0)\rangle = \exp(-iHt/\hbar)\,|\Psi(0)\rangle \tag{2.10}$$

$$= \begin{bmatrix} \cos(\frac{\Omega t}{2}) & i\sin(\frac{\Omega t}{2}) \\ i\sin(\frac{\Omega t}{2}) & \cos(\frac{\Omega t}{2}) \end{bmatrix} |\Psi(0)\rangle \tag{2.11}$$

In particular, if we initially start in the state $|0\rangle$, we get the following wave function

$$|\Psi(t)\rangle = \cos\left(\frac{\Omega t}{2}\right)|0\rangle + i\sin\left(\frac{\Omega t}{2}\right)|1\rangle\,. \tag{2.12}$$

Comparing this result to equation 2.8, we see that our state vector simply rotates around the x-axis on the Bloch sphere, as illustrated in blue in Figure 2.4.

Detuned interaction

Including a detuning $\delta = \omega - \omega_0$ between the frequency of the electric field and the transition frequency makes the analysis slightly more complicated. The Hamiltonian now has an additional term involving $\hat{\sigma}_z$, e.g.

$$\hat{H}_{\text{det}} = \frac{\hbar}{2}(-\Omega\hat{\sigma}_x + \delta\hat{\sigma}_z) \tag{2.13}$$

If we again start in state $|0\rangle$, solving Schrödinger's equation leads to

$$|\Psi(t)\rangle = \left(\cos\left(\frac{\Omega_{\text{d}} t}{2}\right) - i\frac{\delta}{\Omega_{\text{d}}}\sin\left(\frac{\Omega_{\text{d}} t}{2}\right)\right)|0\rangle + \left(i\frac{\Omega}{\Omega_{\text{d}}}\sin\left(\frac{\Omega_{\text{d}} t}{2}\right)\right)|1\rangle \tag{2.14}$$

where $\Omega_d = \sqrt{\Omega^2 + \delta^2}$. Looking at the probability to be in state $|0\rangle$ over time,

$$P(|0\rangle)(t) = 1 - \frac{\Omega^2}{\Omega^2 + \delta^2} \sin^2\left(\frac{\Omega_d t}{2}\right), \tag{2.15}$$

two effects stand out in the detuned case [Fox 05]. First, the oscillations become faster as the detuning increases. Second, the amplitude of the oscillations is reduced.

Figure 2.4 visualizes the trajectories described by equation 2.14 for different detunings on the Bloch sphere. For a given detuning, the trajectory corresponds to rotating the initial state vector around a fixed axis in the x-z plane. For zero detuning, the rotation axis corresponds to the x-axis, and we recover our previous analysis. As the detuning δ becomes larger and larger relative to the Rabi frequency Ω, the rotation axis gradually shifts towards the z-axis.

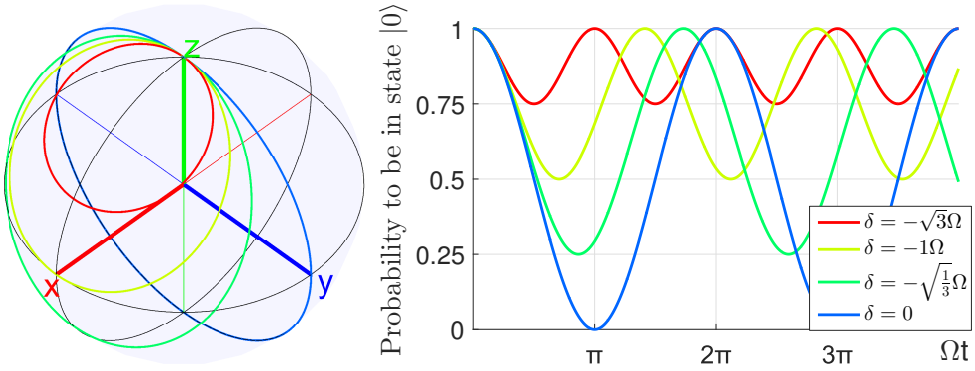


Figure 2.4: Resonant and detuned Rabi oscillations visualized on the Bloch sphere (left), and the probability to be in state $|0\rangle$ plotted against time (right).

Time-varying case

In general, both the Rabi frequency Ω and the detuning δ can be time-varying. We will later encounter this case when doing transport gates with $^{40}Ca^{+}$. The resulting Hamiltonian

$$\hat{H}(t) = \frac{\hbar}{2}(-\Omega(t)\hat{\sigma}_x + \delta(t)\hat{\sigma}_z) \tag{2.16}$$

is then time-varying. For this general Hamiltonian, one can not solve Schrödinger's equation analytically and has to rely on numerical methods.

2.2.4 Transport gates

Transport gates were originally proposed as an addition to the QCCD architecture proposed to build a scalable quantum computer based on trapped ions

[Kielpinski 02, Wineland 98]. In the QCCD architecture, there are different zones for processing and for storing information. It therefore requires the transport of ions between these zones. During processing, single or two-qubit gates are performed by shining laser pulses on static ions. While the trap itself can be scaled using microfabrication techniques, the same can not be said for the optics: To control the laser pulses, one typically uses bulky acousto-optic modulators and associated control electronics.

Transport gates address this scaling problem by shifting the burden of control from the optical to the electrical domain [D. Leibfried 07]. During a transport gate, the laser beam is static in space and constantly on. To perform a the gate, the ion is transported through the laser beam. By carefully controlling the velocity of the ion during transport, one can perform the desired gate. This also combines transport and processing into a single operation. However, we now have to deal with a time-varying detuning $\delta(t)$ as well as a time-varying Rabi frequency $\Omega(t)$.

2.3 Computer control system

Our computer control system consists of several pieces of hardware and software. A rough overview is given in Figure 2.5.

- **Ionizer** This software serves as the GUI to specify parameters of experiments and then run them. It communicates with the Zedboard over Ethernet. Ionizer sends the parameters to the Zedboard, which then runs the experiments. Afterwards, Ionizer receives the experimental results, visualizing them for the user. Ionizer also controls various other devices such as cavities and the shim electrodes not shown in Figure 2.5
- **Zedboard** The Zedboard is the heart of the control system. It has a Xilinx Zynq chip, containing a ARM CPU and an FPGA. It runs all the time-critical parts of the experiment. It is directly connected to the DDS, and can use transistor–transistor logic (TTL) signals to trigger other devices such as the DEATH. It also performs the readout of the photomultiplier tubes (PMT).
- **DDS** The DDS boards synthesize RF signals with a given phase, amplitude and frequency using "direct digital synthesis". These RF signals are then used to drive acousto-optic modulators which control various laser beams.
- **DEATH** The "Direct Ethernet Adjustable Transport Hardware" controls the voltages of the segmented dc electrodes of the ion trap. Currently, it consists of four boards, each containing four arbitrary waveform generators. Before experiments, the user can load several different waveforms into the memory of the DEATH over Ethernet. During ex-

periments, playback of these is then triggered by a TTL signal from the Zedboard.

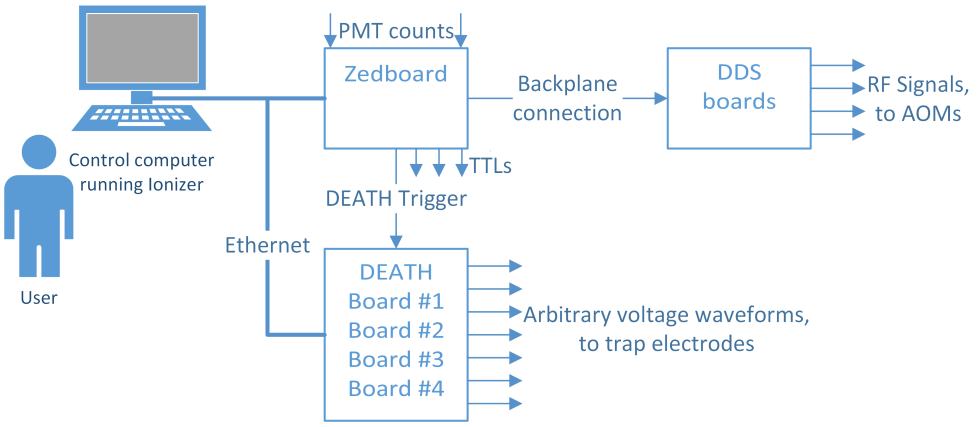


Figure 2.5: Schematic overview of the control system.

2.4 Quantum mechanics and implications for control

The laws of quantum mechanics have profound implications for the control of quantum systems. We therefore quickly review the relevant aspects of quantum mechanics. We then analyze the limitations they impose on the control of quantum systems, and describe how one usually deals with them in practise.

2.4.1 Quantum mechanics

In quantum mechanics, all the information about a quantum state is contained in its complex wave function $|\Psi(t)\rangle$. Its time evolution is governed deterministically through Schrödinger's equation

$$i\hbar\frac{\partial}{\partial t}|\Psi(t)\rangle = \hat{H}(t)\,|\Psi(t)\rangle \tag{2.17}$$

where $\hat{H}(t)$ is the potentially time-varying Hamiltonian. Observables are associated with self-adjoint operators $\hat{O}$. One can easily calculate the expectation value of an observable at time t:

$$\mathbb{E}[\hat{O}(t)] = \langle\Psi(t)|\,\hat{O}\,|\Psi(t)\rangle \tag{2.18}$$

As the operator $\hat{O}$ is self-adjoint, all its eigenvalues O_n are real, and using an appropriate normalization, the associated eigenvectors $|n\rangle$ form an orthonormal basis of the state space. The wave function can thus be expanded as a

complex superposition of these normalized eigenstates:

$$|\Psi(t)\rangle = \sum_n c_n(t)\,|n\rangle\,, \quad \text{with } c_n(t) \in \mathbb{C} \tag{2.19}$$

When we measure the observable $\hat{O}$ at time t, the result is probabilistic. We obtain eigenvalue O_n with probability $|c_n(t)|^2$. Furthermore, upon measurement, the wave function $|\Psi(t)\rangle$ collapses into the eigenstate corresponding to the observed eigenvalue. Performing a measurement thus alters the state itself.

2.4.2 Implications for experimental control

In practise, we usually deal with the destructive nature of measurements in quantum mechanics by doing repeated measurements on identically prepared quantum states. We first initialize the quantum state, then manipulate it for some time t, and finally perform a measurement. The results of a single measurement is typically called a "shot". Repeating this procedure many times allows us to gather statistics about the probabilities $|c_n(t)|^2$ to be in state $|n\rangle$ at time t. In order to measure the evolution of a state over time, we can repeat the above procedure, varying the duration t of the manipulation time.

Experimentally, this requires very good repeatability at the hardware level, as any significant changes in experimental conditions between individual shots would invalidate the above approach. Achieving good repeatability in practise is one of the technical challenges. One typically uses a great number of classical feedback loops to stabilize all the relevant experimental parameters. In our lab for example, we have classical feedback loops to stabilize the frequency and intensity of lasers, magnetic fields and the temperatures of sensitive equipment to name just a few.

The main consequence of repeated measurements on identically prepared quantum states for control is that we have to rely on pure feedforward. Using classical feedback is impossible, because we fundamentally can not measure the state in real-time[1]. The feedforward control inputs are usually calculated based on a model of the system under consideration. Because models are hardly ever perfect, the calculated feedforward control signals usually do not perfectly achieve what they were designed for. If the control signals are particularly simple, e.g. just a pulse of constant strength for a fixed time, we can

[1] In quantum computing, there is also the concept of quantum error correction. It enables fault-tolerant quantum information processing despite imperfect measurements and state preparation, decoherence etc. To do so, the information of a single logical qubit is stored in several physical qubits, using entangled states. By doing measurements on the entangled state, one can determine and correct for errors in the logical qubit. One can thus argue that in this case, one does use real-time feedback. It is however beyond the scope of this work.

easily calibrate them by experimentally scanning either the strength or the duration. However, this does not work for more sophisticated, time-varying control signals.

While we can not use classical, real-time feedback due to the lack of real-time state measurements, we can still use feedback based on the repeated measurements on identically prepared systems: First, we measure the time evolution of the quantum state for a specific set of feedforward control inputs. Based on the error, we can then update the feedforward control inputs and measure again. This process is repeated until the feedforward control inputs achieve our goals and the error is minimized. This procedure is commonly known as "iterative learning control" and is typically used to improve the control of repetitive processes. Not surprisingly, it originated in the control of robots performing repetitive tasks [Arimoto 84]. As outlined above, it is however also well suited to deal with the limitations that the laws of quantum mechanics impose on control.

Chapter 3

Calculation of Feedforward Control Inputs for Transport

In this chapter we develop a method to calculate the time-varying voltages for the segmented dc electrodes used for adiabatic ion transport. Rather than investigating ion transport from a theoretical point of view (as is done in [Lu 14, Fürst 14]), we focus on practical matters: We assume that the desired trapping potential over time is given. We then investigate how one can approximately form the desired potential in practise using a combination of the segmented dc electrodes, while also respecting practical constraints of our electronics and incorporating the effect of filters. Even though the method is presented using transport as an example, it should also be well suited for the splitting of ions or similar tasks.

3.1 Ion transport in practise

One way to transport ions within an ion trap is called adiabatic transport. In it, we move the location of the minimum of the harmonic potential well confining the ion in the axial direction. In this case, adiabatic means that the transport happens on a timescale that is long compared to the axial oscillation period of the ion. We can thus expect the ion to always be located at the minimum of the potential well. The desired potential $V_{\mathrm{des}}(z,t)$ has the form

$$V_{\mathrm{des}}(z,t) = \frac{1}{2}\frac{m\omega_z^2(t)}{q}(z - z_{\mathrm{min}}(t))^2 + C(t) \tag{3.1}$$

where m and q stand for the mass and charge of the ion, $\omega_z(t)$ is the desired axial trapping frequency, $z_{\mathrm{min}}(t)$ the desired position of the potential well and $C(t)$ an overall offset. The offset is included because it influences the orientation of the radial axes [de Clercq 15c].

While adiabatic ion transport is simple enough in theory, actually implementing it in practise is a bit harder. First, we form potentials along the trap axis using a superposition of the potentials of the individual control electrodes and can thus not create arbitrary potentials. Second, there are many constraints related to the limitations of the electronics used to control the voltages of the trap electrodes. Third, there are lowpass filters between the arbitrary waveform generators (AWG) and the electrodes. It is thus necessary to consider these practical matters when calculating the feedforward voltage waveforms used for transport.

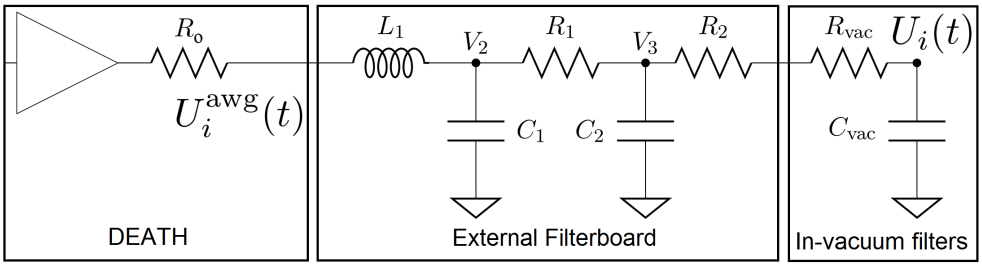


Figure 3.1: Simplified schematic of the electronics controlling the voltages of the segmented dc electrodes. The DEATH is heavily simplified and shows only the part relevant for this work.

A simplified schematic of the control electronics is shown in Figure 3.1. In our transport experiments, the feedforward control inputs are generated by home-built arbitrary waveform generator nicknamed DEATH[1] [de Clercq 15c]. The voltage signal then passes through an external filterboard implementing a third order Butterworth lowpass filter with a cut-off frequency of 250 kHz. Next, the signal enters the vacuum chamber where the ion trap is located. Right next to the trap there is another lowpass filter with a cut-off frequency of 810 kHz. The filters are designed to block signals near and above the motional frequency of the ion. This prevents unwanted motional heating due to electrical noise.

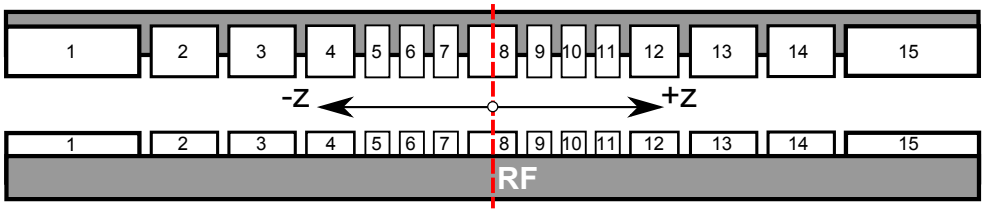


Figure 3.2: Schematic overview of the 2×15 segmented dc electrodes and two rf electrodes. Adapted from [de Clercq 15c].

[1] Direct Ethernet Adjustable Transport Hardware

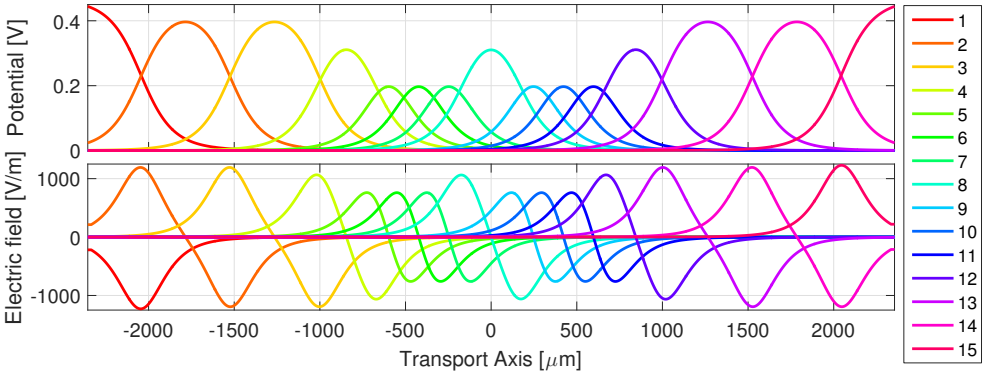


Figure 3.3: The potentials $V_i(z)$ and electric fields $V_i'(z)$ of the individual segmented dc electrodes along the transport axis upon applying 1 V to electrode i and grounding all others, as obtained from simulations. The naming scheme coincides with Figure 3.2.

The electrode structure is shown in Figure 3.2. The two rf electrodes confine the ion in the radial plane. The fifteen pairs of segmented dc electrodes are used to generate a confining potential along the transport axis. Currently, adjacent electrodes are driven by a single arbitrary waveform generator and each electrode has its own filter, but there are plans to change that in the future. Figure 3.3 shows the potentials along the trap axis due to the individual electrodes, as well as the first derivative, corresponding to the electric field. We can form specific potentials $V(z,t)$ along the z-axis using a linear superposition of the potentials $V_i(z)$ due to the individual electrodes, weighted by the time-varying voltages $U_i(t)$ put on the individual electrodes i

$$V_{\text{gen}}(z,t) = \sum_i V_i(z) U_i(t). \tag{3.2}$$

The problem left to solve is thus how to determine the time-dependent voltages $U_i(t)$ such that $V_{\text{gen}}(z,t)$ approximates $V_{\text{des}}(z,t)$ as well as possible while also following all the practical constraints. In particular, these are:

Constraints

i) The voltages $U_i^{\text{awg}}(t)$ at the output of the AWG are constrained to $U_{\min} \leq U_i^{\text{awg}}(t) \leq U_{\max}$. In our current setup $U_{\min} = -9$ V and $U_{\max} = 9$.

ii) The slew rate $\frac{dU_i}{dt}$ of the voltage $U_i(t)$ at the trap electrode is limited to $\mu_{\max} = 4\,\text{V}/\mu\text{s}$.

iii) The voltage $U_i(t)$ at the trap electrode corresponds to $U_i^{\text{awg}}(t)$ after passing the filters.

iv) The trapping wells should be at least a few tens of meV deep in order to avoid losing the ion.

Over the years, our group has used many methods with varying levels of sophistication to tackle this problem. First, we used ordinary least squares fits at various time steps and then built up the overall waveform, interpolating between individual time steps. Because this sometimes produced unreasonably high voltage jumps in violation of constraint (ii) between time steps, an additional regularization term punishing the voltage differences between time steps was later added. This corresponds to using a regularized version of least squares, sometimes also called Tikhonov regularization. This removed the voltage jumps, but the absolute voltage constraint (i) was still not addressed. Next, we switched to using dynamic programming and solved the problem over the whole time-horizon rather than using a one-step lookahead procedure. With the help of additional regularization terms punishing the absolute voltage and slew rate, one could also satisfy constraints (i) and (ii), but only after manually adjusting the weights of the cost terms. Eventually, we realized that all those issues can be addressed by quadratic programming, which is what we are currently using.

3.2 Quadratic programming

The name quadratic program (QP) refers to an optimization problem consisting of a convex quadratic cost function subject to affine constraint functions [Boyd 04]. In particular, one would like to optimize a convex cost function J

$$J = \frac{1}{2} x^T H x + f^T x + c \tag{3.3}$$

subject to the affine constraints

$$A_{\mathrm{eq}}\, x = b_{\mathrm{eq}} \tag{3.4}$$

$$A_{\mathrm{ineq}}\, x \leq b_{\mathrm{ineq}}. \tag{3.5}$$

This optimization problem has two nice properties that allow us to solve it efficiently. First, the cost function is convex. Second, the set of points satisfying the constraints is also convex. Because this kind of problem comes up very often in science and engineering, there are many sophisticated solvers that can handle even large instances reasonably quickly.

Calculating the waveforms corresponds to setting up the right kind of quadratic program that reflects our objectives and practical constraints and then handing it to a solver. After some experimentation, we arrived at the cost function

$$\begin{aligned} J = &\sum_{k=1}^{N} \|V_{\mathrm{des}}^k - V_{\mathrm{gen}}^k\|_{\mathbf{Q}_k}^2 \\ &+ \sum_i \left(r_0 \|U_i^{\mathrm{awg}} - U_i^{\mathrm{def}}\|^2 + r_1 \|\Delta_k U_i^{\mathrm{awg}}\|^2 + r_2 \|\Delta_k^2 U_i^{\mathrm{awg}}\|^2 \right) \end{aligned} \tag{3.6}$$

where we will now explain the various terms.

Overall, the cost function has two main competing terms. The first one punishes differences between the desired and the generated potential, where V_{gen}^k is formed by a discretization of equation 3.2 both in space (index suppressed) and time (index k). The positive semi-definite weighting matrices $\mathbf{Q}_k$ define a window of interest around the desired position of the potential minimum. The second set of terms regularizes the control inputs as follows:

- The term with the weighting factor r_0 sets the default electrode voltage to U_i^{def}. We use this to force all the electrodes that do not particularly influence the desired potential (for example because they are too far away) to a high voltage, resulting in deep potential wells.
- The terms with the weighting factors r_1 and r_2 enforce smoothness of the waveforms by punishing the first and second derivatives using a finite difference approximation based on the finite difference operator Δ_k. This has two advantages. First, it eliminates sudden changes in voltage that the AWG potentially can not follow. Second, the smoother a signal is, the faster its power spectral density rolls off. Because the lowpass filters act only on high frequencies, this thus limits distortion of the waveforms in the time domain.

