

Single-atom heat engine driven by non-thermal and quantum reservoirs

MASTER THESIS

for obtaining the degree Master of Science

Institute of Physics

University of Kassel

Submitted from: Moritz Göb
Address: Wolfsschlucht 19
34117 Kassel

Matriculation Number: 33385187
Email Address: moritzgoeb@physik.uni-kassel.de

Department: Institut für Physik
Experimentalphysik I

Examiner: Prof. Dr. Kilian T. T. Singer
Prof. Dr. Martin E. Garcia

Supervisor: Dr. Daqing Wang, Prof. Dr. Kilian T. T. Singer

Submitted: 03.08.2022

Declaration

Herewith I declare, that I have produced the presented paper myself and solely with the aid of the means permitted by the examination regulations of the University of Kassel. The literature used is indicated in the bibliography. I have indicated literally or correspondingly assumed contents as such.

Kassel, 03.08.2022

Moritz Göb

Abstract

Usually heat engines are used to convert heat into work. This work is then used to drive a load. However, the work extracted from the engine can also be measured to infer information about the thermodynamic processes the engine underwent. This assumes prior knowledge of the engines property.

In a recent proof-of-principle experiment a heat engine was realized with a single atom [Roß16b]. In this thesis, the dynamics are modeled by analytic and numeric means. It will be demonstrated how the single-atom heat engine can be used as to estimate the temperature difference between two thermal reservoirs.

Contents

Declaration	III
1. Introduction	1
2. Ion traps	3
2.1. Funnel-shaped Paul trap	5
2.2. Calcium ion	6
3. Fundamental concepts	7
3.1. The harmonic oscillator	7
3.1.1. Damped harmonic oscillator	7
3.1.2. Driven harmonic oscillator	9
3.1.3. Langevin equation	10
3.1.4. Quantum Harmonic oscillator	12
3.2. Density operators and Liouville equation	15
3.3. Heat engines	17
3.3.1. The Carnot cycle	17
3.3.2. The Otto cycle	19
3.3.3. The Stirling cycle	20
3.3.4. Single-atom heat engine	21
4. Numerical methods	23
4.1. Störmer-Verlet propagator	23
4.2. Monte-Carlo simulation	24
4.3. Newton propagator	24
5. Single-atom heat engine as sensitive probe	27
5.1. Dynamics of the single-ion heat engine	27
5.1.1. Thermodynamic cycle	30
5.2. Numerical simulation	33
5.2.1. Algorithm for the simulation of classical and quantum trajectories	33
5.3. Single-ion heat engine as a sensitive thermal probe	44
5.3.1. Temperature difference estimation	44
5.3.2. The squeezed engine: Enhancing sensitivity using quantum resources	47
6. Summary and Outlook	51

A. Appendix	53
A.1. Laplace Transform	53
A.1.1. Solving the driven harmonic oscillator using the Laplace transform method	53
A.2. Implementation of the Strömer-Verlet method	55
B. List of publications	57
Bibliography	58

1. Introduction

The development of thermal devices started a huge social and economic change. The introduction of the steam engine enabled the development of automatized fabrication plants, which started off the industrial revolution. Furthermore, steam engines were used for transportation such as locomotives and ships. The spread of thermal devices yielded also an interest of the nature of the operation of the devices them self, which interested researchers to develop theoretical description to optimize the operation of the machines. This launched the field which is known as thermodynamics. In recent years technical devices are being miniaturized down to a scale where quantum effects play a non-negligible role. To study especially the behavior of thermal properties of such small devices it is important to find platforms on which one can study the nature of thermodynamics on a quantum level. To investigate thermodynamics at a single quantum level, a robust control over the system is required. Due to recent development in different fields, many different suitable platforms have emerged. Promising platforms for studying the nature of thermodynamics in the quantum regimes are cold atoms [Nie19], nitrogen vacancy centers in diamond [Mic21] and trapped ions [Roß16b; Lin19]. We study a trapped ion confined by a tapered potential in which a heat engine was successfully demonstrated [Roß16a; Roß16b]. However, the operation range of this single-atom heat engine was performed in a classical regime. To observe properties on a quantum level, more sophisticated control and schemes need to be employed. Recently schemes were developed to exceed the adiabatic limits for a single/atom heat engine using short-cuts to adiabaticity (STA) [Tor18; Göb19]. Furthermore, the sideband resolved quadrupole transition in $^{40}\text{Ca}^+$ was recently measured in our laboratory [Mas21].

In this thesis, a brief introduction to ion trapping is given in chapter 2. The underlying theoretical concepts will be elaborated on in chapter 3. In chapter 5 the method to measure small temperature differences with the single-atom heat engine will be demonstrated. An outlook on further research is given in chapter 6.

2. Ion traps

"We never experiment with just one electron or atom or (small) molecule. In thought-experiments we sometimes assume that we do; this invariably entails ridiculous consequences." (Erwin Schrödinger, 1952 [Sch52])

This statement of Erwin Schrödinger does not hold anymore. Research on single atoms are nowadays performed on a daily bases in laboratories. One of such experimental platforms is the quadrupole trap which is often referred to as Paul trap. They have many applications, due to their well controllable dynamics.

The ions can be prepared in the ground state [Lei03] and via coupling to engineered reservoirs [Tur00] the ion can be prepared into a selective state deterministically [Roo00]. This techniques can be utilized for employing high fidelity gates between single and multiple ions [Møl99; Sch03; Lem13], allowing for manipulation of information encode in the internal quantum states of the ions. This allows to generate maximally entangled states [Lin13; Kau17] as well as quantum information processing [Lei03; Gar05; Kau20]. Another field where trapped ions are getting used in metrology. A clock with a systematic uncertainty below 10^{-18} was implemented with trapped $^{27}\text{Al}^+$ [Bre19].

To resolve and address single atoms using visible light, a sufficiently large separation is required. This is not feasible in bulk materials. Wolfgang Paul conceived a technique which made it possible to trap single atoms in vacuum. For this innovation he was awarded the Nobel prize in 1989 [Pau90]. The quadrupole ion trap, which is also referred to as Paul trap, uses a combination of electro-static and dynamic potentials for confinement. This is necessary due to *Earnshaw's theorem*, which prohibits the trapping of a charged particle by solely utilizing static electric potentials. The dynamic voltages are here in the radio frequency (RF) regime. The general time-dependent potential Φ can be expressed as [Lei03]

$$\begin{aligned} \Phi(x, y, z, t) = & \frac{U_{\text{DC}}}{2} (\alpha_{\text{DC}}x^2 + \beta_{\text{DC}}y^2 + \gamma_{\text{DC}}z^2) \\ & + \frac{U_{\text{RF}}}{2} \cos(\omega_{\text{RF}}t) (\alpha_{\text{RF}}x^2 + \beta_{\text{RF}}y^2 + \gamma_{\text{RF}}z^2). \end{aligned} \quad (2.1)$$

U_{DC} and U_{RF} denotes the constant direct current (DC) and RF trapping voltages respectively. Frequency with which the RF voltages are driven are denoted with ω_{RF} . The potential needs to fulfill the Laplace equation in a charge-free space.

$$\Delta\Phi(x, y, z, t) = 0 = U_{\text{DC}} (\alpha_{\text{DC}} + \beta_{\text{DC}} + \gamma_{\text{DC}}) + U_{\text{RF}} \cos(\omega_{\text{RF}}t) (\alpha_{\text{RF}} + \beta_{\text{RF}} + \gamma_{\text{RF}}) \quad (2.2)$$

By separating this equation in a static DC and a dynamic RF part, one can obtain simple relations between the parameters themselves.

$$\begin{aligned} 0 &= \alpha_{\text{DC}} + \beta_{\text{DC}} + \gamma_{\text{DC}} \\ 0 &= \alpha_{\text{RF}} + \beta_{\text{RF}} + \gamma_{\text{RF}} \end{aligned} \quad (2.3)$$

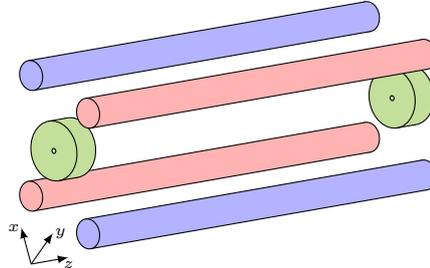


Figure 2.1.: Schematic of the linear Paul trap.

As an example for a possible combination, a linear Paul trap, which is depicted in chapter 2, will be discussed in the following. Many more combinations can be utilized, but this is beyond the scope of this thesis. For further elaborations see [Sin10]. In this linear configuration $\gamma_{\text{RF}} = 0$. Therefore, the parameters can be determined by

$$\begin{aligned} -\alpha_{\text{DC}} &= \beta_{\text{DC}} + \gamma_{\text{DC}} \\ \alpha_{\text{RF}} &= -\beta_{\text{RF}}. \end{aligned} \quad (2.4)$$

The radial degrees of freedom are here confined with an oscillating saddle potential. The axial confinement is implemented using a static harmonic potential. The trajectory of a single ion with mass m and charge q in this potential is described by

$$m\ddot{\vec{r}} = -q\nabla\Phi(\vec{r}, t), \quad (2.5)$$

with $\vec{r} = (x, y, z)^T$. This yields a set of uncoupled Mathieu differential equations in the x and y -direction.

$$\frac{d^2 r_i}{d\xi^2} + [a_i - 2q_i \cos(2\xi)] r_i(\xi) = 0, \quad (2.6)$$

Here $i = x, y$ and $2\xi = \omega_{\text{RF}}t$. The parameter a_i and q_i are determined by the geometry of the trap. In the case of this example, they can be derived to be

$$\begin{aligned} q_x &= \frac{2|q|U_{\text{RF}}\alpha_{\text{RF}}}{m\omega_{\text{RF}}^2}, \quad a_x = -\frac{4|q|U_{\text{DC}}\alpha_{\text{DC}}}{m\omega_{\text{RF}}^2} \\ q_y &= -\frac{2|q|U_{\text{RF}}\beta_{\text{RF}}}{m\omega_{\text{RF}}^2}, \quad a_y = \frac{4|q|U_{\text{DC}}\beta_{\text{DC}}}{m\omega_{\text{RF}}^2}. \end{aligned} \quad (2.7)$$

