

Colloquium: Trapped ions as quantum bits: Essential numerical tools

Kilian Singer* and Ulrich Poschinger

Institut für Physik, Johannes Gutenberg-Universität Mainz, 55099 Mainz, Germany

Michael Murphy

Institut für Quanteninformationsverarbeitung, Universität Ulm, Albert-Einstein-Allee 11, 89069 Ulm, Germany

Peter Ivanov and Frank Ziesel

Institut für Physik, Johannes Gutenberg-Universität Mainz, 55099 Mainz, Germany

Tommaso Calarco

Institut für Quanteninformationsverarbeitung, Universität Ulm, Albert-Einstein-Allee 11, 89069 Ulm, Germany

Ferdinand Schmidt-Kaler

Institut für Physik, Johannes Gutenberg-Universität Mainz, 55099 Mainz, Germany

(Published 14 September 2010)

Trapped laser-cooled atoms and ions are quantum systems which can be experimentally controlled with an as yet unmatched degree of precision. Due to the control of the motion and the internal degrees of freedom, these quantum systems can be adequately described by a well-known Hamiltonian. In this colloquium, powerful numerical tools for the optimization of the external control of the motional and internal states of trapped neutral atoms, explicitly applied to the case of trapped laser-cooled ions in a segmented ion-trap are presented. Inverse problems when optimizing trapping potentials for ions, are solved. The presentation is complemented by a quantum-mechanical treatment of the wave-packet dynamics of a trapped ion. Efficient numerical solvers for both time-independent and time-dependent problems are provided. Shaping the motional wave functions and optimizing a quantum gate is realized by the application of quantum optimal control techniques. The numerical methods presented can also be used to gain an intuitive understanding of quantum experiments with trapped ions by performing virtual simulated experiments on a personal computer. Code and executables are supplied as supplementary online material.

DOI: [10.1103/RevModPhys.82.2609](https://doi.org/10.1103/RevModPhys.82.2609)

PACS number(s): 42.50.-p, 03.67.-a, 37.10.Ty, 02.60.Pn

CONTENTS

	the Schrödinger Equation	2619
	A. Solution of the time-independent Schrödinger equation: The Numerov method	2619
	B. Numerical evaluation of the time-dependent Hamiltonian	2620
	C. The split-operator method	2622
	D. The Chebyshev propagator	2622
	VI. Optimizing Wave-Packet Manipulations: Optimal Control Theory	2623
	A. Krotov algorithm	2623
	1. Constructing the optimization objective	2624
	2. Minimizing the objective	2624
	3. Deriving an iterative scheme	2625
	B. Application	2626
	VII. Improving Quantum Gates: OCT of a Unitary Transformation	2626
	A. The Cirac-Zoller controlled-NOT gate	2626
	B. Krotov algorithm on unitary transformations	2628
	C. Application	2629
	VIII. Conclusion	2629
	A. Applicability to other qubit types	2629
	B. Summary	2630
	Acknowledgments	2630
I. Introduction	2610	
II. Ion-Trap Development: Calculation of Electrostatic Fields	2611	
A. Finite difference method	2612	
B. Finite element method	2613	
C. Boundary element method: Fast multipole method	2614	
D. Application	2614	
III. Ion Trajectories: Classical Equations of Motion	2615	
A. Euler method	2615	
B. Runge-Kutta method	2615	
C. Partitioned Runge-Kutta method	2615	
D. Application	2617	
IV. Transport Operations with Ions: Ill-Conditioned Inverse Problems	2617	
A. Tikhonov regularization	2618	
B. Application	2619	
V. Quantum Dynamics: Efficient Numerical Solution of		

*kilian.singer@uni-mainz.de

I. INTRODUCTION

The information carrier used in computers is a bit, representing a binary state of either zero or one. In the quantum world a two-level system can be in any superposition of the ground and excited states. The basic information carrier encoded by such a system is called a quantum bit (qubit). Qubits are manipulated by quantum gates—unitary transformations on single qubits and on pairs of qubits.

Trapped ions are among the most promising physical systems for implementing quantum computation (Nielsen and Chuang, 2000). Long coherence times and individual addressing allow for the experimental implementation of quantum gates and quantum computing protocols such as the Deutsch-Josza algorithm (Gulde *et al.*, 2003), teleportation (Barrett *et al.*, 2004; Riebe *et al.*, 2004), quantum error correction (Chiaverini *et al.*, 2004), quantum Fourier transform (Chiaverini *et al.*, 2005), and Grover's search (Brickman *et al.*, 2005). Complementary research is using trapped neutral atoms in micropotentials such as magnetic microtraps (Schmiedmayer *et al.*, 2002; Fortágh and Zimmermann, 2007), dipole traps (Grimm *et al.*, 2000; Gaëtan *et al.*, 2009), or optical lattices (Lukin *et al.*, 2001; Mandel *et al.*, 2003) where even individual imaging of single atoms has been accomplished (Nelson *et al.*, 2007; Bakr *et al.*, 2009, 2010; Sherston *et al.*, 2010). The current challenge for all approaches is to scale the technology up for a larger number of qubits, for which several proposals exist (Cirac and Zoller, 2000; Duan *et al.*, 2004; Kielpinski *et al.*, 2004).

The basic principle for quantum computation with trapped ions is to use the internal electronic states of the ion as the qubit carrier. Computational operations can then be performed by manipulating the ions by coherent laser light (Blatt and Wineland, 2008; Häffner *et al.*, 2008). In order to perform entangling gates between different ions, Cirac and Zoller (1995) proposed to use the mutual Coulomb interaction to realize a collective quantum bus (a term used to denote an object that can transfer quantum information between subsystems). The coupling between laser light and ion motion enables the coherent mapping of quantum information between internal and motional degrees of freedom of an ion chain. Two-ion gates are of particular importance since combined with single-qubit rotations they constitute a universal set of quantum gates for computation (DiVincenzo, 1995). Several gate realizations have been proposed (Cirac and Zoller, 1995; Monroe *et al.*, 1997; Poyatos *et al.*, 1998; Mølmer and Sørensen, 1999; Jonathan *et al.*, 2000; Milburn *et al.*, 2000; Mintert and Wunderlich, 2001; García-Ripoll *et al.*, 2005) and realized (DeMarco *et al.*, 2002; Leibfried, DeMarco, *et al.*, 2003; Schmidt-Kaler, Häffner, Riebe, *et al.*, 2003; Benhelm *et al.*, 2008; Kim *et al.*, 2008). When more ions are added to the ion chain, the same procedure can be applied until the different vibrational-mode frequencies

become too close to be individually addressable;¹ the current state of the art is the preparation and read out of a W entangled state of eight ions (Häffner *et al.*, 2005) and a six-ion Greenberger-Horne-Zeilinger (GHZ) state (Leibfried *et al.*, 2005).

A way to solve this scalability problem is to use segmented ion traps consisting of different regions between which the ions are shuttled (transported back and forth) (Kielpinski *et al.*, 2004; Amini *et al.*, 2010). The proper optimization of the shuttling processes and optimization of the laser ion interaction can only be fully performed with the aid of numerical tools (Reichle *et al.*, 2006; Schulz *et al.*, 2006; Hucul *et al.*, 2008; Huber *et al.*, 2010). In our presentation equal emphasis is put on the presentation of the physics of quantum information experiments with ions and the basic ideas of the numerical methods. All tools are demonstrated with ion-trap experiments, such that the reader can easily extend and apply the methods to other fields of physics. Included is supplementary material, e.g., source code and data such that even an inexperienced reader may apply the numerical tools and adjust them for his needs. While some readers might aim at understanding and learning the numerical methods by looking at our specific ion-trap example, others might intend to get a deeper understanding of the physics of quantum information experiments through simulations and simulated experiments. We start in Sec. II with the description of the ion-trap principles and introduce numerical methods to solve for the electrostatic potentials arising from the trapping electrodes. Accurate potentials are needed to numerically integrate the equation of motion of ions inside the trap. Efficient stable solvers are presented in Sec. III. The axial motion of the ion is controlled by changing the dc voltages of the electrodes. However, usually we perform the inverse, such that we find the voltages needed to be applied to the electrodes in order to produce a certain shape of the potential to place the ion at a specific position with the desired trap frequency as described in Sec. IV. This problem belongs to a type of problems known as inverse problems, which are quite common in physics. In Sec. V we enter the quantum world where we first obtain the stationary motional eigenstates of the time-independent Schrödinger equation in arbitrary potentials. We then describe methods to tackle the time-dependent problem and present efficient numerical methods to solve the time-dependent Schrödinger equation. The presented methods are used in Sec. VI where we consider time-dependent electrostatic potentials with the goal to perform quantum control on the motional wave function and present the optimal control algorithm. Finally, we apply these techniques in Sec. VII to the Cirac-Zoller gate. In Sec. VIII, we give a short account on the applicability of the presented numerical methods to qubit implementations other than trapped laser-cooled ions.

¹Although there are schemes that address multiple modes (Zhu *et al.*, 2006a, 2006b; Kim *et al.*, 2009).

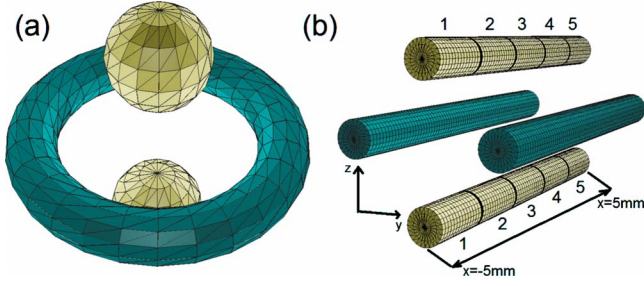


FIG. 1. (Color online) Electrode geometries of ion traps: The rf electrodes (dark shading) and dc electrodes (bright shading), are depicted. (a) Typical electrode configuration for a 3D ring trap with dynamic rf confinement in all three dimensions. (b) Electrode arrangement for a linear Paul trap. The dc electrodes are divided into segments numbered from 1 to 5. For the numerical simulations we assume the following parameters: segments have a width of 2 mm and a radius of 0.5 mm. The central dc electrode is centered at the position $x=0$. The minimum distance of the electrode surface to the trap axis is 1.5 mm.

II. ION-TRAP DEVELOPMENT: CALCULATION OF ELECTROSTATIC FIELDS

The scalability problem for quantum information with ion traps can be resolved with segmented ion traps. The trap potentials have to be tailored to control the position and trapping frequency of the ions. In the following, we describe the mode of operation of a simple ion-trap and then present numerical solvers for the efficient calculation of accurate electrostatic fields. Due to the impossibility of generating an electrostatic potential minimum in free space, ions may be trapped either in a combination of electric and magnetic static fields—a Penning trap (Brown and Gabrielse, 1986) or in a radio-frequency electric field, a Paul trap, where a radio-frequency (rf) voltage U_{rf} , with rf drive frequency ω_{rf} , is applied to some of the ion-trap electrodes (Paul, 1990). In the latter case, we generate a potential

$$\Phi(x, y, z, t) = \frac{U_{\text{dc}}}{2} (\alpha_{\text{dc}} x^2 + \beta_{\text{dc}} y^2 + \gamma_{\text{dc}} z^2) + \frac{U_{\text{rf}}}{2} \cos(\omega_{\text{rf}} t) (\alpha_{\text{rf}} x^2 + \beta_{\text{rf}} y^2 + \gamma_{\text{rf}} z^2), \quad (1)$$

where U_{dc} is a constant trapping voltages applied to the electrodes. The Laplace equation in free space $\Delta\Phi(x, y, z)=0$ puts an additional constraint on the coefficients: $\alpha_{\text{dc}} + \beta_{\text{dc}} + \gamma_{\text{dc}}=0$ and $\alpha_{\text{rf}} + \beta_{\text{rf}} + \gamma_{\text{rf}}=0$. One possibility to fulfill these conditions is to set $\alpha_{\text{dc}} = \beta_{\text{dc}} = \gamma_{\text{dc}}=0$ and $\alpha_{\text{rf}} + \beta_{\text{rf}} = -\gamma_{\text{rf}}$. This produces a purely dynamic confinement of the ion and is realized by an electrode configuration as shown in Fig. 1(a), where the torus-shaped electrode is supplied with radio frequency and the spherical electrodes are grounded. An alternative solution would be the choice $-\alpha_{\text{dc}} = \beta_{\text{dc}} + \gamma_{\text{dc}}$ and $\alpha_{\text{rf}}=0$, $\beta_{\text{rf}} = -\gamma_{\text{rf}}$, leading to a linear Paul trap with dc confinement along the x axis and dynamic confinement in the y - z plane. Figure 1(b) shows a possible setup with cylindrically shaped electrodes and segmented dc electrodes

along the axial direction which we consider in the following. In this trapping geometry, the ions can crystallize into linear ion strings aligned along the x axis. The classical equation of motion for an ion with mass m and charge q is $m\ddot{\mathbf{x}} = -q\nabla\Phi$, with $\mathbf{x}=(x, y, z)$ (James, 1998). For a potential given by Eq. (1), the classical equations of motion are transformed into a set of two uncoupled Mathieu differential equations (Leibfried, Blatt, *et al.*, 2003; Häffner *et al.*, 2008);

$$\frac{d^2 u}{d\xi^2} + [a_u - 2q_u \cos(2\xi)]u(\xi) = 0 \quad u = y, z, \quad (2)$$

with $2\xi = \omega_{\text{rf}} t$. The Mathieu equation belongs to the family of differential equations with periodic boundary conditions and its solution is readily found in textbooks [see, for example, Abramowitz and Stegun (1964)]. For a linear Paul trap, the parameters a_u and q_u in the y - z plane are given by

$$q_y = \frac{2|q|U_{\text{rf}}\beta_{\text{rf}}}{m\omega_{\text{rf}}^2}, \quad a_y = -\frac{4|q|U_{\text{dc}}\beta_{\text{dc}}}{m\omega_{\text{rf}}^2}, \quad (3)$$

$$q_z = -\frac{2|q|U_{\text{rf}}\gamma_{\text{rf}}}{m\omega_{\text{rf}}^2}, \quad a_z = \frac{4|q|U_{\text{dc}}\gamma_{\text{dc}}}{m\omega_{\text{rf}}^2}.$$

The solution is stable in the range $0 \leq \beta_u \leq 1$, where $\beta_u = \sqrt{a_u + q_u^2/2}$ depends only on the parameters a_u and q_u . The solution of Eq. (2) in the lowest-order approximation ($|a_u|, q_u^2 \ll 1$), which implies that $\beta_u \ll 1$, is

$$u(t) = u_0 \cos(\omega_u t) \left(1 + \frac{q_u}{2} \cos(\omega_{\text{rf}} t) \right). \quad (4)$$

The ion undergoes harmonic oscillations at the secular frequency $\omega_u = \beta_u \omega_{\text{rf}}/2$ modulated by small oscillations near the rf-drive frequency (called micromotion). The static axial confinement along the x axis is harmonic with the oscillator frequency given by $\omega_x = \sqrt{|q|U_{\text{dc}}\alpha_{\text{dc}}/m}$. The axial confinement is generated by biasing the dc electrode segments appropriately. Typical potential shapes are shown in Fig. 2(a).

The radial confinement is dominated by the rf potential which can be approximated by an *effective* harmonic potential $\Phi_{\text{eff}}(y, z) = |q| |\nabla\Phi(y, z)|^2 / 4m\omega_{\text{rf}}^2$, where $\Phi(y, z)$ is the potential generated by setting the radio-frequency electrodes to a constant voltage U_{rf} ; see Fig. 2(b). However, this effective potential is only an approximation and does not take the full dynamics of the ion into account. Before we can simulate the motion of the ion we need fast and accurate electrostatic field solvers. In the next section we first present the finite difference method and then the finite element method. If the potentials need to be known on a small scale, a large spatial grid would be needed. Therefore, we introduce the boundary element method and show how the efficiency of this method can be drastically improved by the application of the fast multipole method.