As so often, one needs to make trade-offs between absolute performance (e.g. creating the desired potential) and practical considerations (e.g. creating waveforms that can be generated). Greedily optimizing only for absolute performance is thus too short-sighted.

Next, we cover the constraints. Constraints i) and ii) can be handled by enforcing

$$U_{\min} \leq U_i^{\text{awg}} \leq U_{\max} \tag{3.7}$$

$$-\mu_{\max} \leq \frac{\Delta_k U_i}{T_s} \leq \mu_{\max} \tag{3.8}$$

for all time steps k and electrodes i. Constraint (iii) connecting the voltages U_i^{awg} at the AWG to the voltages U_i at the trap electrodes can be written as

$$U_i = \begin{bmatrix} h_1 & 0 & \dots & 0 \\ h_2 & h_1 & \dots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ h_N & h_{N-1} & \dots & h_1 \end{bmatrix} U_i^{\text{awg}} \tag{3.9}$$

where h stands for the impulse response of the complete filter system (external and in-vacuum) in discrete-time.

The cost function J and constraints presented above can be written as a quadratic program. To do so, we combine all the voltages U_i at the trap

electrodes and U_i^{awg} at the AWG for all sample times into a single vector x. In correspondence to equation 3.5 we then construct H, f, c, A_{eq}, b_{eq}, A_{ineq} and b_{ineq}. When doing so, the use of sparse matrices is advisable, as it greatly reduces the memory requirements and the computation times. To calculate the waveforms, one then hands the quadratic program to a solver[2].

Although direct filter handling can be incorporated via equation 3.9, there are several practical issues. First, it causes the time-varying voltages at the AWG to be obtained via model inversion. Because we rely on pure feedforward, this requires an accurate model of the complete system from the AWG to the individual electrodes in the trap, as encoded in the impulse response $h(k)$. Second, it also requires handling of initial and final conditions, because the filters have internal state variables (e.g. the voltages across capacitors and the current through inductors). Third, when accounting for filters, the waveforms can not be played back with different sampling frequencies or in reverse, both two common practises in our group. Finally, the system does not behave as a perfect LTI system over its whole operating range, as will be discussed in chapter 5.

The practical issues associated with explicitly accounting for filters led us to a pragmatic solution: We enforce smoothness in our waveforms, thus minimizing the distortion of the voltage signals due to filters in the time-domain. Thus the main effect of filters is to delay the voltage signal by roughly 1.25 µs. Because this delay is the same for all electrodes, it poses no problem for our purposes. This approach has worked well at speeds of a few metres per second at which we are currently transporting. For faster transport filter handling becomes increasingly relevant. There is a simple way to check if one needs to account for filters for a given transport sequence: Calculate the waveforms without taking filters into account, and then simulate the effect of filters on them. If there is any significant distortion in the time domain other than a delay, this indicates that filters should be taken into account.

3.3 Results and discussion

Quadratic programming has proven to be well suited to calculate transport waveforms while also handling all the practical constraints. Figure 3.4 shows the electrode voltages over time for a typical transport sequence. It also shows the spatial potential resulting from the superposition of the individual electrodes at varying times during the transport. The generated potential wells are deep and coincide with the desired potential wells, plotted in green. The electrode voltages all vary smoothly over time and respect both the absolute voltage limits as well as the voltage slew rate constraint.

[2] We have made good experiences with Gurobi [Gurobi 15].

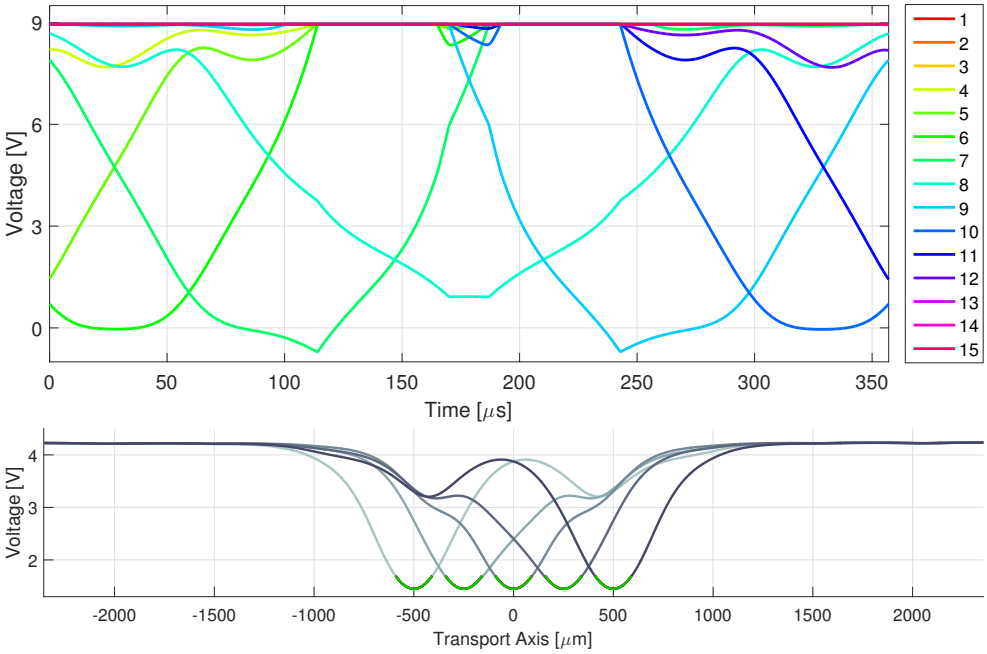


Figure 3.4: Top: Sample waveform showing the voltages $U_i(t)$ of the segmented dc electrodes calculated to transport from $-500\,\mu\text{m}$ to $500\,\mu\text{m}$. Bottom: The resulting potential wells at various times (gray) as well the reference potential (green).

Experimentally, transport of single and multiple ions using these waveforms worked reliably. For example, using this method we have performed parallel transport gates, transporting two ions simultaneously [de Clercq 15a]. Due to the deep wells, losing an ion is unlikely to happen. However, the waveforms are based on first principles modeling. Due to modeling errors, performance is degraded in practise. Nonetheless quadratic programming has proven to be a convenient method. Instead of relying on clipping of voltages or manual tuning of parameters as with previous methods, it can directly handle constraints.

The main difficulty does not lie in solving the quadratic program, but in accurately encoding what one actually cares about as an optimization problem. It can often happen that the solver finds a solution that optimizes a given cost function very well, but that is useless from a practical point of view. Once a suitable cost function and accompanying constraints are found though, the method requires no further effort to generate waveforms for various scenarios.

There are already some ideas for future improvements. One could form the cost function from a combination of cost terms involving the desired potential and its first and second derivative to gain more control about the trade-offs. E.g. one could then specify if one cares more about the well frequency, or the overall

offset. Quadratic programming also allows cost terms based on the 1-norm and the infinity-norm in addition to the 2-norm used here. Regularization terms on the electrode voltages using the 1-norm could be used to limit the number of electrodes active during transport, similar to feature selection in lasso regression [Tibshirani 96].

In the end, the waveforms are only as as good as the underlying system model. There are thus diminishing returns to refining the waveform calculation method. Eventually, one either needs to improve the modeling, use feedback, or e.g. iterative learning control in order to further optimize transport.

Chapter 4

Measuring the Hamiltonian during Transport

This work was carried out together with Ludwig de Clercq.

In order to control the velocity of an ion during transport, one first needs to be able to measure it. In this chapter, we present methods to experimentally determine the velocity of the ion over time. To do so, we measure the evolution of the internal electronic state of the ion as it passes trough a static laser beam. The state evolution depends on the frequency of the laser light as seen by the moving ion. As there is a Doppler shift between the frequency experienced by the moving ion and the frequency in a static frame of reference, this allows us to infer the velocity (as well as the Rabi frequency) from the observed dynamics.

Part of this work previously appeared in [de Clercq 15c, de Clercq 15b], the latter of which is also reproduced in Appendix D. Here, I thus opt to treat the subject from the viewpoint of control engineering and state estimation, which is the dual of control [Todorov 08]. I relate the method developed to infer the time-dependent Hamiltonian from the experimental data to concepts in control. The description of the underlying physics, experimental procedures and practical challenges is thus brief.

4.1 Hamiltonian estimation

4.1.1 Underlying physics

The main reason we care about the velocity of an ion during transport is its influences on the evolution of the internal electronic state. When performing transport quantum logic gates, the goal is to manipulate the state as precisely as possible, thus also requiring good control over the velocity. Specifically, we can write the spin Hamiltonian governing the evolution of the state in an

appropriate rotating frame of reference as

$$\hat{H}(t) = \frac{\hbar}{2}\left(-\Omega(t)\hat{\sigma}_x + \delta(t)\hat{\sigma}_z\right) \tag{4.1}$$

where $\delta(t)$ is the detuning from resonance, and $\Omega(t)$ is the Rabi frequency [de Clercq 15c]. We can write the detuning $\delta(t)$ as

$$\delta(t) = \delta_L - \frac{d}{dt}\left(\vec{k}(\vec{z}(t)) \cdot \vec{z}(t)\right) \tag{4.2}$$

where $\delta_L = \omega_L - \omega_0$ stands for the laser detuning from resonance and $\vec{k}(\vec{z}(t))$ is the wave vector of the laser at position $\vec{z}(t)$. Restricting ourselves to transport along the z-axis and assuming a constant wave vector $\vec{k}$, we can simplify this to

$$\delta(t) = \delta_L - |\vec{k}||\vec{v}(t)|\cos(\alpha) \tag{4.3}$$

where α is the angle between the wave vector $\vec{k}$ and the transport axis. We thus have a Doppler shift between the laser frequency experienced by the moving ion and the laser frequency in the laboratory frame. In reality, matters are a bit more complicated because Gaussian laser beams have curved wave fronts, resulting in a time-dependent angle $\alpha(t)$ [de Clercq 15b]. In this thesis, we opt to neglect treating this effect separately, as it only adds a lot of needless complexity, and because we are mainly interested in $\delta(t)$ (which already includes the effect of the curved wavefronts) rather than the velocity on its own. We thus freely convert between detunings and velocities using equation 4.3. As a rule of thumb, a detuning of 1 MHz corresponds to a velocity of roughly 1 m/s when transporting $^{40}Ca^{+}$ in our current setup.

The task of inferring the velocity over time can thus be accomplished by determining the time-varying Hamiltonian $\hat{H}(t)$, additionally also giving us the Rabi frequency over time. While there are many techniques to experimentally determine a static Hamiltonian, the same can not be said for the time-varying case. One way to infer $\hat{H}(t)$ would be to discretize in time, and to perform process tomography for each time step. However, this would be quite demanding experimentally, as it requires accurately preparing different input states, and measuring the output states in different bases.

In our method, we rely on measuring the time evolution of the state in a single basis (i.e. measuring σ_z) and additionally varying the laser detuning δ_L, which is easy to do in practise. We thus measure e.g. $\langle\hat{\sigma}_z(t, \delta_L)\rangle$. Experimentally this is done by turning the laser off at varying times t, stopping the spin dynamics (see [de Clercq 15b] for details). At a time t, the state will vary for different laser detunings due to the different previous time evolution. This introduces the diversity in the measurements needed to infer the Hamiltonian $\hat{H}(t)$. This is similar to system identification where the test signal needs to be sufficiently rich in order to excite the various modes of the system under investigation, or to the concept of observability in control.

4.1.2 Inferring the Hamiltonian

Inferring the Hamiltonian $\hat{H}(t)$ from the experimental data $\langle \hat{\sigma}_z^{\text{meas}}(t, \delta_L) \rangle$ proved out to be much harder than expected. Here, we first discuss why our initial attempts with various standard techniques for state estimation failed. We then outline how we came up with a solution, combining several concepts from control and adapting them to our specific problem.

Initially, we investigated the use of standard estimation techniques from control engineering to infer the detuning and Rabi frequency over time from the experimental data. Because the Kalman filter can not handle nonlinear dynamics, we first investigated using the extended or unscented Kalman filter. There, we faced several issues. First, we have a unusually large number of states, as we need to store the quantum state for each laser detuning. Second, each quantum state is constrained to lie on the Bloch sphere. After re-parametrizing the states to always respect the constraints and accepting a huge state vector, we noticed that the algorithms were prone to overreacting. For example, after underestimating the Rabi frequency in one timestep, it would massively overestimate it in the next, quickly leading to divergence. The solution to this problem is clear. Rather than overcompensating, the algorithm should have gone back in time and fixed the previous state estimate. We therefore also looked at smoothing techniques that reconstruct the state trajectory over time based on all the data, rather than only using the data up to the current timestep. However, smoothing is usually done by a forward pass, followed by a backward correction pass [Särkkä 13]. As we could not even complete a forward pass due to overcompensating, we abandoned this idea as well.

Next, we investigated the use of optimization-based techniques. Essentially, we want to determine $\delta(t)$ and $\Omega(t)$ such that the resulting simulated spin dynamics matches the experimental data as well as possible. To quantify the match, we can use a standard chi-squared cost function J

$$J = \chi^2 = \frac{1}{N} \sum_t \sum_{\delta_L} \left[\frac{\langle \hat{\sigma}_z^{\text{meas}}(t, \delta_L) \rangle - \langle \hat{\sigma}_z^{\text{sim}}(t, \delta_L) \rangle}{\sigma^{\text{meas}}(t, \delta_L)} \right]^2 \tag{4.4}$$

where $\sigma^{\text{meas}}(t, \delta_L)$ is the standard deviation of $\langle \hat{\sigma}_z^{\text{meas}}(t, \delta_L) \rangle$ assuming shot noise and N is the number of data points.

When minimizing J we face two main difficulties. First, we are optimizing with respect to time-varying functions $\delta(t)$ and $\Omega(t)$. If we simply discretize the problem in time, we end up with a great number of optimization variables. Second, $\langle \sigma_z^{\text{sim}}(t, \delta_L) \rangle$ depends on $\delta(t)$ and $\Omega(t)$ in a non-trivial way, being linked by Schrödinger's equation. We thus have to solve the following

optimization problem

$$\begin{aligned} \min_{\delta(t),\,\Omega(t)} \quad & J(\delta(t), \Omega(t)) \\ \text{subject to} \quad & i\hbar\frac{\partial}{\partial t}\left|\Psi(t,\delta_L)\right\rangle = \hat{H}(t)\left|\Psi(t,\delta_L)\right\rangle, \\ & \left|\Psi(t=0,\delta_L)\right\rangle = \left|0\right\rangle, \\ & \left\langle\hat{\sigma}_z^{\text{sim}}(t,\delta_L)\right\rangle = \left\langle\Psi(t,\delta_L)\right|\hat{\sigma}_z\left|\Psi(t,\delta_L)\right\rangle. \end{aligned} \tag{4.5}$$

This optimization problem also turned out to be difficult. Initial attempts with genetic algorithms for global optimization and nonlinear least squares fitting routines produced mixed results at best and took many hours to run. Two ideas helped to overcome the difficulties. First, we use Basis spline curves [de Boor 01, Bartels 95] to efficiently represent $\delta(t)$ and $\Omega(t)$ using just a few parameters each, assuming both of them to vary smoothly over time. This is reasonable in our case, because we expect the velocity and laser intensity to vary smoothly during transport. Second, we use the causality inherent in our data. If we can not make sense of the data up to some time, it makes no sense to try to look even further. We thus optimize J only over a short time horizon at first, and later iteratively extend it, ultimately covering the whole timespan. For each time horizon we optimize J with the help of a standard nonlinear least squares fitting routine[1]. When extending the time horizon, we can use the previously found $\delta(t)$ and $\Omega(t)$ as a starting point, warm-starting the solver. We call this procedure "extending horizon estimation" (EHE). For details on the actual implementation the reader is referred to [de Clercq 15c, de Clercq 15b].

In retrospect, we can easily relate the ideas used in extending horizon estimation to common concepts in control engineering. To do so, we first need to introduce model predictive control (MPC) and the dual technique of moving horizon estimation (MHE).

In model predictive control, the control input u_k at timestep k is calculated by minimizing a cost function capturing the control objectives with respect to all future control inputs u_i, $i \geq k$. Afterwards, the first control input u_k is enacted. At the next timestep $k+1$, the procedure is repeated, which is why it is often also referred to as "receding horizon control". The main strength of this approach is that it can easily handle practical constraints, much in the same way as we did when we calculated the feedforward control inputs in chapter 3. Because of finite computational resources, the optimization time horizon only extends some fixed time into the future. To ensure stability, one usually introduces a terminal cost, approximating all future costs incurred after the end of the explicit optimization time horizon [Rawlings 09]. The need for such a terminal cost is best seen in analogy to business. If a company

[1] Specifically, we used Matlab's *lsqnonlin* function.

optimizes only for short-term profits, the chances are that it will not be around for very long.

There is a duality between control and state estimation. In control, one aims to minimize the deviations from a desired state (e.g. caused by disturbances), expending as little control effort as possible. In state estimation, the aim is to minimize the difference between the estimated and the true state of the system, making as little unexplained corrections to the state estimate as possible. Because of the similarity of the tasks, a concept used for control can often be adapted to be used for state estimation and vice versa. The most famous example is the correspondence between the linear quadratic regulator and the Kalman filter. Moving horizon estimation is the dual technique to model predictive control for state estimation.

In moving horizon estimation, the aim is to estimate the current state based on the past few measurements. This is again done by solving an optimization problem. The estimated state trajectory is optimized such that the difference between the actual and expected measurements are minimized. Instead of a terminal cost ensuring stability, we now have an arrival cost, penalizing changes in the starting state. Determining the right arrival cost for a general setting is still an open problem. As in MPC, the optimization is performed at each timestep. At the next timestep, we move the optimization time-horizon forward by one sample.

Extending horizon estimation as introduced above is very similar to moving horizon estimation. However, instead of moving both the start and the end of the time-horizon, we leave the start fixed, and extend the end. We can thus avoid having to introduce a heuristic arrival cost, something that would be very difficult for our application. The downside is of course that the computational burden grows over time as the time horizon increases. In MHE, that would be unacceptable, as it is used to estimate the state in real-time (and thus potentially forever), often on severely constrained embedded platforms. But for our application, this is no problem, as the overall time horizon is fixed in length and we have no real-time constraints. Using splines to efficiently represent $\delta(t)$ and $\Omega(t)$ is conceptually similar to move-blocking in MPC, greatly reducing the number of optimization variables. Note however that EHE is not guaranteed to find the global minimum of optimization problem 4.5.

4.1.3 Results

Our approach to reconstruct a time-varying Hamiltonian worked well in practise. Figure 4.1 shows two datasets, together with the best fit and residuals. The only difference between the datasets is the position of the laser beam along the transport axis. The experimental data and the best fit agree very well, indicating that our Hamiltonian $\hat{H}(t)$ models the underlying spin dynamics well. Figure 4.2 shows the inferred time-varying detuning $\delta(t)$ and Rabi fre-

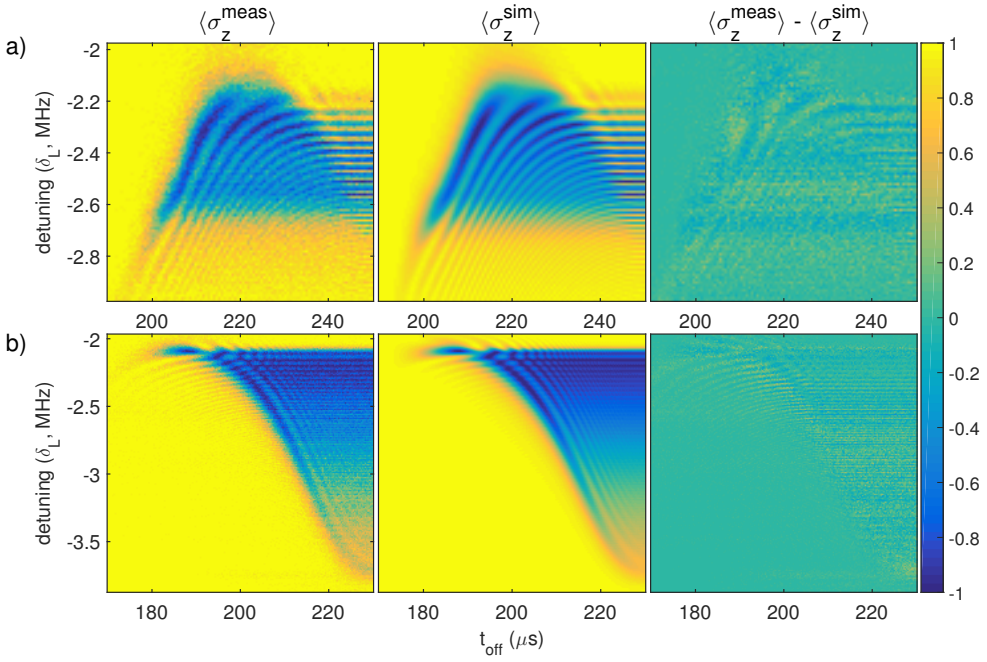


Figure 4.1: The measured spin populations as a function of time and detuning (left), the best fit to it (middle) and residuals (right) for two datasets, corresponding to different positions of the laser beam along the transport axis. Each data point $\langle \hat{\sigma}_z^{\text{meas}}(t, \delta_L) \rangle$ is the result of 100 repeated measurements. Adapted from [de Clercq 15b].

quency $\Omega(t)$. We see that the detunings $\delta(t)$ of the two datasets do not overlap. We suspect that this is due to curvature of the wave fronts of the Gaussian laser beam. Taking this effect into account and calculating the velocity $\dot{z}(t)$ according to equation 4.2, we obtain consistent results. Overall, the difference between the velocity calculated based on equation 4.2 and 4.3 is very small though. The method also works well for datasets with much lower resolutions in both time and detuning. However, one must then be careful to ensure that the timestep used to simulate the spin dynamics is small enough. There is however a problem that limits the accuracy of the method. For high beam powers, we are unable to reproduce the start of the spin dynamics such as the one shown in Figure 4.3, taken at the maximum laser beam power. With our model 4.1, we are unable to reproduce the spin dynamics narrowly confined in detuning in the region $t < 50\,\mu\text{s}$. Currently, the reason for this is unclear, but we suspect that we might have neglected an effect that only becomes relevant in this specific regime. As this problem only occurs in a small region at the start of the fit for high beam powers, it is not very restrictive.