If $0 \leq \beta_i \leq 1$, with $\beta_i = \sqrt{a_i + \frac{q_i^2}{2}}$, the solutions are stable. When $|a_i|, q_i^2 \ll 1$ a solution of Equation (2.6) can be found [Lei03; Sin10].

$$r_i(t) = r_i(0) \cos(\omega_i t) \left(1 + \frac{q_i}{2} \cos(\omega_{\text{RF}} t) \right) \quad (2.8)$$

The ion undergoes a slow secular harmonic oscillation at the frequency $\omega_i = \beta_i \frac{\omega_{\text{RF}}}{2}$ and a fast but small micromotion, which are oscillations at the frequency of the radio drive. The confinement along the z axis is a harmonic potential with the frequency

$$\omega_z = \sqrt{\frac{|q|U_{\text{DC}}\gamma_{\text{DC}}}{m}}. \quad (2.9)$$

Since the micromotion is comparably fast with a small amplitude, during a secular period the influence is negligible small. Furthermore, the amplitude can be further minimized by using additional electrodes [Ber98]. Therefore, the potential can be approximated to be harmonic.

$$\Phi_{\text{p}}(x, y) = \frac{|q|\|\nabla\Phi(x, y, z, 0)\|^2}{4m\omega_{\text{RF}}^2} \quad (2.10)$$

2.1. Funnel-shaped Paul trap

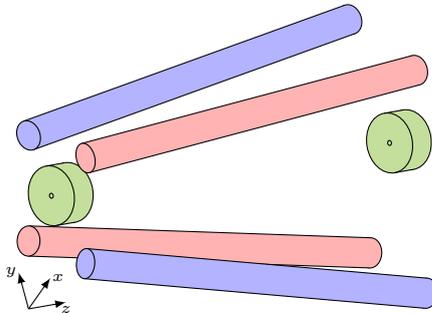


Figure 2.2.: Schematic of a funnel-shaped linear Paul trap.

In constructing linear ion traps it is taken great care of, that the linear RF electrodes are aligned in parallel. Even the slightest deviation gives rise to an undesired coupling of the degrees of freedom. However, configuring the trapping electrodes in a tapered geometry yields interesting and yet unexplored physics. The funnel-shaped trap, which is depicted in fig. 2.2, generates the following RF potential.

$$\Phi(x, y, z, t) \propto \frac{U_{\text{RF}} \cos(\omega_{\text{RF}}t)}{(r_0 + z \tan(\vartheta))^2} (x^2 - y^2) + \frac{U_{\text{DC}}}{z_0^2} z^2. \quad (2.11)$$

Here r_0 describes the distances of the RF electrodes, ϑ the angle with which the electrodes are tilted away from the z -axis and z_0 the distance of the DC electrodes. A pseudo potential can be derived for this set up as well using Equation (2.10).

$$\Phi_{\text{p}} = \frac{m}{2} \frac{\omega_{x,0}^2 x^2 + \omega_{y,0}^2 y^2}{(r_0 + z \tan(\vartheta))^4} + \frac{m}{2} \omega_z^2 z^2 \quad (2.12)$$

Here $\omega_{i,0}$ with $i = x, y$ denotes the radial trapping frequency at $z = 0$. The trapping frequency in radial direction is dependent on the axial position of the ion. Furthermore, the tapering of the trap leads to a coupling of the radial and the axial degrees of freedom [Roß16a]. This will be further discussed in chapter 5.

2.2. Calcium ion

The common atomic ions to trap in a Paul trap are earth-alkali ions. Their remaining single valence electron yields an easy control, because the structure is hydrogen like. In our setup we employ $^{40}\text{Ca}^+$. The electron configuration is $[\text{Ar}]4s^2S_{1/2}$. The corresponding energy levels are depicted in Figure 2.3.

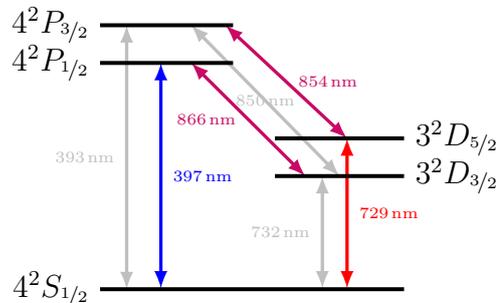


Figure 2.3.: Energy level scheme of $^{40}\text{Ca}^+$. The blue arrow denotes the dipole transition used for Doppler cooling, the red arrow denotes the quadrupole transition for ground state cooling and the magenta arrows denote the transitions addressed by the repump laser. The remaining transitions are not addressed by lasers in the experiment.

Not all transitions in Figure 2.3 are relevant. In fact, it is sufficient to use four of the depicted transitions, which are colored, to obtain full control of the system. The transition $4s^2S_{1/2} \leftrightarrow 4p^2P_{1/2}$ has a short lifetime of $\tau_P = 6.9$ ns [Het15]. This short lifetime can be utilized for laser cooling [Lei03]. The ion decays from the $4p^2P_{1/2}$ to the $3d^2D_{3/2}$ state with a probability of 6.4% [Ram13]. Because further relaxation to the ground state is dipole forbidden, this state has a long lifetime of $\tau_D = 1.18$ s [Kre05]. In order to counteract this relaxation, a light field near 866 nm is used to re-pump the ion to the $4p^2P_{1/2}$ state. However, it can be desirable to exploit this long lifetime. Especially for quantum information processing application a long qubit lifetime will yield a more robust system for executing algorithms. For this the transition $4s^2S_{1/2} \leftrightarrow 3d^2D_{5/2}$ can be used. Due to this long lifetime of the transition, the linewidth is comparably small. This allows for sideband-resolved operations. To close this transition, the repumper at 854 nm transfers the population to the $4p^2P_{3/2}$ state, from which it decays at 393 nm to the ground state. These wavelengths are filtered out by a bandpass filter, to not be detected by the camera.

3. Fundamental concepts

3.1. The harmonic oscillator

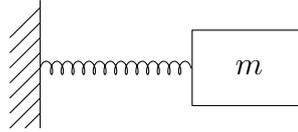


Figure 3.1.: Depiction of a mass m at the end of a coil spring.

One of the most studied problems in physics is the harmonic oscillator. The differential equation reads

$$m\ddot{x} = -kx \quad (3.1)$$

with m being the mass, x the displacement and k the spring constant. This law is also known as Hooke's law. Due to frequent encountering of the differential equation describing the problem in different fields, it has many analogues. With $k = m\omega^2$ it follows

$$m\ddot{x} = -m\omega^2x, \quad (3.2)$$

with the angular frequency ω . The dynamics of this simple case can be obtained by integrating the equation of motion

$$x(t) = x_0 \cos(\omega t + \varphi). \quad (3.3)$$

Here x_0 denotes the maximal elongation and φ the phase, giving the elongation at $t = 0$.

3.1.1. Damped harmonic oscillator

The equations derived in the previous chapter often does not reflect the physical reality. The above assumes a frictionless system. This assumption does often not reflect reality. To account for the loss of energy due to friction a dispersion or damping parameter γ is introduced. Thus, the equation of motion reads

$$\ddot{x} + \gamma\dot{x} + \omega^2x = 0. \quad (3.4)$$

The solution for the ordinary differential equation eq. (3.4) is obtained by making an ansatz for an exponentially decaying function with a free parameter λ .

$$x(t) = x_0 e^{\lambda t} \quad (3.5)$$

Inserting eq. (3.5) into eq. (3.4) yields the equation to determine λ .

$$\begin{aligned} 0 &= \lambda^2 x_0 e^{\lambda t} + \gamma \lambda x_0 e^{\lambda t} + x_0 e^{\lambda t} \\ \Leftrightarrow 0 &= \lambda^2 + \gamma \lambda + \omega^2 \end{aligned} \quad (3.6)$$

λ can now be determined solving eq. (3.6).

$$\lambda_{1,2} = -\frac{\gamma}{2} \pm \sqrt{\frac{\gamma^2}{4} - \omega^2} = -\frac{\gamma}{2} \pm \omega_D \quad (3.7)$$

As λ can take different values, we need to consider the general solution of the homogeneous differential equation, which can be written as

$$x(t) = \eta x_1(t) + \zeta x_2(t). \quad (3.8)$$

Depending on the solution for λ we can distinct three different cases which will be elaborated in the following.

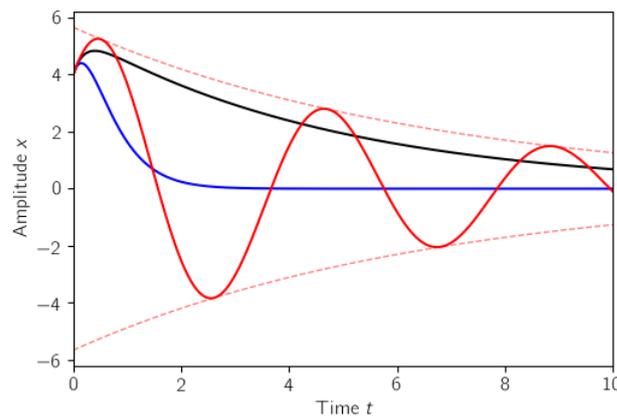


Figure 3.2.: Illustration of an overdamped (black), critical damped (blue) and underdamped (red) harmonic oscillator. The exponentially decaying envelop of the underdamped harmonic oscillator is depicted by the red dashed line.

Overdamped case

In this instance, the damping parameter is $\gamma > 2\omega$. Therefore, the eigenfrequency of the damped harmonic oscillator is $\omega_D \in \mathbb{R}$. The solution eq. (3.8) takes the form

$$x(t) = \eta e^{-\frac{\gamma}{2}t + \omega_D t} + \zeta e^{-\frac{\gamma}{2}t - \omega_D t} = e^{-\frac{\gamma}{2}t} (\eta e^{+\omega_D t} + \zeta e^{-\omega_D t}). \quad (3.9)$$

The parameters η and ζ can be then determined from the initial conditions. The dynamics of the overdamped harmonic oscillator (eq. (3.9)) is depicted in fig. 3.2. The oscillator is approaching a zero elongation, but due to the strong damping very slowly.