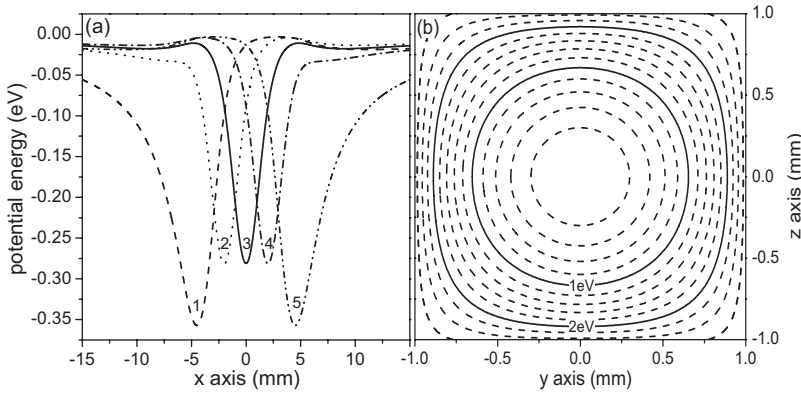


FIG. 2. Trapping potentials. (a) Electrostatic potentials along the x axis generated by each individual electrode from the linear Paul trap geometry of Fig. 1(b). Each curve corresponds to the respective electrode biased to -1 V and all others to 0 V. (b) Equipotential lines of the pseudopotential in the radial plane ($U_{rf} = 200$ V_{pp} and $\omega_{rf} = 2\pi \times 20$ MHz). Potentials are obtained as described in Sec. II.

A. Finite difference method

To obtain the electrostatic potential $\Phi(x, y, z)$ in free space generated by a specific voltage configuration U_i for $i=1, \dots, n$ applied to the n electrodes, we need to solve the Laplace equation $\Delta\Phi(x, y, z)=0$, with the Dirichlet boundary condition $\Phi(x, y, z)=U_i$ for points lying on the i th electrode. There are several approaches to obtain the solution. The most intuitive is the finite difference method (FDM). The principle is that we can write the differential equation in terms of finite differences (Thomas, 1995). To illustrate this, we take the one-dimensional differential equation $d\Phi/dx=F(x)$ with the boundary condition $\Phi(0)=a$, where $F(x)$ is an arbitrary function. If we write

$$\frac{d\Phi}{dx} = \lim_{\Delta x \rightarrow 0} \frac{\Phi(x + \Delta x) - \Phi(x)}{\Delta x} = F(x), \quad (5)$$

take only a finite difference Δx and discretize the x axis by defining $x_i = i\Delta x$, with i running from 0 to N and $x_N = 1$, we obtain a discrete approximation which directly gives an explicit update equation (using the Euler method) for Φ ,

$$\Phi(x_{i+1}) = \Phi(x_i) + \Delta x F(x_i). \quad (6)$$

Equation (6) can then be applied iteratively to solve the differential equation. By comparing the solution with the Taylor expansion and assuming that the higher-order terms are bounded, we see that the error of this finite difference approximation is of order Δx . This is usually written as

$$\frac{\Phi(x + \Delta x) - \Phi(x)}{\Delta x} = \frac{d\Phi}{dx} + \mathcal{O}(\Delta x), \quad (7)$$

which means that there exists a constant d such that

$$\left| \frac{\Phi(x + \Delta x) - \Phi(x)}{\Delta x} - \frac{d\Phi}{dx} \right| < d|\Delta x|$$

for all x . The Laplace equation is of second order but one can transform it into a set of a first-order differential equations

$$\frac{d}{dx} \begin{pmatrix} \Phi \\ v \end{pmatrix} = \begin{pmatrix} v \\ F(x) \end{pmatrix}, \quad (8)$$

from which an explicit update rule can be derived, which is $\mathcal{O}(\Delta x)$. We can obtain a second-order approximation by cancelling the first-order terms which gives a centered-difference approximation for the first derivative

$$\frac{\Phi(x_{n+1}) - \Phi(x_{n-1}))}{2\Delta x} = \frac{d\Phi}{dx} \Big|_{x_n} + \mathcal{O}(\Delta x^2), \quad (9)$$

which is of order $\mathcal{O}(\Delta x^2)$. A centered-difference approximation for the second derivative reads

$$\frac{\Phi(x_{n+1}) - 2\Phi(x_n) + \Phi(x_{n-1}))}{\Delta x^2} = \frac{d^2\Phi}{dx^2} \Big|_{x_n} + \mathcal{O}(\Delta x^2), \quad (10)$$

which is again of order $\mathcal{O}(\Delta x^2)$. The update rule for the one-dimensional Laplace equation $d^2\Phi/dx^2=0$ thus has the form

$$\Phi(x_{n+1}) - 2\Phi(x_n) + \Phi(x_{n-1}) = 0, \quad (11)$$

which is an implicit expression, since the solution now has to be obtained by solving a linear system of algebraic equations. We have to specify two boundary conditions which we assume to be $\Phi(x_0)=U_1$ and $\Phi(x_N)=U_2$, where U_1 and U_2 are the voltages supplied at the boundaries. The matrix equation then has the form

$$\begin{pmatrix} -2 & 1 & 0 & \cdots & 0 \\ 1 & -2 & 1 & & \vdots \\ 0 & 1 & -2 & \ddots & 0 \\ \vdots & & \ddots & \ddots & 1 \\ 0 & \cdots & 0 & 1 & -2 \end{pmatrix} \begin{pmatrix} \Phi(x_1) \\ \Phi(x_2) \\ \Phi(x_3) \\ \vdots \\ \Phi(x_{N-1}) \end{pmatrix} = \begin{pmatrix} -U_1 \\ 0 \\ 0 \\ \vdots \\ -U_2 \end{pmatrix}. \quad (12)$$

This equation has a tridiagonal form and can be most efficiently solved by the Thomas algorithm (Press et al., 2007).² A sparse matrix with more off-diagonal entries is obtained when the two- or three-dimensional (3D) Laplace equation is treated in a similar fashion. The solution is then obtained by either simple Gaussian elimi-

²See package OCTTOOL, function *tridag*.

nation or more efficiently by iterative methods, such as the successive over-relaxation method (SOR) (Press et al., 2007) or the generalized minimum residual (GMRES) method (Saad, 2003).

One advantage of FDM is that it is easy to implement on a uniform Cartesian grid. But in modeling three-dimensional geometries one usually favors a triangular nonuniform mesh, where the mesh spacing is spatially adapted to the local complexity of the geometry structures; e.g., it makes sense to use a finer mesh near the edges.

B. Finite element method

The finite element method (FEM) is better suited for nonuniform meshes with inhomogeneous granularity since it transforms the differential equation into an equivalent variational one: instead of approximating the differential equation by a finite difference, the FEM solution is approximated by a finite linear combination of basis functions. Again, we demonstrate the method with a one-dimensional differential equation $d^2\Phi/dx^2=F(x)$, and for simplicity we take the boundary condition $\Phi(0)=0$ and $\Phi(1)=0$. The variational equivalent is an integral equation integrated between the boundaries at 0 and 1:

$$\int_0^1 \frac{d^2\Phi(x)}{dx^2} v(x) dx \equiv \int_0^1 F(x) v(x) dx, \quad (13)$$

where $v(x)$ is the variational function which can be freely chosen except for the requirement $v(0)=v(1)=0$. Integrating this by parts gives

$$\begin{aligned} \int_0^1 \frac{d^2\Phi(x)}{dx^2} v(x) dx &= \underbrace{\frac{d\Phi(x)}{dx} v(x)}_0 \Big|_0^1 - \int_0^1 \frac{d\Phi(x)}{dx} \frac{dv(x)}{dx} dx \\ &\equiv \int_0^1 F(x) v(x) dx. \end{aligned} \quad (14)$$

We can now discretize this equation by constructing $v(x)$ on a finite-dimensional basis. One possibility is linear interpolation:

$$v_k(x) = \begin{cases} \frac{x - x_{k-1}}{x_k - x_{k-1}} & \text{if } x_{k-1} \leq x \leq x_k \\ \frac{x_{k+1} - x}{x_{k+1} - x_k} & \text{if } x_k < x \leq x_{k+1} \\ 0 & \text{otherwise,} \end{cases} \quad (15)$$

with $x_0=0$, $x_N=1$, and x_k are the (not necessarily equidistant) sequential points in between and k ranges from 1 to $N-1$. These functions are shown in Fig. 3. The advantage of this choice is that the inner products of the basis functions $\int_0^1 v_k(x) v_j(x) dx$ and their derivatives $\int_0^1 v_k'(x) v_j'(x) dx$ are only nonzero for $|j-k| \leq 1$. The functions $\Phi(x)$ and $F(x)$ are then given by $\Phi(x) \approx \sum_{k=0}^N \Phi(x_k) v_k(x)$ and $F(x) \approx \sum_{k=0}^N F(x_k) v_k(x)$ which lin-

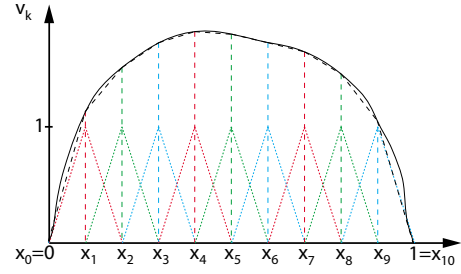


FIG. 3. (Color online) Overlapping basis functions from Eq. (15) $v_k(x)$ (dotted lines) for the finite element method providing linear interpolation (black dashed line) of an arbitrary function (black solid line).

early interpolates the initial functions (see Fig. 3). With

$$\frac{d\Phi(x)}{dx} \approx \sum_{k=0}^N \Phi_k \frac{dv_k(x)}{dx},$$

Eq. (14) is now recast into the form

$$\begin{aligned} - \sum_{k=0}^N \Phi(x_k) \int_0^1 \frac{dv_k(x)}{dx} \frac{dv_j(x)}{dx} dx \\ = \sum_{k=0}^N F(x_k) \int_0^1 v_k(x) v_j(x) dx, \end{aligned} \quad (16)$$

where the terms in square brackets are sparse matrices. This matrix equation can then again be solved by iterative matrix solvers (such as GMRES).

For the Laplace problem, we need to extend this method to higher dimensions. In this case, instead of the integration by parts in Eq. (14) we have to use the Green's theorem (Jackson, 2009):

$$\begin{aligned} \int_V \Delta\Phi(\mathbf{x}) v(\mathbf{x}) dV &= \int_V \underbrace{\frac{\partial\Phi}{\partial\nu} \frac{\partial v}{\partial n}}_0 ds - \int_V \nabla\Phi \nabla v dV \\ &\equiv \int_V F(\mathbf{x}) v(\mathbf{x}) dV, \end{aligned} \quad (17)$$

where V is the volume of interest and δV the bounding surface of the volume. Now space is discretized by three-dimensional basis functions and we can proceed in an analogous manner as in the one-dimensional case described above.

Potentials obtained by FDM and FEM usually result in unphysical discontinuities (i.e., numerical artifacts) and must be smoothed in order to be useful for ion trajectory simulations. Additionally, in order to obtain high accuracy trajectory simulations needed to simulate the trajectory extend of a trapped ion of less than 100 nm, the potentials that are calculated have to be interpolated since computing with a grid with nanometer spacing would involve an unbearable computational overhead: the whole space including the typically centimeter-sized trap would have to be meshed with a nanometer-spaced grid. FEM would allow for a finer mesh in the region where the ion would be located reducing the overhead somewhat, but this does not increase the accuracy of the

surrounding coarser grid. Avoiding to give a wrong expression, we stress that the FEM method finds wide applications in engineering and physics especially when complicated boundary conditions are imposed but for our accuracy goals FEM and FDM are inadequate.

C. Boundary element method: Fast multipole method

We proceed to show a different way of solving the Laplace problem with a method which features a high accuracy and gives smooth potentials that perform well in high-resolution ion-ray-tracing simulations.

To begin with, we divide the electrodes into small surface elements s_i of uniform surface charge density σ_i , with i numbering all surface elements from 1 to N . The potential at any point in space caused by a charge distribution of these elements can be easily obtained from Coulomb's law: one must simply sum up all the contributions from each surface element. Hence the voltage U_j on the surface element s_j is generated by a linear superposition of the surface charge densities $\sigma_i = \partial\Phi(x_i)/\partial n$ also expressed as the normal derivative of the potential Φ (as obtained from the Maxwell equations) on all surface elements (including s_j) additionally weighted by geometry factors given by the Coulomb law (represented by a matrix \hat{C}) providing the following simple matrix equation:

$$\begin{pmatrix} U_1 \\ U_2 \\ \vdots \\ U_N \end{pmatrix} = \hat{C} \begin{pmatrix} \sigma_1 \\ \sigma_2 \\ \vdots \\ \sigma_N \end{pmatrix}. \quad (18)$$

Now we want to solve for the surface charge densities in terms of the applied voltages. The surface charge densities for a given voltage configuration can then be obtained by finding the matrix inversion of \hat{C} . This is the basic idea of the boundary element method (BEM) (Pozrikidis, 2002). In the case of metallic surface elements where either the potential or the charge density is fixed, we have to exploit the Green's second identity

$$\Phi(\mathbf{x}_j) = -2 \sum_{i=1}^N \alpha_i(\mathbf{x}_j) \frac{\partial\Phi(\mathbf{x}_i)}{\partial n} + 2 \sum_{i=1}^N \beta_i(\mathbf{x}_j) \Phi(\mathbf{x}_i). \quad (19)$$

This equation then has to be solved for the unknown parameters which are the surface charge density $\partial\Phi(\mathbf{x}_i)/\partial n$ on surfaces with given potential or the potential $\Phi(\mathbf{x}_i)$ on surfaces with given charge density. Now we can choose the \mathbf{x}_i and \mathbf{x}_j to be representative points on each surface element, e.g., the center of gravity. This corresponds to the approximation that the potential and charge density are constant on each surface element. Equation (19) is a matrix equation equivalent to Eq. (18). α_i is obtained by performing a surface integral over the surface elements s_i of the two-dimensional Green's function $G(\mathbf{x}, \mathbf{x}_j) = -(1/2\pi) \ln|\mathbf{x} - \mathbf{x}_j|$ (for two-dimensional problems) or the three-dimensional Green's function $G(\mathbf{x}, \mathbf{x}_j) = -(1/4)\pi|\mathbf{x} - \mathbf{x}_j|^{-1}$ (for three-dimensional prob-

lems). β_i is obtained by performing a surface integral over the surface elements s_i of the gradient of the Green's function multiplied by the surface norm \mathbf{n} ,

$$\alpha_i(\mathbf{x}_j) = \oint_{s_i} G(\mathbf{x}, \mathbf{x}_j) ds, \quad (20)$$

$$\beta_i(\mathbf{x}_j) = \oint_{s_i} \mathbf{n}(\mathbf{x}) \cdot \nabla G(\mathbf{x}, \mathbf{x}_j) ds. \quad (21)$$

Analytical expressions for these integrals over triangular surface elements can be found in Davey and Hinduja (1989) or via Gauss-Legendre quadrature over a triangle. Equation (19) is now solved for the unknown parameters such as the surface charge densities $\partial\Phi(\mathbf{x}_i)/\partial n$. Once this is achieved, we can calculate the potential

$$\Phi(\mathbf{x}) = - \sum_{i=1}^N \alpha_i(\mathbf{x}) \frac{\partial\Phi(\mathbf{x}_i)}{\partial n} + \sum_{i=1}^N \beta_i(\mathbf{x}) \Phi(\mathbf{x}_i) \quad (22)$$

at any position \mathbf{x} , with $\alpha_i(\mathbf{x})$ and $\beta_i(\mathbf{x})$ evaluated at the same position. BEM is very accurate and the implementation is quite straight forward, but the complexity of the matrix inversion scales prohibitively as $\mathcal{O}(N^3)$. Different to the finite element method, we cannot use sparse matrix solvers for this matrix inversion.

Fortunately, Greengard and Rokhlin (1988) came up with an innovative method for speeding up the matrix vector multiplication needed for iterative matrix inversion, which they termed the fast multipole method (FMM). FMM can solve the BEM problem with $\mathcal{O}(N)$ complexity, giving a drastic increase in speed and making BEM applicable to more complex systems. The algorithm was further improved (Carrier *et al.*, 1988; Cheng *et al.*, 1999; Nabors *et al.*, 1994; Greengard and Rokhlin, 1997; Gumerov and Duraiswami, 2005; Shen and Liu, 2007) and extended to work with the Helmholtz equation (Gumerov and Duraiswami, 2004). The basic idea was to use local and far-field multipole expansions together with efficient translation operations to calculate approximations of the fields where the three-dimensional space is recursively subdivided into cubes. A detailed description of the method is beyond the scope of this paper, and we refer the interested reader to the cited literature.