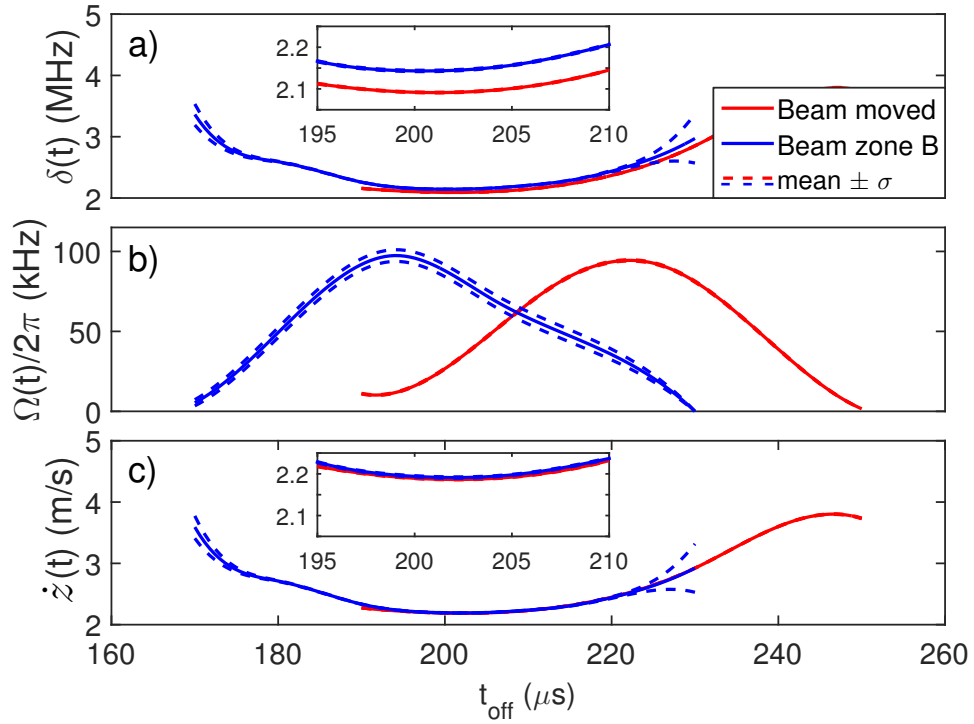


Figure 4.2: Estimates of the time-varying coefficients: a) Detuning $\delta(t)$, b) Rabi frequency $\Omega(t)$ and c) Velocity $\dot{z}(t)$ of the ion, taking the curvature of the wave fronts into account.

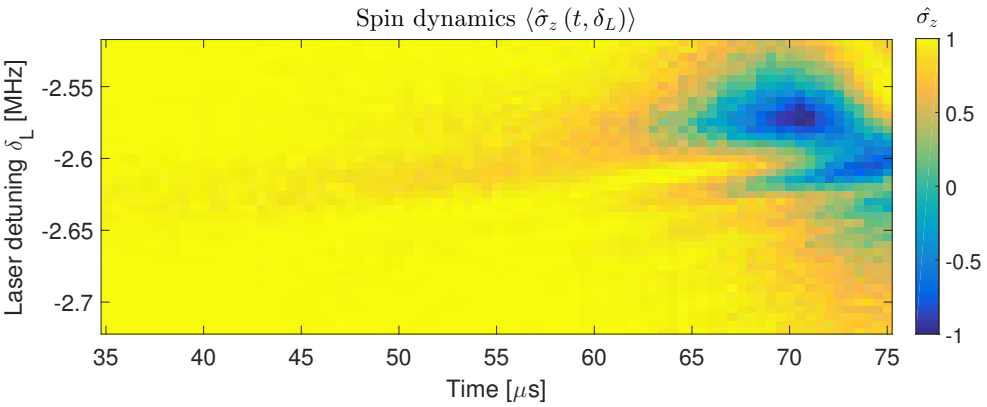


Figure 4.3: Start of the spin dynamics at maximum beam power. While we can easily reproduce the dynamics for $t > 50\,\mu s$ with our model, we fail to do so for the narrow features in the region $t < 50\,\mu s$.

4.2 Limiting case: Direct measurement of the Hamiltonian

In limiting cases, one can also directly measure the Rabi frequency $\Omega(t)$ and detuning $\delta(t)$ at a time t upon making some approximations. So far, we have always measured the spin dynamics with the laser beam turned on already before the start of the transport, and then turning it off at varying times into the transport. The spin evolution at time t thus depends on the previous evolution, making it hard to interpret. After looking at many datasets, we noticed that the initial dynamics upon entering the beam are essentially always the same, and hence easy to interpret. This led to the idea of probing the spin dynamics with a short laser pulse of length T at varying times t.

Keeping the pulse duration T short compared to the timescale at which $\Omega(t)$ and $\delta(t)$ fluctuate allows us to approximate them as constant over the course of the pulse. Because we additionally always prepare the state to initially be in $|0\rangle$, this corresponds exactly to the detuned interaction described in section 2.2.3. Using equation 2.15, we can write the spin population in state $|0\rangle$ after a pulse of duration T as

$$P(|0\rangle) = 1 - \frac{\Omega^2}{\Omega^2 + \delta^2} \sin^2\left(\frac{\sqrt{\Omega^2 + \delta^2}\, T}{2}\right). \tag{4.6}$$

Figure 4.4 plots equation 4.6 for different values of ΩT. Choosing the pulse duration T such that $\Omega T = \pi$ results in the highest contrast. In practise, we can not always do so, because we have limited beam power, resulting in a limited Rabi frequency Ω. However, we can simply choose a shorter pulse duration T, leading to lower contrast, but keeping our underlying assumptions valid.

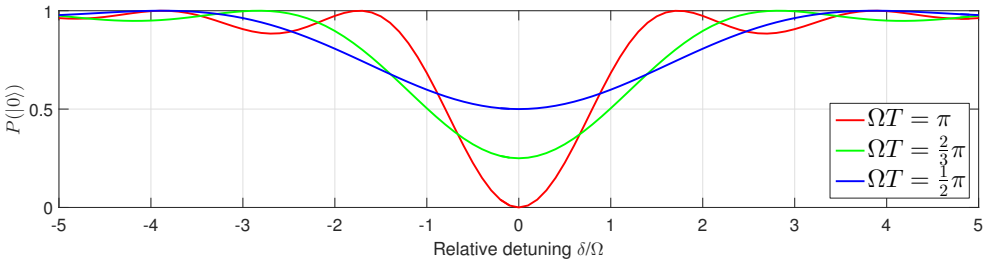


Figure 4.4: Probability to find the spin in state $|0\rangle$ after a laser pulse of duration T as given by equation 4.6.

In practise, the method has worked well. Figure 4.5 presents experimental data and a best fit to it, showing that there is a good agreement. We can readily read off that the resonance occurs for a laser detuning δ_L of roughly

-2.3 MHz, corresponding to a speed of roughly 2.3 m/s. Using this method, we can thus measure $\delta(t)$ and $\Omega(t)$ at a specific time t. If we now also vary the starting time of the short laser pulse, we can sample $\delta(t)$ and $\Omega(t)$ at different times. Figure 4.6 shows an example of that. Note that the detuning is now plotted along the vertical axis. We can interpret each vertical slice in the same way as before. We can thus readily read off that the velocity started out at roughly 2.4 m/s, then increased to 3 m/s and finally decreased down to 2.6 m/s.

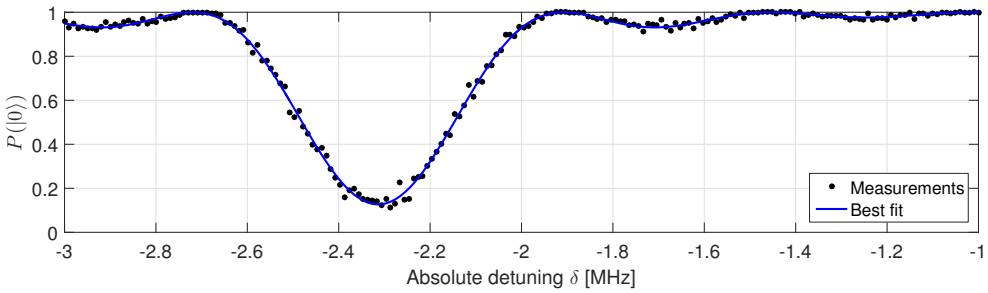


Figure 4.5: Measured probability to find the ion in state $|0\rangle$ after a laser pulse lasting 1.4 µs.

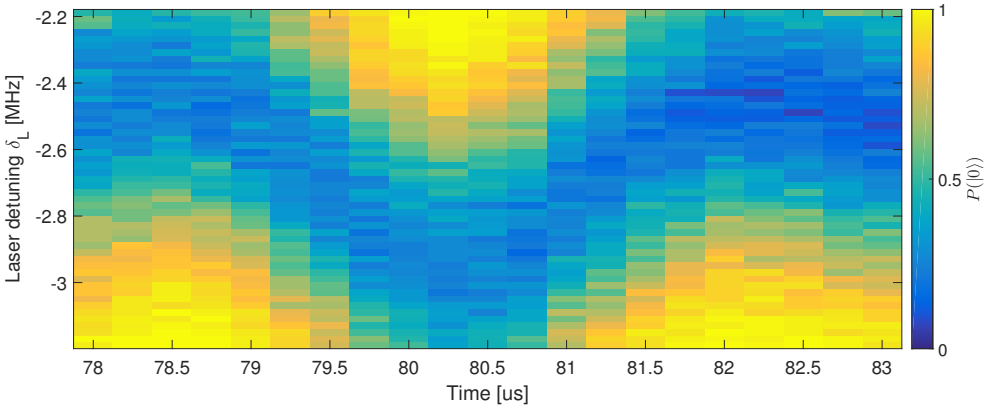


Figure 4.6: Measured probability to find the ion in state $|0\rangle$ after a laser pulse lasting 1.4 µs. The horizontal axis marks the time at which the pulse was turned on.

4.3 Discussion

The two methods described above to determine the time-varying coefficients $\delta(t)$ and $\Omega(t)$ of the Hamiltonian $\hat{H}(t)$ both have advantages and disadvan-

tages, making them complimentary to each other.

The first method is based on measuring the time evolution of the spin degree of freedom and then inferring the Hamiltonian $\hat{H}(t)$ using optimization techniques. It is easy to implement in practise and works well for a general setting, not requiring any strict assumptions. However, the experimental data is hard to interpret without running an extended analysis. Also, if we want to measure the velocity at a particular time t, we still have to measure the complete time evolution leading up to that time. Furthermore, we can not use this method indefinitely in time, as eventually noise accumulated during the time-evolution will drown out the signal. In practise, we were operating far from that regime though, making it no concern for now. Fundamentally, the analysis method relies on quickly being able to simulate the spin dynamics many times. It will thus not scale well to more complex quantum systems with a large number of states.

The second method is based on probing the spin dynamics only for a short time. The main disadvantage is that it requires a set of simplifying assumptions to hold, limiting the regime for which it can be used in practise. For example, the method does not work at low beam powers, because as it would require long pulse durations, invalidating our assumptions that $\delta(t)$ and $\Omega(t)$ are constant over the pulse duration. Allowing time-varying $\delta(t)$ and $\Omega(t)$ in the analysis would then again lead to the first method. In practise, making sure that the assumptions are met is not always easy. The method allows us to measure $\delta(t)$ and $\Omega(t)$ at a specific time, thus enabling pointwise measurements. The resulting data is easy to interpret by eye and simple to process.

In practise, we have mostly used the first method because it has no restrictive underlying assumptions. Another practical reason was that our analysis process was much more refined for the first method. The second method proved to be very useful to quickly check the effects of modifications on the velocity without requiring any sophisticated processing. It thus proved very helpful over the course of this work in order to quickly try and test various changes.

Chapter 5

Hardware Modifications: Filters

The technique presented in the previous chapter is nice in that it enable us to measure the velocity of the ion during transport. What is not so nice, however, is that even though we were aiming for a constant velocity of 2.8 m/s, the measured velocity profile is far from constant, with deviations of up to 1 m/s (see Figure 4.2c). Originally, the plan was to use iterative learning control in order to compensate for those deviations. However, after some initial tests, we discovered a serious flaw in the design of the electronics driving the trap electrodes. We therefore first investigated this issue in detail, and modified the hardware to fix the problem at its root, benefiting all future experiments.

5.1 The problem: Limits to the LTI assumption

Discovering the issue was tricky mainly because all our measurements were very indirect, and because we did not anticipate it at all. The problem was uncovered when we tried to determine the time at which the ion is at the exact center of the trap. To do so, we made use of the fact that electrode 8 coincides with the center. As visible in Figure 3.3, the electric field due to electrode 8 has a zero crossing at $z = 0$. Therefore, the effect of a positive voltage step on electrode 8 depends on the position of the ion: If it is left of the center ($z < 0$), it will move the trapping well to the left. If we are transporting from left to right and apply a positive voltage step while the ion is still to the left of the center, the velocity is thus reduced. If the ion is to the right of the center, the well is shifted to the right, thus speeding up the ion. The closer the ion is to the center of the trap, the weaker this effect becomes.

Using this effect, we experimentally determined the time at which the ion is at the center. We transported the ion from $z = -200$ µm to $z = 200$ µm using normal voltage waveforms. On top of those, we added a voltage step of 0.5 V lasting for 4 µs to electrode 8, centered at varying times. Using the direct technique from chapter 4, we measured the effects of this voltage step on the

velocity. Figure 4.6 shows one such measurement: At $t = 78\,\mu s$ the ion was to the left of the center, and thus the velocity was slowed down. Then, after $4\,\mu s$ at $t = 82\,\mu s$ the negative voltage step from 0.5V back to 0 V again causes a slow down of the velocity because the ion is now located right of the trap center. Thus the ion crossed the trap center between those two times. Using this technique, we found a delay of around $14\,\mu s$ between the time when we would have expected the ion to be at the center and the time when it actually was there. Because this delay was inexplicably high, we had a closer look at the electronics, which was previously introduced in chapter 3 in Figure 3.1.

The electronics consists of a home-built arbitrary waveform generator nicknamed DEATH[1], external lowpass filters, and another lowpass filter within the vacuum can, close to the trap electrodes. The issue lies subtly at the interface between those blocks. When testing the DEATH in isolation, they perform well in that they nicely output the desired waveform. The same is true for the external filters: The measured transfer function matches the design value well (see Figure 5.3). Even though both subsystems perform well in isolation, the overall system performed in unexpected ways. Step testing on the combined system of DEATH and external filter (but excluding the in-vacuum filters) revealed two issues. First, the output series resistor R_o on the DEATH drastically lowered the corner frequency from 250 kHz to roughly 20 kHz. Second, the voltage slew rate was severely limited. This is due to the non-ideal behaviour of real op-amps. In order to change the output voltage at the end of the filter, its capacitors need to be charged. This in turn requires a current, which is supplied by the op-amps. Because real op-amps can only supply a limited current, the slew rate of the filter output is also limited. We were thus accidentally operating in the nonlinear regime of our hardware, saturating the op-amp output current.

The effect of both issues on transport is visualized in Figure 5.1 and can easily be identified. First, the delay between the signals before and after the filters is much larger than the $1.25\,\mu s$ one would expect for a third order Butterworth filter with a cut-off frequency of 250 kHz. Second, for electrodes 7 and 9, we run into the slew rate limit. This heavily distorts the waveform in the time domain. From the reference voltage waveforms, we would expect the ion to be at the trap center around $t = 160\,\mu s$, which is when electrodes 7 and 9 have equal voltages. However, we see that due to the filters, the time when both voltages actually are equal is delayed by roughly $14\,\mu s$, explaining the delay measured previously.

Once the problems were identified, resolving them was quite straightforward. The first issue was solved by shorting out the output series resistor on the DEATH. To address the second problem, the external filterboard was redesigned. When designing analog filters, one usually chooses a fixed topology,

[1] Direct Ethernet Adjustable Transport Hardware

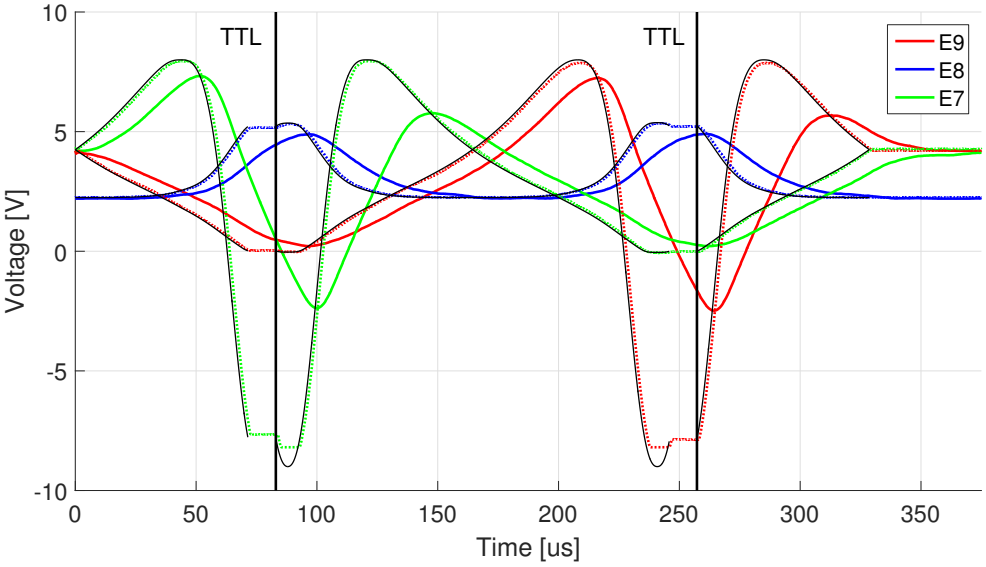


Figure 5.1: Voltages during a typical transport operation for electrodes 7, 8 and 9 measured at varying points along the electronics: The reference signal (thin black line), the signal measured when the DEATH is connected to a high-impedance load (e.g. and oscilloscope, dotted colored line), and the signal of the system consisting of DEATH and the old external filters (but excluding the in-vacuum filters, solid colored line).

and then finds suitable values for capacitors, inductors and resistors to match the transfer function to the desired one. The transfer function only depends on certain ratios of the components, but not on their absolute values. Using C, L, and R, one can form three time constants: $\tau_1 = RC$, $\tau_2 = \sqrt{LC}$ and $\tau_3 = {}^{L}/_{R}$. A combination of these will determine the overall transfer function. The time constants are invariant upon multiplying all the inductors and resistors by a factor of α while simultaneously also dividing all the capacitor values by the same value α. Thus the overall transfer function is also invariant with respect to this transformation. What does change however is the current drawn by the filters: Because the capacitors are lower by a factor of α, they also require a factor of α less current for the same voltage swing.

The new external filterboards are based on the old design, but with the components scaled by a factor $\alpha = 10$. This reduces the current required by the filters by a factor of 10. With the new filters, the voltage slew rate measured at the output of the external filterboard for a voltage step at the DEATH from -9 V to +9 V improved from $0.7\,\mathrm{V/\mu s}$ to $4\,\mathrm{V/\mu s}$. For such steps, we still run into the current limit (as is illustrated by the perfect LTI step response in 5.2), but its effect is reduced. To avoid this limit, one can easily incorporate it as a constraint in the calculation of the waveforms, as outlined in chapter 3.

Increasing the obtainable voltage slew rate even more would necessitate larger changes, as we can not simply increase the scaling factor α indefinitely: First, larger resistors result in higher Johnson–Nyquist noise, as its power spectral density is proportional to resistance. Currently, the noise is still dominated by the resistor of the in-vacuum filters. Second, moving to higher and higher inductances is impractical due to the limited availability of components in small packages, and due to the non-idealities of real inductors.

5.2 Results and discussion

The new filters were tested thoroughly before installing them in the overall setup. The results are shown in Figure 5.2 and 5.3. Figure 5.3 shows the transfer function of the new and old filters, measured with a Digilent analog discovery kit. The measured transfer functions are quite similar, as one would expect. The transfer function of both filters compare favourably with the design value up to roughly 1 MHz. From then on, the attenuation is limited to roughly 40 dB. The exact reason for this is unclear, but we suspect that non-ideal components might explain it. The behavior of the combined system of DEATH and external filterboard in the time domain is shown in Figure 5.2. Rescaling the component values of the external filters and shorting the output series resistor on the DEATH both clearly improved the step response.

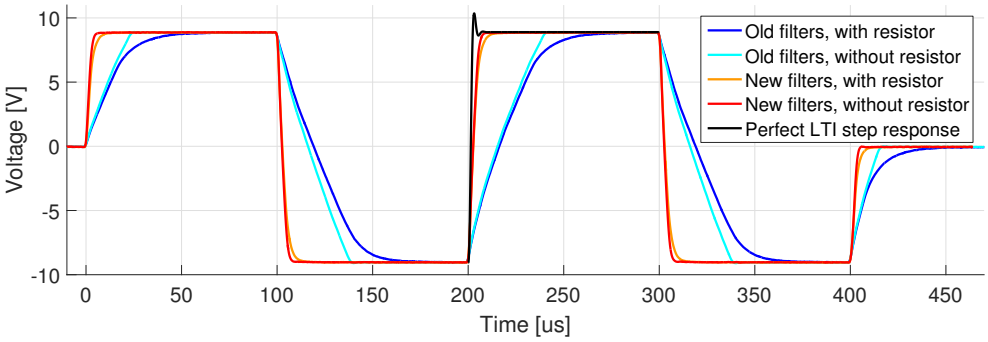


Figure 5.2: Time Domain: Step response of the combined system consisting of the DEATH AWG and external filterboard, measured with and without series output resistor R_o. Also shown is the ideal step response of the filters, indicating that we still run into the current limit.

After installing the new filters, we also measured the heating rate using blue sideband flops. At first, when only one of the three new external filterboards was directly connected to ground, the heating rate was markedly worse when compared to the old filters. After connecting each of the three new filterboards directly to ground, the heating rate dropped to the value of the old filterboards.

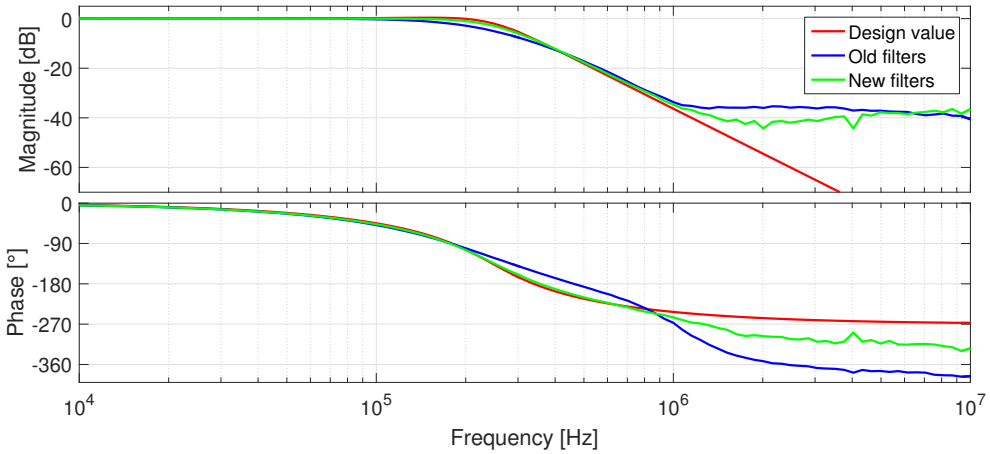


Figure 5.3: Frequency Domain: Measured and desired filter transfer functions.