Critical damped

The oscillator is critically damped in the case that $\gamma = 2\omega$. The eigenfrequency of the damper oscillator becomes $\omega_D = 0$. Therefore, we choose the ansatz

$$x(t) = e^{-\frac{\gamma}{2}t} (\eta + \zeta t) \quad (3.10)$$

to solve the differential equation. This equation is plotted in fig. 3.2 as the blue solid line. The trajectory in this case yields a steeper decay of the initial elongation x_0 compared to the previous case. This behavior is often desired in mechanical devices which need to damp sudden accelerations. For example, shock absorbers in cars are designed in such a way.

Underdamped case

In the underdamped case ω_D takes an imaginary value, which means that $\gamma < 2\omega$. The solution thus reads

$$x(t) = x_0 e^{-\frac{\gamma}{2}t} \cos(\omega_D t + \varphi), \quad (3.11)$$

where φ denotes the phase of the oscillation. Thus, the dynamics consist of oscillation which is exponentially decaying. The exponentially decaying envelope is depicted in fig. 3.2 with the dashed red line.

3.1.2. Driven harmonic oscillator

In the previous section we examined a damped harmonic oscillator. In this section we will examine the properties of such an oscillator when it is driven by an external time-dependent force $F_{\text{ext}}(t)$. This force adds an inhomogeneity to eq. (3.4).

$$\ddot{x} + \gamma\dot{x} + \omega^2 x = F_{\text{ext}}(t) \quad (3.12)$$

This external force can take many different forms, but for the sake of illustration we will examine a periodic force with a driving frequency ω_{ext} , phase of the force φ_{ext} and an amplitude F_0 .

$$F_{\text{ext}}(t) = F_0 \cos(\omega_{\text{ext}} t + \varphi_{\text{ext}}) \quad (3.13)$$

In section 3.1.3 we will examine the properties of a harmonic oscillator driven by a stochastic random force. The general solution can be split into a transient and steady-state solution.

$$x(t) = x_{\text{tr}}(t) + x_{\text{st}}(t) \quad (3.14)$$

Dependent on the parameters, it is necessary to find a suitable solution analogous to the in section 3.1.1 discussed cases. However, we will restrict ourselves solely to the underdamped or weakly damped case. In this case the damping $\gamma < 4\omega$, thus the solution reads:

$$x(t) = x_0 e^{-\frac{\gamma}{2}t} \sin(\omega_D t + \varphi) + x_{\text{ext}} \cos(\omega_{\text{ext}} t + \varphi_{\text{ext}}), \quad (3.15)$$

where x_0 denotes the elongation of the transient solution and x_{ext} the elongation in the steady state. These parameters can be determined from the initial conditions. However, the algebraic problem is involved and the solution can be found in the literature.

3.1.3. Langevin equation

In this section we will examine the behavior of a damped harmonic oscillator driven by statistical forces. A description by an ordinary differential equation is thus not sophisticated enough, as it only describes the first order statistical moments (e.g., the mean position $\langle x \rangle$ and mean velocity $\langle v \rangle$). The physical interesting behavior is described by the second order statistical moments (e.g., the mean squared position $\langle x^2 \rangle$). To properly describe this type of dynamics we need to employ stochastic differential equations (SDE).

Brownian motion was first described in detailed by Robert Brown in 1827. The botanist was studying the transfer of pollen to the ovulum of aquatic plants. The pollen underwent rapid oscillatory motion while observed under a microscope. The initial explanation of Brown was that this motion was due to bio-mechanical properties, analogous to the male gamete, but discarded this expectation rather quickly after observing similar dynamics with inorganic grains in aqueous suspension.

The first theoretical prediction for this effect was done by Albert Einstein in 1905. He employed the stochastic process of random collision and the Maxwell-Boltzmann distribution to describe the dynamics of the system, which was experimentally verified 1908 by Jean-Baptiste Perrin. Einstein utilized a probability diffusion equation, which has become known as Fokker-Planck equation. For a more in detail description of the solution and a discussion of different approaches see ref. [Cof12].

To accurately describe a system driven by random noises, one can employ Newton's second law to describe the problem. The acceleration on the particle is

$$m\ddot{x} + \gamma\dot{x} + m\omega^2x = \xi(t), \quad (3.16)$$

where γ denotes the damping coefficient, ω the frequency of the harmonic oscillator and $\xi(t)$ the acceleration due to the random force. This random force is independent of the position $x(t)$ of the particle. Furthermore, the noise fluctuates comparably fast to the change of the position $x(t)$. The average amplitude of the noise is $\langle \xi(t) \rangle = 0$. The correlation function of the noise reads $\langle \xi(t)\xi(t') \rangle = C(t-t')$, which is symmetric.

$$C(t-t') = C(t'-t) \quad (3.17)$$

Due to the stochastic properties of the noise it is not sufficient to look only at the first order stochastic moments. The correlation of the Gaussian white noise fulfills in the steady state the fluctuation-dissipation theorem.

$$\langle \xi(t)\xi(t') \rangle = 2k_B\gamma T\delta(t-t') = 2D\delta(t-t') \quad (3.18)$$

k_B denotes the Boltzmann constant, T the temperature and D the noise strength. However, it is only in the steady state that eq. (3.18) holds. Especially in thermal machines running with a final cycle time it is not always guaranteed that equilibrium is reached in each stroke. For such non-equilibrium dynamics, it is necessary to examine the correlation matrix \mathbf{M} .

$$\mathbf{M} = \begin{pmatrix} \sigma_{xx}^2 & \sigma_{xv}^2 \\ \sigma_{vx}^2 & \sigma_{vv}^2 \end{pmatrix} = \begin{pmatrix} \langle x(t)x(t+\tau) \rangle & \langle x(t)v(t+\tau) \rangle \\ \langle v(t)x(t+\tau) \rangle & \langle v(t)v(t+\tau) \rangle \end{pmatrix} \quad (3.19)$$

The elements of this matrix are the autocorrelation of position σ_{xx} and velocity σ_{vv} and the cross-correlations σ_{xv} and σ_{vx} . In the underdamped case $\gamma < 2\omega$, the position auto correlation takes the form

$$\sigma_{xx}^2(t) = 2 \int_0^t dt_1 \int_0^{t_1} dt_2 H(t_1) H(t_2) C(t_1 - t_2), \quad (3.20)$$

where $H(t)$ denotes the Laplace transform associated with eq. (3.16) [Wan96; Mas93]. The explicit expression is derived in appendix A.1.1. The time derivative of this expression is denoted with $h(t) = \partial_t H(t)$. The remaining matrix elements read:

$$\sigma_{vv}^2(t) = 2 \int_0^t h(t_1) dt_1 \int_0^{t_1} h(t_2) C(t_1 - t_2) dt_2 \quad (3.21)$$

$$\sigma_{xv}^2(t) = \int_0^t H(t_1) dt_1 \int_0^{t_1} h(t_2) C(t_1 - t_2) dt_2 \quad (3.22)$$

$$\sigma_{vx}^2(t) = \int_0^t h(t_1) dt_1 \int_0^{t_1} H(t_2) C(t_1 - t_2) dt_2. \quad (3.23)$$

The steady state distribution, the distribution after $t \rightarrow \infty$, reads

$$P_{\text{ss}}(x, v) = \frac{1}{2\pi\sigma_{xx}\sigma_{vv}} \exp \left[-\frac{1}{2} \left(\frac{x^2}{\sigma_{xx}^2} + \frac{v^2}{\sigma_{vv}^2} \right) \right], \quad (3.24)$$

where $\sigma_{ij} = \lim_{t \rightarrow \infty} \sigma_{ij}(t)$. Note that in the steady-state case the cross-correlation terms σ_{xv} and σ_{vx} are vanishing. The auto-correlation elements can be rewritten in terms of the Laplace transformed auto-correlation function of the driving noise $\tilde{C}(s)$.

$$\sigma_{xx}^2 = \frac{1}{2\gamma_0\omega^2} \text{Re}[\tilde{C}(s)] + \frac{1}{2\Omega\omega^2} \text{Im}[\tilde{C}(s)] \quad (3.25)$$

$$\sigma_{vv}^2 = \frac{1}{2\gamma_0} \text{Re}[\tilde{C}(s)] - \frac{1}{2\Omega} \text{Im}[\tilde{C}(s)] \quad (3.26)$$

Here $\Omega = \omega^2 - \gamma_0^2$ with $\gamma_0 = \gamma/(2m)$ and $s = \gamma_0 \pm i\Omega$ [Mén13]. The energy in the steady-state is computed from the mean energy.

$$\langle E \rangle_{\text{ss}} = \frac{m}{2} \langle v^2 \rangle + \frac{m}{2} \omega^2 \langle x^2 \rangle = \frac{m}{2} \sigma_{vv}^2 + \frac{m}{2} \omega^2 \sigma_{xx}^2 \quad (3.27)$$

$$= \frac{1}{2\gamma_0} \text{Re} \left[\tilde{C}(s = \gamma_0 - i\Omega) \right] = k_B T \quad (3.28)$$

This relation demonstrates the relation of steady-state temperatures between different types of noises. The most common found type of noise is Gaussian white noise with a correlation function of

$$C(t, t') = D\delta(t, t') = 2\gamma_0 k_B T \delta(t, t'). \quad (3.29)$$

However, it is sometimes more interesting to look at noises, which are different from the Gaussian white noise. As one can see in eq. (3.29), the noise $\xi(t)$ is independent

of the noise at a different time t' . This lack of time dependence is often described as having no memory in the system, due to the state of $\xi(t)$ being independent on the time which has already passed. This is also described as Markovian. In recent years, research found a special interest in non- or semi-Markovian processes in quantum information [Uta21; Wu20] and quantum thermodynamics [Uzd16]. Especially exponentially correlated noise and colored noise is of great interest.