D. Application

We have used the FMM implementation from Nabors *et al.* (1994) and combined it with a scripting language for geometry description and the ability to read AutoCAD files for importing geometrical structures. Any small inaccuracies due to numerical noise on the surface charges are "blurred out" at large distances due to the Coulomb law's $1/r$ scaling. In this regard, we can assert that the surface charge densities obtained by FMM are accurate enough for our purposes. If special symmetry properties are needed (such as rotational symmetry for ion-lens systems or mirror symmetry), then one can additionally symmetrize the surface charge den-

sities. We have implemented symmetrization functions in our code to support these calculations (Fickler et al., 2009). As FMM is used to speed up the matrix vector multiplication it can be also used to speed up the evaluation of Eq. (22) to obtain the potentials in free space. However, if accurate potentials in the submicrometer scale are needed (such as for our application), it is better to use FMM for the calculation of the surface charge densities, i.e., for the inversion of the matrix equation [Eq. (19)] and then use conventional matrix multiplication for the field evaluations as described by Eq. (22). Figure 2(a) shows the smooth potentials calculated by solving for the surface charge densities with FMM. Depicted are the potentials for each electrode when biased to -1 V with all others grounded. A trapping potential is then generated by taking a linear superposition of these potentials. Figure 2(b) shows the equipotential lines of the pseudopotential. The full implementation can be found inside our BEMSOLVER package together with example files for different trap geometries.

With the calculated potentials from this section we can now solve for the motion of an ion in the dynamic trapping potential of the Paul trap which will be the focus of the next section.

III. ION TRAJECTORIES: CLASSICAL EQUATIONS OF MOTION

The electrostatic potentials obtained with the methods presented in the previous section are used in this section to simulate the trajectories of ions inside a dynamic trapping potential of a linear Paul trap. We present the Euler method and the more accurate Runge-Kutta integrators. Then we show that the accuracy of trajectories can be greatly enhanced using phase-space area conserving and energy conserving solvers such as the Störmer-Verlet method, which is a partitioned Runge-Kutta integrator.

A. Euler method

The equation of motion of a charged particle with charge q and mass m in an external electrical field can be obtained by solving the ordinary differential equation

$$\ddot{\mathbf{x}}(t) = \mathbf{f}(t, \mathbf{x}),$$

$$\dot{\mathbf{y}}(t) \equiv \begin{pmatrix} \dot{\mathbf{x}} \\ \dot{\mathbf{v}} \end{pmatrix} = \begin{pmatrix} \mathbf{v} \\ \mathbf{f}(\mathbf{x}, t) \end{pmatrix} \equiv \mathbf{F}(t, \mathbf{y}),$$

where $\mathbf{f}(t, \mathbf{x}) = (q/m)\mathbf{E}(t, \mathbf{x}) = -(q/m)\nabla\Phi(t, \mathbf{x})$ is the force arising from the electric field. The vectors \mathbf{y} and \mathbf{F} are six-dimensional vectors containing the phase-space coordinates. As in the previous section, the equation of motion can be solved by means of the explicit Euler method with the update rule $\mathbf{y}_{n+1} = \mathbf{y}_n + h\mathbf{F}(t_n, \mathbf{y}_n)$, where we use the notation $\mathbf{y}_n = \mathbf{y}(t_n)$. If ε is the absolute tolerable error, then the time step $h = t_{n+1} - t_n$ should be chosen as $h = \sqrt{\varepsilon}$, which gives the best compromise between numerical errors caused by the method and floating

point errors accumulated by all iterations.

An implicit variation in the Euler method is given by the update rule $\mathbf{y}_{n+1} = \mathbf{y}_n + h\mathbf{F}(t_{n+1}, \mathbf{y}_{n+1})$. Neither of the methods is symmetric, which means that under time inversion ($h \rightarrow -h$ and $\mathbf{y}_n \rightarrow \mathbf{y}_{n+1}$) a slightly different trajectory is generated. A symmetric update rule is given by the implicit midpoint rule $\mathbf{y}_{n+1} = \mathbf{y}_n + h\mathbf{F}((t_n + t_{n+1})/2, (\mathbf{y}_n + \mathbf{y}_{n+1})/2)$, which has the additional property that it is *symplectic*, meaning that it is area preserving in phase-space. The explicit and implicit Euler methods are of order $\mathcal{O}(h)$, whereas the implicit midpoint rule is of order $\mathcal{O}(h^2)$ (Hairer et al., 2002). These methods belong to the class of one-stage Runge-Kutta methods (Greenspan, 2006).

B. Runge-Kutta method

The general s -stage Runge-Kutta method is defined by the update equation

$$\mathbf{y}_{n+1} = \mathbf{y}_n + h \sum_{i=1}^s b_i \mathbf{k}_i, \quad (23)$$

with

$$\mathbf{k}_i = \mathbf{F}\left(t_n + c_i h, \mathbf{y}_n + h \sum_{j=1}^s a_{ij} \mathbf{k}_j\right), \quad c_i = \sum_{j=1}^s a_{ij} \quad (24)$$

for $i=1, \dots, s$ and b_i and a_{ij} are real numbers, which are given in *Butcher tableaux* for several Runge-Kutta methods. Note that in the general case k_i are defined *implicitly* such that Eq. (24) have to be solved at each time step. However, if $a_{ij}=0$ for $i \leq j$ then the Runge-Kutta method is known as *explicit*. The standard solver used in many numerical packages is the explicit fourth- and fifth-order Dormand-Price Runge-Kutta, whose values are given in the Butcher tableau in Table I.

The difference between the fourth- and fifth-order terms can be used as error estimate for dynamic step size adjustment.

C. Partitioned Runge-Kutta method

A significant improvement can be achieved by partitioning the dynamical variables into two groups, e.g., position and velocity coordinates, and to use two *different* Runge-Kutta methods for their propagation. To illustrate this we are dealing with a classical nonrelativistic particle of mass m which can be described by the Hamiltonian function $H(\mathbf{x}, \mathbf{v}) = T(\mathbf{v}) + \Phi(\mathbf{x})$, with $T(\mathbf{v}) = m\mathbf{v}^2/2$. The finite difference version of the equation of motion generated by this Hamiltonian reads $\mathbf{x}_{n+1} - 2\mathbf{x}_n + \mathbf{x}_{n-1} = h^2\mathbf{f}(\mathbf{x}_n)$. The only problem is that we cannot start this iteration, as we do not know \mathbf{x}_{-1} . The solution is to introduce the velocity $\mathbf{v} = \dot{\mathbf{x}}$ written as a symmetric finite difference,

TABLE I. Butcher tableau for the fourth- and fifth-order Runge-Kutta methods: the leftmost column contains the c_i coefficients, the last two rows under the separation line contain the b_i coefficients to realize a fourth-order or fifth-order Dormand-Price Runge-Kutta. The a_{ij} coefficients are given by the remaining numbers in the central region of the tableau. Empty entries correspond to $a_{ij}=0$.

0	$\leftarrow c_1$						
$\frac{1}{5}$	$\frac{1}{5}$	$\leftarrow a_{21}$					
$\frac{3}{10}$	$\frac{3}{40}$	$\frac{9}{40}$	$\leftarrow a_{32}$				
$\frac{4}{5}$	$\frac{44}{45}$	$-\frac{56}{15}$	$\frac{32}{9}$				
$\frac{8}{9}$	$\frac{19372}{6561}$	$-\frac{25360}{2187}$	$\frac{64448}{6561}$	$-\frac{212}{729}$		a_{76}	
1	$\frac{9017}{3168}$	$-\frac{355}{33}$	$\frac{46732}{5247}$	$\frac{49}{176}$	$-\frac{5103}{18656}$	\downarrow	b_7
1	$\frac{35}{384}$	0	$\frac{500}{1113}$	$\frac{125}{192}$	$-\frac{2187}{6784}$	$\frac{11}{84}$	\downarrow
b^{4th}	$\frac{35}{384}$	0	$\frac{500}{1113}$	$\frac{125}{192}$	$-\frac{2187}{6784}$	$\frac{11}{84}$	0
b^{5th}	$\frac{5179}{57600}$	0	$\frac{7571}{16695}$	$\frac{393}{640}$	$-\frac{92097}{339200}$	$\frac{187}{2100}$	$\frac{1}{40}$

$$\mathbf{v}_n = \frac{\mathbf{x}_{n+1} - \mathbf{x}_{n-1}}{2h}, \tag{25}$$

and the initial conditions $\mathbf{x}(0)=\mathbf{x}_0, \dot{\mathbf{x}}(0)=\mathbf{v}_0$ such that we can eliminate \mathbf{x}_{-1} to obtain $\mathbf{x}_1=\mathbf{x}_0+h\mathbf{v}_0+(h^2/2)\mathbf{f}(\mathbf{x}_0)$. Now we can use the following recursion relation:

$$\mathbf{v}_{n+1/2} = \mathbf{v}_n + \frac{h}{2}\mathbf{f}(\mathbf{x}_n), \tag{26}$$

$$\mathbf{x}_{n+1} = \mathbf{x}_n + h\mathbf{v}_{n+1/2}, \tag{27}$$

$$\mathbf{v}_{n+1} = \mathbf{v}_{n+1/2} + \frac{h}{2}\mathbf{f}(\mathbf{x}_{n+1}). \tag{28}$$

One should not be confused by the occurrence of half-integer intermediate time steps. Equations (26) and (28) can be merged to $\mathbf{v}_{n+1/2}=\mathbf{v}_{n-1/2}+h\mathbf{f}(\mathbf{x}_n)$ if \mathbf{v}_{n+1} is not of interest. The described method is the Störmer-Verlet method and is popular in molecular-dynamics simulations where the Hamiltonian has the required properties, e.g., for a conservative system of N particles with two-body interactions:

$$H(\mathbf{x}, \mathbf{v}) = \frac{1}{2} \sum_{i=1}^N m_i \mathbf{v}_i^T \mathbf{v}_i + \sum_{i=2}^N \sum_{j=1}^{i-1} V_{ij}(|\mathbf{x}_i - \mathbf{x}_j|). \tag{29}$$

In the case of the simulation of ion crystals, V_{ij} would be the Coulomb interaction between ion pairs. The popularity of this method is due to the fact that it respects the

conservation of energy and is a symmetric and symplectic solver of order 2. It belongs to the general class of partitioned Runge-Kutta methods where the s -stage method for the partitioned differential equation $\dot{\mathbf{y}}(t) = \mathbf{f}(t, \mathbf{y}, \mathbf{v})$ and $\dot{\mathbf{v}}(t) = \mathbf{g}(t, \mathbf{y}, \mathbf{v})$ is defined by

$$\mathbf{y}_{n+1} = \mathbf{y}_n + h \sum_{i=1}^s b_i \mathbf{k}_i, \tag{30}$$

$$\mathbf{v}_{n+1} = \mathbf{v}_n + h \sum_{i=1}^s b'_i \mathbf{l}_i, \tag{31}$$

$$\mathbf{k}_i = \mathbf{f} \left(t_n + c_i h, \mathbf{y}_n + h \sum_{j=1}^s a_{ij} \mathbf{k}_j, \mathbf{v}_n + h \sum_{j=1}^s a'_{ij} \mathbf{l}_j \right),$$

$$\mathbf{l}_i = \mathbf{g} \left(t_n + c'_i h, \mathbf{y}_n + h \sum_{j=1}^s a_{ij} \mathbf{k}_j, \mathbf{v}_n + h \sum_{j=1}^s a'_{ij} \mathbf{l}_j \right),$$

with b_i and a_{ij} ($i, j=1, \dots, s$) being real numbers and $c_i = \sum_{j=1}^s a_{ij}$ and analogous definitions for b'_i, a'_{ij} , and c'_i . As an example Table II shows the Butcher tableau for the fourth-order partitioned Runge-Kutta method consisting of a three-stage Lobatto IIIA–IIIB pair. A collection of more Butcher tableaux can be found in Hairer et al. (2002) and Greenspan (2006).

TABLE II. Butcher tableau for the fourth-order partitioned Runge-Kutta method consisting of a three-stage Lobatto IIIA–IIIB pair (Hairer et al., 2002): Left table shows the b_i , a_{ij} , and c_i coefficients corresponding to the table entries as described in the legend of Table I. Right table shows corresponding primed b'_i , a'_{ij} , and c'_i coefficients.

0	0	0	0	0	$\frac{1}{6}$	$-\frac{1}{6}$	0
$\frac{1}{2}$	$\frac{5}{24}$	$\frac{1}{3}$	$-\frac{1}{24}$	$\frac{1}{2}$	$\frac{1}{6}$	$\frac{1}{3}$	0
1	$\frac{1}{6}$	$\frac{2}{3}$	$\frac{1}{6}$	1	$\frac{1}{6}$	$\frac{5}{6}$	0
	$\frac{1}{6}$	$\frac{2}{3}$	$\frac{1}{6}$		$\frac{1}{6}$	$\frac{2}{3}$	$\frac{1}{6}$

D. Application

We have simulated the phase-space trajectories of ions in the linear Paul trap of Fig. 1(b) for 500 000 steps with the Euler method and with the Störmer-Verlet method. A trajectory in the radial plane obtained with the Euler method is shown in Fig. 4(a); it clearly shows an unphysical instability. When simulated by the Störmer-Verlet method, the trajectories are stable, as shown in Fig. 4(b). The small oscillations are due to the micromotion caused by the rf drive. The Euler method does not lead to results which obey energy and phase-space area conservation [see Fig. 4(c)], whereas the Störmer-Verlet method conserves both quantities [see Fig. 4(d)]. The Euler method should be avoided when more than a few simulation steps are performed. The Störmer-Verlet integrator is implemented in the BEM-SOLVER package.

In the next section we discover how we can control the position and motion of the ion.

IV. TRANSPORT OPERATIONS WITH IONS: ILL-CONDITIONED INVERSE PROBLEMS

If we wish to transport ions in multielectrode geometries, the electrostatic potential has to be controlled. An important experimental constraint is that the applicable voltage range is always limited. Additionally, for the dynamic change of the potentials, voltage sources are generally band limited and therefore we need the voltages of the electrodes to change smoothly throughout the transport process. The starting point for the solution of this problem is to consider $A(x_i, j) = A_{ij}$, a unitless matrix representing the potential on all points x_i when each electrode j is biased to 1 V, whereas all other electrodes are kept at 0 V [see Fig. 2(a)]. Hence we can calculate the generated total potential $\Phi(x_i) = \Phi_i$ at any position x_i by the linear superposition

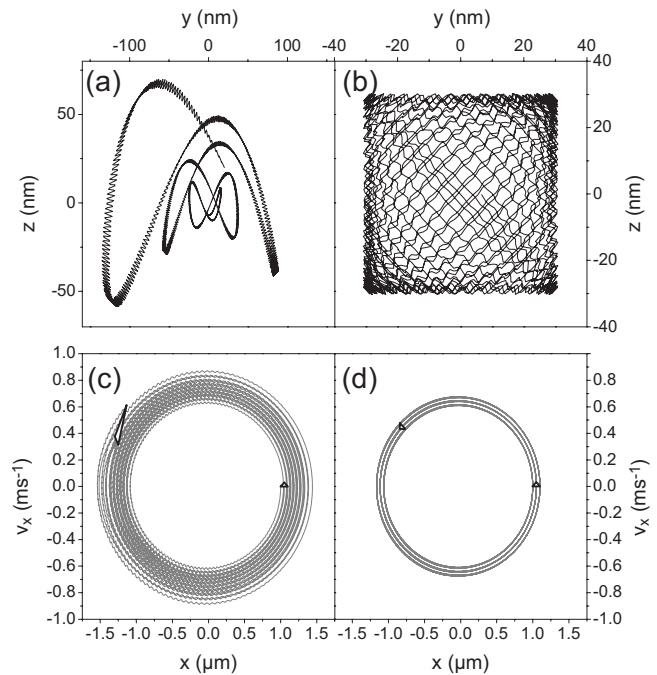


FIG. 4. Comparison of the Euler and Störmer-Verlet simulation methods. (a) Trajectories in the radial plane simulated with the explicit Euler method. It can be seen that the trajectories are unstable. (b) Simulation for the same parameters with the Störmer-Verlet method results in stable trajectories. The small oscillations are due to the micromotion caused by the rf drive. (c) Phase-space trajectories of the harmonic axial motion of an ion numerically integrated with the explicit Euler method. The simulation was performed for a set of three different initial phase-space coordinates, as indicated by the triangles. Energy and phase-space area conservations are violated. (d) Equivalent trajectories numerically integrated with the Störmer-Verlet method which respects energy and phase-space area conservation. The parameters are simulation time $80 \mu\text{s}$, number of simulation steps 4000, $U_{\text{rf}} = 400 \text{ V}_{\text{pp}}$, $\omega_{\text{rf}} = 2\pi \times 12 \text{ MHz}$, and $U_{\text{dc}} = (0, 1, 0, 1, 0) \text{ V}$ for the trap geometry shown in Fig. 1(b).

$$\Phi_i = \sum_{j=1}^N A_{ij} U_j, \quad i = 1, \dots, M, \quad (32)$$

with N denoting the number of separately controllable electrodes and M being the number of grid points in space, which could be chosen, for example, on the trap axis. We want to position the ion at a specific location in a harmonic potential with a desired curvature, i.e., trap frequency. This is a matrix inversion problem since we have specified Φ_i over the region of interest but we need to find the voltages U_j . The problem here is that M is much larger than N , such that the matrix A_{ij} is overdetermined and (due to some unrealizable features or numerical artifacts) the desired potential $\Phi(x_i)$ might not lie in the solution space. Hence a usual matrix inversion, in most cases, will give divergent results due to singularities in the inverse of A_{ij} .