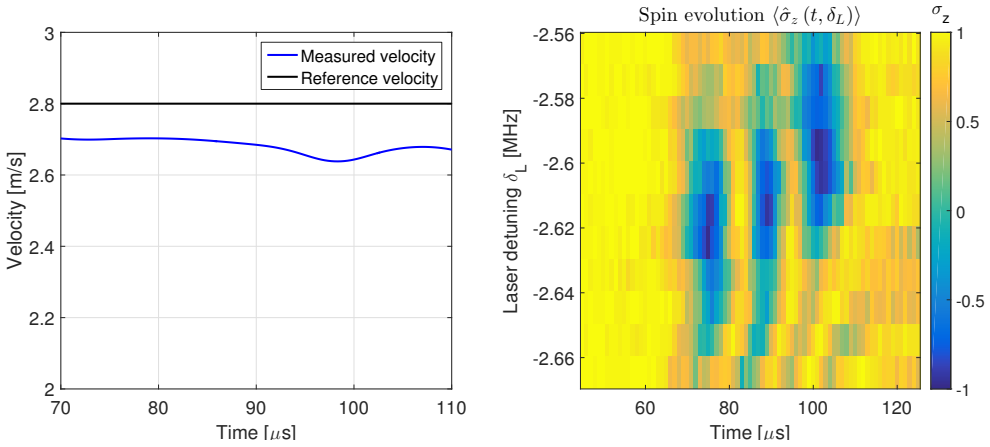


Figure 5.4: Velocity profile and spin dynamics after improving the hardware.

The velocity profiles during transport are also much improved. Figure 5.4 shows the velocity profile measured after modifying the hardware but with a waveform generated using the same method and model as before. There is still a deviation from the desired velocity of 2.8 m/s, but the velocity fluctuations are reduced down to roughly 0.1 m/s. Compared to the fluctuations of roughly 1 m/s from before the modifications (see e.g. Figure 4.2) this is an improvement by a factor of roughly 10.

We can already think of two possible future improvements. Currently, each AWG channel of the DEATH drives two filters, each leading to the two opposite dc electrodes. We could thus gain a factor of two by having a separate AWG channel for each filter. To limit distortion of the voltage waveforms by the filters in the time domain, it would also be interesting to switch the filter type. Currently, we are using a third order Butterworth filter, which has a

maximally flat response in the passband. Instead, we could switch to Bessel filters, which have a maximally flat group delay, minimizing the time-domain distortion of the input signal, only delaying it. However, implementing a Bessel filter would require considerably higher inductor values [Blakestad 10].

The improvements to the hardware will be beneficial for any kind of operation involving quickly varying voltages. This is in contrast to ILC, which only improves specific operations 'on demand', i.e. upon actually running ILC. Furthermore, staying away from the current saturation regime of the op-amps enables us to treat the combined system of DEATH and filters as a standard linear, time-invariant (LTI) system. As LTI systems are the staple of control engineering, this simplifies the task of implementing ILC.

Chapter 6

Iterative Learning Control

While the hardware changes described in the previous chapter have improved the velocity profile during transport, there are still some remaining velocity fluctuations. Instead of further improving our modeling or hardware, we now use iterative learning control (ILC) to optimize the feedforward control inputs in order to reduce the velocity fluctuations. We do so by combining the technique for calculating the feedforward control inputs developed in chapter 3, and the technique for measuring the velocity of the ion over time developed in chapter 4. We present two different approaches for ILC, the corresponding experimental results and discuss ideas for future studies.

6.1 Iterative Learning Control

In control engineering one often uses a combination of feedforward and feedback in order to control a process. The feedforward control inputs are usually calculated in advance based on a mathematical model of the process and are designed to make the process track a reference trajectory. The feedforward control inputs are directly fed to the actuators. Solely relying on feedforward is unreliable because of potential modeling errors and disturbances. One therefore often also uses feedback to minimize the tracking error on the real system based on real-time measurements of the tracking error.

Up to now we have only used feedforward control in this thesis. While the results have been reasonably good, there are remaining velocity fluctuations. It would therefore be tempting to use real-time feedback in order to minimize them. However, we can only infer the velocity of the ion based on retrospectively analyzing its spin dynamics. We therefore have no real-time measurements and thus can not use real-time feedback. However, our goal of transporting the ion from A to B at a constant velocity does not change over time. We can therefore measure the velocity for a specific set of feedforward control inputs and afterwards update the control inputs iteratively.

Introducing an iteration domain thus allows us to close the loop. This process of iteratively optimizing the feedforward control inputs is commonly known as iterative learning control.

Next, we present two different ILC schemes, closing the loop at different points as illustrated in Figure 6.1. The first method closes the loop by modifying the reference position over time used to calculate the feedforward control inputs as described in chapter 3. The second method operates directly on the voltage waveforms. The main motivation for the first approach is its simplicity. Compared to directly operating on the voltage waveforms, modifying the reference position is easier conceptually. In contrast to the second approach, it requires no additional modeling and identification efforts, although that also makes it susceptible to modeling errors.

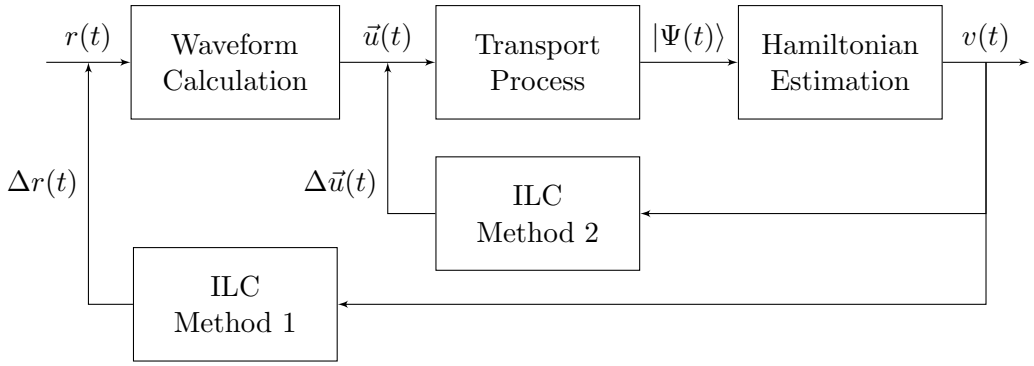


Figure 6.1: Conceptual overview of the two ILC methods investigated.

6.2 Method 1: Modifying the reference

The main motivation for the first approach is the observation that although we have so far failed to transport exactly at the reference velocity, we are already quite close, as illustrated in Figure 5.4. The model used to calculate the feedforward control inputs as outlined in chapter 3 might thus be already reasonably accurate. Iteratively adjusting the reference position over time based on the measured velocity error and then recalculating the feedforward control inputs might thus already be all we need to do.

Figure 6.2 presents a conceptual overview of the approach[1]. At iteration $m, 0 \leq m$ the reference position $z^m_{\text{ref}}(t)$ is used to calculate the feedforward control inputs $u^m_i(t)$, collectively referred to as the waveforms. This is done as outlined in chapter 3, inverting our estimated system model $\hat{G}$. The waveforms are then used on the real system G, and we infer the velocity $v^m_{\text{meas}}(t)$ from the

[1]We opt to present the analysis in continuous-time, resulting in simple formulas. The actual implementation is however done in discrete-time via a straightforward discretization.

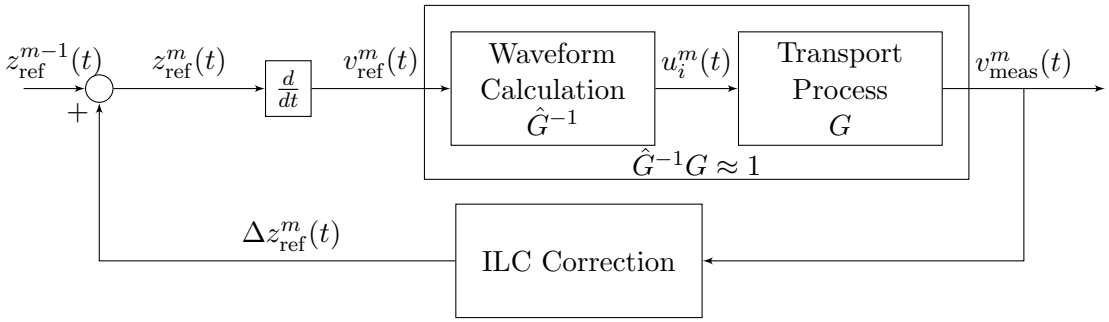


Figure 6.2: Outline of the first ILC method and the signals involved in it.

time evolution of the spin as outlined in chapter 4. We can then use $v^m_{\mathrm{meas}}(t)$ to calculate an updated reference position

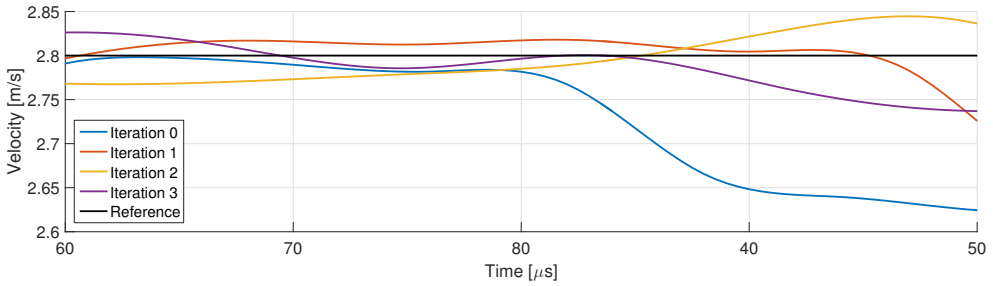
$$z^{m+1}_{\mathrm{ref}}(t) = z^m_{\mathrm{ref}}(t) + \int_{-\infty}^{t} (v_{\mathrm{ref}}(\tau) - v^m_{\mathrm{meas}}(\tau)) d\,\tau \tag{6.1}$$

effectively closing the loop. The assumption underlying this procedure is that our estimated system model $\hat{G}$ is a good representation of the real system G such that $\hat{G}^{-1}G \approx 1$.

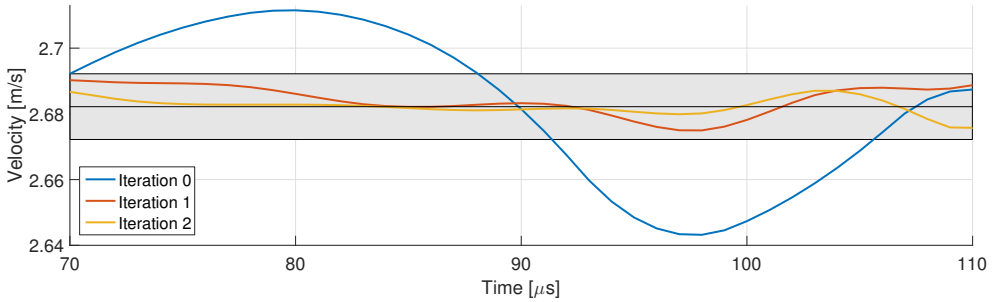
For our application, we made some slight adjustments to the update law 6.1 in order to deal with practical constraints. First, we can only reliably infer the velocity $v^m_{\mathrm{meas}}(t)$ while the ion is in the laser beam. We thus restrict ourselves to this region in time. Second, we keep the start and end positions fixed. Third we keep the time t_{center} at which the ion passes the exact center of the trap fixed by setting $z^{m+1}_{\mathrm{ref}}(t_{\mathrm{center}}) = z^m_{\mathrm{ref}}(t_{\mathrm{center}}) = 0$. Finally, for simplicity, we opt to only flatten out velocity fluctuations, but allow for a small constant offset to the original reference velocity.

The results of running update law 6.1 on two different days are shown in Figure 6.3a and Figure 6.3b. On both days, we observed clear improvements in the velocity profiles after one iteration. The initially substantial velocity deviations are much reduced. After further iterations the results are mixed. On one hand, iterations 2 and 3 did not result in additional improvements in Figure 6.3a . On the other hand, the velocity profile in Figure 6.3b converges to within the measurement uncertainty indicated by the area shaded in gray. However, despite the good results in Figure 6.3b, the pattern of initial improvements followed by stagnation (or even regression) from Figure 6.3a is much more representative of our experiences with this approach to ILC. So while the method improved the flatness of the velocity profile to some degree, it did not do so reliably, and there are usually still some remaining velocity fluctuations.

We have two possible explanations for this behavior. As outlined in chapter 3 the waveforms are calculated by optimizing a cost function with competing



(a) Data taken on 05.09.2015. While iteration 1 reduced the velocity fluctuations, iteration 2 and 3 led to no further improvements.



(b) Data taken on 19.10.2015. The procedure converged to within measurement uncertainty (± 0.01 m/s, indicated by the grey area) after two iterations.

Figure 6.3: Velocity profiles obtained with update law 6.1 on two different days.

objectives (e.g. not only the position $z_{\text{ref}}(t)$ is optimized, but also the frequency $\omega_z(t)$ and the offset $C(t)$ of the potential well) as well as constraints. The effects of changing the reference position on the velocity are therefore not straightforward and the validity of the assumption $\hat{G}^{-1}G \approx 1$ is thus questionable. Second, the experimental apparatus exhibited different behavior from day to day (as will be discussed in more detail in the next section). For the current approach, we always rely on the same, original system model. Drifts in the real system will thus not be accounted for, potentially causing problems.

6.3 Method 2: Directly modifying the waveforms

6.3.1 ILC with a quadratic cost criterion

Addressing the main weakness of our first approach to ILC naturally leads to the second method. The first approach critically depends on the accuracy of our model $\hat{G}$ of the complete system. With some courage, we just boldly hoped that $\hat{G}^{-1}G \approx 1$ is a good approximation, but never actually verified it. While we could perform system identification on the combined system $\hat{G}^{-1}G$, it would be needlessly complicated due to having two concatenated nonlinear

systems $\hat{G}^{-1}$ and G interacting in non-trivial ways, especially because the waveforms are calculated by solving an optimization problem. We therefore opt to instead identify G in isolation. Doing so then forces us to directly modify the voltage waveforms instead of the reference position over time.

In general, the transport process G is nonlinear. Identifying it for a general setting would therefore be very challenging. Luckily, we do not need a general system model for our purposes. Instead, we linearize around the transport trajectory, resulting in a linear time-varying system ΔG. We can then write the change in velocity $\Delta v \in \mathbb{R}^N$ as

$$\Delta v = \sum_i \Delta G_i \Delta u_i \tag{6.2}$$

where $\Delta G_i \in \mathbb{R}^{N \times N}$ is the linear time-varying model for electrode i , and $\Delta u_i \in \mathbb{R}^N$ is the change to control input u_i over time with N being the number of samples in time[2].

We can use our model ΔG to calculate the corrections Δu_i^m to the voltage waveforms u_i^m at iteration m. Roughly speaking, we want to use the corrections Δu_i^m to compensate for the error in velocity measured at iteration m, $e_{\text{meas}}^m = v_{\text{ref}} - v_{\text{meas}}^m$. Updating the voltage waveforms with

$$u_i^{m+1} = u_i^{m+1} + \Delta u_i^m \tag{6.3}$$

we can predict the error at the next iteration using equation 6.2 as

$$e_{\text{pred}}^{m+1} = e_{\text{meas}}^m - \sum_i \Delta G_i \Delta u_i^m. \tag{6.4}$$

Ideally, we would like to minimize the predicted error, while also keeping the corrections Δu_i^m small. We thus introduce a cost function

$$J^m = \|e_{\text{pred}}^{m+1}\|_{\mathbf{Q}}^2 + \sum_i \|\Delta u_i^m\|_{\mathbf{R}}^2 \tag{6.5}$$

capturing both these objectives. Here, $\mathbf{Q}$ and $\mathbf{R}$ are positive-definite weighting matrices. To determine the corrections Δu_i^m we minimize J^m with respect to them, i.e.

$$\Delta u_i^m = \arg \min_{\{\Delta u_i^m\}} J^m. \tag{6.6}$$

In practice, we have used a more elaborate cost function and introduced additional constraints to suit our application. The details are described in Appendix C.1. The approach outlined here is based on [Amann 95, Lee 00].

[2] In contrast to the first method, we treat the second ILC method in discrete-time, using k as the time index.

6.3.2 Derivation of the linear time-varying model

We now derive the linear time-varying model needed in update law 6.6. Here, we present a simplified but intuitive derivation. A more rigorous treatment can be found in Appendix C.2.

At any given time during transport, the ion is confined by an approximately harmonic potential well which is formed by a superposition of the individual electrodes, e.g.

$$V(z) \approx \frac{1}{2}\frac{m\omega_z^2}{q}\left(z - z_{\min}\right)^2 + C \approx \sum_i V_i(z) U_i \tag{6.7}$$

where we have dropped the time-dependence for convenience. If we now increase the voltage of electrode j by dU_j, we get an additional field $dV(z)$

$$dV(z) = V_j(z) dU_j \approx \left[V_j(z_{\min}) + V_j'(z_{\min})(z - z_{\min})\right] dU_j \tag{6.8}$$

where we expanded to first order around the ion position $z_{\min}$ in the undisturbed potential $V(z)$. We can then calculate the shift in the equilibrium position due to dU_j by finding the minimum of $V(z) + dV(z)$, i.e.

$$0 = \frac{d}{dz}\left(V(z) + dV(z)\right) = \frac{m\omega_z^2}{q}\left(z - z_{\min}\right) + V_j'(z_{\min}) dU_j \tag{6.9}$$

Solving for the new equilibrium position $z_{\min}^{\text{new}}$, we obtain

$$z_{\min}^{\text{new}} = z_{\min} - \frac{qV_j'(z_{\min}) dU_j}{m\omega_z^2}. \tag{6.10}$$

The small voltage dU_j thus shifts the equilibrium position by

$$dz = z_{\min}^{\text{new}} - z_{\min} = -\frac{qV_j'(z_{\min}) dU_j}{m\omega_z^2}. \tag{6.11}$$

There is a simple interpretation for this result. Originally, the ion is attached to a spring with a spring constant of $k = m\omega_z^2$, oscillating at frequency of ω_z around the equilibrium position $z_{\min}$. The change dU_j to the voltage on electrode j causes a force $dF = qV_j'(z_{\min}) dU_j$. This force then results in a small change $dz = -dF/k$ in the equilibrium position of the spring.

Now recall that we are primarily interested in changes in velocity dv rather than in position. To obtain dv, we calculate the derivative of dz with respect to time

$$dv = \frac{d(dz)}{dt} \approx -\frac{qV_j'(z_{\min})}{m\omega_z^2}\frac{d(dU_j)}{dt} \tag{6.12}$$

where we have neglected the time-dependence of $V_j'(z_{\min})$ and ω_z^2. We thus see that the change in velocity is proportional to the first derivative of the

control input change dU_j with respect to time, with the proportionality constant being time varying. By evaluating the proportionality constant for a specific trajectory $z_{\min}(t)$, we arrive at the linear time-varying model ΔG_i (see Appendix C.2 for the details).

6.3.3 Verification of the linear time-varying model

Before proceeding with ILC, we now compare the linear time-varying model we just derived from first principles with measurements on the real system.

In practice, we are mainly concerned with precisely transporting in the center of the trap, i.e. roughly from $z = -100\,\mu\text{m}$ to $z = 100\,\mu\text{m}$, as the laser beam is usually centered there. In this region, the main electrodes having a strong influence are number 7, 8 and 9, with electrode 8 providing most of the axial confinement (see e.g. Figure 3.3). We therefore opt to only use electrodes 7 and 9 for corrections, and thus only need to verify models ΔG_7 and ΔG_9 when the ion is in the laser beam.

To test our modeling, we measured the velocity of the ion at various times t_n first for the unperturbed base waveform, and then once more after adding a perturbation $dU_j(t)$. Subtracting the measured velocities gives us $dv(t_n)$ from equation 6.12. In order to make the results easy to interpret, we chose a perturbation dU_j with a constant slew rate μ_{slew}, centered around t_n. The resulting change in velocity $dv(t_n)$ is then approximately given by

$$dv(t_n) \approx -\frac{qV_j'(z_{\min}(t_n))}{m\omega_z(t_n)^2}\mu_{\text{slew}} \tag{6.13}$$

where j stands for electrode 7 or 9. We can thus compare the difference in velocity $dv(t_n)$ measured on the real system to the one predicted by equation 6.13 at various times t_n.

The measured and predicted changes in velocity are shown in Figure 6.4. For electrode 9 the measurements (green markers) and the predicted values (solid green line) agree well, except for $t = 62\,\mu\text{s}$. For electrode 7, the measurements (blue markers) and the predicted values (solid blue line) agree well only for $t > 80\,\mu\text{s}$. For $t < 80\,\mu\text{s}$ the measured and predicted values differ considerably. This is quite surprising because electrodes 7 and 9 are both identical in shape, with electrode 7 being located to the left of the trap center and electrode 9 to the right. We therefore would have expected them to show the same behavior in their respective regions of influence. For the time being, we therefore opted to only use electrode 9 for corrections. Furthermore, there also seem to be drifts from day to day, as repeating the measurements on different days gave different results. Currently, we have no explanation for the model mismatch and the drifts and can only speculate about possible reasons. Performing additional investigations to identify the underlying causes would seem worthwhile.

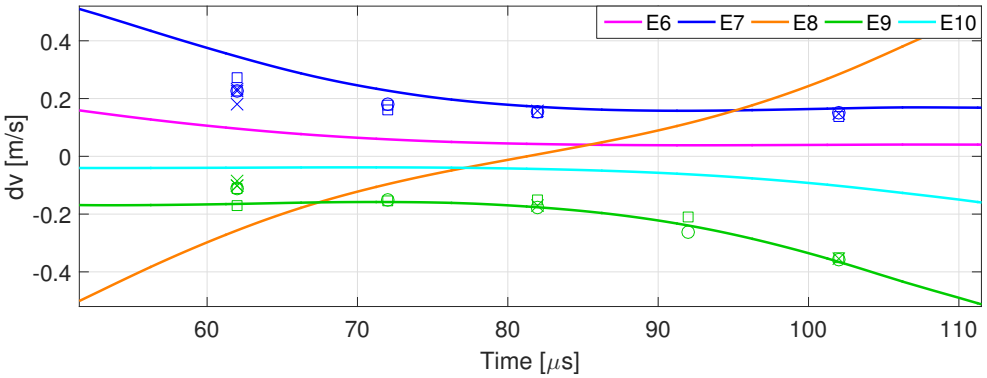


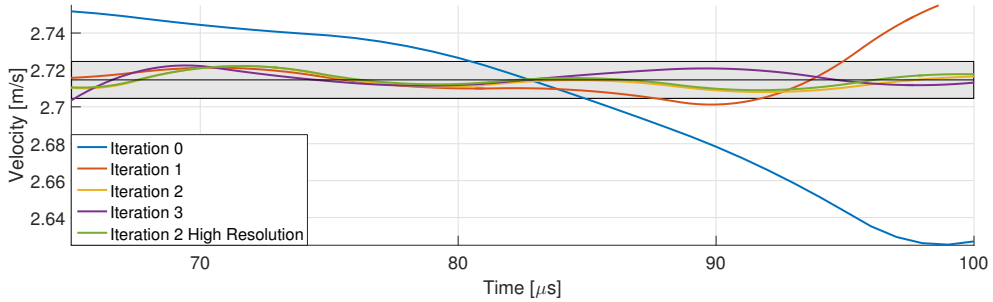
Figure 6.4: Measured change in velocity (markers □/○/×) against predicted values (solid lines) for electrodes 7 and 9 at various times. The measurements were repeated on three different days, each corresponding to a different marker type. Also shown are the predictions for electrodes 6, 8 and 10 in order to illustrate their respective strengths.