Exponentially correlated noise follows the correlation function

$$C(t, t') = \frac{D}{\tau_c} \exp\left(-\frac{|t - t'|}{\tau_c}\right) \quad (3.30)$$

with the correlation time τ_c . This noise can be generated by employing the Fokker-Planck equation describing this process [Gil96]. The second type of noise, we only take flicker noise or $1/f$ noise

$$C(t) = \frac{D\gamma}{\ln(\tau_2/\tau_1)} \int_{\tau_1}^{\tau_2} \frac{1}{\tau} \exp(-t/\tau) d\tau, \quad (3.31)$$

with $\tau_1 < \tau < \tau_2$ are the corresponding times to the respective angular frequency $\omega_i = 1/(2\pi\tau_i)$. This noise is generated via a Fourier domain multiplication method [Kas95].

3.1.4. Quantum Harmonic oscillator

The quantum harmonic oscillator is one of the example problems studied in undergraduate courses. It is an often used approximation for more complex systems, i.e. Morse potential. The first approach to describe harmonic oscillators in a quantized manner was done by Max Planck in his work about the black body radiator by allowing for discrete energy states [Pla01]. The Hamiltonian $\hat{\mathcal{H}}$ takes the form

$$\hat{\mathcal{H}} = \frac{m}{2} \hat{p}^2 + \frac{m}{2} \omega^2 \hat{x}^2, \quad (3.32)$$

with $\hat{p}, \hat{x} \in \mathcal{H}$ denoting the momentum and position operator respectively. Instead of using the stationary Schrödinger equation to compute the wave function, we'll use the algebraic method to solve the problem. This derivation was first done by Paul Dirac [Dir27]. We define the annihilator operator \hat{a} and the conjugated operator \hat{a}^\dagger , which is also called creation operator, as follows

$$\hat{a} = \sqrt{\frac{m\omega}{2\hbar}} \left(\hat{x} + \frac{i}{m\omega} \hat{p} \right) \quad (3.33)$$

$$\hat{a}^\dagger = \sqrt{\frac{m\omega}{2\hbar}} \left(\hat{x} - \frac{i}{m\omega} \hat{p} \right). \quad (3.34)$$

This allows to express the momentum and position operator in terms of the creation and annihilation operator.

$$\hat{x} = \sqrt{\frac{\hbar}{2m\omega}} (\hat{a} + \hat{a}^\dagger) \quad (3.35)$$

$$\hat{p} = i\sqrt{\frac{\hbar m\omega}{2}} (\hat{a} - \hat{a}^\dagger) \quad (3.36)$$

By inserting eq. (3.35) and 3.36 into eq. (3.32) and using the following canonical commutation relations

$$[\hat{a}, \hat{a}^\dagger]_- = \frac{1}{2\hbar} (-i[\hat{x}, \hat{p}] + i[\hat{p}, \hat{x}]) = \frac{1}{2\hbar} ((-i)i\hbar\hat{1} + i(-i)\hbar\hat{1}) = \hat{1} \quad (3.37)$$

$$[\hat{n}, \hat{a}]_- = [\hat{a}\hat{a}^\dagger, \hat{a}]_- = \hat{a}^\dagger[\hat{a}, \hat{a}]_- + [\hat{a}^\dagger, \hat{a}]_- \hat{a} = -\hat{a} \quad (3.38)$$

$$[\hat{n}, \hat{a}^\dagger]_- = [\hat{a}\hat{a}^\dagger, \hat{a}^\dagger]_- = \hat{a}[\hat{a}^\dagger, \hat{a}^\dagger]_- + [\hat{a}, \hat{a}^\dagger]_- \hat{a}^\dagger = \hat{a}^\dagger \quad (3.39)$$

where $\hat{n} = \hat{a}\hat{a}^\dagger$ denotes the number operator, the Hamiltonian can be rewritten as

$$\hat{\mathcal{H}} = \hbar\omega \left(\hat{n} + \frac{1}{2}\hat{1} \right). \quad (3.40)$$

The eigenstates of the number operator $|n\rangle$ yields the basis, which eigenvalues corresponds to the phonon numbers. Furthermore, the operators affect these eigenstates $|n\rangle$, which are also referred to as Fock states,

$$\hat{a} |n\rangle = \sqrt{n} |n-1\rangle \quad (3.41)$$

$$\hat{a}^\dagger |n\rangle = \sqrt{n+1} |n+1\rangle \quad (3.42)$$

$$\hat{n} |n\rangle = n |n\rangle \quad (3.43)$$

In the following, we will discuss two special states which are of high interest, since both are suggested to be able to enhance the performance of heat engines [Aba12; Kos13; Gel15; Aba16; Dan19].

Coherent states

Coherent states are quantum mechanical states which resemble a classical harmonic oscillator. A coherent state $|\alpha\rangle$ in the Fock basis reads

$$|\alpha\rangle = \exp\left(-\frac{|\alpha|^2}{2}\right) \sum_{n=0}^{\infty} \frac{\alpha^n}{\sqrt{n!}} |n\rangle, \quad (3.44)$$

where $\alpha \in \mathbb{C}$. This state is generated by displacing a vacuum state using the displacement operator $\hat{D}(\alpha)$

$$\hat{D}(\alpha) = \exp[\alpha\hat{a} + \alpha^*\hat{a}^\dagger]. \quad (3.45)$$

The action of this operator is illustrated in fig. 3.3. As it is depicted, the action of the displacement operator is illustrated on an initial vacuum state. Representing $\alpha = |\alpha|e^{i\varphi}$ the displacement in phase space can be given.

$$\langle\alpha|\hat{x}|\alpha\rangle = \sqrt{\frac{m\omega}{2\hbar}} |\alpha| \cos(\varphi) \quad (3.46)$$

$$\langle\alpha|\hat{p}|\alpha\rangle = \sqrt{\frac{1}{2\hbar m\omega}} |\alpha| \sin(\varphi) \quad (3.47)$$

Furthermore, it is also of interest to examine the expectation values of the number operator as well as its variance.

$$\langle\hat{n}\rangle = |\alpha|^2, \quad (\Delta\hat{n})^2 = |\alpha|^2 \quad (3.48)$$

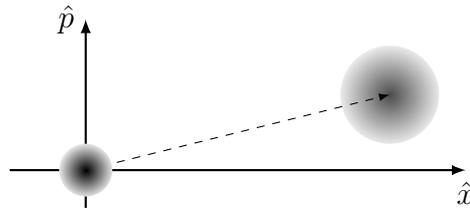


Figure 3.3.: Illustration of a generation of a coherent state by displacing a vacuum state in phase space.

Squeezed states

Further states of interest are squeezed states. These can be generated by applying squeezing operator $\hat{S}(\zeta)$ to the vacuum state. The squeezing operator reads

$$\hat{S}(\zeta) = \exp \left[\frac{1}{2} \left(\zeta^* \hat{a}^2 - \zeta \hat{a}^{\dagger 2} \right) \right]. \quad (3.49)$$

To note here, that the creation and annihilation operators are in the second order in exponent. By examining the action of this operator, it becomes clear that this operator acts on the second statistical moments. With $\zeta = |\zeta| \exp(2i\varphi)$, the variances of position and momentum are given with

$$(\Delta \hat{x})^2 = \frac{\hbar}{2m\omega} e^{-2\zeta} \quad \text{and} \quad (\Delta \hat{p})^2 = \frac{m\hbar\omega}{2} e^{2\zeta}. \quad (3.50)$$

The action of the operator is sketched in fig. 3.4. The center of the distribution is not shifted for an initial vacuum state. Note that it is possible to reduce one second order moment, but the other second order moment is thus increased. Therefore, the Heisenberg limit can't be overcome, but it is a useful technique to enhance the sensitivity with respect to one of the two parameters.

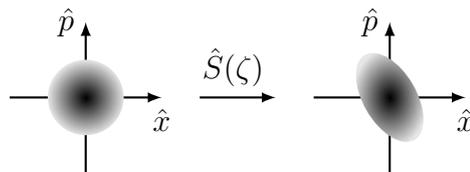


Figure 3.4.: Illustration of the action of the squeezing operator.

3.2. Density operators and Liouville equation

One usually describes the dynamics of a quantum system in an Hilbert space \mathcal{H} . However, due to the nature of this description it neglects decays and it is not feasible to describe more complex states as for example mixed states. To fully describe a quantum system with entanglement and decoherences one must move to the Liouville space. The Liouville space is defined as the tensor product of two Hilbert spaces as follows

$$\mathcal{L} = \mathcal{H} \otimes \mathcal{H} \quad (3.51)$$

The state of the system is denoted by the density operator $\hat{\rho}$. In complete orthonormal basis $\{|\psi_\ell\rangle\}$ the density operator can be described as

$$\hat{\rho} = \sum_{\ell} p_{\ell} |\psi_{\ell}\rangle\langle\psi_{\ell}|. \quad (3.52)$$

The factor p_{ℓ} denotes the probability for each state. This constitutes a probability distribution, $\forall \ell : p_{\ell} \geq 0$ and $\sum_{\ell} p_{\ell} = 1$. For finite dimensional Hilbert spaces the density operator is also referred as density matrix [Rei15]. To compute the expectation values of operators on the Liouville space the Hilbert-Schmidt norm $\langle \cdot, \cdot \rangle_{\text{HS}}$ is used.

$$\langle \hat{A} \rangle = \langle \hat{A}, \hat{\rho} \rangle_{\text{HS}} = \text{tr} [\hat{A} \hat{\rho}] \quad (3.53)$$

To obtain the dynamics of a system, one must consider the time evolution of the density matrix. The generator of the time evolution is the Hamiltonian $\hat{\mathcal{H}}$. The time evolution can be obtained from the time derivative of a density operator.

$$\frac{\partial \hat{\rho}}{\partial t} = \frac{\partial}{\partial t} \sum_{\ell} p_{\ell} |\psi_{\ell}\rangle\langle\psi_{\ell}| \quad (3.54)$$

$$= \sum_{\ell} p_{\ell} \frac{\partial |\psi_{\ell}\rangle}{\partial t} \langle\psi_{\ell}| + \sum_{\ell} p_{\ell} |\psi_{\ell}\rangle \frac{\partial \langle\psi_{\ell}|}{\partial t} \quad (3.55)$$