This class of problems is called *inverse problems*, and if singularities occur then they are called *ill-conditioned inverse problems*.

In our case, we wish to determine the electrode voltages for a potential well moving along the trap axis, therefore a series of inverse problems have to be solved. As an additional constraint we require that the electrode-voltage update at each transport step is limited. In the following, we describe how the Tikhonov regularization method can be employed for finding a matrix inversion avoiding singularities and fulfilling additional constraints (Tikhonov and Arsenin, 1977; Press et al., 2007).

A. Tikhonov regularization

For notational simplicity, we always assume that A is a $M \times N$ dimensional matrix, the potentials along the trap axis are given by the vector

$$\Phi = (\Phi(x_1), \Phi(x_2), \dots, \Phi(x_M))^T \quad (33)$$

and $\mathbf{u} = (U_1, U_2, \dots, U_N)^T$ is a vector containing the electrode voltages. Instead of solving the matrix equation $A\mathbf{u} = \Phi$, the Tikhonov method minimizes the residual $\|A\mathbf{u} - \Phi\|^2$ with respect to the Euclidean norm. This alone could still lead to diverging values for some components of \mathbf{u} which is cured by imposing an additional minimization constraint through the addition of a regularization term such as

$$\alpha \|\mathbf{u}\|^2, \quad (34)$$

which penalizes diverging values. The larger the real-valued weighting parameter α is chosen, the more this penalty is weighted, and large values in \mathbf{u} are suppressed at the expense that the residual might increase. Instead of resorting to numerical iterative minimizers a faster and more deterministic method is to perform a singular-value decomposition which decomposes the $M \times N$ matrix A into a product of three matrices $A = USV^T$, where U and V are unitary matrices of dimensions $M \times M$ and $N \times N$ and S is a diagonal (however not quadratic) matrix with diagonal entries s_i and dimension $M \times N$.

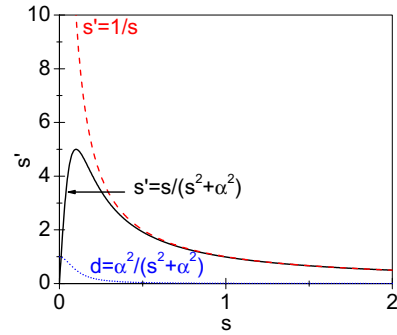


FIG. 5. (Color online) Illustration of the regularization technique: Suppression of the divergence at zero (dashed curve) of the $1/s$ term. The solid curve shows the Tikhonov regularization term, the singular behavior of the $1/s$ inverse is avoided and diverging values are replaced by values near zero. The dotted curve tends towards 1 for $s \rightarrow 0$ where $1/s$ is diverging. It is used to force diverging inverses towards a given value (see text).

Singular-value decomposition routines are contained in many numerical libraries, for example, LAPACK.³ The inverse is then given by $A^{-1} = VS'U^T$, where $S' = S^{-1}$. The diagonal entries of S' are related to those of S by $s'_i = 1/s_i$ such that the singular terms can now be directly identified. The advantage of the singular value decomposition now becomes clear: all the effects of singularities are contained only in the diagonal matrix S' . By addressing these singularities (i.e., when $s_i \rightarrow 0$) in the proper way, we avoid any divergent behavior in the voltages U_j . The easiest way to deal with the singularities would be to set all terms s'_i above a certain threshold to zero. Tikhonov, however, uses the smooth truncation function

$$s'_i = \frac{s_i}{s_i^2 + \alpha^2} \quad (35)$$

(see Fig. 5, solid curve), which behaves as $1/s_i$ for large s_i but tends to zero for vanishing s_i , providing a gradual cutoff. The truncation parameter α has the same meaning as above in the regularization term: the larger α the more the diverging values are forced toward zero, and if $\alpha = 0$ then the exact inverse will be calculated and diverging values are not suppressed at all. The required voltages are now obtained by

$$\mathbf{u} = VS'U^T\Phi. \quad (36)$$

These voltages fulfill the requirement to lie within some given technologically accessible voltage range, which can be attained by iteratively adjusting α .

In the remainder of this section, we present an extension of this method which is better adapted to our specific task of smoothly shuttling an ion between two trapping sites: instead of generally minimizing the electrode voltages, we rather like to limit the changes in the voltage with respect to the voltage configuration \mathbf{u}_0 which is

³<http://www.netlib.org/lapack>.

already applied prior to a single shuttling step. Therefore the penalty function of Eq. (34) is to be replaced by

$$\alpha \|\mathbf{u} - \mathbf{u}_0\|^2. \quad (37)$$

The application of this penalty is achieved through an additional term in Eq. (36)

$$\mathbf{u} = VS'U^T\Phi + VDV^T\mathbf{u}_0. \quad (38)$$

D is a $N \times N$ diagonal matrix with entries $d_i = \alpha^2/(s_i^2 + \alpha^2)$ which shows an opposite behavior as the Tikhonov truncation function of Eq. (35): where the truncation function leads to vanishing voltages avoiding divergencies, d_i tends towards 1 (see Fig. 5, dotted curve) such that the second term in Eq. (38) keeps the voltages close to \mathbf{u}_0 . By contrast for large values of s_i the corresponding values of d_i are vanishing as s_i^{-2} such that the second term in Eq. (35) has no effect. The $N \times N$ matrix V is needed to transform the vector \mathbf{u}_0 into the basis of the matrix D .

A remaining problem now arises when, e.g., a single electrode, i.e., a column in Eq. (32) leads to a singularity. This might be due to the fact that it does generate only small or vanishing fields at a trap site of interest. Even though the singularity is suppressed by the regularization it nevertheless leads to a global reduction of all voltages which adversely affects the accuracy of the method. This can be resolved by the additional introduction of weighting factors $0 < w_j < 1$ for each electrode such that the matrix A is replaced by $A'_{ij} = A_{ij}w_j$. The voltages u'_j obtained from Eq. (38) with accordingly changed matrices U , V , S' , and D are then to be rescaled $1/w_j$. A reasonable procedure would now be to start with all $w_j = 1$ and to iteratively decrease the w_j for electrodes for which the voltage u_j is out of range.

B. Application

A full implementation of the algorithm can be found in the supplied numerical tool box in the SVDREG package. Figure 6 shows the obtained voltages which realize a harmonic trapping potential $\Phi(x) = a(x - x_0)^2$, with $a = 0.03 \text{ V/mm}^2$ at different positions x_0 with a given voltage range of $-10 \leq U_i \leq 10$ for the linear five-segmented trap of Fig. 1(b). We have also experimentally verified the accuracy of the potentials using the ion as a local field probe with subpercent agreement to the numerical results (Huber et al., 2010).

V. QUANTUM DYNAMICS: EFFICIENT NUMERICAL SOLUTION OF THE SCHRÖDINGER EQUATION

As the trapped atoms can be cooled close to the ground state, their motional degrees of freedom have to be described quantum mechanically. In this section we present methods to solve the time-independent Schrödinger equation and the time-dependent Schrödinger equation. The presented tools are used in Secs. VI and VII on optimal control. For trapped ions, the relevance of treating the quantum dynamics of the

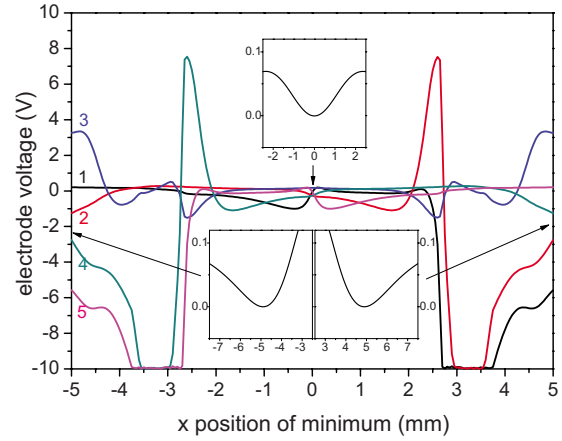


FIG. 6. (Color online) Voltage configurations to put the minimum of a harmonic potential with fixed curvature into different positions. The insets show the resulting potentials obtained by linear superposition of the individual electrode potentials.

motional degrees is not directly obvious as the trapping potentials are extremely harmonic, such that (semi)classical considerations are often fully sufficient. However, full quantum dynamical simulations are important for experiments outside the Lamb-Dicke regime (McDonnell et al., 2007; Poschinger et al., 2010) and for understanding the sources of gate infidelities (Kirchmair et al., 2009). For trapped neutral atoms, the confining potentials are generally very anharmonic such that quantum dynamical simulations are of fundamental importance.

A. Solution of the time-independent Schrödinger equation: The Numerov method

The stationary eigenstates for a given external potential are solutions of the time-independent Schrödinger equation (TISE)

$$\left(-\frac{\hbar^2}{2m} \frac{d^2}{dx^2} - \Phi(x) \right) \psi(x) = E\psi(x). \quad (39)$$

For the harmonic potentials generated with the method of the previous section these solutions are the harmonic oscillator eigenfunctions. But how can we obtain the eigenfunctions and eigenenergies for an arbitrary potential? A typical textbook solution would be choosing a suitable set of basis functions and then diagonalizing the Hamiltonian with the help of linear algebra packages such as LAPACK to obtain the eigenenergies and the eigenfunctions as linear combination of the basis functions.

A simple approach is exploiting the fact that physical solutions for the wave function have to be normalizable. This condition for the wave function leads to the constraint that the wave function should be zero for $x \rightarrow \pm\infty$. Thus it can be guessed from the potential shape where the nonzero parts of the wave function are located in space, and Eq. (39) can be integrated from a starting point outside this region with the eigenenergy as an initial guess parameter. For determining the correct

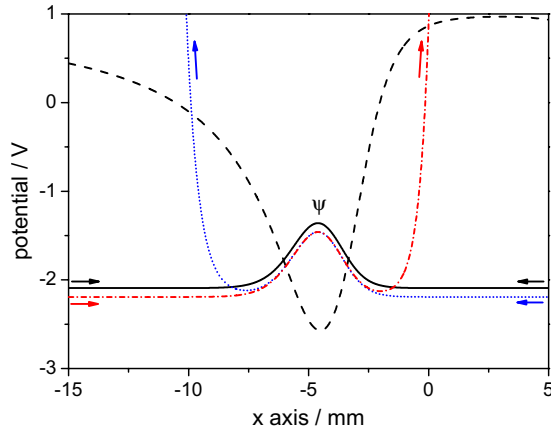


FIG. 7. (Color online) Illustration of the Numerov algorithm for the numerical solution of the TISE. The dashed line shows the trapping potential for a particle. The black solid line shows the ground-state eigenfunction. Dotted and dash-dotted lines show the result of numerical integration starting from right and left if the energy does not correspond to an energy eigenvalue.

energy eigenvalues, we make use of the freedom to start the integration from the left or from the right of this region of interest. Only if correct eigenenergies are chosen as initial guess, the two wave functions will be found to match (see Fig. 7). This condition can then be exploited by a root-finding routine to determine the proper eigenenergies. If the Schrödinger equation is rewritten as $(d^2/dx^2)\psi(x) = g(x)\psi(x)$, then the Numerov method (Blatt, 1967) can be used for integration. Dividing the x axis into discrete steps of length Δx , the wave function can be constructed using the recurrence relation

$$\psi_{n+1} = \frac{\psi_n \left(2 + \frac{10}{12} g_n \Delta x^2 \right) - \psi_{n-1} \left(1 - \frac{1}{12} g_{n-1} \Delta x^2 \right)}{1 - (\Delta x^2/12) g_{n+1}} + \mathcal{O}(\Delta x^6), \tag{40}$$

where $\psi_n = \psi(x+n\Delta x)$ and $g_n = g(x+n\Delta x)$. With this method, the stationary energy eigenstates are obtained. The source code is contained in the OCTTOOL package.

B. Numerical evaluation of the time-dependent Hamiltonian

In order to understand the behavior of quantum systems under the influence of external control field and to devise strategies for their control, we perform numerical simulations of the time evolution. In the case of systems with very few degrees of freedom, the task is simply to solve linear first-order differential equations and/or partial differential equations, depending on the representation of the problem. For mesoscopic systems, the Hilbert space already becomes so vast that one has to find suitable truncations to its regions which are of actual rel-

evance. Here we only deal with the simple case of only one motional degree of freedom and possible additional internal degrees of freedom.

An essential prerequisite for the propagation of quantum systems in time is to evaluate the Hamiltonian; first, one must find its appropriate matrix representation, and second, one needs to find an efficient way to describe its action on a given wave function. The first step is decisive for finding the eigenvalues and eigenvectors, which is often important for a meaningful analysis of the propagation result, and the second step will be necessary for the propagation itself. We assume that we are dealing with a particle without any internal degrees of freedom moving along one spatial dimension x . A further assumption is that the particle is to be confined to a limited portion of configuration space $0 \leq x \leq L$ during the time interval of interest. We can then set up a homogeneous grid

$$x_i = i\Delta x, \quad i = 1, \dots, N, \quad \Delta x = L/N. \tag{41}$$

A suitable numerical representation of the wave function is given by a set of N complex numbers

$$\psi_i = \psi(t, x_i). \tag{42}$$

The potential-energy part of the Hamiltonian is diagonal in position space, and with $V_i = V(x_i)$ it is straightforwardly applied to the wave function:

$$\hat{V}\psi(x) \rightarrow V_i\psi_i. \tag{43}$$

One might now wonder how many grid points are necessary for a faithful representation of the wave function. The answer is given by the Nyquist-Shannon sampling theorem. This theorem states that a band-limited waveform can be faithfully reconstructed from a discrete sampled representation if the sampling period is not less than half the period pertaining to the highest frequency occurring in the signal. Returning to our language, we can represent the wave function *exactly* if its energy is limited and we use at least one grid point per antinode. Of course, one still has to be careful and consider the possible minimum distance of antinodes for setting up a correct grid. Equation (42) then gives an exact representation, and Eq. (43) becomes an equivalence.

The kinetic-energy operator, however, is not diagonal in position space, because the kinetic energy is given by the variation in the wave function along the spatial coordinate, i.e., its second derivative,

$$\hat{T} = -\frac{\hbar^2}{2m} \frac{d^2}{dx^2}. \tag{44}$$

One could then apply \hat{T} by means of finite differences (see Sec. II) which turns out to be extremely inefficient as one would have to use small grid steps (large N) in order to suppress errors. At the very least, we would have to be sure that the grid spacing is much smaller than the minimum oscillation period, which is in complete contrast to the sampling theorem above. In order to circumvent this problem, we consider that \hat{T} is diag-

onal in momentum representation, with the matrix elements

$$\tilde{T}_{nn'} = \langle k_n | \hat{T} | k_{n'} \rangle = \frac{\hbar^2 k_n^2}{2m} \delta_{nn'}. \quad (45)$$

Thus, we can directly apply the kinetic-energy operator on the wave function in momentum space,

$$\hat{T} \tilde{\psi}(k) \rightarrow \tilde{T}_{ii} \tilde{\psi}_i, \quad i = 1, \dots, M, \quad (46)$$

where $\tilde{\psi}(k)$ is the momentum representation of the wave function.