The observed model mismatch and drifts are not much of a problem for ILC though. While the predicted and measured velocity changes do not match perfectly, at least the sign is always correct. In practice, we can deal with inaccurate models by decreasing the step size of the corrections. For example, instead of trying to correct the complete error e^m_{meas} when going from iteration m to $m+1$, we can opt to only correct part of it. This can easily be accomplished by choosing appropriate weighting matrices $\mathbf{Q}$ and $\mathbf{R}$ in 6.5 such that the changes Δu_i^m are strongly penalized. Furthermore, the drifts pose no problems as long as they occur on a slow timescale compared to the ILC iterations, which is currently the case.

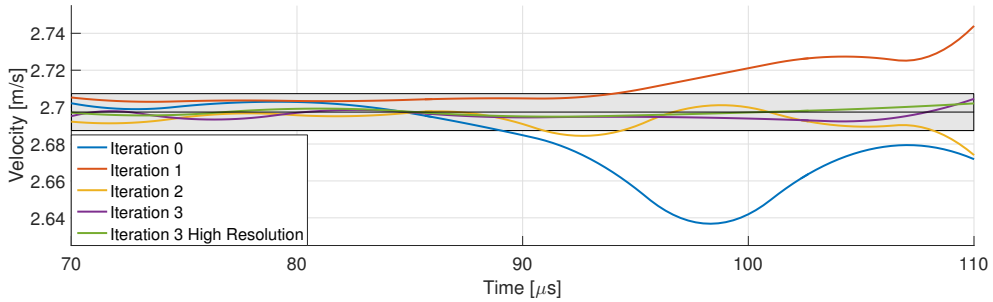
6.3.4 Results

The results upon applying update law 6.3 on two different days are shown in Figure 6.5. In both cases the results are very similar. After one iteration, the velocity is already much improved, although towards the end of the time window the corrections overshoot, e.g. leading from too slow to too fast velocities. In contrast to the first approach to ILC, the next few iterations then consistently lead to further improvements. Within two to three iterations, we reliably converge to within the measurement uncertainty. From then on, we do not have a reliable error signal $e^m_{\text{meas}}(t)$ anymore and thus stop the procedure.

The resulting spin evolution after three iterations is shown in Figure 6.6. Note that the features are very narrow in the laser detuning δ_L, covering only roughly 200 kHz, indicating that the velocity varies little. This is in stark contrast to e.g. Figure 4.1b where the detuning varied by roughly 1.5 MHz.



(a) Data taken on 08.10.2015. The velocity after iteration 2 was remeasured using a higher resolution for the spin dynamics.



(b) Data taken on 19.10.2015. The velocity after iteration 3 was remeasured using a higher resolution for the spin dynamics.

Figure 6.5: Velocity profiles obtained with update law 6.3. The area shaded in grey indicates the uncertainty in the velocity measurements (± 0.01 m/s). On both days, the procedure converged to within measurement uncertainty after 3 iterations.

Second, the spin evolution is almost symmetric with respect to the horizontal axis. This also indicates that the velocity fluctuations are very small. In the limiting case of a truly constant velocity, the data would be perfectly symmetric.

Finally, Figure 6.7 shows the original voltage waveform from iteration 0, as well as the final waveform from iteration 3 from Figure 6.5b. The absolute changes to the voltage of electrode 9 are very small compared to the overall voltages. Linearizing around a particular trajectory is thus well justified.

6.4 Discussion and Outlook

Overall, ILC was successfully used to reduce velocity fluctuations during transport from about 0.1 m/s down to roughly 0.01 m/s. The first approach produced mixed results, improving the velocity only to some degree. The second approach reliably reduced the velocity fluctuations down to within the measurement accuracy in a few iterations.

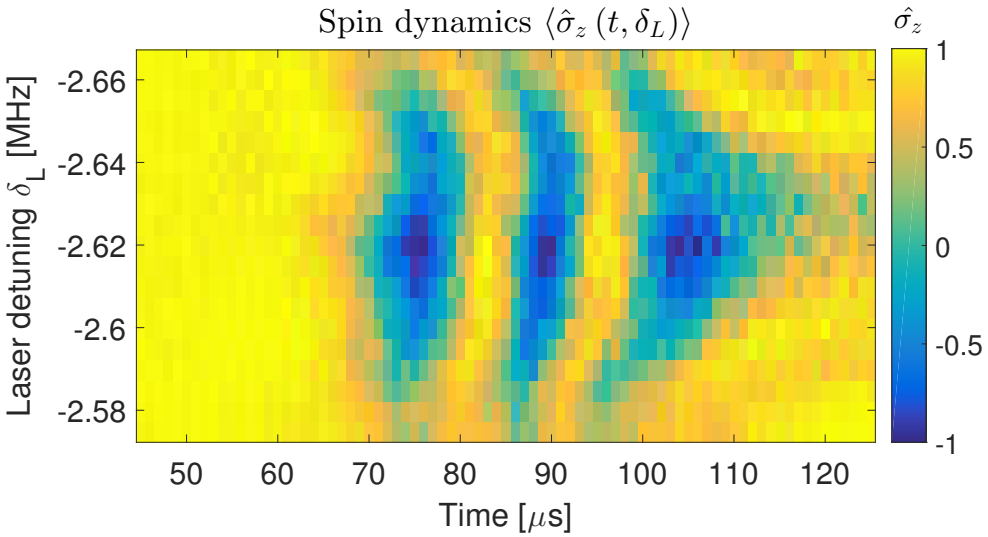


Figure 6.6: Spin dynamics obtained after iteration 3 on 19.10.2015.

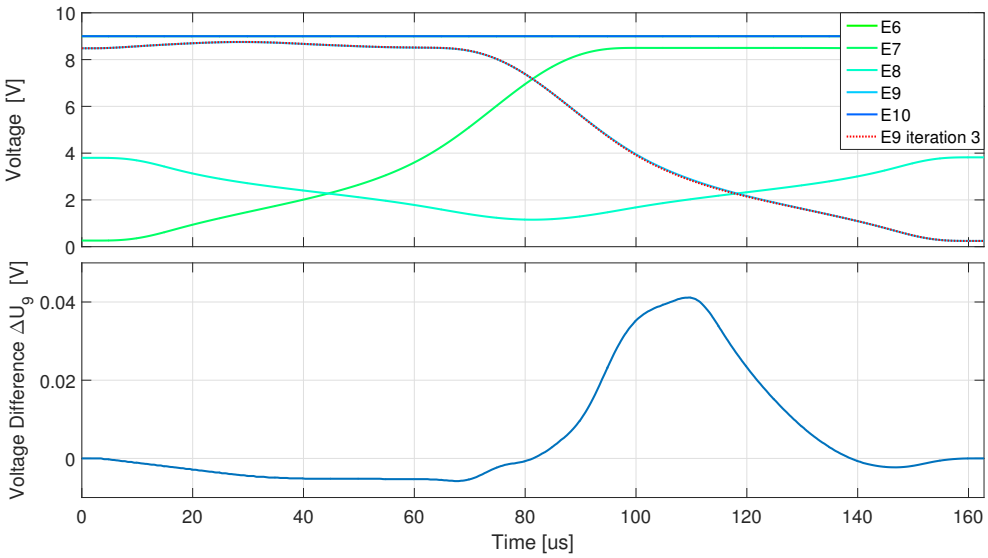


Figure 6.7: Top: Initial waveform from iteration 0 together with the waveform from iteration 3 from 19.10.2015. Note that only the input to electrode 9 was modified. Bottom: Overall changes to the voltage of electrode 9 from iteration 0 to iteration 3.

Currently, the main limitation is not the ILC technique, but rather the accuracy of the velocity measurement. As soon as one is unable to reliably infer an error signal, ILC cannot be used to make further progress. There are two factors limiting the accuracy of the velocity measurement. First, for some experimental settings our model can not accurately reproduce the spin dynamics at the very beginning. Second, choosing the number of fitting parameters is challenging. When using many parameters one obtains very fine features in the velocity profile. However, it is hard to tell if they are real or the result of overfitting. Using cross-validation could help in this regard. If those two problems can be overcome, we expect further improvements to the flatness of the velocity, as simulations show that we should be able to measure the velocity more accurately.

The improved control over the velocity also revealed new aspects of the experimental apparatus. First, we observed day-to-day fluctuations. Sometimes the measured results changed over the timescale of hours, but at other times the setup was stable over the course of a full day. Second, while there generally is a good agreement between model and measurement, we observed a discrepancy between the measured and expected effectiveness of electrodes 7 and 9 left of the trap center. Now that our control of the velocity is good enough for these effects to become significant, it would be interesting to study them in more detail and then also get them under control.

While ILC currently is not the limiting factor, there are some possibilities for future improvements. First, one could use the data between iterations to perform on-the-fly system identification as in [Rezaeizadeh 15]. Second, one could further adapt the method to the constraints posed by quantum mechanics. Specifically, we rely on repeated measurements on identically prepared systems. One could therefore correct the velocity profile in a progressive way from the start to the end. This could reduce the time spent measuring, as one then only has to measure at a specific time rather than over the whole time horizon. Currently, the time spent measuring is not a limiting factor though. Third, with the recent introduction of scripting to our control software, we could automate much of what is currently done manually.

Chapter 7

Conclusions and Outlook

Transport of ions is a key component of the proposed QCCD architecture for building a scalable ion trap quantum computer. Transport quantum logic gates are another promising building block as they shift the burden of control from the optical to the electrical domain, which has an excellent track record when it comes to scaling. By controlling the velocity of an ion during transport, we can control the time it is in a laser beam as well as the detuning of the laser as seen by the ion due to a Doppler shift. This thesis investigated how to implement ion transport in practice, paying special attention to the control of the velocity over time.

Quadratic programming was used to calculate feedforward control signals for adiabatic ion transport. The strength of this approach is that it can easily handle various practical constraints. We also expect it to be useful e.g. for splitting of ions. However, in the end, the experimental results will always be limited by the accuracy of the underlying model. Because models are never perfect, iterative learning control was later used to close the gap between predicted and actual performance.

In order to control something, one first needs to be able to measure it. To this end, we devised experimental methods and associated analysis techniques to reconstruct a time-dependent Hamiltonian from experimental data, allowing us to infer the ion's velocity. The method only requires being able to quickly turn off the dynamics, and to add a controlled perturbation to the Hamiltonian. We thus expect it to be useful for many different physical systems. One of its key strengths is the experimental simplicity, as it only requires measurements in a single basis. However, we do not expect it to scale to more complicated quantum systems, as it fundamentally relies on being able to rapidly simulate the quantum dynamics, one of the original motivations for building a quantum computer.

Control of the velocity during transport was improved, reducing velocity fluc-

tuations from the order of 1 m/s down to roughly 0.01 m/s at a constant transport speed of 2.8 m/s. Finding and eliminating a flaw in the control electronics contributed roughly a factor of 10. Another factor of 10 came from iteratively updating the feedforward control inputs using iterative learning control. The current limiting factor is the uncertainty of the velocity measurements, but further improvements seem possible.

Precise control over the velocity enables follow-up investigations. First, randomized benchmarking could be used to assess the fidelity of single-qubit transport gates. Second, it would be interesting to see if one can also carry out two-qubit gates solely using transport, e.g. implementing a transport version of a Mølmer-Sørensen gate. However, doing so might also require stabilizing the axial trapping frequency during transport, which one could also do with ILC.

We expect iterative learning control to be a useful tool in the quest for better control over quantum systems. Particularly when time-varying control inputs are used, calibration becomes difficult. Furthermore, quantum mechanics puts strict limitations on what one can measure, with measurements influencing the quantum state. One therefore often relies on repeated measurements on identically prepared systems. In this context, iterative learning control seems to be well suited for optimizing time-varying feedforward control inputs directly on the experimental apparatus. ILC could also be further adapted to the constraints posed by the control of quantum systems; doing so would seem worthwhile

Appendix A

List of Abbreviations and Symbols

Abbreviations

AOM	Acousto-Optic Modulator
AWG	Arbitrary Waveform Generator
DEATH	Direct Ethernet Adjustable Transport Hardware
EHE	Extending Horizon Estimation
ILC	Iterative Learning Control
LTI	Linear Time-Invariant
LTV	Linear Time-Varying
MHE	Moving Horizon Estimation
MPC	Model Predictive Control
RF	Radio Frequency
QCCD	Quantum Charge-Coupled Device
QP	Quadratic Program

Symbols

Indices

i	Electrode index
k	Discrete-time sample index
m	Iteration index

Physics

δ	Detuning
ω_x, ω_y, ω_z	Trap frequency along a given axis
Ω	Rabi frequency

$\vec{k}$	Laser wave vector
$\hat{H}$	Hamiltonian
$\hat{\sigma}_x$, $\hat{\sigma}_y$, $\hat{\sigma}_z$	Pauli matrices
$\|\Psi\rangle$	Wave function
$\|0\rangle, \|1\rangle$	Energy eigenstates of a two level system

Transport

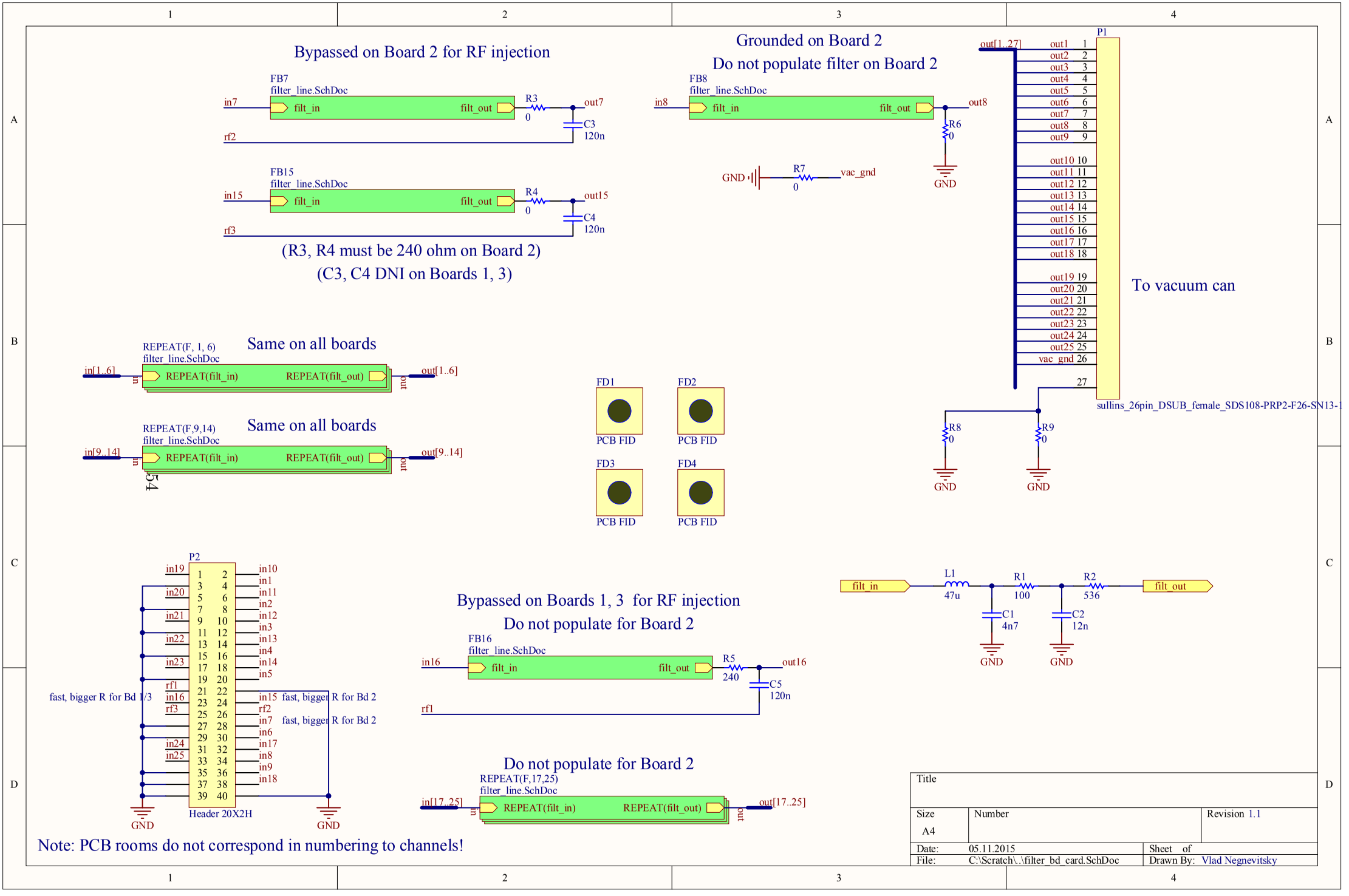
$V(z,t)$	Voltage along the trap axis z at time t
$V_i(z)$	Voltage along the trap axis z due to a unit voltage on electrode i
$U_i^{\text{awg}}(t)$	Voltage at the output of the AWG controlling electrode i
$U_i(t)$	Voltage on trap electrode i at time t
$u_i(k)$	Voltage on trap electrode i at sample index k
$\Delta u_i(k)$	Change to the voltage on trap electrode i at sample index k
$z(t)$	Position along transport axis z

Mathematics

Δ_k	Finite difference operator with respect to index k
du	Small change to continuous-time quantity u
Δu	Small change to discrete-time quantity u

Appendix B

Schematic of new external Filterboards


Bypassed on Board 2 for RF injection
FB7
filter_line.SchDoc
in7
filt_in
filt_out
R3
0
out7
C3
120n
rf2
FB15
filter_line.SchDoc
in15
filt_in
filt_out
R4
0
out15
C4
120n
rf3
(R3, R4 must be 240 ohm on Board 2)
(C3, C4 DNI on Boards 1, 3)
Grounded on Board 2
Do not populate filter on Board 2
FB8
filter_line.SchDoc
in8
filt_in
filt_out
out8
R6
0
GND
R7
0
vac_gnd
GND
P1
out[1..27]
out1 1
out2 2
out3 3
out4 4
out5 5
out6 6
out7 7
out8 8
out9 9
out10 10
out11 11
out12 12
out13 13
out14 14
out15 15
out16 16
out17 17
out18 18
out19 19
out20 20
out21 21
out22 22
out23 23
out24 24
out25 25
vac_gnd 26
27
sullins_26pin_DSUB_female_SDS108-PRP2-F26-SN13-1
To vacuum can
R8
0
R9
0
GND
GND
Same on all boards
REPEAT(F, 1, 6)
filter_line.SchDoc
in[1..6]
REPEAT(filt_in)
REPEAT(filt_out)
out[1..6]
Same on all boards
REPEAT(F,9,14)
filter_line.SchDoc
in[9..14]
REPEAT(filt_in)
REPEAT(filt_out)
out[9..14]
54
FD1
PCB FID
FD2
PCB FID
FD3
PCB FID
FD4
PCB FID
P2
in19
in20
in21
in22
in23
rf1
in16
rf3
in24
in25
1 2
3 4
5 6
7 8
9 10
11 12
13 14
15 16
17 18
19 20
21 22
23 24
25 26
27 28
29 30
31 32
33 34
35 36
37 38
39 40
in10
in1
in11
in2
in12
in3
in13
in4
in14
in5
in15 fast, bigger R for Bd 2
rf2
in7 fast, bigger R for Bd 2
in6
in17
in8
in9
in18
fast, bigger R for Bd 1/3
Header 20X2H
GND
GND
Bypassed on Boards 1, 3 for RF injection
Do not populate for Board 2
FB16
filter_line.SchDoc
in16
filt_in
filt_out
R5
240
out16
C5
120n
rf1
Do not populate for Board 2
REPEAT(F,17,25)
filter_line.SchDoc
in[17..25]
REPEAT(filt_in)
REPEAT(filt_out)
out[17..25]
filt_in
L1
47u
R1
100
R2
536
filt_out
C1
4n7
C2
12n
GND
GND
Note: PCB rooms do not correspond in numbering to channels!
Title
Size
A4
Number
Revision 1.1
Date: 05.11.2015
Sheet of
File: C:\Scratch\..\filter_bd_card.SchDoc
Drawn By: Vlad Negnevitsky

Appendix C

Iterative Learning Control Details

C.1 Adjustments to the cost function and additional constraints

Here we present the modifications we made to the cost function J^m from equation 6.5 in order to adapt it to our application. First, we introduced additional cost terms, arriving at

$$\begin{aligned} J^m =& \|e_{\text{pred}}^{m+1}\|_{\mathbf{Q}}^2 + r_{\text{off}} \| \sum_i V_i(z_{\min}(k)) \Delta u_i^m \|_{\mathbf{R}}^2 \\ &+ \sum_i \left[r_0 \|\Delta u_i^m\|_{\mathbf{R}}^2 + r_1 \|\Delta_k \Delta u_i^m\|_{\mathbf{R}}^2 + r_2 \|\Delta_k^2 \Delta u_i^m\|_{\mathbf{R}}^2 \right] \end{aligned} \tag{C.1}$$

where Δ_k stands for the finite difference operator in time.

The term with the weighting factor r_{off} minimizes changes to the offset of the potential well. The terms prefaced by r_0, r_1 and r_2 punish the changes Δu_i^m as well as their first and second derivatives. This results in small and smooth changes Δu_i^m, the latter of which is important to prevent distortion of the corrections by the filters in the time domain. Furthermore, the term penalizing the first derivative controls how much of the error e_{meas}^m gets corrected by Δu_i^m, because velocity changes are proportional to the derivative of the control input changes with respect to time[1].

Furthermore, we also added constraints. To enforce the absolute voltage limits on the control inputs, we set

$$U_{\min} \leq u_i^m + \Delta u_i^m \leq U_{\max} \tag{C.2}$$

for all electrodes i. We could easily also incorporate slew rate constraints, but there currently is no need to do so because we are far from the limit.

[1] This result is derived intuitively in section 6.3.2 and rigorously in Appendix C.2.

Second, we want to keep the time at which the ion passes through the exact center of the trap ($z = 0\,\mu m$) constant in order to have a fixed reference point. Denoting the sample at which the ion is in the middle by k_{mid}, we can write this constraint as

$$\sum_i k_i(k_{\text{mid}})\Delta u_i^m(k_{\text{mid}}) = 0 \tag{C.3}$$

where $k_i(k_{\text{mid}})$ is the proportionality constant between the change in position Δz and the change in control input Δu_i^m as introduced in equation C.13 below. Finally, in order to keep the start and end position of the transport fixed, we constrain the changes Δu_i^m to be equal to zero for a fixed number of samples at the start and end of the waveform.

The choice of the weighting matrices $\mathbf{Q}$ and $\mathbf{R}$ is also important. We primarily care about the velocity of the ion while it is in the beam, which is also when we can actually measure it. We therefore use a window corresponding to the time the ion is in the beam as the matrix $\mathbf{Q}$ weighing the predicted error. For the matrix $\mathbf{R}$ weighing the control effort over time, we also use a similar time window. However, we still want to keep the changes to the control inputs small even at times when the ion is not in the beam. We therefore only reduce the weights outside the beam, but do not set them to zero.