The time derivatives of the states are given by the time-dependent Schrödinger equation

$$i\hbar \frac{\partial}{\partial t} |\psi_{\ell}\rangle = \hat{\mathcal{H}} |\psi_{\ell}\rangle \quad \text{and} \quad -i\hbar \frac{\partial}{\partial t} \langle\psi_{\ell}| = \langle\psi_{\ell}| \hat{\mathcal{H}}. \quad (3.56)$$

The time derivative of the density operator can thus be rewritten as

$$\frac{\partial \hat{\rho}}{\partial t} = \sum_{\ell} p_{\ell} \left(-\frac{i}{\hbar} \right) \hat{\mathcal{H}} |\psi_{\ell}\rangle \langle\psi_{\ell}| + \sum_{\ell} p_{\ell} |\psi_{\ell}\rangle \langle\psi_{\ell}| \frac{i}{\hbar} \hat{\mathcal{H}} \quad (3.57)$$

$$= -\frac{i}{\hbar} \left(\hat{\mathcal{H}} \sum_{\ell} p_{\ell} |\psi_{\ell}\rangle \langle\psi_{\ell}| - \sum_{\ell} p_{\ell} |\psi_{\ell}\rangle \langle\psi_{\ell}| \hat{\mathcal{H}} \right) \quad (3.58)$$

$$= -\frac{i}{\hbar} \left(\hat{\mathcal{H}} \hat{\rho} - \hat{\rho} \hat{\mathcal{H}} \right). \quad (3.59)$$

The last step is now to rewrite the commutator of the Hamiltonian and the density operator.

$$\boxed{\frac{\partial \hat{\rho}}{\partial t} = -\frac{i}{\hbar} [\hat{\mathcal{H}}, \hat{\rho}]_-} \quad (3.60)$$

This equation is referred to as the Liouville-von Neumann equation. Note that eq. (3.60) resembles Liouville's theorem from classical statistical mechanics. Therefore, it is regarded as quantum equivalent of it.

This equation only holds for closed quantum system. In the case of open quantum systems, i.e. systems where energy is dissipated, the dissipation is modelled often phenomenologically. The super-operator $\hat{\mathcal{L}}_D [\hat{\rho}(t)]$, which is referred to as Lindbladian or Lindblad operator, denotes this dissipation. A super-operator denotes an operator acting upon an operator. For ease of notation the Liouvillian super-operator $\hat{\mathcal{L}} [\hat{\rho}(t)]$ is defined as

$$\hat{\mathcal{L}} [\hat{\rho}(t)] = \frac{d\hat{\rho}(t)}{dt} = \frac{\partial \hat{\rho}(t)}{\partial t} + i\hat{\mathcal{L}}_D [\hat{\rho}(t)]. \quad (3.61)$$

Further instructions on how this super-operator is used is found in section 4.3. Note that the dissipation can take different forms. For most systems, this dissipation process is Markovian. This means that the dynamics of the dissipation is solely dependent on the current density operator. This is the case for dynamics like the decay of electrons of an excited atomic state or the process taking place in the regime in thermodynamics. However, it is of great interest to especially examine dynamics governed by non-Markovian processes, which are processes depending on previous states. Because dynamics of such systems is of great interest, recent research effort is put into investigating such systems [Bas17; Veg17; Tam18; Cia19; Luc20; Bas21].

3.3. Heat engines

Heat engines are of interest in research, since they are used widely throughout society. In general, they consist in general of a working agent, which is connected to a flywheel, and a cold and a hot reservoir. The working agent is brought into the hot reservoir and heat Q is transferred from the bath to the working agent. This heat is transferred to the cold reservoir while the working agent is performing work W . This work can be stored in the flywheel. Depending on the type of cycle employed by the heat engine these processes can differ vastly. In this section we will elaborate on the Carnot section 3.3.1, Otto section 3.3.2 and the Stirling section 3.3.3 cycle. Furthermore, the Single atom heat engine [Roß16b; Roß16a] will be discussed briefly in section 3.3.4

3.3.1. The Carnot cycle

The Carnot cycle is one of the most theoretical studied heat engine cycles, since it is closely linked to the second law of thermodynamics. The engine cycle is named after Sadi Carnot, who described this cycle [Car72]. Even today this cycle is still of interest for ongoing research [Cur75; Gev92; Ben00; Esp10a; Esp10b; Ma17; Ma18; Dan19].

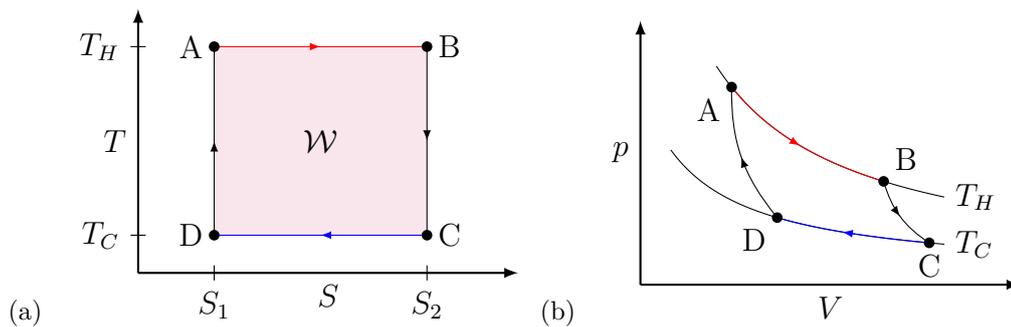


Figure 3.5.: Depiction of the Carnot cycle in a (a) T - S diagram and in a (b) p - V diagram.

The Carnot cycle consists of four strokes, which are depicted in fig. 3.5. The four strokes are described below in table 3.1.

To evaluate the performance of heat engines, it is of interest to compare the energy which is put in as heat Q and the work extracted by the system W . Such a measure is the efficiency η . The heat between point A to B Q_{AB} is computed by

$$Q_{AB} = T_H \Delta S, \quad (3.62)$$

with $\Delta S = (S_2 - S_1)$. This term correlates to the heat, which is put into the system. The work performed by the working agent is computed by

$$W = (T_H - T_C) \Delta S. \quad (3.63)$$

This is due to the working agent being decoupled from the environment, leaving the internal energy unchanged. From these expressions, the efficiency of the Carnot

A → B	isothermal expansion	The working agent is in contact with the hot reservoir at temperature T_H and heat is transferred from the hot reservoir to the working agent.
B → C	isentropic expansion	The working agent is decoupled from the bath and adiabatically expanded until it reaches the temperature of the cold reservoir T_C . Here work is performed by the working agent.
C → D	isothermal compression	The working agent is in contact with the cold bath at temperature T_C , to which heat is transferred from the working agent to the cold bath.
D → A	isentropic compression	The working agent is decoupled from the cold reservoir and adiabatically compressed until it reaches the temperature of the hot reservoir T_H .

Table 3.1.: Description of the four strokes which together build the Carnot cycle.

cycle η_C can be computed as the ratio of the work extracted from the engine and the energy put into the system.

$$\eta_C = \frac{W}{Q_{AB}} = \frac{(T_H - T_C)}{T_H} = 1 - \frac{T_C}{T_H} \quad (3.64)$$

It can be concluded from the second law of thermodynamics, that the Carnot efficiency yields a tight classical bound for any heat engine. That is due to the use of solely reversible adiabatic and isothermal processes. This reversibility necessitates not to generate entropy. Therefore, any classical heat engine can only reach the Carnot efficiency, but not exceed it.

$$\boxed{\eta \leq 1 - \frac{T_H}{T_C}} \quad (3.65)$$

To achieve reversible thermalization, an infinite time must be waited for the equilibration between the working agent and the thermal reservoirs. Thus the engine yields a vanishing power due to the infinite cycle time. This is not of importance for theoretical considerations. However, for experimental realization of a Carnot heat engine an infinite cycle time is not feasible. Furthermore, for experimental machines it is of interest to obtain the highest power at an optimal efficiency. For the so called *endoreversible engine* a bound can be obtained. This bound describes the maximal efficiency at maximal power. This bound is also known as the Curzon-Ahlborn-Novikov limit [Cur75; Nov58], which is takes the form

$$\eta_{CA} = 1 - \sqrt{\frac{T_H}{T_C}}. \quad (3.66)$$

3.3.2. The Otto cycle

The Otto cycle is one of the most spread in modern applications. It approximates the cycle of an internal combustion engine, which are commonly found in cars. As well as the Carnot cycle, the Otto cycle is investigated in recent research [Den13; Roß14; Kos17; Aba19; Çak19]. The Otto cycle is depicted in fig. 3.6.

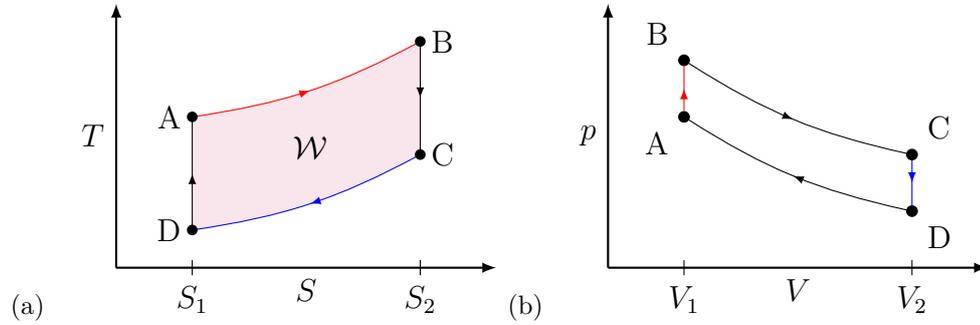


Figure 3.6.: Depiction of the Otto cycle in a (a) T - S diagram and in a (b) p - V diagram.

The four strokes of the Otto cycle are described in table 3.2.

$A \rightarrow B$	isochoric heating	Heat Q_{AB} is transferred from the hot reservoir to the working agent while the volume V_1 is kept constant.
$B \rightarrow C$	adiabatic expansion	The working agent is adiabatically expanded from V_1 to V_2 .
$C \rightarrow D$	isochoric cooling	Heat Q_{CD} is transferred from the working agent to the cold reservoir while the volume V_2 is kept constant.
$D \rightarrow A$	adiabatic compression	The working agent is adiabatically compressed from V_2 to V_1 .

Table 3.2.: Description of the four strokes of which the Otto cycle consist of.