The quantity we need is the position representation of $\psi(x)$ with the kinetic-energy operator applied to it, which gives

$$\begin{aligned} (\hat{T}\psi)_l &= \langle x_l | \hat{T} | \psi \rangle = \sum_{j=1}^N \langle x_l | \hat{T} | x_j \rangle \langle x_j | \psi \rangle \\ &= \sum_{j=1}^N \sum_{n=1}^M \langle x_l | k_n \rangle \langle k_n | \hat{T} | k_n \rangle \langle k_n | x_j \rangle \langle x_j | \psi \rangle \\ &= \frac{1}{M} \sum_{n=1}^M e^{ik_n x_l} \frac{\hbar^2 k_n^2}{2m} \sum_{j=1}^N e^{-ik_n x_j} \psi_j = \sum_{j=1}^N T_{lj} \psi_j, \end{aligned} \quad (47)$$

where $\mathcal{F}_{nj} = \langle k_n | x_j \rangle = e^{-ik_n x_j} / \sqrt{M}$. An explicit expression for the matrix T_{lj} will be given below. After addition of the diagonal potential-energy matrix one obtains the total Hamiltonian in the position representation, it then can be diagonalized by means of computer algebra programs or efficient algorithms as the DSYEVD routine of the computational algebra package LAPACK.

An interesting perspective on the propagation problem is seen when the second last line of Eq. (47) is read from right to left, which gives a direct recipe for the efficient application of the kinetic-energy operator:

- (1) Transform the initial wave function to momentum space by performing the matrix multiplication

$$\tilde{\psi}_n = \sum_{j=1}^N \mathcal{F}_{nj} \psi_j. \quad (48)$$

- (2) Multiply with the kinetic-energy matrix elements

$$\tilde{\psi}'_n = \tilde{T}_{nn} \tilde{\psi}_n. \quad (49)$$

- (3) Transform back to position space by performing another matrix multiplication

$$(\hat{T}\psi)_l = \sum_{n=1}^M \mathcal{F}_{ln}^* \tilde{\psi}'_n. \quad (50)$$

These three steps can, of course, be merged into only one matrix multiplication, but the crucial point here is to notice that the matrix multiplications are nothing more

than a discrete Fourier transform (DFT), which can be performed on computers with the fast Fourier transform (FFT) algorithm (Cooley and Tukey, 1965). This has the tremendous advantage that instead of the N^2 scaling for matrix multiplication, the scaling is reduced to $N \log N$.

Up to now, we have made no statement on how the grid in momentum space defined by k_n and M is to be set up. The usage of FFT algorithms strictly requires $M=N$. The grid in momentum space is then set up by

$$k_n = n\Delta k, \quad -\frac{N}{2} + 1 \leq n \leq \frac{N}{2}, \quad \Delta k = \frac{2K}{N}, \quad (51)$$

analogously to Eq. (41). The maximum kinetic energy is then $T_{\max} = \hbar^2 K^2 / 2m$. The remaining free parameter for a fixed position space grid determined by L and N is then the maximum wave number K . Its choice is motivated by the sampling theorem: the position space step Δx is to be smaller than the minimum nodal distance $\lambda_{\min} / 2 = \pi / K$, which establishes

$$\Delta x = \beta \frac{\pi}{K} = \beta \frac{2\pi}{\Delta k N}. \quad (52)$$

β is a ‘‘safety factor’’ which is to be chosen slightly smaller than 1 to guarantee the fulfillment of the sampling theorem. For clarity, we note that Eq. (52) is equivalent to

$$L/N = \beta \pi / K. \quad (53)$$

The optimum number of grid points N_{opt} for efficient but accurate calculations is then determined by energy conservation, i.e., the grid should provide a maximum possible kinetic energy T_{\max} equal to the maximum possible potential energy V_{\max} . The latter is determined by the specific potential pertaining to the physical problem under consideration, whereas T_{\max} is directly given by the grid step. We can therefore state that

$$\begin{aligned} V_{\max} = T_{\max} &= \frac{\hbar^2 K^2}{2m} = \frac{1}{2m} \left(\frac{\beta \pi \hbar N}{L} \right)^2 \\ \Rightarrow N_{\text{opt}} &= \frac{L}{\beta \pi \hbar} \sqrt{2m V_{\max}}. \end{aligned} \quad (54)$$

If more grid points are chosen, the computational effort increases without any benefit for the accuracy. In turn the results become inaccurate for fewer grid points. If we consider a harmonic oscillator with $V(x) = \frac{1}{2} m \omega^2 (x - L/2)^2$, we obtain $V_{\max} = \frac{1}{8} m \omega^2 L^2$ and therefore

$$N_{\text{opt}} = m \omega L^2 / \beta \hbar, \quad (55)$$

which is (for $\beta=1$) exactly the number of eigenstates sustained by the grid.

Now, the recipe for the application of the kinetic-energy operator can be safely performed, while in Eqs. (48) and (50) the matrix multiplications are simply to be

replaced by forward and backward FFTs, respectively.⁴ This fast Fourier grid method was first presented by [Feit *et al.* \(1982\)](#) and [Kosloff and Kosloff \(1983\)](#). For a review on propagation schemes, see [Kosloff \(1988\)](#).

With the grid having been set up, we are in good shape to calculate the eigenstates of a given Hamiltonian by matrix diagonalization as mentioned. The explicit expression for the position space matrix elements of Eq. (47) is ([Tannor, 2007](#))

$$T_{lj} = \frac{\hbar^2}{2m} \begin{cases} \frac{K^2}{3} \left(1 + \frac{2}{N^2}\right) & \text{for } l=j \\ \frac{2K^2}{N^2} \frac{(-1)^{j-l}}{\sin^2[\pi(j-l)/N]} & \text{otherwise.} \end{cases} \quad (56)$$

The efficiency of the Fourier method can be significantly increased for problems with anharmonic potentials, as is the case with the Coulomb problem in molecular systems. If one is looking at the classical trajectories in phase-space, these will have distorted shapes, such that only a small fraction of phase-space is actually occupied by the system. This leads to an inefficient usage of the grid, unless an inhomogeneous grid is used. The Nyquist theorem can be invoked *locally*, such that the local de Broglie wavelength $\lambda_{dB} = 2\pi[2m\{E_0 - V(x)\}]^{-1/2}$ is to be used as the grid step. Further information can be found in [Fattal *et al.* \(1996\)](#), [Kokoouline *et al.* \(1999\)](#), and [Willner *et al.* \(2004\)](#).

The general propagator for time-dependent Hamiltonians is given by

$$\hat{U}(t, t_0) = \hat{T} \exp \left[-\frac{i}{\hbar} \int_{t_0}^t \hat{H}(t') dt' \right], \quad (57)$$

with the time-ordering operator \hat{T} and a Hamiltonian consisting of a static kinetic-energy part and a time-dependent potential-energy part, which one may write as

$$\hat{H}(t) = \hat{T} + \hat{V}(t). \quad (58)$$

We discretize the problem by considering a set of intermediate times t_n , which are assumed to be equally spaced. The propagator is then given by

$$\hat{U}(t, t_0) = \hat{T} \prod_n e^{-(i/\hbar)\hat{H}(t_n)\Delta t}. \quad (59)$$

It is important to state that the time-ordering operator now just keeps the factors ordered, with decreasing t_n from left to right. Here the first approximation has been

⁴Two caveats for handling FFTs shall be mentioned. First, some FFT algorithms do not carry out the normalization explicitly, such that one has to multiply the resulting wave function with the correct normalization factor after the backward FFT. Second, one is initially often confused by the way the data returned by the FFT is stored: the first component typically pertains to $k=0$, with k increasing by $+\Delta k$ with increasing array index. The negative k components are stored from end to beginning, with $k=-\Delta k$ as the last array element, with k changing by $-\Delta k$ with decreasing array index.

made by replacing the control field by its piecewise constant simplification $\hat{V}(t_n)$. The short term propagators in the product of Eq. (59) have to be subsequently applied to the initial wave function. The remaining problem with the application of the short time propagators then arises due to the noncommutativity of \hat{T} and $\hat{V}(t_n)$. A possible solution would be the diagonalization of $\hat{T} + \hat{V}(t_n)$ in matrix representation, which is highly inefficient due to the unfavorable scaling behavior of matrix diagonalization algorithms. Two main solutions for this problem are widely used, namely, the split-operator technique and polynomial expansion methods, which are to be explained in the following.

C. The split-operator method

The basic idea of the split-operator method is to simplify the operator exponential using the product

$$e^{-(i/\hbar)\hat{H}(t_n)\Delta t} \approx e^{-(i/2\hbar)\hat{V}(t_n)\Delta t} e^{-(i/\hbar)\hat{T}\Delta t} e^{-(i/2\hbar)\hat{V}(t_n)\Delta t} \quad (60)$$

at the expense of accuracy due to violation of the noncommutativity of the kinetic- and potential-energy operators ([Tannor, 2007](#)). The error scales as Δt^3 if $\hat{V}(t_n)$ is taken to be the averaged potential over the time interval Δt ([Kormann *et al.*, 2008](#)) with t_n being the midpoint of the interval. Additional complexity arises if one is dealing with internal degrees of freedom, such that distinct states are coupled by the external control field. This is exactly the case for light-atom or light-molecule interaction processes. In these cases no diagonal representation of \hat{V} exists in position space, meaning that it has to be diagonalized.

D. The Chebyshev propagator

A convenient way to circumvent the problems associated with the split-operator method is to make use of a polynomial expansion of the propagator,

$$\exp \left(-\frac{i}{\hbar} \int_{t_0}^t \hat{H}(t') dt' \right) = \sum_k a_k \mathcal{H}_k(\hat{H}), \quad (61)$$

and exploit the properties of these polynomials. As we will see, an expansion in terms of Chebyshev polynomials \mathcal{H}_k leads to a favorable convergence behavior and a simple implementation due to the recurrence relation

$$\mathcal{H}_{k+1}(x) = 2x\mathcal{H}_k(x) - \mathcal{H}_{k-1}(x), \quad (62)$$

with $\mathcal{H}_0(x)=1$ and $\mathcal{H}_1(x)=x$. As Chebyshev polynomials are defined on an interval $x \in [-1, 1]$, the energy has to be mapped on this interval by shifting and rescaling:

$$\hat{H}' = 2 \frac{\hat{H} - E_{<}}{E_{>} - E_{<}} - 1, \quad (63)$$

where $\mathbb{1}$ is the unity matrix, and $E_{>}$ and $E_{<}$ denote the maximum and minimum eigenvalues of the unscaled

Hamiltonian, respectively. The propagation scheme is then as follows:

- (1) Given the initial wave function $\psi(t=t_i)$, set

$$\phi_0 = \psi(t=t_i), \quad \phi_1 = -i\hat{H}'\phi_0. \quad (64)$$

- (2) Calculate

$$\phi_{n+1} = -2i\hat{H}'\phi_n + \phi_{n-1}, \quad (65)$$

for all $n < n_{\max}$, which is recursion relation (62) applied on the wave function.

- (3) Sum the final wave function according to

$$\psi(t_n + \Delta t) = e^{-i(2\hbar)(E_{<}+E_{>})\Delta t} \sum_{n=0}^{n_{\max}} a_n \phi_n. \quad (66)$$

The phase factor in front of the sum corrects for the energy rescaling and the expansion coefficients are given by Bessel functions:

$$a_n = \begin{cases} J_0\left(\frac{(E_{>} - E_{<})\Delta t}{2\hbar}\right) & \text{for } n = 0 \\ 2(-i)^n J_n\left(\frac{(E_{>} - E_{<})\Delta t}{2\hbar}\right) & \text{for } n > 0. \end{cases} \quad (67)$$

It is interesting to note that the Chebyshev polynomials are not used explicitly in the scheme. Due to the fact that the Bessel functions $J_n(z)$ converge to zero exponentially fast for arguments $z > n$, the accuracy of the propagation is only limited by the machine precision as long as enough expansion coefficients are used. For time-dependent Hamiltonians, however, the accuracy is limited by the finite propagation time steps Δt , which should be much smaller than the time scale at which the control fields are changing. A detailed account on the accuracy of the Chebyshev method for time-dependent problems is given by [Ndong et al. \(2010\)](#) and [Peskin et al. \(1994\)](#). The suitable number of expansion coefficients can easily be found by simply plotting their magnitudes. The most common error source in the usage of the propagation scheme is an incorrect normalization of the Hamiltonian. One has to take into account that the eigenvalues of the Hamiltonian might change in the presence of a time-dependent control field. A good test if the scheme is working at all is to initialize the wave function in an eigenstate of the static Hamiltonian and then check if the norm is conserved and the system stays in the initial state upon propagation with the control field switched off. Another important point is that the propagation effort is relatively independent of the time step Δt . For larger time steps, the number of required expansion coefficients increases linearly, while the total number of steps only decreases. For extremely small steps, the computational overhead of the propagation steps will lead to a noticeable slowdown. All presented numerical propagators are contained in the OCTTOOL package.

VI. OPTIMIZING WAVE-PACKET MANIPULATIONS: OPTIMAL CONTROL THEORY

Now that we know how to efficiently simulate wave packets in our quantum system and how to manipulate the potentials, we can begin to think about *designing* our potentials to produce a desired evolution of our wave packet, whether for transport or for more complex operations (such as quantum gates). In this section, we discuss one method for achieving this in detail, namely, optimal control theory (OCT) ([Peirce et al., 1988](#); [Kosloff et al., 1989](#); [Tannor et al., 1992](#); [Somloi et al., 1993](#); [Krotov, 1996, 2008](#); [Zhu and Rabitz, 1998](#); [Sklarz and Tannor, 2002](#); [Khaneja et al., 2005](#)). These methods belong to a class of control known as open loop, which means that we specify everything about our experiment beforehand in our simulation, and then apply the results of optimal control directly in our experiment. This has the advantage that we should not need to acquire constant feedback from the experiment as it is running (an often destructive operation in quantum mechanics). We focus on one particular method prevalent in the literature, known as the Krotov algorithm ([Tannor et al., 1992](#); [Somloi et al., 1993](#); [Krotov, 1996](#)).

A. Krotov algorithm

Optimal control theory came about as an extension of the classical calculus of variations subject to a differential equation constraint. Techniques for solving such problems were already known in the engineering community for some years, but using OCT to optimize quantum-mechanical systems only began in the late 1980s with the work of Rabitz and co-workers ([Peirce et al., 1988](#); [Shi et al., 1988](#)), where they applied these techniques to numerically obtain optimal pulses for driving a quantum system towards a given goal. At this time, the numerical approach for solving the resulting set of coupled differential equations relied on a simple gradient method with line search. In the years that followed, the field was expanded by the addition of more sophisticated techniques that promised improved optimization performance. One of the most prominent amongst these is the Krotov method ([Tannor et al., 1992](#); [Somloi et al., 1993](#); [Krotov, 1996](#)) developed by Tannor and co-workers for problems in quantum chemistry around the beginning of the 1990s, based on Krotov's initial work. This method enjoyed much success, being further modified by Rabitz ([Zhu and Rabitz, 1998](#)) in the late 1990s.

Until this point, OCT had been applied mainly to problems in quantum chemistry, which typically involved driving a quantum state to a particular goal state (known as state-to-state control) or maximizing the expectation value of an operator. The advent of quantum information theory at the beginning of the new millennium presented new challenges for control theory, in particular, the need to perform quantum gates, which are not state-to-state transfers, but rather full unitary operations that map whole sets of quantum states into the desired final states ([Palao and Kosloff, 2002](#); [Calarco et al., 2004](#);

Hsieh and Rabitz, 2008). Palao and Kosloff (2002) extended the Krotov algorithm to deal with such unitary evolutions, showing how the method can be generalized to deal with arbitrary numbers of states. This will become useful for us later when we want to optimize the Cirac-Zoller gate. Other methods for optimal control besides Krotov have also been extensively studied, most notably perhaps being gradient ascent pulse engineering (GRAPE) (Khaneja et al., 2005), but these methods will not be discussed here.