In order to handle the time delay introduced by the lowpass filters, we shift the measured velocity error accordingly in time. For example, if we have a time delay of t_d and originally measure the velocity error $e^m_{\text{orig}}(t)$, we set $e^m_{\text{meas}}(t) = e^m_{\text{orig}}(t - t_d)$.

The implementation was done in Matlab. We write the cost function and the constraints as a quadratic program, being careful to use sparse matrices. We can then carry out the minimization and calculate the changes to the control inputs by handing the quadratic program to a solver such as [Gurobi 15].

Sample results of the calculation process are presented in Figure C.1. Assuming we can reliably infer the velocity between $t = 50\,\mu s$ to $t = 110\,\mu s$, corresponding to the time the ion is in the beam, we ramp the measured velocity error before and afterwards. We then use e^m_{meas} to calculate the changes to the control inputs by minimizing the cost function given in equation C.1 subject to the constraints outlined above. Because we only used electrode 9 for corrections in the experiment, we also restrict ourselves to electrode 9 here. We see that the derivative of Δu_9^m varies depending on the measured error. This counteracts the measured velocity error, resulting in a much lower predicted error. When the measured error is zero around $t = 88\,\mu s$, the derivative of Δu_9^m is also zero. For this example, the ion is expected to be in the middle of the trap at $t_{\text{mid}} = 81.5\,\mu s$. Because we only use electrode 9 for corrections, $\Delta u_9^m(t_{\text{mid}})$ is zero in order to satisfy constraint C.3. Finally, we note that the predicted error is nonzero at the beginning and towards the end, i.e. when the ion is not in the beam. This is due to the corrections Δu_9^m having to

ramp down to zero at the start and at the end, resulting in velocity changes. However, we do not mind those velocity changes, as they happen outside the beam.

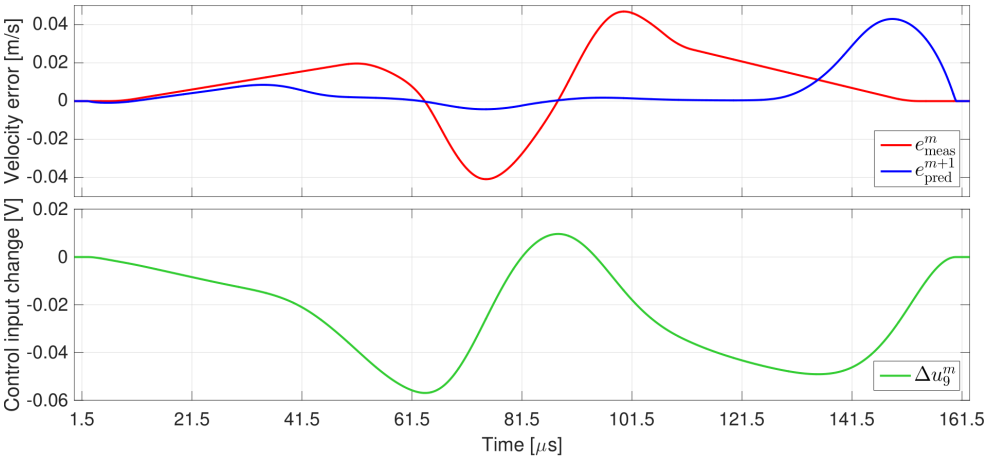


Figure C.1: Illustration of the calculation of the changes to electrode voltage 9 for a sample velocity error. The top shows the measured velocity error as well as the predicted velocity error at the next iteration upon applying the corrections Δu_9^m shown in the bottom half of the Figure.

C.2 Detailed derivation of the linear time-varying model

Here, we derive the linear time-varying models ΔG_i by linearizing the general process model G from chapter 3 around a particular trajectory of the ion. First, we recall that in adiabatic transport the ion is always located at the minimum of the confining potential $V(z,t)$. We can write the position $z_{\min}(t)$ of the ion at time t as

$$z_{\min}(t) = \arg\min_z V(z,t). \tag{C.4}$$

The confining potential $V(z,t)$ is the result of a superposition of the spatial potentials $V_i(z)$ due to the individual electrodes, weighted by the electrode voltages $U_i(t)$ (see equation 3.2):

$$V(z,t) = \sum_i V_i(z) U_i(t) \tag{C.5}$$

To perform the minimization in equation C.4, we take the derivative with respect to z and set it to zero, resulting in

$$0 = \sum_i V_i'(z_{\min}(t)) U_i(t) \tag{C.6}$$

which implicitly defines $z_{\min}(t)$. For general $V_i'(z)$ we can not explicitly solve for $z_{\min}(t)$. Luckily, for our current task of linearizing around the trajectory $z_{\min}(t)$ there is no need to do so. Instead of solving for the ion position $z_{\min}(t)$ for given electrode voltages $U_i(t)$, we want to determine the small change in position $dz(t)$ caused by small changes $dU_i(t)$ to the electrode voltages $U_i(t)$. Modifying equation C.6 and dropping the time argument, we get

$$0 = \sum_i V_i'(z_{\min} + dz)\,(U_i + dU_i) \tag{C.7}$$

$$\approx \sum_i \left(V_i'(z_{\min}) + dz V_i''(z_{\min})\right)(U_i + dU_i) \tag{C.8}$$

$$\approx \sum_i \left[V_i'(z_{\min})U_i + V_i'(z_{\min})dU_i + dz V_i''(z_{\min})\,(U_i + dU_i)\right] \tag{C.9}$$

$$\approx \sum_i \left[V_i'(z_{\min})dU_i + dz V_i''(z_{\min})\,(U_i + dU_i)\right] \tag{C.10}$$

where we first expanded $V_i'(z)$ in a Taylor series around $z_{\min}$, then multiplied out the terms, and finally used equation C.6 to eliminate the term $V_i'(z_{\min})U_i$. Solving for dz and approximating $U_i + dU_i$ as U_i, we obtain

$$dz \approx \frac{-\sum_i V_i'(z_{\min})dU_i}{\sum_i V_i''(z_{\min})U_i} \tag{C.11}$$

Next, we re-introduce the time argument and bring the result into a more readable form

$$dz(t) \approx \sum_i k_i(t) dU_i(t) \tag{C.12}$$

by introducing

$$k_i(t) = \frac{-V_i'(z_{\min}(t))}{\sum_i V_i''(z_{\min}(t))U_i(t)}. \tag{C.13}$$

We thus see that the small changes dU_i to the electrode voltages U_i lead to a small displacement dz. So far, we have worked in continuous-time. We now switch to discrete-time by replacing time t with sample index k, $dz(t)$ with $\Delta z(k)$, $k_i(t)$ with $k_i(k)$ and $dU_i(t)$ with $\Delta u_i(k)$ in accordance to our definitions in equation 6.2. We thus have

$$\Delta z(k) \approx \sum_i k_i(k)\Delta u_i(k). \tag{C.14}$$

Finally, we recall that we were originally interested in the change in velocity $\Delta v(k)$ rather than the change in position $\Delta z(k)$. To obtain the change in velocity, we can simply use a finite difference approximation. Using e.g. first central differences we obtain

$$\Delta v(k) \approx \frac{\Delta z(k+1) - \Delta z(k-1)}{2T_s} \tag{C.15}$$

$$\approx \frac{\sum_i k_i(k+1)\Delta u_i(k+1) - \sum_i k_i(k-1)\Delta u_i(k-1)}{2T_s} \tag{C.16}$$

where T_s stands for the sampling time. By comparing equations 6.2 and C.16, we can read off the matrices ΔG_i representing the linear time-varying models obtained after linearizing around a particular trajectory $\{z_{\min}(t), U_i(t)\}$.

Appendix D

arXiv:1509.07083 Time-dependent Hamiltonian estimation for Doppler velocimetry of trapped ions

Time-dependent Hamiltonian estimation for Doppler velocimetry of trapped ions

L. E. de Clercq, R. Oswald, C. Flühmann, B. Keitch, D. Kienzler,
H.-Y. Lo, M. Marinelli, D. Nadlinger, V. Negnevitsky, J. P. Home
Institute for Quantum Electronics, ETH Zürich, Otto-Stern-Weg 1, 8093 Zürich, Switzerland

The time evolution of a closed quantum system is connected to its Hamiltonian through Schrödinger's equation. The ability to estimate the Hamiltonian is critical to our understanding of quantum systems, and allows optimization of control. Though spectroscopic methods allow time-independent Hamiltonians to be recovered, for time-dependent Hamiltonians this task is more challenging [1–6]. Here, using a single trapped ion, we experimentally demonstrate a method for estimating a time-dependent Hamiltonian of a single qubit. The method involves measuring the time evolution of the qubit in a fixed basis as a function of a time-independent offset term added to the Hamiltonian. In our system the initially unknown Hamiltonian arises from transporting an ion through a static, near-resonant laser beam [7]. Hamiltonian estimation allows us to estimate the spatial dependence of the laser beam intensity and the ion's velocity as a function of time. This work is of direct value in optimizing transport operations and transport-based gates in scalable trapped ion quantum information processing [8–10], while the estimation technique is general enough that it can be applied to other quantum systems, aiding the pursuit of high operational fidelities in quantum control [11, 12].

Estimation of the underlying dynamics which drive the evolution of systems is a key problem in many areas of physics and engineering. This knowledge allows control inputs to be designed which account for imperfections in the physical implementation. For closed quantum systems, the time dependence of a system is driven by the Hamiltonian through Schrödinger's equation. If the Hamiltonian is static in time, a wide range of techniques have been proposed for recovering the Hamiltonian [1–3, 13], which have been applied to a variety of systems including chemical processes [4] and quantum dots [5, 6]. These methods often involve estimation of the eigenvectors and eigenvalues of the Hamiltonian via spectroscopy, or through pulse-probe techniques for which a Fourier transform of the time-evolution gives information about the spectrum. However these methods are not directly applicable to time-dependent Hamiltonians. Such Hamiltonians are becoming of increasingly important as quantum engineering pursues a combination of high operational fidelities and speed, often involving fast variation of control fields which are particularly susceptible to distortion before reaching the quantum device [9, 10, 14–17].

In this Letter, we propose and demonstrate a method for reconstructing a general time-dependent Hamiltonian with two non-commuting terms which drives the evolution of a single qubit. The method works with any single qubit Hamiltonian $\hat{H} = \sum_i f_i(t)\hat{\sigma}_i$, where the $f_i(t)$ are arbitrary time-dependent functions and $\hat{\sigma}_i$ are the Pauli operators. In our experiments, a Hamiltonian with two non-commuting time-dependent terms arises when we try to perform quantum logic gates by transporting an ion through a static laser beam [7, 18]. In this case, the Hamiltonian describing the interaction between the ion and the laser can be written in an appropriate rotating frame as

$$\hat{H}_I(t) = \frac{\hbar}{2}\left(-\Omega(t)\hat{\sigma}_x + \delta(t)\hat{\sigma}_z\right) \tag{1}$$

which includes a time-varying Rabi frequency $\Omega(t)$, and an effective detuning $\delta(t)$ which is related to the first-order Doppler shift of the laser in the rest frame of the moving ion (see [19] for details). For a Hamiltonian of this type with unspecified time-dependent coefficients, no analytical solution to Schrödinger's equation exists [20, 21]. In order to reconstruct the Hamiltonian we make use of two additional features of our experiment. The first is that we can switch off the Hamiltonian at time t_{off} on a timescale which is fast compared to the evolution of the qubit. Secondly we are able to offset one of the terms in the Hamiltonian, in our case by adding a static detuning term $\hat{H}_{\text{s}} = \hbar\delta_L\hat{\sigma}_z/2$ such that the total Hamiltonian is $\hat{H}_I(t) + \hat{H}_s$. We then measure the expectation value of the qubit in the $\hat{\sigma}_z$ basis as a function of δ_L and t_{off}. Repeating the experiment with identical settings many times, we obtain an estimate of the expectation value which we denote as $\langle\hat{\sigma}_z^{\text{meas}}(t_{\text{off}}, \delta_L)\rangle$.

Hamiltonian extraction involves theoretically generating the qubit populations $\langle\hat{\sigma}_z^{\text{sim}}(t_{\text{off}}, \delta_L)\rangle$, and attempting to find the Hamiltonian for which this most closely matches the data. In order to provide a simple parameterization, we represent $\delta(t)$ and $\Omega(t)$ as a linear weighted combination of basis splines [22, 23]. $\langle\hat{\sigma}_z^{\text{sim}}(t_{\text{off}}, \delta_L)\rangle$ is compared to the measured data using a weighted least-squares cost function, which we optimize with respect to the weights of the basis-splines used to parameterize $\delta(t)$ and $\Omega(t)$. Solving this optimization problem in general is hard because the cost function is subject to strong constraints imposed by quantum mechanics, producing a non-trivial relation between the weights and the spin populations [19]. We overcome this problem by making use of the inherent causality of the quantum-mechanical evolution, and by assuming that the parameters of the Hamiltonian vary smoothly. We call our technique "Extending the Horizon Estimation", in analogy to estab-

lished methods in engineering [24] (a detailed description of our method can be found in [19]). Rather than optimizing over the whole data set at once, we build up the solution by initially fitting the data over a limited region of time $0 < t_{\rm off} < T_0$. The solution obtained over this first region can be extrapolated over a larger time span $0 < t < T_1$ where $T_1 = T_0 + \tau$, which we use as a starting point to find an optimal solution for this extended region. This procedure is iterated until $T_{n_{\rm max}} = \max(t_{\rm off})$. The method allows us to choose a reduced number of basis spline functions to represent $\delta(t)$ and $\Omega(t)$, and also reduces the amount of data considered in the early stages of the fit, when the least is known about the parameters. This facilitates the use of non-linear minimization routines, which are based on local linearization of the problem and converge faster near the optimum. More details regarding the optimization routine can be found in [19].

In the experimental work, we demonstrate reconstruction of the spin Hamiltonian for an ion transported through a near-resonant laser beam. Our qubit is encoded in the electronic states of a trapped calcium ion, which is defined by $|0\rangle \equiv \left|^2S_{1/2}, M_J = 1/2\right\rangle$ and $|1\rangle \equiv \left|^2D_{5/2}, M_J = 3/2\right\rangle$. This transition is well resolved from all other transitions, and has an optical frequency $\omega_0/(2\pi) \simeq 411.0420$ THz. The laser beam points at 45 degrees to the transport axis, and has an approximately Gaussian spatial intensity distribution. The time-dependent velocity $\dot{z}(t)$ of the ion is controlled by adiabatic translation of the potential well in which the ion is trapped. This is implemented by applying time-varying potentials to multiple electrodes of a segmented ion trap, which are generated using a multi-channel arbitrary waveform generator, each output of which is connected to a pair of electrodes via a passive third order low-pass Butterworth filter. The result is that the ion experiences a time-varying Rabi frequency $\Omega(t)$ and a laser phase which varies with time as $\Phi(t) = \phi(z(t)) - \omega_L t$, where $\phi(z(t)) = k_z(z(t))z(t)$ with $k_z(z(t))$ the laser wavevector projected onto the transport axis at position $z(t)$ and ω_L the laser frequency. The spatial variation of $k_z(z(t))$ accounts for the curvature of the wavefronts of the Gaussian laser beam. In order to create a Hamiltonian of the form of equation 1, we work with the differential of the phase, which gives a detuning $\delta(t) = \delta_L - \dot{\phi} = (k_z'(z)z + k_z(z))\,\dot{z}$ with $\delta_L = \omega_L - \omega_0$ the laser detuning from resonance. For planar wavefronts $k_z'(z) = 0$, and $\delta(t)$ corresponds to the familiar expression for the first-order Doppler shift (see [19] for details).

The experimental sequence is depicted in figure 1. We start by cooling all motional modes of the ion to $\bar{n} < 3$ using a combination of Doppler and electromagnetically-induced-transparency cooling [25], and then initialize the internal state by optical pumping into $|0\rangle$. The ion is then transported to zone A, and the laser beam used to implement the Hamiltonian is turned on in zone B. The ion is then transported through this laser beam to zone C. During the passage through the laser beam, we rapidly

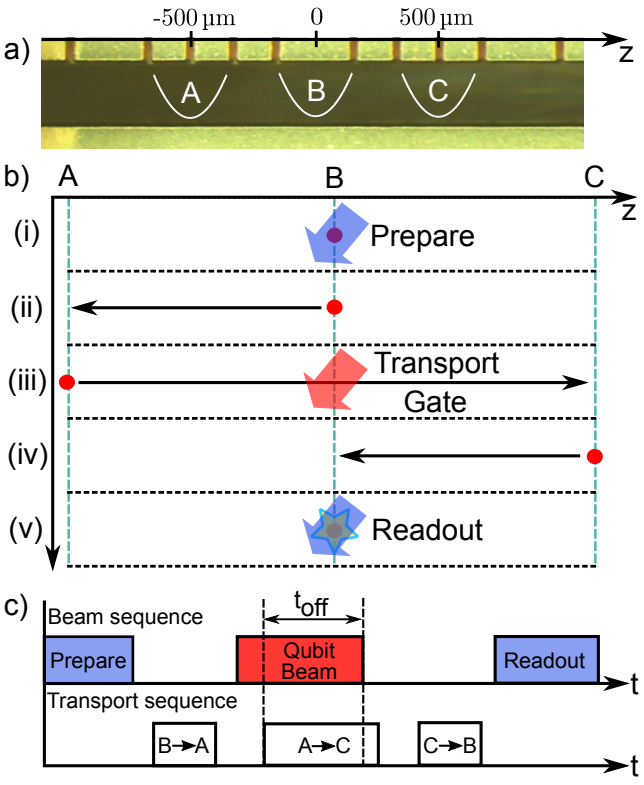


FIG. 1: **Experimental sequence and timing:** a) The experiment is carried out in three zones of the trap indicated by A, B and C. b) The experimental sequence involves steps (i) through (v). Preparation and readout are carried out on the static ion in zone B. The qubit evolves while the ion is transported from zone A to zone C, via the laser beam in zone B. c) Experimental sequence showing the timing of applied laser beams and ion transport, including shutting off the laser beam during transport.

turn the beam off at time $t_{\rm off}$ and thus stop the qubit dynamics. The ion is then returned to the central zone B in order to perform state readout, which measures the qubit in the computational basis (for more details see [19]). The additional Hamiltonian $\hat{H}_{\rm s}$ is implemented by offsetting the laser frequency used in the experiment by a detuning δ_L. For each setting of $t_{\rm off}$ and δ_L the experiment is repeated 100 times, allowing us to obtain an estimate for the qubit populations $\langle \hat{\sigma}_z\,(t_{\rm off}, \delta_L)\rangle$.

Experimental data is shown in figure 2 for two different beam positions, alongside the results of fitting performed using our iterative method. The beam positions used for each data set differ by around 64 µm, but the transport waveform used was identical. The reconstructed velocities should therefore agree in the region where the data overlap. It can be seen from the residuals that the estimation is able to find a Hamiltonian which results in a close match to the data. In order to get an estimate of the relevant error bars for our reconstruction, we have performed non-parametric resampling with replacement, optimizing for the solution using the same set of B-spline

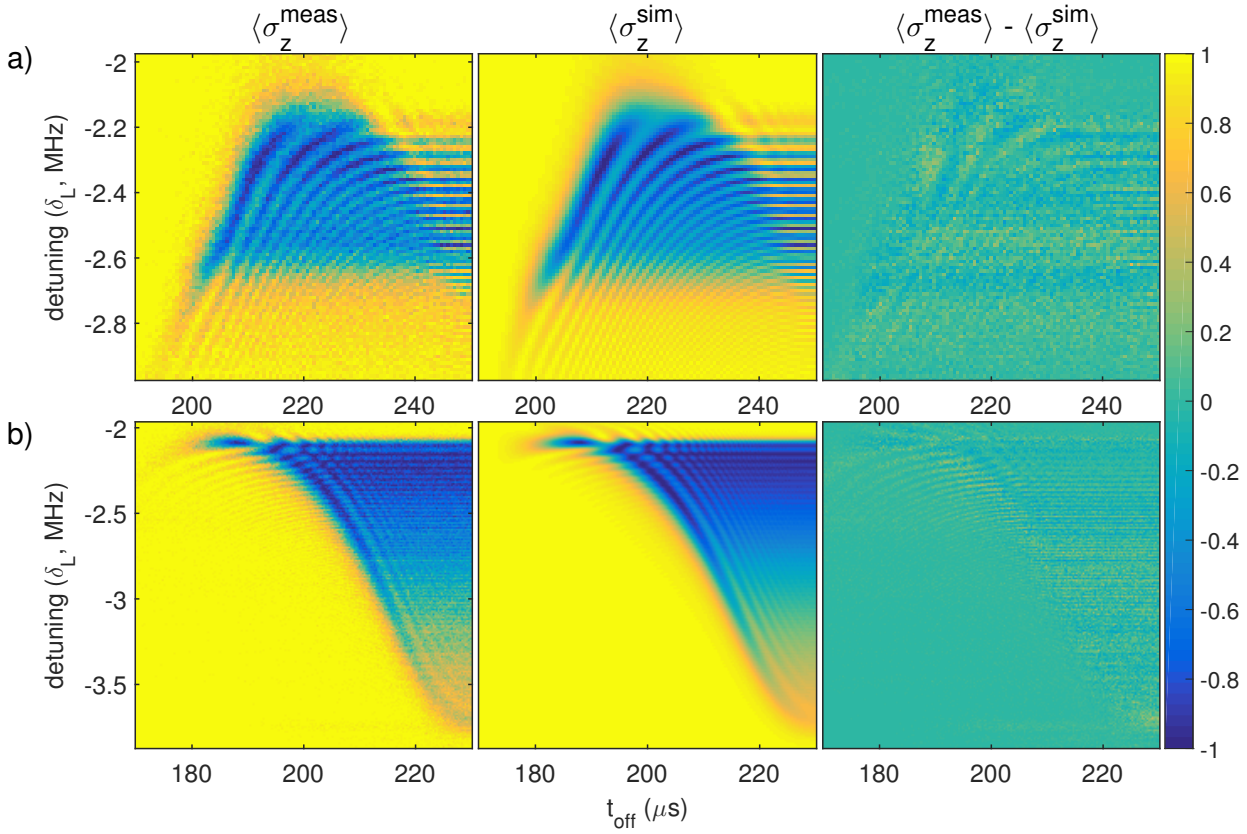


FIG. 2: **Measured data, estimation and residuals:** Spin population as a function of detuning and switch-off time of the laser beam. a) is for a laser beam centered in zone B, while for b) the beam was displaced towards zone C by 64 μm. From left to right are plotted the experimental data, the populations generated from the best fit Hamiltonian, and the residuals. Each data point results from 100 repetitions of the experimental sequence. The data in a) consist of an array of 100×101 experimental settings, while that shown in b) consists of an array of 201×201 settings. This leads to smaller error bars in the reconstructed Hamiltonian for the latter. For the Hamiltonian estimation the data was weighted according to quantum projection noise.

functions as was used for the experimental data to provide a new estimate for the Hamiltonian. This is repeated for a large number of samples, resulting in a distribution for the estimated values of $\delta(t)$ and $\Omega(t)$ from which we extract statistical properties such as the standard error. The error bounds shown in figures 3 correspond to the standard error on the mean obtained from these distributions (see [19] for further details).