To compute the heat transferred between the reservoirs and the working agent, one need to integrate over the isochoric heating/cooling stroke. The work done by the working agent can be analogous determined as in section 3.3.1. The efficiency of the Otto heat engine cycle can be derived to be

$$\eta_{\text{Otto}} = 1 - \left(\frac{V_2}{V_1} \right)^{\frac{c_p - c_v}{c_v}}, \quad (3.67)$$

where c_p denotes the isobaric and c_v the isochoric heat capacity [Cal85].

Quantum harmonic Otto cycle

The quantum harmonic Otto cycle describes the quantum working agent described as a harmonic oscillator. The volume of a harmonic oscillator can be understood as the confinement of the particle in the harmonic potential [Kos17]. This confinement is described by the frequency ω . The heat transfer is expressed as the difference of the occupation number $\Delta n = n_h - n_c$, with $n_i = 1/(\exp[\hbar\omega_i/k_bT_i] - 1)$. The work per cycle is derived as

$$\mathcal{W}_{\text{cyc}} = \hbar\Delta\omega\Delta n, \quad (3.68)$$

with $\Delta\omega = \omega_h - \omega_c$ [Kos17]. The efficiency of the quantum harmonic Otto engine becomes

$$\eta_{\text{Otto}} = 1 - \frac{\omega_c}{\omega_h} \leq \eta_{\text{CA}}. \quad (3.69)$$

3.3.3. The Stirling cycle

The Stirling cycle, even though not as prominent as the Otto cycle, has still many applications. Due to the Stirling engine not requiring combustion for the operation, Stirling engines can be used in hazardous atmospheres where the oxygen which would be required for combustion is a sparse resource. Such an environment can be found in submarines, where Stirling engines are employed to propel the craft [Nil88]. Another application for Stirling engines is to use the sun's radiation as a hot reservoir to convert the radiation energy into mechanical work, which can be used to propel a generator [Kon03]. Also it is still subject to ongoing research [Wu98; Yin17; Yin18].

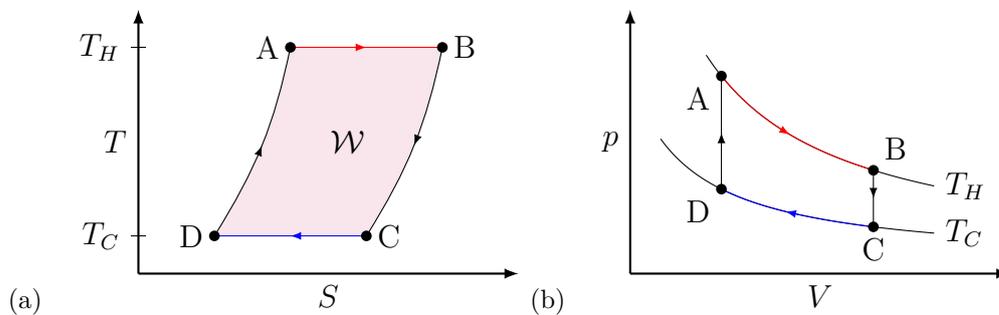


Figure 3.7.: Depiction of the Stirling cycle in a (a) T - S diagram and in a (b) p - V diagram.

The Stirling cycle, which is depicted in fig. 3.7, is composed of four processes, which are described in table 3.3.

Since the Stirling cycle includes isothermal expansion and compression like a Carnot cycle, the same notion as discussed in section 3.3.1 applies here.

A \rightarrow B	isothermal expansion	The working agent is in contact with the hot reservoir at temperature T_H and work is transferred reversibly from the hot reservoir to it.
B \rightarrow C	isochoric expansion	The working agent is decoupled from bath and adiabatically expanded until it reaches the temperature of the cold reservoir T_C .
C \rightarrow D	isothermal compression	The working agent is in contact with the cold bath at temperature T_C , to which work is transferred from the working agent.
D \rightarrow A	isochoric compression	The working agent is decoupled from the cold reservoir and adiabatically compressed until it reaches the temperature of the hot reservoir T_H .

Table 3.3.: Description of the four strokes of which the Stirling cycle consist of.

3.3.4. Single-atom heat engine

To investigate the behavior of thermal devices in the quantum regime, a miniaturization of such devices is required. To still be able to perform experiments a sophisticated control needs to be employed. One such a system is trapped ions, which has been demonstrated to be used for thermal devices [Roß16b; Roß16a; Lin19]. In this thesis we will focus on the usage of a tapered ion trap, which was used in [Roß16b; Roß16a]. This trapping geometry is described in section 2.1.

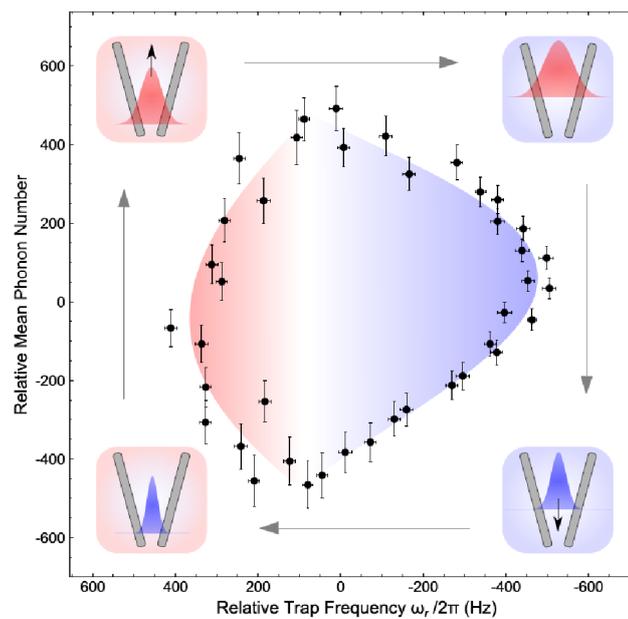


Figure 3.8.: Working cycle of the single-atom heat engine [Roß16b].

The working cycle of the single-atom heat engine is depicted in fig. 3.8. Since the working agent consists of two harmonic oscillators in radial direction, the dimension to examine the thermodynamic cycle is the trapping frequency ω_r and the relative mean phonon number \bar{n}_r in radial direction. As a cold reservoir a Doppler cooling laser was utilized. The hot bath was emulated using white noise together with the Doppler cooling laser. The laser damps the movement of the ion, yielding dissipation. After some time, the ion will arrive at the final steady state temperature following the dissipation fluctuation theorem. As we will see in section 5.1, the heating and cooling of the ion yields a changing displacement force along the axial direction. The ion follows this force, lowering/raising the confinement of the ion in radial direction. The single-atom heat engine performed at an efficiency of $\eta_{\text{SIHE}} = 2.8\text{‰}$ with a power of $P = 3.42 \cdot 10^{-22}$ J/s [Roß16b].

4. Numerical methods

Not every problem has an exact analytical solution. To still examine the properties, one might choose to solve the problem numerically. In the following will be presented a classical propagator and a propagator for quantum dynamics.

4.1. Störmer-Verlet propagator

Obtained equations of motion can not always be solved analytical or only with a substantial time effort. To obtain the trajectories of such complex systems one can employ numerical propagators. The straightforward implementation of such a propagator is the Euler-method, which just integrates the equation of motion in discrete steps directly. However, often this propagator does not yield a good stability and leads especially for long integration time to large errors [Sin10]. A more sophisticated solution is to employ the Störmer-Verlet or inter-leaping frog method [Hai03]. The basic idea behind this method is that the integration takes place on two partitioned groups of variables. These groups are position $\vec{r}(t)$ and velocity $\vec{v}(t)$. Let the particle with mass m and charge q be put in an arbitrary continuous potential $V(\vec{r}, t)$. The Hamiltonian $H(\vec{r}, \vec{v}, t)$ takes the form

$$H(\vec{r}, \vec{v}, t) = \frac{m}{2} |\vec{v}|^2 + V(\vec{r}, t). \quad (4.1)$$

The canonical equation of motion is discretized with the finite time step θ .

$$\vec{r}_{n+1} - 2\vec{r}_n + \vec{r}_{n-1} = -\theta^2 \frac{q}{m} \nabla V(\vec{r}_n, t_n). \quad (4.2)$$

This relation allows to find the next position \vec{r}_{n+1} for $n \leq 1$. However, the position before the time propagation has started \vec{r}_{-1} is not known. To account for this pre-initial condition, the velocity needs to be included. The velocity \vec{v} is defined as $\vec{v} = \dot{\vec{r}}$. To also put this equation into a discrete time frame, it can be rewritten as

$$\vec{v}_n = \frac{\vec{r}_{n+1} - \vec{r}_{n-1}}{2\theta}. \quad (4.3)$$

By solving eq. (4.3) for \vec{r}_{n-1} and inserting this expression in eq. (4.2), this parameter can be eliminated.

$$\vec{r}_{n+1} = \vec{r}_n + \theta \vec{v}_n + \frac{\theta^2}{2} \frac{q}{m} \nabla V(\vec{r}_n, t_n) \quad (4.4)$$

Instead of computing the required of the position, one can compute the velocity at the intermediate point in the time step $\vec{v}_{n+\frac{1}{2}}$. The following relations are used to

update the positions and velocities.

$$\vec{v}_{n+\frac{1}{2}} = \vec{v}_n + \frac{\theta}{2} \frac{q}{m} \nabla V(\vec{r}_n, t_n) \quad (4.5)$$

$$\vec{r}_{n+1} = \vec{r}_n + \theta \vec{v}_{n+\frac{1}{2}} \quad (4.6)$$

$$\vec{v}_{n+1} = \vec{v}_{n+\frac{1}{2}} + \frac{\theta}{2} \frac{q}{m} \nabla V(\vec{r}_n, t_n) \quad (4.7)$$

4.2. Monte-Carlo simulation

For deterministic problems trajectories can be obtained from numerical propagators. However, the simulation of systems undergoing random events which are not deterministic is not as straight-forward. Due to the encountering of problems where underlying randomness in different sectors like finances, fluid dynamics or molecular dynamics. By utilizing random sampling algorithms, it is feasible to obtain predictions for complex dynamics. Such algorithms are referred to as Monte-Carlo algorithms.