1. Constructing the optimization objective

We now proceed to outline the basics of optimal control theory as it is used for the optimization later in this paper.⁵ We always begin by defining the *objective* which is a mathematical description of the final outcome we want to achieve. For simplicity, we take as our example a state-to-state transfer of a quantum state $|\psi(t)\rangle$ over the interval $t \in [0, T]$. We begin with the initial state $|\psi(0)\rangle$, and the evolution of this state takes place in accordance with the Schrödinger equation

$$i\hbar \frac{\partial}{\partial t} |\psi(t)\rangle = \hat{H}(t) |\psi(t)\rangle, \quad (68)$$

where \hat{H} is the Hamiltonian. (Note that we often omit explicit variable dependence for brevity.) Now assume that the Hamiltonian can be written as

$$\hat{H}(t) = \hat{H}_0(t) + \sum_i \varepsilon_i(t) \hat{H}_i(t), \quad (69)$$

where H_0 is the *uncontrollable* part of the Hamiltonian (meaning physically the part we cannot alter in the laboratory), and the remaining H_i are the *controllable* parts, in that we may affect their influence through the (real) functions $\varepsilon_i(t)$, which we refer to interchangeably as “controls” or “pulses” (the latter originating from the early days of chemical control where interaction with the system was performed with laser pulses). We take as our goal that we should steer the initial state into a particular final state at our final time T , which we call the goal state $|\phi\rangle$. A measure of how well we have achieved the final state is given by the *fidelity*

$$J_1[\psi] \equiv -|\langle \phi | \psi(T) \rangle|^2, \quad (70)$$

which can be seen simply as the square of the inner product between the goal state and the final evolved state. Note that $J_1[\psi]$ is a *functional* of ψ .

The only other constraint to consider in our problem is the dynamical one provided by the time-dependent Schrödinger equation (TDSE) in Eq. (68). We require that the quantum state must satisfy this equation at all times, otherwise the result is clearly nonphysical. If a quantum state $|\psi(t)\rangle$ satisfies Eq. (68), then we must have

$$(\partial_t + i\hat{H}/\hbar) |\psi(t)\rangle = 0, \quad \forall t \in T, \quad \text{where } \partial_t = \partial/\partial t. \quad (71)$$

We can introduce a Lagrange multiplier to cast our constrained optimization into an unconstrained one. Here we introduce the state $|\chi(t)\rangle$ to play the role of our Lagrange multiplier, and hence we write our constraint for the TDSE as

$$\begin{aligned} J_2[\varepsilon_i, \psi, \chi] &\equiv \int_0^T \left[\langle \chi(t) | \left(\partial_t + \frac{i}{\hbar} \hat{H} \right) | \psi(t) \rangle + \text{c.c.} \right] dt \\ &= 2 \operatorname{Re} \int_0^T \langle \chi(t) | \left(\partial_t + \frac{i}{\hbar} \hat{H} \right) | \psi(t) \rangle dt, \end{aligned} \quad (72)$$

where we have imposed that both $|\psi(t)\rangle$ and $\langle \psi(t)|$ must satisfy the TDSE.

2. Minimizing the objective

Now that we have defined our goal and the constraints, we can write our *objective* $J[\varepsilon_i, \psi, \chi]$ as

$$J[\varepsilon_i, \psi, \chi] = J_1[\psi] + J_2[\varepsilon_i, \psi, \chi]. \quad (73)$$

The goal for the optimization is to find the minimum of this functional with respect to the parameters $\psi(t)$, $\chi(t)$, and the controls $\varepsilon_i(t)$. In order to find the minimum, we consider the stationary points of the functional J by setting the total variation $\delta J = 0$. The total variation is simply given by the sum of the variations δJ_ψ (variation with respect to ψ), δJ_χ (variation with respect to χ), and δJ_{ε_i} (variation with respect to ε_i), which we set individually to zero. For our purposes, we define the variation of a functional

$$\delta_\psi F[\psi] = F[\psi + \delta\psi] - F[\psi]. \quad (74)$$

This can be thought of as the change brought about in F by perturbing the function ψ by a small amount $\delta\psi$.

Considering δJ_{ψ} , we have

$$\delta J_{\psi} = \delta J_{1,\psi} + \delta J_{2,\psi}.$$

Using our definition from Eq. (74) for $\delta J_{1,\psi}$ results in

$$\begin{aligned} \delta J_{1,\psi} &= J_1[\psi + \delta\psi] - J_1[\psi] \\ &= -|\langle \phi | \psi(T) + \delta\psi(T) \rangle|^2 + |\langle \phi | \psi(T) \rangle|^2 \\ &= -\langle \psi(T) + \delta\psi(T) | \phi \rangle \langle \phi | \psi(T) + \delta\psi(T) \rangle \\ &\quad + |\langle \phi | \psi(T) \rangle|^2 \\ &= -\langle \delta\psi(T) | \phi \rangle \langle \phi | \psi(T) \rangle - \langle \psi(T) | \phi \rangle \langle \phi | \delta\psi(T) \rangle \\ &\quad - |\langle \phi | \delta\psi(T) \rangle|^2 \\ &= -2 \operatorname{Re}\{\langle \psi(T) | \phi \rangle \langle \phi | \delta\psi(T) \rangle\} - |\langle \phi | \delta\psi(T) \rangle|^2. \end{aligned} \quad (75)$$

The last term is $\mathcal{O}(\delta\psi(T)^2)$, and since $\delta\psi(T)$ is small we set these terms to zero. Hence, we have finally

$$\delta J_{1,\psi} = -2 \operatorname{Re}\{\langle \psi(T) | \phi \rangle \langle \phi | \delta\psi(T) \rangle\}. \quad (76)$$

Repeating this treatment for $\delta J_{2,\psi}$, we have

⁵For a tutorial on quantum optimal control theory which covers the topics presented here in more detail, see Werschnik and Gross (2007).

$$\begin{aligned}
\delta J_{2,\psi} &= 2 \operatorname{Re} \int_0^T \langle \chi(t) | \left(\partial_t + \frac{i}{\hbar} \hat{H} \right) | \psi(t) + \delta \psi(t) \rangle dt \\
&\quad - 2 \operatorname{Re} \int_0^T \langle \chi(t) | \left(\partial_t + \frac{i}{\hbar} \hat{H} \right) | \psi(t) \rangle dt \\
&= 2 \operatorname{Re} \int_0^T \langle \chi(t) | \left(\partial_t + \frac{i}{\hbar} \hat{H} \right) | \delta \psi(t) \rangle dt \\
&= 2 \operatorname{Re} \left\{ \langle \chi(T) | \delta \psi(T) \rangle - \langle \chi(0) | \delta \psi(0) \rangle \right. \\
&\quad \left. - \int_0^T \left(\langle \chi(t) | \frac{i}{\hbar} \hat{H} - \langle \partial_t \chi(t) | \right) | \delta \psi(t) \rangle dt \right\}. \quad (77)
\end{aligned}$$

Noting that the initial state is fixed, we must have $\delta \psi(0)=0$. Thus setting $\delta J_{\psi}=0$, we obtain

$$\langle \psi(T) | \phi \rangle \langle \phi | \delta \psi(T) \rangle + \langle \chi(T) | \delta \psi(T) \rangle = 0, \quad (78)$$

$$\left(\langle \chi(t) | \frac{i}{\hbar} \hat{H} - \langle \partial_t \chi(t) | \right) | \delta \psi(t) \rangle = 0. \quad (79)$$

Since these must be valid for an arbitrary choice of $|\delta \psi\rangle$, we obtain from Eq. (78) the boundary condition

$$|\chi(T)\rangle = |\phi\rangle \langle \phi | \psi(T)\rangle, \quad (80)$$

and from Eq. (79) the equation of motion

$$i\hbar \partial_t |\chi(t)\rangle = \hat{H} |\chi(t)\rangle. \quad (81)$$

We now continue the derivation by finding the variation δJ_{χ} , which results in the condition already given in Eq. (68), namely, that $|\psi\rangle$ must obey the Schrödinger equation. The variation δJ_{ε_i} results in the condition

$$-\frac{2}{\lambda_i} \operatorname{Im} \left\{ \langle \chi(t) | \frac{1}{\hbar} \frac{\partial \hat{H}}{\partial \varepsilon_i} | \psi(t) \rangle \right\} = 0, \quad (82)$$

where λ_i can be used to suppress updates at $t=0$ and $t=T$. It can be clearly seen that Eq. (82) cannot be solved directly for the controls ε_i since we have a system of split boundary conditions: $|\psi(t)\rangle$ is only specified at $t=0$, and similarly $|\chi(t)\rangle$ only at $t=T$. Hence we require an iterative scheme which will solve the equations self-consistently.

3. Deriving an iterative scheme

The goal of any iterative method will be to reduce the objective J at each iteration while satisfying the constraints. Written mathematically, we simply require that $J^{k+1}-J^k < 0$, where J^k is the value of the functional J evaluated at the k th iteration of the algorithm. We also attach this notation to other objects to denote which iteration of the algorithm we are referring to. Looking at Eq. (82) and taking into account our constraints, the optimization algorithm presents itself as follows:

- (1) Make an initial guess for the control fields $\varepsilon_i(t)$.
- (2) At the k th iteration, propagate the initial state $|\psi(0)\rangle$ until $|\psi^k(T)\rangle$ and store it at each time step t .

- (3) Calculate $|\chi^k(T)\rangle$ from Eq. (80). Our initial guess from step (1) should have been good enough that the final overlap of the wave function with the goal state is not zero otherwise Eq. (80) would give us $|\chi^k(T)\rangle=0$.

- (4) Propagate $|\chi^k(T)\rangle$ backward in time in accordance with Eq. (81). At each time step, calculate the new control at time t ,

$$\varepsilon_i^{k+1}(t) = \varepsilon_i^k(t) + \gamma \frac{2}{\lambda_i} \operatorname{Im} \left\{ \langle \chi^k(t) | \frac{1}{\hbar} \frac{\partial \hat{H}}{\partial \varepsilon_i} | \psi^k(t) \rangle \right\}, \quad (83)$$

which one can identify as a gradient-type algorithm, using Eq. (82) as the gradient. The parameter γ is determined by a line search to ensure our condition that $J^{k+1}-J^k < 0$.

- (5) Repeat steps (2)–(4) until the desired convergence has been achieved.

This gradient-type method, while guaranteeing convergence, is rather slow. A much faster method is what is known as the Krotov method. Here the modified procedure is as follows:

- (1) Make an initial guess for the control fields $\varepsilon_i(t)$.
- (2) Propagate the initial state $|\psi(0)\rangle$ until $|\psi(T)\rangle$.
- (3) At the k th iteration, calculate $|\chi^k(T)\rangle$ from Eq. (80) (again taking care that the final state overlap with the goal state is nonzero).
- (4) Propagate $|\chi^k(T)\rangle$ backward in time in accordance with Eq. (81) to obtain $|\chi(0)\rangle$, storing it at each time step.
- (5) Start again with $|\psi(0)\rangle$ and calculate the new control at time t ,

$$\varepsilon_i^{k+1}(t) = \varepsilon_i^k(t) + \frac{2}{\lambda_i} \operatorname{Im} \left\{ \langle \chi^k(t) | \frac{1}{\hbar} \frac{\partial \hat{H}}{\partial \varepsilon_i} | \psi^{k+1}(t) \rangle \right\}. \quad (84)$$

Use these new controls to propagate $|\psi^{k+1}(0)\rangle$ to obtain $|\psi^{k+1}(T)\rangle$.

- (6) Repeat steps (3)–(5) until the desired convergence has been achieved.

This new method looks similar to the gradient method, except that now we see that we must not use the “old” $|\psi^k(t)\rangle$ in the update, but the “new” $|\psi^{k+1}(t)\rangle$. This is achieved by immediately propagating the current $|\psi^{k+1}(t)\rangle$ with the newly updated pulse, and not the old one. To make this explicit, take $|\psi^{k+1}(0)\rangle = |\psi(0)\rangle$ at $t=0$. We use this to calculate the first update to the controls $\varepsilon_i^{k+1}(0)$ from Eq. (84). We use these controls to find $|\psi^{k+1}(\Delta t)\rangle$, where Δt is one time step of our simulation. We then obtain the next update $\varepsilon_i^{k+1}(\Delta t)$ by again using Eq. (84) where we use the old $|\chi^k(\Delta t)\rangle$ that we had saved from the previous step. In other words, $|\chi(t)\rangle$ is always propagated with the old controls, and $|\psi(t)\rangle$ is always propagated with the new controls.

For a full treatment of this method, see [Sklarz and Tannor \(2002\)](#). For our purposes, we simply note that the method is proven to be convergent, meaning $J^{k+1} - J^k < 0$, and that it has a fast convergence when compared to many other optimization algorithms, notably the gradient method ([Somloi *et al.*, 1993](#)).

B. Application

We now show how the wave function can be controlled via tailored time-dependent external potentials. In ion-trap experiments quantum control of the wave function via individual electrodes is difficult, as these are several orders of magnitude larger than the size of the wave function. However, matching length scales occur for cold atoms trapped in optical lattice potentials ([Calarco *et al.*, 2004](#)) or magnetic microtrap. We nonetheless focus on the ion-trap system and present this as a generic example for quantum control where we want to transport the ion from one place to another without exciting higher motional states. The transport is performed by applying time-dependent voltages $u_i(t)$ to the electrodes generating a total potential

$$\Phi(x, u_1(t), u_2(t), \dots, u_n(t)) = \sum_{i=1}^n \Phi_i(x) u_i(t). \quad (85)$$

The Hamiltonian of this system is

$$H_0(x, u_1(t), \dots, u_n(t)) = -\frac{\hbar^2}{2m} \frac{d^2}{dx^2} + \Phi(x, u_1(t), \dots, u_n(t)). \quad (86)$$

As a target wave function $|\phi\rangle$, we choose a harmonic oscillator ground-state wave function centered at the target position. We thus have to maximize the wave-function overlap $|\langle \phi | \psi(T) \rangle|^2$, where $|\psi(T)\rangle$ is the wave function at the final time T obtained by application of the time-dependent voltages. This is exactly the fidelity functional of Eq. (70) and the initial condition for the Lagrange multiplier of Eq. (80). The update [Eq. (84)] for the control parameters $u_i^k(t)$ from iteration step k to $k+1$ has the form

$$u_i^{k+1}(t) = u_i^k(t) + \frac{2}{\lambda_i} \langle \chi^k(t) | \frac{1}{\hbar} \Phi_i(x) | \psi^{k+1}(t) \rangle. \quad (87)$$

Starting with a sinusoidal-shaped initial guess for the time-dependent voltage configuration $u_i^0(t)$ [see Fig. 8(a)] the wave function at the final time T is excited to the first excited state, leading to a wave-function overlap of 0.3 as seen in Fig. 8(b). After 100 steps of optimization the wave-function overlap has been iteratively increased to 0.997 (Fig. 8) and the motional ground state has been preserved [see Fig. 8(d)]. The optimized time-dependent voltages are shown in Fig. 8(c). The full source code of the optimal control algorithm to perform the presented optimization is contained in the OCTTOOL package.

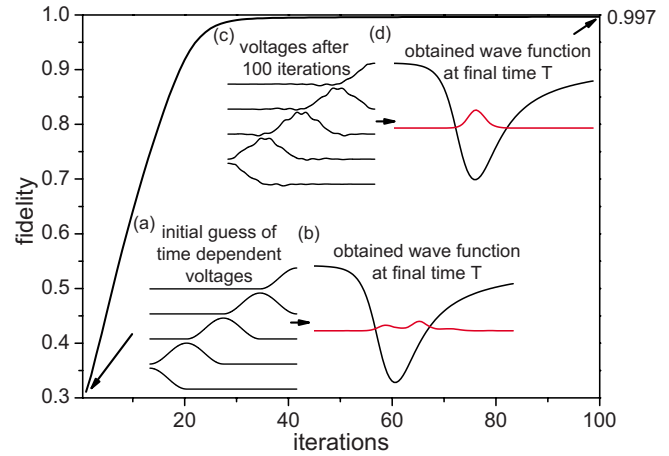


FIG. 8. (Color online) Fidelity increase after iterative optimization steps. (a) Initial guess of the time-dependent voltages applied to the electrodes. (b) Resulting wave function at the final time T . The fidelity is only 0.3. (c) Voltage configurations obtained after 100 iterative calls of the Krotov optimal control method. (d) The final wave function obtained at time T with optimized voltages agrees well with the target ground-state wave function. The fidelity is increased to 0.997.

VII. IMPROVING QUANTUM GATES: OCT OF A UNITARY TRANSFORMATION

Up to now we have only considered the motional degrees of freedom of trapped ions. The internal electronic degrees of freedom serve as information storage for the qubit and are manipulated by laser or microwave radiation. They are subject to quantum dynamics where a certain degree of sophistication can arise due to the presence of interference effects. Joint manipulation of the internal and external degrees of freedom is therefore a promising playground for the application of OCT theory along with the quantum dynamical simulation methods presented in Sec. V. In the following we present the Cirac-Zoller controlled-NOT gate as a case study. We first explain the gate mechanism, then we use the OCT algorithm to derive a laser pulse sequence for the proper realization of the quantum gate.