The estimated coefficients of the Hamiltonian extracted from the two data sets are shown in figure 3a). It can be seen that the values of $\delta(t)$ for the two different beam positions differ for the region where the reconstructions overlap. We think that this effect arises from the non-planar wavefronts of the laser beam. Inverting the expression for $\delta(t)$ to obtain the velocity of the ion, we find $\dot{z}(t) = \delta(t)/(k'_z(z)z + k_z(z))$. Using this correction, we find that the two velocity profiles agree if we assume that the ion passes through the center of the beam at a distance of 2.27 mm before the minimum beam waist, a value which is consistent with experimental uncertainties due to beam propagation and possible mis-positioning of the ion trap with respect to the fixed final focusing lens. The velocity estimates taking account of this effect are shown in figure 3b).

Figure 4 shows the results of a reconstruction for a second pair of data sets taken using two different velocity profiles but with a common beam position. The resolution in both time and detuning were lower in this case than for the data shown in figure 2 (see [19] for the data). We observe that the estimated Rabi frequency profiles agree to within the error bars of the reconstruction. One interesting feature of this plot is that the error bars produced from the resampled data sets are notably higher at the peak than on the sides of the beam. We think that this happens because the sampling time of the data is 0.5 μs, which is not high enough to accurately resolve the fast population dynamics resulting from the high Rabi frequency (the Nyquist frequency is 1 MHz). In order to optimize the efficiency of our method, it would be advantageous to run the reconstruction method in parallel with data taking, thus allowing updating of the sampling time and frequency resolution of points based on the current

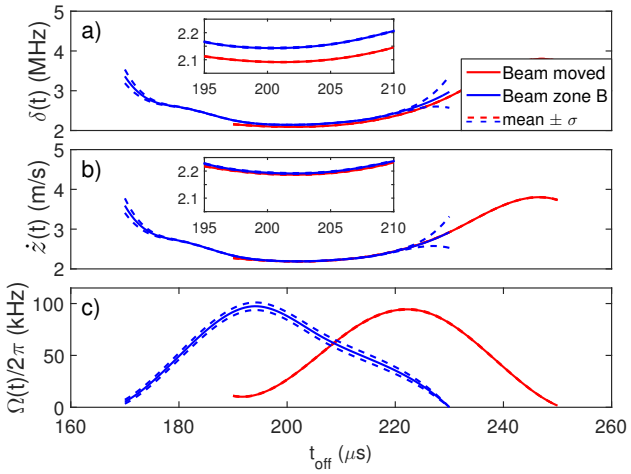


FIG. 3: **Estimates of time-dependent co-efficients:** a) The effective detuning $\delta(t)$ and c) Rabi frequency $\Omega(t)$ obtained from the two data sets, along with dashed lines indicating the standard error on the mean of these estimates obtained from resampling. For a), the inset shows a close up of the estimated $\delta(t)$ in the regions where the estimates overlap, showing that these do not give the same value. b) The estimated velocity $\dot{z}(t)$ of the ion obtained after applying wavefront correction. The inset shows that this can produce consistent results.

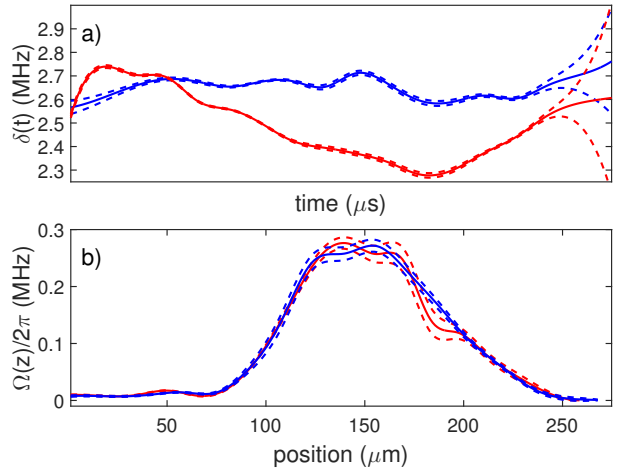


FIG. 4: **Spatial Rabi frequency:** a) The estimated $\delta(t)$ obtained from the second pair of data sets (Figure 8 in [19]). b) The estimated Rabi frequency $\Omega(t)$ for the same two data sets.

estimates of parameter values.

Our method for directly obtaining a non-commuting time-dependent Hamiltonian uses straightforward measurements of the qubit state in a fixed basis as a function of time and a controlled offset to the Hamiltonian. This simplicity means that the method should be applicable in a wide range of physical systems where such control is available, including many technologies considered for quantum computation [1, 5, 6, 26, 27]. A process-tomography based approach would require that for every time step multiple input states be introduced, and a measurement made in multiple bases [28–30]. An effective modulation of the measurement basis arises in our approach due to the additional detuning δ_L. It is worth noting that tomography provides more information than our method: it makes no assumptions about the dynamics aside from that of a completely positive map while we require coherent dynamics. Extensions to our work are required in order to provide a rigorous estimation of the efficiency of the method in terms of the precision obtained for a given number of measurements, and to see whether a similar approach could be taken to non-unitary dynamics. Using this method on considerably lower resolution data sets, we have recently been able to improve the control over the velocity, which will be necessary in order to realize multi-qubit transport gates in our current setup [7].

We thank Lukas Gerster, Martin Sepiol and Karin Fisher for contributions to the experimental apparatus. We acknowledge support from the Swiss National Science Foundation under grant numbers 200021_134776 and 200020_153430, ETH Research grant under grant no. ETH-18 12-2, and from the National Centre of Competence in Research for Quantum Science and Technology (QSIT).

Author Contributions: Experimental data was taken by LdC, RO and MM using apparatus built up by all authors. Data analysis was performed by LdC and RO. The paper was written by JPH, LdC and RO, with input from all authors. The work was conceived by JPH and LdC.

[1] J. Zhang and M.Sarovar. Quantum hamiltonian identification from measurement time traces. *Phys. Rev. Lett.*, 113(080401), 2014.

[2] A. Mitra and H. Rabitz. Identifying mechanisms in the control of quantum dynamics through hamiltonian encoding. *Phys. Rev A.*, 67(033407), 2003.

[3] R. Rey de Castro and H. Rabitz. Laboratory implementation of quantum-control-mechanism identification through hamiltonian encoding and observable decoding. *Phys. Rev A.*, 81(063422), 2010.

[4] R. Rey de Castro, R. Cabrera, D. I Bondar, and H. Rabitz. Time-resolved quantum process tomography using hamiltonian-encoding and observable-decoding. *New J. Phys.*, 15(025032), 2013.

[5] S. J. Devitt, J. H. Cole, and L. C. L. Hollenberg. Scheme for direct measurement of a general two-qubit hamiltonian. *Phys. Rev A.*, 73(052317):1–5, 2006.

[6] A. Shabani, M. Mohseni, S. Lloyd, R. L. Kosut, and H. Rabitz. Estimation of many-body quantum hamiltonians via compressive sensing. *Phys. Rev A.*, 84(012107):1–8, 2011.

[7] D. Leibfried, E. Knill, C. Ospelkaus, and D. J. Wineland. Transport quantum logic gates for trapped ions. *Phys. Rev. A*, 76:032324, Sep 2007.

[8] D. J. Wineland, C. Monroe, W. M. Itano, D. Leibfried, B. E. King, and D. M. Meekhof. Experimental issues in coherent quantum-state manipulation of trapped atomic ions. *J. Res. Natl. Inst. Stand. Technol.*, 103:259–328, 1998.

[9] R. Bowler, J. Gaebler, Y. Lin, T.R. Tan, D. Hanneke, J.D. Jost, J.P. Home, D. Leibfried, and D.J. Wineland. Coherent diabatic ion transport and separation in a multizone trap array. *Phys. Rev. Lett.*, 109:080502, Aug 2012.

[10] A. Walther, F. Ziesel, T. Ruster, S.T. Dawkins, K. Ott, M. Hettrich, K. Singer, F. Schmidt-Kaler, and U. Poschinger. Controlling fast transport of cold trapped ions. *Phys. Rev. Lett.*, 109:080501, Aug 2012.

[11] J.M. Martinis and M.R. Geller. Fast adiabatic gates using only σ_z control. *Phys. Rev A.*, 90(022307), 2014.

[12] R. Schutjens, F. Abu Dagga, D.J. Egger, and F.K. Wilhelm. Single-qubit gates in frequency-crowded transmon systems. *Phys. Rev. A*, 88:052330, Nov 2013.

[13] J. H. Cole, S. G. Schirmer, A. D. Greentree, C. J. Wellard, D. K. L. Oi, and L. C. L. Hollenberg. Identifying an experimental two-state hamiltonian to arbitrary accuracy. *Phys. Rev A.*, 71(6):1–11, 2005.

[14] N. Zhao, J. Honert, B. Schmid, M. Klas, J. Isoya, M. Markham, D. Twitchen, F. Jelezko, R-B. Liu, Helmut Fedder, and Jörg Wrachtrup. Sensing single remote nuclear spins. *Nature Nanotech.*, 7:657–662, 2012.

[15] A. Blais, , R.-S. Huang, A. Wallraff, S. M. Girvin, and R. J. Schoelkopf. Cavity quantum electrodynamics for superconducting electrical circuits: An architecture for quantum computation. *Phys. Rev. A*, 69(062320), 2004.

[16] M. H. Devoret and R. J. Schoelkopf. Superconducting circuits for quantum information: An outlook. *Science*, 339:1169–1174, 2013.

[17] H. Ball and M. J. Biercuk. Walsh-synthesized noise filters for quantum logic. *EPJ Quantum Technology*, 2(1):1–45, 2015.

[18] L. de Clercq, H.-Y. Lo, M. Marinelli, D. Nadlinger, R. Oswald, V. Negnevitsky, D. Kienzler, B. Keitch, and J. P. Home. Parallel transport-based quantum logic gates with trapped ions. *arXiv:1509.06624*, 2015.

[19] Materials and methods are available online.

[20] E. Barnes and S. Das Sarma. Analytically solvable driven time-dependent two-level quantum systems. *Phys. Rev. Lett.*, 109(060401), 2012.

[21] E. Barnes. Analytically solvable two-level quantum systems and landau-zener interferometry. *Phys. Rev. A*, 88(013818), 2013.

[22] R. H. Bartels, J. C. Beatty, and B. Barsky. *An introduction to splines for use in computer graphics and geometric modeling*. Elsevier, 1995.

[23] Carl de Boor. B(asic)-spline basics. `http://www.researchgate.net/publication/228398058_B_(asic)-spline_basics`. Notes that are kept up to date by the author.

[24] K. R. Muske and J. B. Rawlings, editors. *Methods of Model Based Process Control*. Springer Netherlands, 1995.

[25] C. F. Roos, D. Leibfried, A. Mundt, F. Schmidt-Kaler, J. Eschner, and R. Blatt. Experimental demonstration of ground state laser cooling with electromagnetically induced transparency. *Phys. Rev. Lett.*, 85(26):5547–5550, 2000.

[26] R. Barends, L. Lamata, J. Kelly, L. Garcia-Alvarez, A.G. Fowler, A. Megrant, E. Jeffrey, T.C. White, D. Sank, J.Y. Mutus, B. Campbell, Yu Chen, Z. Chen, B. Chiaro, A. Dunsworth, I.-C. Hoi, C. Neill, P.J.J. O'Malley, C. Quintana, P. Roushan, A. Vainsencher, J. Wenner, E. Solano, and John M. Martinis. Digital quantum simulation of fermionic models with a superconducting circuit. *Nat. Commun.*, 6(7654), 2015.

[27] C. Bonato, M. S. Blok, H. T. Dinani, D. W. Berry, M. L. Markham, D. J. Twitchen, and R. Hanson. Optimized quantum sensing with a single electron spin using real-time adaptive measurements. *arXiv preprint arXiv:1508.03983*, 2015.

[28] I. L. Chuang and M. A. Nielsen. Prescription for experimental determination of the dynamics of a quantum black box. *J. Mod. Opt.*, 44:2455–2467, 1997.

[29] J. F. Poyatos, J. I. Cirac, and P. Zoller. Complete characterization of a quantum process: the two-bit quantum gate. *Phys. Rev. Lett.*, 78(2):390–393, 1997.

[30] M. Riebe, K. Kim, P. Schindler, T. Monz, P. O. Schmidt, T. K. Körber, W. Hänsel, H. Häffner, C. F. Roos, and R. Blatt. Process tomography of ion trap quantum gates. *Phys. Rev. Lett.*, 97:220407, 2006.

[31] B.E.A. Saleh and M.C. Teich. *Fundamentals of photonics*. Wiley, 2007.

[32] K. P. Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

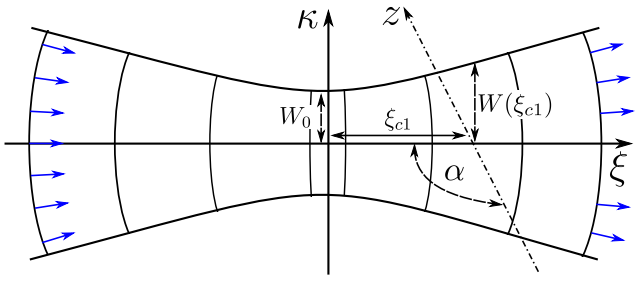


FIG. 5: **Beam and ion transport:** The beam propagation direction lies along the ξ-axis and the ion is transported along the z-axis lying the $\kappa\xi$-plane as indicated. Normalized vectors representing $\vec{e}_l(\kappa,\xi)$ lying perpendicular to the wavefronts are indicated by the blue arrows.

I. SUPPLEMENTARY MATERIAL

A. Derivation of Hamiltonian

The interaction of a laser beam with frequency ω_L and wave vector $\vec{k}(\vec{z}(t))$ with a two-level atom with resonant frequency ω_0 and time-dependent position of the ion $\vec{z}(t) = (0, 0, z(t))$ can be described in the Schrödinger picture by the Hamiltonian

$$\hat{H}_S = -\frac{\hbar\omega_0}{2}\hat{\sigma}_z - \hbar\Omega(z(t))\cos\left(\vec{k}(\vec{z}(t))\cdot\vec{z}(t) - \omega_L t\right)\hat{\sigma}_x \,, \quad (2)$$

where the Rabi frequency $\Omega(z(t))$ gives the interaction strength between the laser and the two atomic levels. We can define the laser phase at the position of the ion as $\Phi(t) = \phi(t) - \omega_L t$ with $\phi(t) = \vec{k}(\vec{z}(t))\cdot\vec{z}(t) = k_z(z(t))z(t)$ and $k_z(z(t)) = |\vec{k}|\cos(\theta(t))$ being the projection of the laser beam onto the z-axis along which the ion is transported. Here $\theta(t)$ is the angle between the wave-vector $\vec{k}(z(t))$ and the transport axis evaluated at position $z(t)$. Moving to a rotating frame using the unitary transformation $U = e^{-i\frac{\Phi(t)}{2}}$ and applying the rotating wave approximation with respect to optical frequencies, we obtain

$$\hat{H}_I = \frac{\hbar}{2}\left(-\Omega(t)\hat{\sigma}_x + \left(-\omega_0 - \dot{\Phi}(t)\right)\hat{\sigma}_z\right). \quad (3)$$

Defining a static detuning $\delta_L = \omega_L - \omega_0$ we obtain

$$\hat{H}_I = \frac{\hbar}{2}\left(-\Omega(t)\hat{\sigma}_x + \left(\delta_L - \dot{\phi}(t)\right)\hat{\sigma}_z\right). \quad (4)$$

with

$$\delta(t) = \delta_L - \dot{\phi}(t) \quad (5)$$

which is the expression used in the main text.

B. Wavefront correction

For plane waves we find that $\dot{\phi}(t) = \vec{k}\cdot\vec{v}(t)$ which is the well-known expression for the first-order Doppler shift. For transport through a real Gaussian beam, the wave-vector direction changes with position. Taking this into account, the derivative of $\phi(t)$ becomes

$$\dot{\phi}(t) = \left[k_z'(z(t))z(t) + k_z(z(t))\right]\dot{z}(t) \quad (6)$$

where $k_z' = dk_z/dz$ and $\dot{z}(t)$ is the component of the ion's velocity which lies along the z-axis. We extract $\delta(t)$ using our Hamiltonian estimation procedure, thus to obtain the velocity of the ion we use

$$\dot{z}(t) = \frac{-\delta(t) + \delta_L}{k_z'(z(t))z(t) + k_z(z(t))} \,. \quad (7)$$

As the ion moves through the beam it experiences the same magnitude of the wave vector $|\vec{k}| = 2\pi/\lambda$, but the angle θ between the ion's direction and the wave vector changes. Written as a function of this angle, the velocity becomes

$$\dot{z}(t) = \frac{-\delta(t) + \delta_L}{-|\vec{k}|\sin\left(\theta(z(t))\right)\theta'(z(t))z(t) + |\vec{k}|\cos(\theta(z(t)))} \quad (8)$$

where $\theta'(z(t)) = d\theta(z(t))/dz(t)$. We parameterize our Gaussian beam according to figure 5. The phase is given as a function of both the position along the beam axis ξ and the perpendicular distance from this axis κ by [31]

$$\varphi(\kappa,\xi) = |\vec{k}|\xi - \zeta(\xi) + \frac{|\vec{k}|\kappa^2}{2R(\xi)} \,. \quad (9)$$

where the Gaussian beam parameters include the beam waist $W(\xi)$, the radius of curvature $R(\xi)$ the Rayleigh range ξ_R and the Guoy phase shift $\zeta(\xi)$. These are given by the expressions

$$\begin{aligned} W(\xi) &= W_0\sqrt{1 + \left(\frac{\xi}{\xi_R}\right)^2} \\ R(\xi) &= \xi\left(1 + \left(\frac{\xi_R}{\xi}\right)^2\right) \\ \zeta(\xi) &= \tan^{-1}\left(\frac{\xi}{\xi_R}\right) \\ \xi_R &= \frac{\pi W_0^2}{\lambda} \\ k &= \frac{2\pi}{\lambda} \end{aligned} \quad (10)$$

where W_0 is the minimum beam waist and λ the laser wavelength. The ion moves along the z-axis shown in figure 5. In the $\kappa\xi$-plane a unit vector $\vec{e}_l(\kappa,\xi)$ perpendicular to the wavefronts is given by

$$\vec{e}_l(\kappa,\xi) = \frac{\nabla\varphi(\kappa,\xi)}{||\nabla\varphi(\kappa,\xi)||} \quad (11)$$

and the unit vector $\vec{e}_v$ pointing along the direction of transport is given by

$$\vec{e}_v = \begin{bmatrix}\cos(\alpha)\\ \sin(\alpha)\end{bmatrix} \quad (12)$$

The angle $\theta(\xi)$ between the wave and position vector is then given by the dot product

$$\theta(\kappa) = \cos^{-1}\left(\vec{e}_n \cdot \vec{e}_v\right). \tag{13}$$

which can be written in terms of the full set of parameters above as

$$\begin{aligned}
\theta(\kappa) &= \cos^{-1}\left(\gamma_1 + \gamma_2\right) \\
\gamma_1 &= \frac{\cos(\alpha)\left(-2\xi_R\left(\xi^2+\xi_R^2\right)+k\kappa^2\left(\xi_R^2-\xi^2\right)+2k\left(\xi^2+\xi_R^2\right)^2\right)}{\eta(\kappa)} \\
\gamma_2 &= \frac{\sin(\alpha)2k\kappa\xi\left(\xi^2+\xi_R^2\right)}{\eta(\kappa)} \\
\eta(\kappa,\xi) &= \left(\xi^2+\xi_R^2\right)\left[4\left(\frac{k\kappa\xi}{\xi^2+\xi_R^2}\right)^2+\left(-\frac{2\xi_R}{\xi^2+\xi_R^2}+k\left(2+\frac{\kappa^2\left(\xi_R^2-\xi^2\right)}{\left(\xi^2+\xi_R^2\right)^2}\right)\right)^2\right] \\
\kappa(t) &= z(t)\sin(\alpha)
\end{aligned} \tag{14}$$

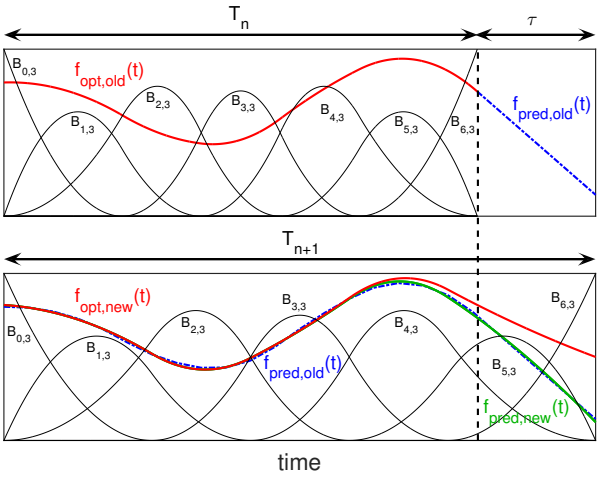


FIG. 6: **Extending the Horizon Estimation:** The steps performed when extending the time horizon from T_n to $T_{\mathrm{n}+1}$ are illustrated. We first predict in the old basis, then move to the new basis, and finally optimize again. The figure also shows the basis splines $B_{i,k}(t)$.

where in our experiments $\alpha = 3\pi/4$.

Using Eq. 8 and 14 we examined the value of ξ_cl required for the velocity to match for our two beam positions. We find that they agree for $\xi_\mathrm{cl} = -2.27$ mm, which is within the experimental uncertainties for our setup.

C. Basis spline curves

One challenge in obtaining an estimate for the Hamiltonian is that we must optimize over continuous functions $\delta(t)$ and $\Omega(t)$. To address this, we represent $\delta(t)$ and $\Omega(t)$ with basis spline curves. Basis spline curves allow the construction of smooth functions using only a few parameters. This is achieved by introducing a set of polynomial Basis (B)-spline functions $B_{i,k}(t)$ of order k [23]. A smooth curve $S(t)$ can then be represented as a linear combination of these B-spline basis functions [22]

$$S(t) = \sum_{i=0}^{n} \alpha_i B_{i,k}(t). \tag{15}$$

The B-splines $B_{i,k}(t)$ of order k are recursively defined over the index i over a set of points $\vec{K} = \{t_0, t_1, ..., t_{n+k}\}$ which is referred to as the knot vector [23].

$$\begin{aligned}
B_{i,1}(t) &= \begin{cases} 1 & t_i \le t \le t_{i+1} \\ 0 & \text{otherwise} \end{cases} \\
B_{i,k}(t) &= \omega_{i,k}(t) B_{i,k-1}(t) + \left(1 - \omega_{i+1,k}(t)\right) B_{i+1,k-1}(t). \\
\omega_{i,k}(t) &= \begin{cases} \frac{t-t_i}{t_{i+k-1}-t_i} & \text{if } t_i \neq t_{i+k-1} \\ 0 & \text{otherwise} \end{cases}
\end{aligned} \tag{16}$$

Figure 6 gives a visualization of the B-splines $B_{i,k}(t)$ and a basis spline curve. The B-spline construction ensures that any linear combination of the B-splines is continuous and has $(k-2)$ continuous derivatives. The knot vector $\vec{K}$ determines how the basis functions are positioned within the interval $[t_0, t_{n+k}]$. We notice that for our Hamiltonian the spacing of the B-splines is not critical, which we think is due to the smoothness of the variations in our Hamiltonian parameters $\delta(t)$ and $\Omega(t)$. We therefore used the Matlab function *spap2* to automatically choose a suitable knot vector and restricted ourselves to optimizing the coefficients α_i. We collect all coefficients α_i for $\delta(t)$ and $\Omega(t)$ and store them in a single vector $\vec{\alpha}$.