For particles which are undergoing random accelerations from fluctuating forces, the numerical modeling is not as straightforward as in section 4.1. Such a problem is for example the Brownian motion. Due to the thermal fluctuations one needs to consider the microscopic processes and obtain the dynamic properties by averaging over each iteration. To obtain each iteration, the fluctuating noise needs to be generated. These are generated from distributions of pseudo random numbers.

Randomness and computers

It might seem to be contradictory to generate randomness on a deterministic device such as computers. The acquiring of true random data, for example from quantum jump measurements, is quite slow. Especially compared to state-of-the-art compute process unit (CPU), which are becoming faster and faster. Therefore, a lot of effort went into developing of algorithms, which can deterministically generate random numbers. Such a procedure is referred to as pseudo random number generation (pRNG). The numbers generated are dependent on an input parameter which is called seed. From this seed, the algorithm is generating a string of random numbers deterministically. There are different algorithms available. One of most used is the implementation of the Mersenne Twister algorithm as provided by the GNU scientific library (GSL) [Gal09]. For the more interested reader, a detailed description of this algorithm can be found in the paper by Matsumoto and Nishimura [Mat98].

4.3. Newton propagator

The newton propagator is utilized to propagate quantum systems in the Liouville space. The dynamic of the system is described by the Liouvillian super-operator \mathcal{L} eq. (3.61). The time-dependent density operator $\hat{\rho}(t)$ is obtained by solving this

equation.

$$\hat{\rho}(t) = e^{-\frac{i}{\hbar}\hat{\mathcal{L}}t}\hat{\rho}(0) \quad (4.8)$$

The state can be therefore determined at a given time using the above expression. However, it is computationally expensive to perform the exponentiation of matrices, because the expansion of the Taylor series is not converging fast. Therefore, one can expand the exponential functions into Newton polynomials. The Newton polynomials $R_n(z)$ can express an arbitrary complex function $f(z) : \mathbb{C} \rightarrow \mathbb{C}$.

$$f(z) \approx \sum_{n=0}^{N-1} a_n R_n(z), \quad R_n(z) = \prod_{m=0}^{n-1} (z - z_j) \quad (4.9)$$

$\{z_j\}$ denotes the set of the N sampling points. At these points the interpolation is exact. The coefficients a_n are determined by the iterative method referred to as *divided difference* [Ash95; Goe15].

$$a_0 = f(z_0) \quad (4.10)$$

$$a_1 = f(z_1) - f(z_0) \quad (4.11)$$

$$a_n = \frac{f(z_n) - \sum_{m=0}^{n-1} a_m \prod_{\ell=0}^{m-1} (z_n - z_\ell)}{\prod_{m=0}^{n-1} (z_n - z_m)} \quad (4.12)$$

Therefore, the time evolution of a density operator can be obtained by expanding eq. (3.60) in Newton polynomials.

$$\hat{\rho}(t) = e^{-\frac{i}{\hbar}\hat{\mathcal{L}}t}\hat{\rho}_0 \approx \sum_{n=0}^{N-1} a_n \left[\hat{\mathcal{L}} - z_n \hat{\mathbb{1}} \right] \hat{\rho}_0 \quad (4.13)$$

By iterative utilization of this expansion, any arbitrary point in time can be reached. To obtain a fast convergence of the series, the proper interpolation points must be chosen. It has been demonstrated that fastest convergence is reached when using the complex eigenvalues of $\hat{\mathcal{L}}$ [Goe15]. But these are often not easily determinable, especially with high dissipation terms. For this problem sophisticated estimation methods have been developed using suitable sampling points. In this thesis, we use the Newton propagator developed by Michael Goerz [Goe15; Goe18].

5. Single-atom heat engine as sensitive probe

A major part of this chapter has been published in:

Single-atom heat engine as a sensitive thermal probe,

A. Levy, M. Göb, B. Deng, K. Singer, E. Torrontegui, D. Wang, *New Journal of Physics* **22**, 093020 (2020).

Usually, work is extracted from two thermal reservoirs by a heat engine and stored in the flywheel. Usually, this work is used to propel a load. The performance of a heat engine is determined by the thermodynamic properties of the system. Instead of using the work to drive a device, the work can be measured to obtain information about the performance of the heat engine. This principle is depicted in fig. 5.1 (a) and (b).

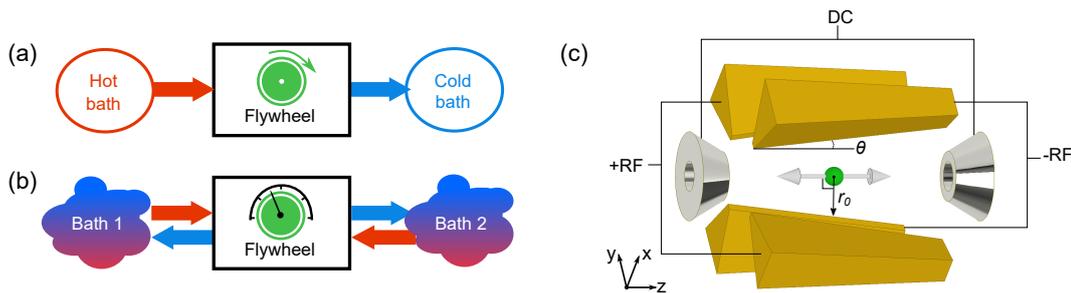


Figure 5.1.: (a) A heat engine takes in heat from a hot thermal bath, converts part of the thermal energy into mechanical work and releases the rest to a cold bath. The work can be stored in a flywheel. (b) A calibrated heat engine can be employed to measure the temperature difference between two baths by monitoring the energy in the flywheel. (c) A modified linear Paul trap with four tapered blade electrodes facilitating the operation of a single-ion heat engine.

5.1. Dynamics of the single-ion heat engine

In our setup we investigate a $^{40}\text{Ca}^+$ ion confined in a tapered Paul trap, which is depicted in fig. 5.1 (c). This trap consists out of four blade electrodes, which are connected to RF voltages, and two endcap electrodes connected to DC voltages [Roß16b]. Compared to conventional linear Paul traps, the RF electrodes are tilted with respect to the z axis with an angle θ . Due to this angle, the harmonic

confinement in radial (x, y) direction is in dependence of the axial (z) direction.

$$\omega_{x,y}(z) = \frac{\omega_{x0,y0}}{(1 + \tan \theta \cdot z/r_0)^2} \quad (5.1)$$

Here r_0 denotes the radial distance of the ion to the blade electrodes and $\omega_{x0,y0}$ the radial trapping frequencies at $z = 0$. In our setup we need to introduce a small anisotropy of the potential to lift the degeneracy in the radial direction. The confinement in radial direction is here usually an order of magnitude larger than the axial trapping frequency. The Hamiltonian \hat{H} of an ion inside the tapered potential reads

$$\hat{H} = \frac{\hat{p}_x^2}{2m} + \frac{1}{2}m\omega_x^2(z)\hat{x}^2 + \frac{\hat{p}_y^2}{2m} + \frac{1}{2}m\omega_y^2(z)\hat{y}^2 + \frac{\hat{p}_z^2}{2m} + \frac{1}{2}m\omega_z^2 z^2, \quad (5.2)$$

with m the mass of the ion, \hat{x} , \hat{y} , \hat{z} and \hat{p}_x , \hat{p}_y , \hat{p}_z the position and momentum operators in the corresponding directions. To formulate the dynamics in terms of a heat engine, one can recontextualize the radial degrees of freedom as the working agent. This working agent is coupled to external thermal reservoirs periodically. The resulting displacement, as we will see later in this section, yields a displacement force. This displacement force acts analog to a piston, because it drives an adiabatic compression/expansion of the working medium. The work performed by the atom is stored in the axial direction, which is therefore analogous to the flywheel of an engine. Here, we will consider the radial degrees of freedom in a full quantum picture and the axial motion to be classical. This is due to the oscillation in axial direction being coherent with a large elongation. The Hamiltonian can be rewritten in terms of annihilation and creation operators \hat{a}_x^\dagger (\hat{a}_y^\dagger) and \hat{a}_x (\hat{a}_y) in x (y) direction.

$$\begin{aligned} \hat{H} &= \hbar\omega_{x0}(\hat{a}_x^\dagger\hat{a}_x + \frac{1}{2}) + \hbar\omega_{y0}(\hat{a}_y^\dagger\hat{a}_y + \frac{1}{2}) \\ &+ \frac{1}{2}m(\omega_{x0}^2\hat{x}^2 + \omega_{y0}^2\hat{y}^2) \left(\frac{1}{(1 + \gamma z)^4} - 1 \right) + \frac{p_z^2}{2m} + \frac{1}{2}m\omega_z^2 z^2 \end{aligned} \quad (5.3)$$

The first two terms in eq. (5.3) describe the energy transferred to the working medium. The coupling of the working medium to the flywheel is denoted in the third term, where we define $\gamma = \tan \theta / r_0$. The last two terms denote energy along the axial direction of the atom. p_z and x_z denote the classical momentum and position of the calcium ion in the z direction. As the keen reader might notice that the two radial directions couple each individually with the axial degree of freedom. This is indicated in eq. (5.3). For sake of brevity, we consider here only the x direction and assume that the oscillator in y direction is kept at a constant and low constant temperature, not contributing to the work done by the heat engine. The creation (annihilation) operator is denoted as \hat{a}^\dagger (\hat{a}). We define the moments $\hat{X}_1(t)$ and $\hat{X}_2(t)$, the number operator $\hat{N}(t)$ in x direction as well as the coupling relation $g(t)$ between the quantum harmonic oscillator and the axial degree of freedom as

follows

$$\begin{aligned}
\hat{X}_1(t) &= (\hat{a}^{\dagger 2}(t) + \hat{a}^2(t)) \\
\hat{X}_2(t) &= i(\hat{a}^{\dagger 2}(t) - \hat{a}^2(t)) \\
\hat{N}(t) &= \hat{a}^{\dagger}(t)\hat{a}(t) \\
g(t) &= \frac{\hbar\omega_{x0}}{4} \left(\frac{1}{[1 + \gamma z(t)]^4} - 1 \right).
\end{aligned} \tag{5.4}$$