A. The Cirac-Zoller controlled-NOT gate

Atomic qubits are suitable carriers of quantum information because one can exploit the internal electronic degrees of freedom for realizing a near-perfect two-level system ([Cohen-Tannoudji *et al.*, 2004](#)), where coherent transitions between states can be driven by laser radiation. Suitable long-lived states are found to be sublevels of the electronic ground state connected via stimulated Raman transitions ([Poschinger *et al.*, 2009](#)) or metastable electronic states excited on narrow dipole-forbidden transitions ([Schmidt-Kaler, Gulde, *et al.*, 2003](#)). For neutral atoms the additional possibility of employing Rydberg states exists ([Gaëtan *et al.*, 2009](#)). In the following we explain the [Cirac and Zoller \(1995\)](#) controlled-NOT (CNOT) gate as realized by [Schmidt-Kaler, Häffner, Gulde, *et al.* \(2003\)](#). To understand each single step of

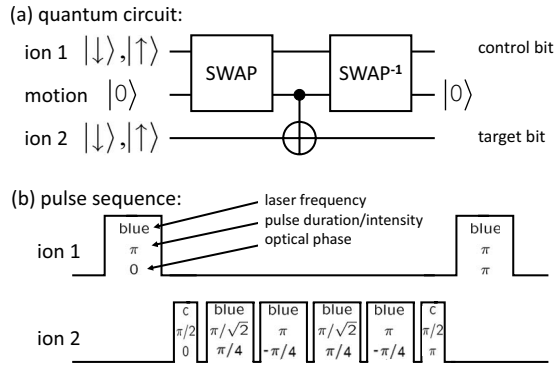


FIG. 9. CNOT gate. (a) Quantum circuit for a CNOT gate between two ions realized by a SWAP operation on the control ion, a CNOT gate between the motional state, and the target ion and a SWAP⁻¹ operation. (b) Composite pulse sequence to realize the total CNOT gate between two ions (Schmidt-Kaler, Häffner, Gulde, *et al.*, 2003; Schmidt-Kaler, Häffner, Riebe, *et al.*, 2003).

wave function. This parameter sets the coupling strength between the laser radiation and the atomic motion. Atomic excitation on the red sideband is accompanied by the lowering of the harmonic oscillator energy by one vibrational quantum.

For a blue laser detuning with $\delta = \omega_{tr}$, the blue sideband interaction is realized. On the blue sideband transitions between the states $|\downarrow, n\rangle$ and $|\uparrow, n+1\rangle$ are driven with Rabi frequency $\Omega \eta \sqrt{n+1}$. When applying a π pulse to the $|\downarrow, 0\rangle$ state we excite one motional quantum and obtain the state $|\uparrow, 1\rangle$. On the other hand, if applied to the $|\uparrow, 0\rangle$ state, no motional quantum can be excited thus this state does not couple to the blue sideband. A π pulse on the blue sideband transition can therefore be used to map quantum information back and forth between the internal state of a specific ion and the motional state of an ion chain if a collective vibrational mode is driven. This operation is referred in the following as a SWAP operation.

Now we have all the tools at hand to put the quantum gate together. In the following one ion will be referred to as control ion whose internal state is denoted by $|\cdot\rangle_c$, and a second target ion with internal state $|\cdot\rangle_t$. We want to flip the state of the target ion conditionally to the state of the control ion, realizing the CNOT truth table,

$$\begin{aligned} |\downarrow\rangle_c |\downarrow\rangle_t &\rightarrow |\downarrow\rangle_c |\uparrow\rangle_t, & |\downarrow\rangle_c |\uparrow\rangle_t &\rightarrow |\downarrow\rangle_c |\downarrow\rangle_t, \\ |\uparrow\rangle_c |\downarrow\rangle_t &\rightarrow |\uparrow\rangle_c |\downarrow\rangle_t, & |\uparrow\rangle_c |\uparrow\rangle_t &\rightarrow |\uparrow\rangle_c |\uparrow\rangle_t. \end{aligned} \quad (94)$$

This gate is performed by the following steps: First the state of the control ion is mapped on a collective vibrational mode of the two ions by means of a SWAP operation. The remaining task is to perform a CNOT gate between the vibrational mode and the target ion and finally perform the SWAP⁻¹ operation to restore the state of the control ion [see Fig. 9(a)]. The CNOT gate between the motional mode and the internal state of the target ion corresponds to the truth table

$$\begin{aligned} |\downarrow, 0\rangle_t &\rightarrow |\uparrow, 0\rangle_t, & |\uparrow, 0\rangle_t &\rightarrow |\downarrow, 0\rangle_t, \\ |\downarrow, 1\rangle_t &\rightarrow |\downarrow, 1\rangle_t, & |\uparrow, 1\rangle_t &\rightarrow |\uparrow, 1\rangle_t, \end{aligned} \quad (95)$$

where the motional state now acts as the control. The key element of this operation is a *controlled phase* gate between these two degrees of freedom, which corresponds to the truth table

$$\begin{aligned} |\downarrow, 0\rangle_t &\rightarrow -|\downarrow, 0\rangle_t, & |\uparrow, 0\rangle_t &\rightarrow |\uparrow, 0\rangle_t, \\ |\downarrow, 1\rangle_t &\rightarrow -|\downarrow, 1\rangle_t, & |\uparrow, 1\rangle_t &\rightarrow -|\uparrow, 1\rangle_t. \end{aligned} \quad (96)$$

As explained, mapping of a superposition phase to the states is accomplished by means of resonant $\pi/2$ pulses. If the controlled phase is sandwiched between two such pulses, then the internal state of the target ion is given by the conditional phase from the phase gate operation. This realizes the CNOT gate of Eq. (95).

The phase gate itself is performed by exploiting, on the one hand, the fact that the blue sideband does not couple to the $|\uparrow, 0\rangle$ state and, on the other hand, that a 2π pulse flips the phase of any given state by π as can be seen from Eq. (93). Therefore a 2π pulse on the blue sideband changes the phase for the states $|\downarrow, 0\rangle$, $|\uparrow, 1\rangle$, and $|\downarrow, 1\rangle$, whereas $|\uparrow, 0\rangle$ is left unchanged. This would result in the conditional phase gate of Eq. (96) such that the whole CNOT gate sequence is complete.

Additional complications arise due to the fact that the blue sideband Rabi frequency on the $|\downarrow, 1\rangle \rightarrow |\uparrow, 2\rangle$ state is larger than the one on the $|\downarrow, 0\rangle \rightarrow |\uparrow, 1\rangle$ transition by a factor of $\sqrt{2}$. The problem was theoretically resolved from a proposal by Childs and Chuang (2000) and realized experimentally by Schmidt-Kaler, Häffner, Gulde, *et al.* (2003) and Schmidt-Kaler, Häffner, Riebe, *et al.* (2003) through the application of a composite pulse sequence of blue sideband pulses with different durations and phases as shown in Fig. 9(b). In the next section we demonstrate how such sequences can be automatically obtained by application of quantum optimal control techniques.

B. Krotov algorithm on unitary transformations

Now we show how the optimal control algorithm from Sec. VI finds a control sequence which realizes the controlled phase gate. The input for the optimal control algorithm is the system dynamics governed by the TDSE and the Hamiltonian H_I from Eq. (92). The subject to be controlled is the unitary transform of Eq. (96). The state of the system is represented by two distinct wave functions in position space for the states $|\uparrow\rangle$ and $|\downarrow\rangle$. We now perform the optimal control algorithm for a unitary transformation along the lines of Palao and Kosloff (2002). Instead of one initial and one target state, we now have four initial states $|\psi_s(0)\rangle$ and four target states $|\phi_s\rangle$ ($s=1, \dots, 4$) corresponding to the states in Eq. (96). Additionally, we have to change the fidelity objective of Eq. (70) to a phase sensitive one with

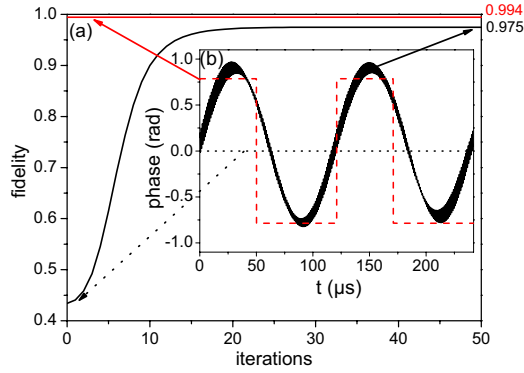


FIG. 10. (Color online) Optimal control of a CNOT gate. (a) Increase in fidelity from 0.43 to 0.975 after 50 iterations (black). The composite pulse sequence realizes a fidelity of 0.994. (b) Time-dependent phase applied on the blue sideband. The dotted curve shows the initial guess $\phi(t)=0$ and the black curve shows the result obtained after 50 iterations of the Krotov algorithm. Note the similarity in the obtained phases concerning amplitude and period when compared to the composite pulse sequence as used in Schmidt-Kaler, Häffner, Gulde, *et al.*, 2003 and Schmidt-Kaler, Häffner, Riebe, *et al.*, 2003 (dashed curve).

$$\tilde{F}[\psi] \equiv \frac{1}{8} \left(\sum_{s=1}^4 \langle \phi_s | \psi_s(T) \rangle \right) + \frac{1}{2}, \quad (97)$$

such that $J_1[\psi] \equiv -\tilde{F}[\psi]$. The constraint is now that the TDSE is to be fulfilled for all four states, thus we introduce four Lagrange multipliers $|\chi_s(t)\rangle$. Equation (72) is now changed into

$$J_2[\varepsilon_i, \psi, \chi] \equiv 2 \sum_{s=1}^4 \operatorname{Re} \int_0^T \langle \chi_s(t) | \left(\partial_t + \frac{i}{\hbar} \hat{H}_I \right) | \psi_s(t) \rangle dt. \quad (98)$$

From these equations we derive the initial condition for the Lagrange multipliers $\langle \chi_s(T) | = \langle \phi_s |$ and the update equation

$$\varepsilon_i^{k+1}(t) = \varepsilon_i^k(t) + \frac{2}{\lambda_{i,s=1}} \sum \operatorname{Im} \left\{ \langle \chi_s^k(t) | \frac{i}{\hbar} \frac{\partial \hat{H}_I}{\partial \varepsilon_i^k} | \psi_s^{k+1}(t) \rangle \right\}. \quad (99)$$

C. Application

We now compare the four pulses on the blue sideband [see Fig. 9(b)] to the pulse found by the optimal control algorithm. For our gate optimization problem we need only one control parameter which is the phase ϕ of the laser on the blue sideband $\varepsilon_1(t) \equiv \phi(t)$ with the initial guess $\phi(t)=0$. Figure 10(a) shows the increase in the fidelity from 0.43 to 0.975 after 50 iterations. The composite pulse sequence used by Schmidt-Kaler, Häffner, Gulde, *et al.* (2003) and Schmidt-Kaler, Häffner, Riebe, *et al.* (2003) achieves a fidelity of 0.994. In both cases, the deviation from unity is caused by off-resonant excitation

on the carrier transition. It is quite remarkable that the Krotov algorithm finds a time-dependent phase $\phi(t)$ with similar amplitude and period as the composite pulse sequence [see Fig. 10(b)]. We have illustrated the OCT method on an example of a quantum control problem where a solution is already known. For more complex control problems, however, this might not be the case, such that OCT allows for tackling control problems where the vastness of Hilbert space and the complexity of quantum interference are hard to handle manually. All sources can be found in the OCTOOL package, where additionally a simulation of the gate operation on the wave function is visualized. Further application of optimal control techniques to quantum gates with ions can be found in the literature (García-Ripoll *et al.*, 2003; Zhu *et al.*, 2006a).

VIII. CONCLUSION

A. Applicability to other qubit types

Currently, trapped ion quantum systems are leading experimental efforts in quantum information theory. But with the growing maturity of quantum information experiments with trapped neutral atoms or solid state systems, we stress that the methods presented here are applicable also to these systems with minor modifications. In this section we elucidate how far each of the methods presented is applicable to each type of qubit and indicate, where appropriate, the connections between them with relevant citations. The methods from Sec. III for the numerical calculation of particle trajectories are directly applicable to neutral atoms, where it might also be interesting to analyze their motional behavior in trap structures such as magnetic microchip traps (Fortágh and Zimmermann, 2007), which have grown greatly in complexity, essentially realizing laboratories on atom chips. The ion-trap community has mimicked the success story of atom chips by the development of microstructured ion traps. For neutral atoms, the full quantum dynamical simulations are of an even higher importance than for ion-trap systems, which is due to the fact that neutral atoms are generally more weakly confined and are therefore much more sensitive to anharmonicities of the trap potentials. Quantum dynamical simulations have been used in conjunction with the optimal control method from Sec. VI to investigate the perspectives for transport and splitting operations in magnetic microtraps and optical lattices (Calarco *et al.*, 2004; Treutlein *et al.*, 2006; Hohenester *et al.*, 2007; De Chiara *et al.*, 2008; Riedel *et al.*, 2010). The optimal control method might turn out to be of essential importance for the realization of robust high-fidelity quantum gates in artificial atom systems such as Josephson-junction-based qubits or impurity-based qubits in solid state host materials (Kane, 1998; Neumann *et al.*, 2008), where the level scheme, the coupling to external control fields, and the decoherence mechanisms are generally more complicated than for trapped ion qubits (Montangero *et al.*, 2007; Spörl *et al.*, 2007). In general, the ability to calcu-

late the quantum dynamics for any kind of engineered quantum system is of fundamental importance, and the methods presented in Sec. V can be directly adapted to any given Hamiltonian. The methods for the precise and efficient calculation of electrostatic fields from Sec. II.A are also of interest for the optimization of electrode geometries for quantum dot qubits based on a two-dimensional electron gas (Koppens *et al.*, 2006). Furthermore, the Laplace equation is of a similar mathematical structure as the Helmholtz equation for ac electromagnetic fields, such that these fields may be calculated in miniaturized microwave traps for neutral atoms or microstructured Josephson devices based on adapted fast multipole methods (Gumerov and Duraiswami, 2004).

B. Summary

We have presented the whole process of ion-trap quantum computing starting from trap design, trapping, and transport of ions, and ending with laser-ion interactions and the simulation and optimization of the Cirac-Zoller CNOT gate starting from basic principles. The explanation of the physics is complemented by a detailed description of the numerical methods needed to perform precise simulations, which are carefully selected such that both precision and efficiency are maintained. Additionally, the numerical methods are presented in a general manner, such that they may be applied to solve physical problems outside of the particular focus of this paper.¹⁰ For fast testing of the routines we have bundled them with the C++ scripting library ROOT.¹¹ By making these libraries accessible to the public we want to inspire students to experiment and provide researchers with a foundation to perform more sophisticated simulations.

ACKNOWLEDGMENTS

This work was supported by the European Commission projects EMALI and FASTQUAST, the Bulgarian NSF Grants No. VU-F-205/06, No. VU-I-301/07, and No. D002-90/08, AQUOTE, SFB TR21, and the Elite program of the Landesstiftung Baden-Württemberg. The authors thank Christiane Koch, Ronnie Kosloff, and David Tannor for useful discussions and Rainer Reichle for his contributions at an early stage of the project.