D. Extending the Horizon Estimation

The task of inferring the time-dependent Hamiltonian of the form (1) from the measured data can be cast into an optimization problem for which we use a reduced chi-squared cost function

$$J = \frac{1}{\nu} \sum_{t_{\rm off}} \sum_{\delta_L} \left[\frac{\langle \hat{\sigma}_z^{\rm meas}(t_{\rm off}, \delta_L)\rangle - \langle \hat{\sigma}_z^{\rm sim}(t_{\rm off}, \delta_L)\rangle}{\sigma^{\rm meas}(t_{\rm off}, \delta_L)} \right]^2 \quad (17)$$

where $\nu = N - n - 1$ is the degrees of freedom with N the number of data points and n the number of fitting parameters, and $\sigma^{\rm meas}(t_{\rm off}, \delta_L)$ is the standard error on the estimated $\langle \hat{\sigma}_z^{\rm meas}(t_{\rm off}, \delta_L)\rangle$ which we obtain assuming quantum projection noise. In our case the Hamiltonian $\hat{H}_I$ (Eq. 1) with offset $\hat{H}_{\rm s} = \hbar\delta_L \hat{\sigma}_z/2$ is parametrized by $\Omega(t)$ and $\delta(t)$. We can thus write the problem as

$$\min_{\delta(t),\,\Omega(t)} J(\delta(t), \Omega(t)) \quad (18)$$

subject to

$$\begin{aligned} i\hbar \frac{\partial}{\partial t} |\Psi(t,\delta_L)\rangle &= \left(\hat{H}_I(t) + \hat{H}_s\right) |\Psi(t,\delta_L)\rangle, \\ |\Psi(t=0,\delta_L)\rangle &= |0\rangle, \\ \langle \hat{\sigma}_z^{\rm sim}(t,\delta_L)\rangle &= \langle \Psi(t,\delta_L)| \hat{\sigma}_z |\Psi(t,\delta_L)\rangle \end{aligned} \quad (19)$$

for all δ_L.

This optimization problem is hard to efficiently solve in general, because it is nonlinear and non-convex due to the nature of Schrödinger's equation and the use of projective measurements. In order to overcome this challenge, we have implemented a method which we call "Extending the Horizon Estimation" (EHE) in analogy to a well-established technique called "Moving Horizon Estimation" (MHE) [24].

The key idea is that because our measurement data arises from a causal evolution, we can also estimate the Hamiltonian in a causal way. We define a time span ranging from the initial time to some later time which we call the time horizon. Instead of optimizing J over the complete time span at once, we first restrict ourselves to a small, initial time horizon reaching only up to the start of the qubit dynamics. Optimizing J over this short time horizon requires fewer optimization parameters and is simpler than attempting to optimize over the full data set. Once we have solved this small sub-problem, we extend the time horizon and re-run the optimization, extrapolating the results of the initial time window into the extended window in order to provide good starting conditions for the subsequent optimization. This is greatly advantageous for the use of non-linear least squares optimization, which typically works by linearizing the problem and converges much faster near the optimum. The extension of the horizon is used repeatedly until the time window covers the full data set.

Conceptually EHE is very similar to MHE. The main difference is that in MHE the time span has a fixed length and thus its origin gets shifted forwards in time along with the horizon. In EHE the origin stays fixed at the expense of having to increase the time span under consideration. MHE avoids this by introducing a so-called arrival cost to approximate the previous costs incurred before the start of the time span. This keeps the computational burden fixed over time, which is very important as MHE is usually used to estimate the state of a system in real-time, often on severely constrained embedded platforms. Since neither constraint applies to our problem, we decided to extend the horizon rather than finding an approximate arrival cost. This is advantageous since finding the arrival cost in the general case is still an open problem. Due to the similarity between MHE and EHE, we anticipate future improvements by adapting techniques used in MHE to EHE.

Next, we present a more detailed algorithmic summary of our implementation of the method outlined above.

1. **Searching for a starting point.** Here we reconstruct the Hamiltonian for a first, minimal time horizon such that we can then use this as a starting point to iteratively extend the horizon as described in step 2.

   (a) Choose an initial time horizon such that it contains the region where the first discernible qubit dynamics occur.

   (b) Cut down the number of fitting parameters as much as possible, e.g. by using few Basis splines of low order. This amounts to choosing empirically a low number of basis splines (and thus the length of $\vec{\alpha}_0$) which might represent $\delta(t)$ and $\Omega(t)$ over the given region.

   (c) Use a nonlinear least squares fitting routine to minimize J by varying the parameters $\vec{\alpha}_0$. In the case that the initial fit is not good or no minimum is found, try new initial conditions, change the number of B-spline functions, or manually adjust the function using prior knowledge of the physical system under consideration.

   This procedure is used to provide a starting point for the optimization over the initially chosen window, which is typically performed with a higher order set of B-splines. From this starting point, we iteratively extend the fitting method to the full data set as follows.

2. **Extend the horizon** This step is repeated until the whole time horizon is covered. It consists of the following sequence, which is illustrated in figure 6.

   (a) Extend the time horizon by τ from $T_{\rm n}$ to $T_{\rm n+1} = T_{\rm n} + \tau$.

   (b) Extrapolate $f_{\rm opt,old}(t)$ within τ, e.g. using *fnxtr* in Matlab.

(c) Adapt the Basis splines to the new time horizon $T_{\mathrm{n+1}}$ and represent $f_{\mathrm{pred,old}}(t)$ in the new basis, giving $f_{\mathrm{pred,new}}(t)$. In Matlab one can use *spap2* to do this.

(d) Use $f_{\mathrm{pred,new}}(t)$ as the initial guess for a weighted nonlinear least squares fit over the extended time span up to $T_{\mathrm{n+1}}$.

(e) Judge the results of the fit based on its reduced chi-squared value χ^2_{red}. If it is below a specified bound, continue with an additional iteration of steps a)-d), repeating until the full region of the data is covered. Otherwise, try the following fall-back procedures:

i. Reduce τ, the time by which the time horizon is extended, and try again

ii. Increase the number of Basis splines and try again

iii. Try again using a different starting point.

If all of those fail, we have to resort to increasing the bound on χ^2_{red}.

3. **Post-processing**. The following steps are optional and were performed manually in cases where we desired to improve the fit, or examine its behaviour.

(a) The optimization over the whole time horizon was re-run using different numbers of Basis splines for $\delta(t)$ and $\Omega(t)$. This served as a useful check on the sensitivity of the fit.

(b) The optimization over the whole time horizon was re-run using a starting point based on the previously found optimum plus randomized deviations. This examines robustness of the final fit.

E. Error estimation

To obtain error bars of the time-dependent functions we use non-parametric bootstrapping [32]. The process is summarized as follow:

1. **Estimate initial solution** Estimate the time dependent functions from the original data using Hamiltonian estimation.

2. **Resampling** Create N_{s} sample solutions for all time-dependent functions in the following way:

(a) Form a sample set by randomly picking with replacement from the photon count data used in qubit detection.

(b) Re-estimate new time-dependent functions by optimizing over the full time span, using the solution found in (1) as a starting point.

(c) Record the reduced chi-squared values $\chi^2_{\mathrm{red,r}}$ for each sample r along with the B-spline curve coefficients $\vec{\alpha}_{\mathrm{r}}$

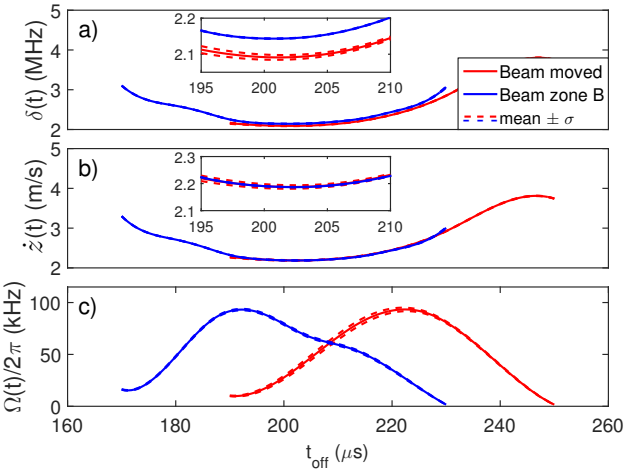


FIG. 7: **Parametric bootstrap resampling:** Predictions for $\delta(t)$, $\dot{z}(t)$ and $\Omega(t)$ with error bounds obtained using parametric bootstrap resampling, assuming quantum projection noise. This can be compared with the error bounds obtained from the non-parametric method which are shown in Figure 3 in the main text. The bounds are tighter for the parametric bootstrapping.

3. **Post-process samples**

(a) Form a histogram of the chi-squared values $\chi^2_{\mathrm{red,r}}$.

(b) Find and fit a normal-like distribution to the histogram with preference to the spread with lowest lying $\chi^2_{\mathrm{red,r}}$ in the case of a multi-modal distribution. From the fit obtain the mean reduced-chi squared value $\langle\chi^2_{\mathrm{red,r}}\rangle$ as well as the standard deviation σ_χ.

(c) Eliminate the outlier samples by removing all $\vec{\alpha}_r$ with $\chi^2_{\mathrm{red,r}}$ values that are 3-5σ_χ from the mean $\langle\chi^2_{\mathrm{red,r}}\rangle$.

(d) Form a matrix $\boldsymbol{Y}$ where each row vector is a sample set of coefficients $\vec{\alpha}_r$ that remained after step 3(c).

4. **Obtain statistics**

(a) Find the mean B-spline coefficients $\langle\vec{\alpha}\rangle$ of equation 15 by taking the mean over the column vectors of $\boldsymbol{Y}$ with each element of the mean given by $\langle\vec{\alpha}\rangle_i = \langle\alpha_i\rangle$.

(b) Find the covariance matrix $\boldsymbol{\Sigma} = \mathrm{cov}(\boldsymbol{Y}^\alpha)$ with $\boldsymbol{\Sigma}_{ij} = \mathrm{E}\left[(\alpha_i - \langle\alpha_i\rangle)(\alpha_j - \langle\alpha_j\rangle)\right]$ with E the expectation operator. The standard deviations of each of the mean coefficients $\langle\alpha_i\rangle$ is given by $\sigma_{\langle\alpha_i\rangle} = \sqrt{\boldsymbol{\Sigma}_{ii}}$. We record these values in a row vector $\vec{\sigma}_{\langle\alpha_i\rangle}$.

We have also applied parametric bootstrapping in order to obtain the error bounds shown in figure 7. The difference to the non-parametric case is that in point (2)

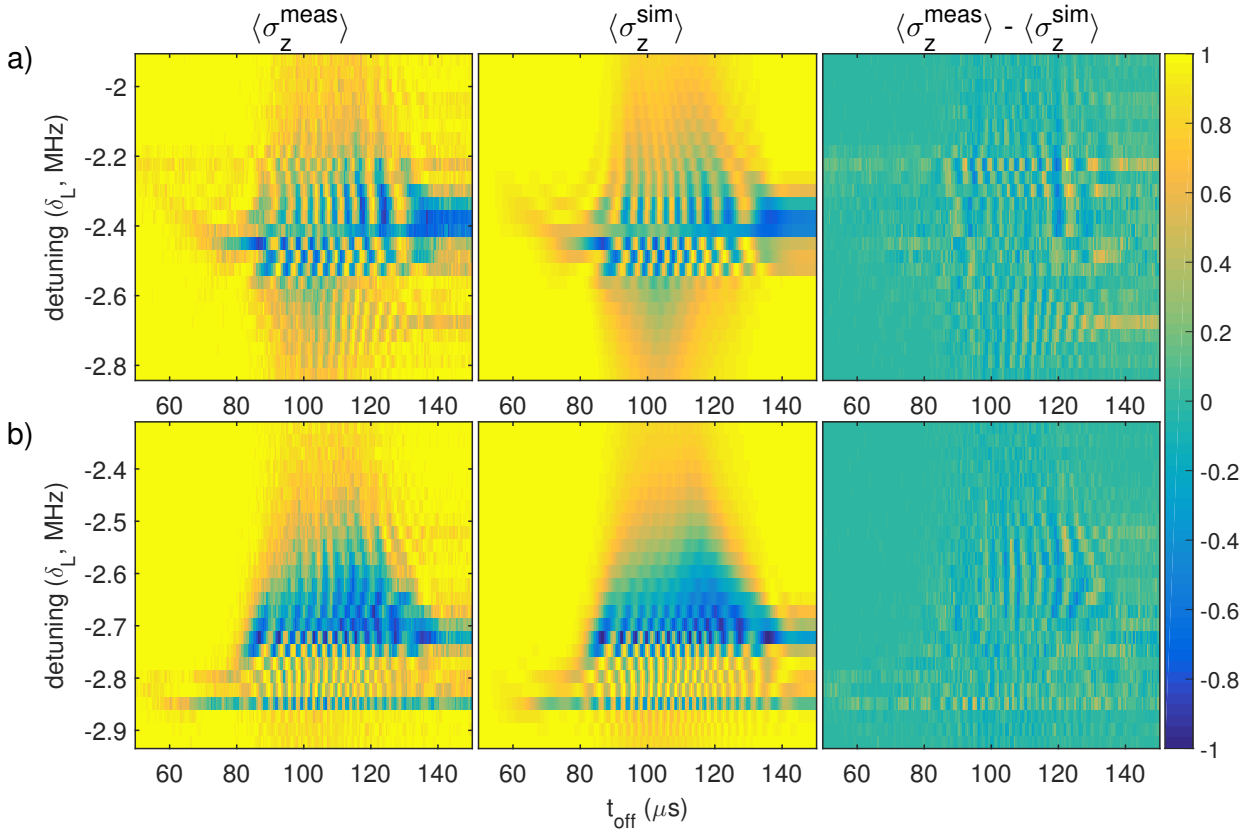


FIG. 8: **Measured data, estimation and residuals:** Spin population as a function of detuning and switch-off time of the laser beam, for the data sets used to obtain the reconstructed parameters shown in figure 4. a) uses a velocity profile with only small variations. b) A second data run in which large variations in the velocity profile were used. Each data point results from 100 repetitions of the experimental sequence. For the Hamiltonian estimation the data was weighted according to quantum projection noise.

the samples are created using the solutions obtained from (1) and adding quantum projection noise. For each sample the Hamiltonian is estimated. The estimates from multiple samples are used to construct error bounds in the same manner as for the non-parametric resampling. We have found that the error bounds obtained from parametric bootstrapping are lower compared to that of the non-parametric case as shown in figure 3. We think this is due to the latter exploring deviations around a single minimum in the optimization landscape, while the case resampling arrives at different local minima which are spread over a wider region.

F. Single beam profile with two different velocity profiles.

As a check that our method is also able to produce consistent results for the Rabi frequency profile, we measured a second pair of data sets in which we take two different velocity profiles using the same beam position. This data is shown in Figure 8. Also shown are the best-fits obtained from the reconstructed Hamiltonians. The parameter variations obtained from the reconstructed Hamiltonians for these data sets can be found in the main text in Figure 4. The sampling rate of the data in these data sets was 2 MHz, resulting in a Nyquist frequency of 1 MHz.

Bibliography

[Amann 95] Notker Amann, David H. Owens & Eric Rogers. *Iterative Learning Control using Optimal Feedback and Feedforward Actions*, 1995.

[Arimoto 84] Suguru Arimoto, Sadao Kawamura & Fumio Miyazaki. *Bettering operation of Robots by learning.* Journal of Robotic Systems, vol. 1, no. 2, pages 123–140, 1984.

[Bartels 95] R. H. Bartels, J. C. Beatty & B. Barsky. An introduction to splines for use in computer graphics and geometric modeling. Elsevier, 1995.

[Blakestad 09] R. B. Blakestad, C. Ospelkaus, A. P. VanDevender, J. M. Amini, J. Britton, D. Leibfried & D. J. Wineland. *High-Fidelity Transport of Trapped-Ion Qubits through an X-Junction Trap Array.* Phys. Rev. Lett., vol. 102, page 153002, 2009.

[Blakestad 10] R. B. Blakestad. *Transport of Trapped-Ion Qubits within a Scalable Quantum Processor.* PhD thesis, University of Colorado, 2010.

[Bowler 15] R. Bowler. *Coherent Ion Transport in a Multi-electrode Trap Array.* PhD thesis, University of Colorado, 2015.

[Boyd 04] S. Boyd & L. Vandenberghe. Convex optimization. Cambridge University Press, 2004.

[D. Leibfried 07] C. Ospelkaus D. Leibfried E. Knill & D. J. Wineland. *Transport quantum logic gates for trapped ions.* Phys. Rev. A, 2007.

[de Boor 01] Carl de Boor. A practical guide to splines; rev. ed. Applied mathematical sciences. Springer, Berlin, 2001.

[de Clercq 15a] L. de Clercq, H.-Y. Lo, M. Marinelli, D. Nadlinger, R. Oswald, V. Negnevitsky, D. Kienzler, B. Keitch & J.P. Home. *Parallel transport quantum logic gates with trapped ions.* arXiv:1509.06624, 2015.

[de Clercq 15b] L.E. de Clercq, R. Oswald, C. Flühmann, B. Keitch, D. Kienzler, H.-Y. Lo, M. Marinelli, D. Nadlinger, V. Negnevitsky & J.P. Home. *Time-dependent Hamiltonian estimation for Doppler velocimetry of trapped ions.* arXiv:1509.07083, 2015.

[de Clercq 15c] Ludwig de Clercq. *Transport Quantum Logic Gates for Trapped Ions.* PhD thesis, ETH Zürich, 2015.

[Feynman 82] R. P. Feynman. *Simulating physics with computers.* International Journal of Theoretical Physics, vol. 21, page 467, 1982.

[Fox 05] M. Fox. Quantum optics. Oxford University Press, 2005.

[Fürst 14] H. A. Fürst, M. H. Goerz, U. G. Poschinger, M. Murphy, S. Montangero, T. Calarco, F. Schmidt-Kaler, K .Singer & C. P. Koch. *Controlling the transport of an ion: classical and quantum mechanical solutions.* New Journal of Physics, vol. 16, no. 7, page 075007, 2014.

[Gurobi 15] Gurobi Optimization Inc. *Gurobi optimizer reference manual.* http://www.gurobi.com, 2015.

[Home 09] J. P. Home, D. Hanneke, J. D. Jost, J. M. Amini, D. Leibfried & D. J. Wineland. *Complete Methods Set for Scalable Ion Trap Quantum Information Processing.* Science, vol. 325, page 1227, 2009.

[Kielpinski 02] D. Kielpinski, C.Monroe & D. Wineland. *Architecture for a large-scale ion-trap quantum computer.* Nature, vol. 417, pages 709–711, 2002.

[Kienzler 15] Daniel Kienzler. *Quantum harmonic oscillator state synthesis by reservoir engineering.* PhD thesis, ETH Zürich, 2015.

[Lee 00] Jay H. Lee, Kwang S. Lee & Won C. Kim. *Model-based iterative learning control with a quadratic criterion for time-varying linear systems.* Automatica, vol. 36, no. 5, pages 641 – 657, 2000.

[Lo 15] Hsiang-Yu Lo. *Creation of squeezed schrödinger's cat states in a mixed-species ion trap.* PhD thesis, ETH Zürich, 2015.

[Lu 14] Xiao-Jing Lu, J. G. Muga, Xi Chen, U. G. Poschinger, F. Schmidt-Kaler & A. Ruschhaupt. *Fast shuttling of a trapped ion in the presence of noise.* Phys. Rev. A, vol. 89, no. 063414, 2014.

[Rawlings 09] James Blake Rawlings & David Q. Mayne. Model predictive control : theory and design. Nob Hill Publishing, 2009.

[Rezaeizadeh 15] A. Rezaeizadeh & R. Smith. *An Adaptive Iterative Learning Control Scheme for Reducing CO2 Emission in Gasoline Engines.* In European Control Conference, Linz, Austria, July 2015.

[Rowe 02] M. A. Rowe, A. Ben-Kish, B. DeMarco, D. Leibfried, V. Meyer, J. Beall, J. Britton, J. Hughes, W. M. Itano, B. Jelenković, C. Langer, T. Rosenband & D. J. Wineland. *Transport of quantum states and separation of ions in a dual rf ion trap.* Quantum Information and Computation, vol. 2, page 257, 2002.

[Särkkä 13] Simo Särkkä. Bayesian filtering and smoothing. Cambridge University Press, 2013.

[Sasura 02] M. Sasura & V. Buzek. *Cold trapped ions as quantum information processors.* J. Mod. Opt., vol. 49, no. 10, 2002.

[Tibshirani 96] Robert Tibshirani. *Regression Shrinkage and Selection via the Lasso.* Journal of the Royal Statistical Society. Series B (Methodological), vol. 58, no. 1, pages 267–288, 1996.

[Todorov 08] E. Todorov. *General duality between optimal control and estimation.* In Decision and Control, 2008. CDC 2008. 47th IEEE Conference on, pages 4286–4292, Dec 2008.

[Wineland 98] D. J. Wineland, C. Monroe, W. M. Itano, D. Leibfried, B. E. King & D. M. Meekhof. *Experimental issues in coherent quantum-state manipulation of trapped atomic ions.* J. Res. Natl. Inst. Stand. Technol., vol. 103, pages 259–328, 1998.

[Wright 13] K. Wright, J. M Amini, D.L. Faircloth, C. Volin, S.C. Doret, H. Hayden, C-S Pai, D.W. Landgren, D. Denison, T. Killian, R.E. Slusher & A.W. Harter. *Reliable transport through a microfabricated X-junction surface-electrode ion trap.* New J. Phys., vol. 15, no. 033004, 2013.

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Declaration of originality

The signed declaration of originality is a component of every semester paper, Bachelor's thesis, Master's thesis and any other degree paper undertaken during the course of studies, including the respective electronic versions.

Lecturers may also require a declaration of originality for other written papers compiled for their courses.

I hereby confirm that I am the sole author of the written work here enclosed and that I have compiled it in my own words. Parts excepted are corrections of form and content by the supervisor.

Title of work (in block letters):

Velocity Control of Trapped Ions for Transport Quantum Logic Gates

Authored by (in block letters):

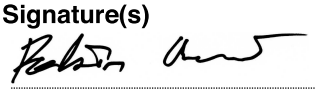
For papers written by groups the names of all authors are required.

Name(s):	**First name(s):**
Oswald	Robin

With my signature I confirm that

- I have committed none of the forms of plagiarism described in the 'Citation etiquette' information sheet.
- I have documented all methods, data and processes truthfully.
- I have not manipulated any data.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for plagiarism.

Place, date	**Signature(s)**
Zurich, 23.11.2015	

For papers written by groups the names of all authors are required. Their signatures collectively guarantee the entire content of the written paper.