REFERENCES

- Abramowitz, M., and I. A. Stegun, 1964, *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables* (Dover, New York).
- Amini, J. M., H. Uys, J. H. Wesenberg, S. Seidelin, J. Britton, J. J. Bollinger, D. Leibfried, C. Ospelkaus, A. P. VanDevender, and D. J. Wineland, 2010, *New J. Phys.* **12**, 033031.
- Bakr, W. S., J. I. Gillen, A. Peng, S. Fölling, and M. Greiner, 2009, *Nature (London)* **462**, 74.
- Bakr, W. S., A. Peng, M. E. Tai, R. Ma, J. Simon, J. I. Gillen, S. Fölling, L. Pollet, and M. Greiner, 2010, *Science* **329**, 547.
- Barrett, M. D., J. Chiaverini, T. Schaetz, J. Britton, W. M. Itano, J. D. Jost, E. Knill, C. Langer, D. Leibfried, R. Ozeri, and D. J. Wineland, 2004, *Nature (London)* **429**, 737.
- Benhelm, J., G. Kirchmair, C. F. Roos, and R. Blatt, 2008, *Nat. Phys.* **4**, 463.
- Blatt, J. M., 1967, *J. Comput. Phys.* **1**, 382.
- Blatt, R., and D. Wineland, 2008, *Nature (London)* **453**, 1008.
- Brickman, K.-A., P. C. Haljan, P. J. Lee, M. Acton, L. Deslauriers, and C. Monroe, 2005, *Phys. Rev. A* **72**, 050306(R).
- Brown, L. S., and G. Gabrielse, 1986, *Rev. Mod. Phys.* **58**, 233.
- Calarco, T., U. Dorner, P. S. Julienne, C. J. Williams, and P. Zoller, 2004, *Phys. Rev. A* **70**, 012306.
- Carrier, J., L. Greengard, and V. Rokhlin, 1988, *SIAM J. Sci. Stat. Comput. (USA)* **9**, 669.
- Cheng, H., L. Greengard, and V. Rokhlin, 1999, *J. Comput. Phys.* **155**, 468.
- Chiaverini, J., J. Britton, D. Leibfried, E. Knill, M. D. Barrett, R. B. Blakestad, W. M. Itano, J. D. Jost, C. Langer, R. Ozeri, and D. J. Wineland, 2005, *Science* **308**, 997.
- Chiaverini, J., D. Leibfried, T. Schaetz, M. D. Barrett, R. B. Blakestad, J. Britton, W. M. Itano, J. D. Jost, E. Knill, C. Langer, R. Ozeri, and D. J. Wineland, 2004, *Nature (London)* **432**, 602.
- Childs, A. M., and I. L. Chuang, 2000, *Phys. Rev. A* **63**, 012306.
- Cirac, J. I., and P. Zoller, 1995, *Phys. Rev. Lett.* **74**, 4091.
- Cirac, J. I., and P. Zoller, 2000, *Nature (London)* **404**, 579.
- Cohen-Tannoudji, C., J. Dupont-Roc, and G. Grynberg, 2004, *Atom-Photon Interactions* (Wiley, Weinheim).
- Cooley, J. W., and J. W. Tukey, 1965, *Math. Comput.* **19**, 297.
- Davey, K., and S. Hinduja, 1989, *Appl. Math. Model.* **13**, 450.
- De Chiara, G., T. Calarco, M. Anderlini, S. Montangero, P. J. Lee, B. L. Brown, W. D. Phillips, and J. V. Porto, 2008, *Phys. Rev. A* **77**, 052333.
- DeMarco, B., A. Ben-Kish, D. Leibfried, V. Meyer, M. Rowe, B. Jelenković, W. Itano, J. Britton, C. Langer, T. Rosenband, and D. J. Wineland, 2002, *Phys. Rev. Lett.* **89**, 267901.
- DiVincenzo, D. P., 1995, *Phys. Rev. A* **51**, 1015.
- Duan, L.-M., B. B. Blinov, D. L. Moehring, and C. Monroe, 2004, *Quantum Inf. Comput.* **4**, 165.
- Fattal, E., R. Baer, and R. Kosloff, 1996, *Phys. Rev. E* **53**, 1217.
- Feit, M. D., J. J. A. Fleck, and A. Steiger, 1982, *J. Comput. Phys.* **47**, 412.
- Fickler, R., W. Schnitzler, N. M. Linke, F. Schmidt-Kaler, and K. Singer, 2009, *J. Mod. Opt.* **56**, 2061.
- Fortágh, J., and C. Zimmermann, 2007, *Rev. Mod. Phys.* **79**, 235.
- Gaëtan, A., Y. Miroshnychenko, T. Wilk, A. Chotia, M. Vitteau, D. Comparat, P. Pillet, A. Browaeys, and P. Grangier, 2009, *Nat. Phys.* **5**, 115.
- García-Ripoll, J. J., P. Zoller, and J. I. Cirac, 2003, *Phys. Rev. Lett.* **91**, 157901.
- García-Ripoll, J. J., P. Zoller, and J. I. Cirac, 2005, *Phys. Rev. A* **71**, 062309.
- Greengard, L., and V. Rokhlin, 1988, in *Vortex Methods*, edited by C. Anderson and C. Greengard (Springer, Berlin), p. 121.
- Greengard, L., and V. Rokhlin, 1997, *Acta Numerica* **6**, 229.
- Greenspan, D., 2006, *Numerical Solution of Ordinary Differential Equations* (Wiley, Weinheim).

¹⁰The source code of all methods together with all needed libraries packed into a single installation file can be downloaded from <http://kilian-singer.de/ent> for both Linux and Windows operating systems.

¹¹Website at <http://root.cern.ch>

- Grimm, R., M. Weidemüller, and Y. Ovchinnikov, 2000, *Adv. At., Mol., Opt. Phys.* **42**, 95.
- Gulde, S., M. Riebe, G. P. T. Lancaster, C. Becher, J. Eschner, H. Häffner, F. Schmidt-Kaler, I. L. Chuang, and R. Blatt, 2003, *Nature (London)* **421**, 48.
- Gumerov, N. A., and R. Duraiswami, 2004, *Fast Multipole Methods for the Helmholtz Equation in Three Dimensions* (Elsevier, Amsterdam).
- Gumerov, N. A., and R. Duraiswami, 2005, University of Maryland, Technical Report No. CS-TR-4701.
- Häffner, H., C. F. Roos, and R. Blatt, 2008, *Phys. Rep.* **469**, 155.
- Häffner, H., et al., 2005, *Nature (London)* **438**, 643.
- Hairer, E., C. Lubich, and G. Wanner, 2002, *Geometrical Numerical Integration* (Springer, Berlin).
- Hohenester, U., P. K. Rekdal, A. Borzi, and J. Schmiedmayer, 2007, *Phys. Rev. A* **75**, 023602.
- Hsieh, M., and H. Rabitz, 2008, *Phys. Rev. A* **77**, 042306.
- Huber, G., F. Ziesel, U. Poschinger, K. Singer, and F. Schmidt-Kaler, 2010, *Appl. Phys. B: Lasers Opt.* (to be published).
- Hucul, D., M. Yeo, W. K. Hensinger, J. Rabchuk, S. Olmschenk, and C. Monroe, 2008, *Fortschr. Phys.* **8**, 0501.
- Jackson, J. D., 2009, *Classical Electrodynamics* (Wiley, Weinheim).
- James, D. F. V., 1998, *Appl. Phys. B: Lasers Opt.* **66**, 181.
- Jonathan, D., M. B. Plenio, and P. L. Knight, 2000, *Phys. Rev. A* **62**, 042307.
- Kane, B. E., 1998, *Nature (London)* **393**, 133.
- Khaneja, N., T. Reiss, C. Kehlet, T. Schulte-Herbrüggen, and S. J. Glaser, 2005, *J. Magn. Reson.* **172**, 296.
- Kielpinski, D., C. Monroe, and D. J. Wineland, 2002, *Nature (London)* **417**, 709.
- Kim, K., M.-S. Chang, R. Islam, S. Korenblit, L.-M. Duan, and C. Monroe, 2009, *Phys. Rev. Lett.* **103**, 120502.
- Kim, K., C. F. Roos, L. Aolita, H. Häffner, V. Nebendahl, and R. Blatt, 2008, *Phys. Rev. A* **77**, 050303(R).
- Kirchmair, G., J. Benhelm, F. Zähringer, R. Gerritsma, C. F. Roos, and R. Blatt, 2009, *New J. Phys.* **11**, 023002.
- Kokoouline, V., O. Dulieu, R. Kosloff, and F. Masnou-Seeuws, 1999, *J. Chem. Phys.* **110**, 9865.
- Koppens, F. H. L., C. Buizert, K. J. Tielrooij, I. T. Vink, K. C. Nowack, T. Meunier, L. P. Kouwenhoven, and L. M. K. Vandersypen, 2006, *Nature (London)* **442**, 766.
- Kormann, K., S. Holmgren, and H. O. Karlsson, 2008, *J. Chem. Phys.* **128**, 184101.
- Kosloff, D., and R. Kosloff, 1983, *J. Comput. Phys.* **52**, 35.
- Kosloff, R., 1988, *J. Phys. Chem.* **92**, 2087.
- Kosloff, R., S. Rice, P. Gaspard, S. Tersigni, and D. Tannor, 1989, *Chem. Phys.* **139**, 201.
- Krotov, V. F., 1996, *Global Methods in Optimal Control Theory* (Dekker, New York).
- Krotov, V. F., 2008, *Dokl. Math.* **78**, 949.
- Leibfried, D., R. Blatt, C. Monroe, and D. Wineland, 2003, *Rev. Mod. Phys.* **75**, 281.
- Leibfried, D., B. DeMarco, V. Meyer, D. Lucas, M. Barrett, J. Britton, W. M. Itano, B. Jelenkovic, C. Langer, T. Rosenband, and D. J. Wineland, 2003, *Nature (London)* **422**, 412.
- Leibfried, D., et al., 2005, *Nature (London)* **438**, 639.
- Lukin, M. D., M. Fleischhauer, R. Cote, L. M. Duan, D. Jaksch, J. I. Cirac, and P. Zoller, 2001, *Phys. Rev. Lett.* **87**, 037901.
- Mandel, O., M. Greiner, A. Widera, T. Rom, T. W. Hansch, and I. Bloch, 2003, *Nature (London)* **425**, 937.
- McDonnell, M. J., J. P. Home, D. M. Lucas, G. Imreh, B. C. Keitch, D. J. Szwer, N. R. Thomas, S. C. Webster, D. N. Stacey, and A. M. Steane, 2007, *Phys. Rev. Lett.* **98**, 063603.
- Milburn, G. J., S. Schneider, and D. F. James, 2000, *Fortschr. Phys.* **48**, 801.
- Mintert, F., and C. Wunderlich, 2001, *Phys. Rev. Lett.* **87**, 257904.
- Mølmer, K., and A. Sørensen, 1999, *Phys. Rev. Lett.* **82**, 1835.
- Monroe, C., D. Leibfried, B. E. King, D. M. Meekhof, W. M. Itano, and D. J. Wineland, 1997, *Phys. Rev. A* **55**, R2489.
- Montangero, S., T. Calarco, and R. Fazio, 2007, *Phys. Rev. Lett.* **99**, 170501.
- Nabors, K., F. T. Korsmeyer, F. T. Leighton, and J. White, 1994, *SIAM J. Sci. Comput. (USA)* **15**, 713.
- Ndong, M., H. Tal-Ezer, R. Kosloff, and C. P. Koch, 2010, *J. Chem. Phys.* **132**, 064105.
- Nelson, K. D., X. Li, and D. S. Weiss, 2007, *Nat. Phys.* **3**, 556.
- Neumann, P., N. Mizuochi, F. Rempp, P. Hemmer, H. Watanabe, S. Yamasaki, V. Jacques, T. Gaebel, F. Jelezko, and J. Wrachtrup, 2008, *Science* **320**, 1326.
- Nielsen, M. A., and I. L. Chuang, 2000, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, England).
- Palao, J. P., and R. Kosloff, 2002, *Phys. Rev. Lett.* **89**, 188301.
- Paul, W., 1990, *Rev. Mod. Phys.* **62**, 531.
- Peirce, A. P., M. A. Dahleh, and H. Rabitz, 1988, *Phys. Rev. A* **37**, 4950.
- Peskin, U., R. Kosloff, and N. Moiseyev, 1994, *J. Chem. Phys.* **100**, 8849.
- Poschinger, U. G., G. Huber, F. Ziesel, M. Deiss, M. Hettrich, S. A. Schulz, G. Poulsen, M. Drewsen, R. J. Hendricks, K. Singer, and F. Schmidt-Kaler, 2009, *J. Phys. B* **42**, 154013.
- Poschinger, U. G., A. Walther, K. Singer, and F. Schmidt-Kaler, 2010, e-print [arXiv:1005.5547](https://arxiv.org/abs/1005.5547).
- Poyatos, J. F., J. I. Cirac, and P. Zoller, 1998, *Phys. Rev. Lett.* **81**, 1322.
- Pozrikidis, C., 2002, *A Practical Guide to Boundary Element Methods with the Software Library BEMLIB* (Chapman & Hall/CRC, Boca Raton, FL).
- Press, W. H., S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, 2007, *Numerical Recipes the Art of Scientific Computing*, 3rd ed. (Cambridge University Press, Cambridge, England).
- Reichle, R., D. Leibfried, R. B. Blakestad, J. Britton, J. D. Jost, E. Knill, C. Langer, R. Ozeri, S. Seidelin, and D. J. Wineland, 2006, *Fortschr. Phys.* **54**, 666.
- Riebe, M., H. Häffner, C. F. Roos, W. Hänsel, J. Benhelm, G. P. T. Lancaster, T. W. Körber, C. Becher, F. Schmidt-Kaler, D. F. V. James, and R. Blatt, 2004, *Nature (London)* **429**, 734.
- Riedel, M. F., P. Böhi, Y. Li, T. W. Hänsch, A. Sinatra, and P. Treutlein, 2010, *Nature (London)* **464**, 1170.
- Saad, Y., 2003, *Quantum Computation and Quantum Information* (SIAM, Philadelphia).
- Šašura, M., and V. Bužek, 2002, *J. Mod. Opt.* **49**, 1593.
- Schmidt-Kaler, F., S. Gulde, M. Riebe, T. Deuschle, A. Kreuter, G. Lancaster, C. Becher, J. Eschner, H. Häffner, and R. Blatt, 2003, *J. Phys. B* **36**, 623.
- Schmidt-Kaler, F., H. Häffner, S. Gulde, M. Riebe, G. Lancaster, T. Deuschle, C. Becher, W. Hänsel, J. Eschner, C. Roos, and R. Blatt, 2003, *Appl. Phys. B: Lasers Opt.* **77**, 789.
- Schmidt-Kaler, F., H. Häffner, M. Riebe, S. Gulde, G. P. T. Lancaster, T. Deuschle, C. Becher, C. F. Roos, J. Eschner, and R. Blatt, 2003, *Nature (London)* **422**, 408.
- Schmiedmayer, J., R. Folman, and T. Calarco, 2002, *J. Mod.*

- [Opt. 49, 1375.](#)
- Schulz, S., U. Poschinger, K. Singer, and F. Schmidt-Kaler, 2006, [Fortschr. Phys. 54, 648.](#)
- Shen, L., and Y. J. Liu, 2007, *Acta Numerica* **39**, 681.
- Sherson, J. F., C. Weitenberg, M. Endres, M. Cheneau, I. Bloch, and S. Kuhr, 2010, [Nature \(London\)](#) (to be published).
- Shi, S., A. Woody, and H. Rabitz, 1988, [J. Chem. Phys. 88, 6870.](#)
- Sklarz, S. E., and D. J. Tannor, 2002, [Phys. Rev. A 66, 053619.](#)
- Somlóí, J., V. A. Kazakov, and D. J. Tannor, 1993, [Chem. Phys. 172, 85.](#)
- Spörl, A., T. Schulte-Herbrügge, S. J. Glaser, V. Bergholm, M. J. Storcz, J. Ferber, and F. K. Wilhelm, 2007, [Phys. Rev. A 75, 012302.](#)
- Tannor, D. J., 2007, *Introduction to Quantum Mechanics: A Time-Dependent Perspective* (University Science Books, Sausalito, CA).
- Tannor, D. J., V. A. Kazakov, and V. Orvlov, 1992, in *Time-Dependent Quantum Molecular Dynamics*, edited by J. Broeckhove and L. Lathouwers (Plenum, New York), pp. 347–360.
- Thomas, J. W., 1995, *Numerical Partial Differential Equations* (Springer, New York).
- Tikhonov, A. N., and V. A. Arsenin, 1977, *Solution of Ill-Posed Problems* (Winston & Sons, Washington, D.C.).
- Treutlein, P., T. W. Hänsch, J. Reichel, A. Negretti, M. A. Cirone, and T. Calarco, 2006, [Phys. Rev. A 74, 022312.](#)
- Werschnik, J., and E. K. U. Gross, 2007, [J. Phys. B 40, R175.](#)
- Willner, K., O. Dulieu, and F. Masnou-Seeuws, 2004, [J. Chem. Phys. 120, 548.](#)
- Zhu, S.-L., C. Monroe, and L.-M. Duan, 2006a, [Europhys. Lett. 73, 485.](#)
- Zhu, S.-L., C. Monroe, and L.-M. Duan, 2006b, [Phys. Rev. Lett. 97, 050505.](#)
- Zhu, W., and H. Rabitz, 1998, [J. Chem. Phys. 109, 385.](#)