The equations of motion of the operators in the x direction read

$$\begin{aligned}
\frac{d}{dt}\hat{X}_1(t) &= 2(\hbar\omega_{x0} + 2g(t))\hat{X}_2(t) \\
\frac{d}{dt}\hat{X}_2(t) &= -2(\hbar\omega_{x0} + 2g(t))\hat{X}_1(t) - 8g(t)\hat{N}(t) - 4g(t) \\
\frac{d}{dt}\hat{N}(t) &= -2g(t)\hat{X}_2(t).
\end{aligned} \tag{5.5}$$

The trajectory of the ion in axial direction is obtained by the following classical

$$\begin{aligned}
\dot{z}(t) &= p_z(t)/m \\
\dot{p}_z(t) &= \mathcal{F}(t),
\end{aligned} \tag{5.6}$$

with \mathcal{F} denoting the force acting on the axial oscillator. In the mean-field approximation, \mathcal{F} can be expressed as

$$\mathcal{F}(t) = -\frac{\partial\langle\hat{H}\rangle}{\partial z} = -m\omega_z^2 z(t) + F(t). \tag{5.7}$$

The first term denotes the restoring force of the harmonic potential in axial direction. The radial-axial coupling is described by $F(t)$.

$$F(t) = \frac{\gamma\hbar\omega_{x0}R(t)}{(1 + \gamma z(t))^5} - \frac{\hbar\omega_{x0}}{4} \frac{\partial R}{\partial z} \left(\frac{1}{(1 + \gamma z(t))^4} - 1 \right). \tag{5.8}$$

Here, we define $R(t) = \langle (\hat{a}^{\dagger}(t) + \hat{a}(t))^2 \rangle$. Instead of assuming the temperature of the ion being constant, alternatively one could consider the temperature being the same in the y direction. This would increase the force $F(t)$ by a factor of two. Therefore, the derivation for this case is analog to the one presented in this thesis. In the experiment $z(t) \ll r_0$ [Roß16b], which allows due to $\gamma z(t) \ll 1$ an approximation for the force $\mathcal{F}(t)$

$$\mathcal{F}(t) \approx -m\omega_z^2 z(t) + \gamma\hbar\omega_{x0}R(t), \tag{5.9}$$

with

$$\begin{aligned}
R(t) &= \langle \hat{X}_1(t) + 2\hat{N}(t) + 1 \rangle \\
&= 1 + 2N_0 + X_1(0) \cos(2\omega_{x0}t) + X_2(0) \sin(2\omega_{x0}t),
\end{aligned} \tag{5.10}$$

and $X_1(0)$, $X_2(0)$, N_0 the expectation values of $\hat{X}_1(t)$, $\hat{X}_2(t)$ and $\hat{N}(t)$ at $t = 0$.

5.1.1. Thermodynamic cycle

The operation of the system can be described as an Otto engine [Aba12; Rez06; Tor13; Kos17], which is described in section 3.3.2. This cycle is depicted in fig. 5.2 (a) and (b). For the sake of clarity, here the principal working cycle will be elaborated on our specific system.

Hot isochore [A]: The ion is kept confined in a constant trapping frequency ω_h , while it is coupled to the hot reservoir. The radial degree of freedom thermalizes to the temperature T_h after a time τ_h . The time the interaction takes is comparably brief to the axial oscillation, so that the trajectory of the ion is well assumed to be negligible in axial direction.

Isentropic expansion [B]: The ion is decoupled from the hot bath and the ion evolves over a time τ_z freely. During this time the frequency changes from the initial ω_h to the frequency ω_c . This is due to the axial displacing force caused by the now higher temperature. This evolution is usually adiabatic, but it can be also driven faster using optimal control techniques [Tor18; Göb19]. This method also allows for optimizing with respect to experimental limitations and minimization of errors from different sources [Lev17; Lev18]. This process is unitary, meaning that it is not increasing the entropy of the motional state of the ion.

Cold isochore [C]: The ion is coupled to the cold bath and thermalizes with it to the temperature T_c . During this interaction the trapping frequency is kept constant at ω_c .

Isentropic compression [D]: in the last stroke, the ion is again isolated from the baths and evolves isentropically in the trapping potential for another time of τ_z .

In previous work, the baths were realized by utilizing Doppler cooling lasers and electric noise to emulate thermal baths [Roß16b]. The Doppler cooling laser serves here as cold reservoir as well as dissipation of the sent in electrical noise. This allows the ion to thermalize following the fluctuation-dissipation theorem. The Doppler cooling laser is here a red detuned laser from the ions $^2S_{1/2} - ^2P_{1/2}$ transition and the noise is generated by an arbitrary waveform generator. With this technique temperatures ranging from 1 mK to 4 K were obtained.

To obtain further insight into the dynamics of the above-described heat engine, we will examine analytical approximation as well as a full numerical simulation of this system. First, we will derive the analytical description. The numerical model will be explained in section 5.2. For both descriptions we will use parameters found in the experiment. The ion in the trap is a $^{40}\text{Ca}^+$ ion (see section 2.2) with $m = 40$ amu confined in a tapered Paul trap with $\theta = \pi/6$, $r_0 = 1$ mm and trap frequencies of $\omega_{x0} = 2\pi \times 1$ MHz and $\omega_z = 2\pi \times 0.1$ MHz. The ion is initialized in a thermal distribution by using the Doppler laser to cool the ion to the Doppler temperature, which is ~ 1 mK. The average phonon number corresponding to this temperature in axial direction is ≥ 200 . This well justifies the classical treatment of the axial degree of freedom. At time $t = 0$ the ion is at an axial position z_0 with an initial axial velocity $v_{z0} = p_{z0}/m$. The ion is coupled to a hot reservoir with a temperature of $\beta_h = 1/k_B T_h$. After the thermalization to the reservoir the working medium has

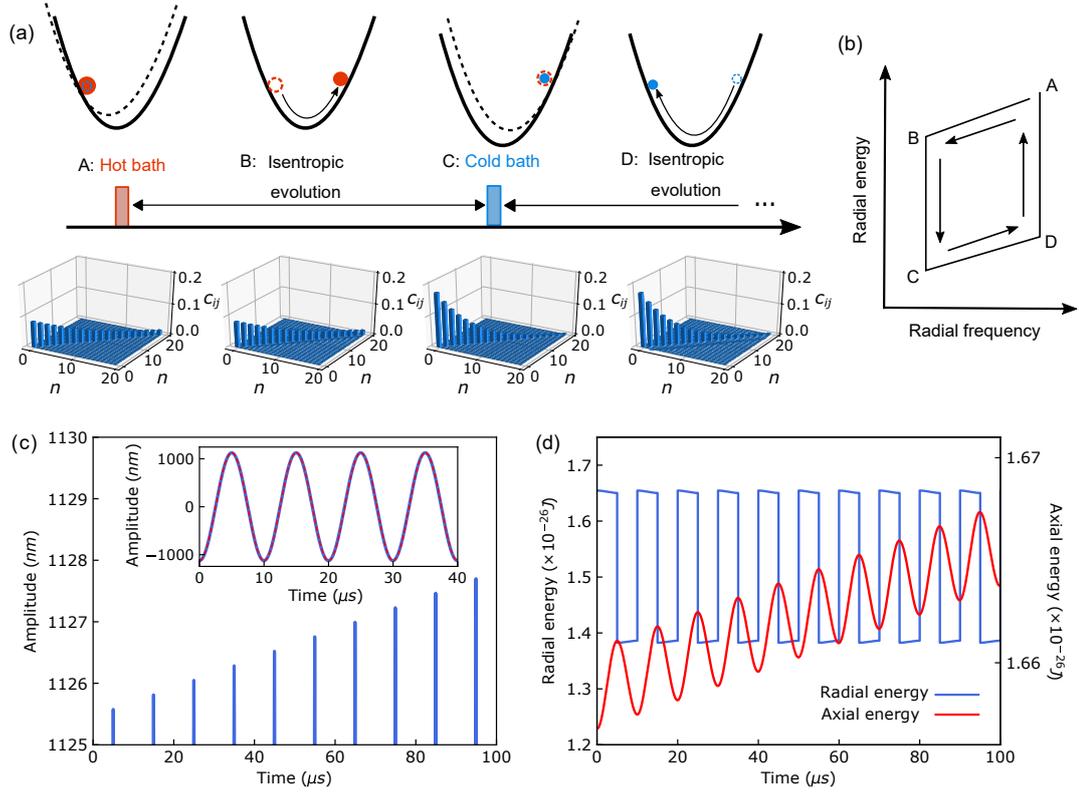


Figure 5.2.: (a) Upper: illustration of the radial and axial states of the ion that undergoes one four-stroke (A-B-C-D) cycle. Red (blue) circle represents the radial state after interaction with the hot (cold) bath. A different temperature of the radial state leads to a different amount of displacement of the axial potential. Dashed and solid black lines illustrate the axial potential before and after the bath interaction. Lower panels show the density matrix elements (c_{ij}) of the radial state in phonon number (n) basis at the end of each stroke. Only the first 20 levels are displayed. (b) Energy-frequency diagram of the radial state. The cycle is not closed due to the accumulation of energy in the flywheel. (c) Blue line shows a close-up of the axial trajectory of the ion as a function of time with $T_c = 1.0$ mK, $T_h = 1.2$ mK and $z_0 = -1.1 \mu\text{m}$. The oscillation amplitude grows linearly with the number of engine cycles. Inset: the full axial trajectory over four engine cycles. The results from numerical simulations (solid blue) and analytical calculations using section 5.3.1 (dashed red) show excellent agreement. (d) Energy in the working medium (blue) and the flywheel (red) under the same condition as in (c).

$X_1(0) = X_2(0) = 0$, $N_0 = (e^{\beta_h \hbar \omega_{x0}} - 1)^{-1}$. In the small axial displacement limit the force in axial direction due to the variance in radial direction becomes

$$F_h = \gamma \hbar \omega_{x0} R_h, \quad (5.11)$$

with

$$R_h = \coth\left(\frac{\beta_h \hbar \omega_{x0}}{2}\right). \quad (5.12)$$

As illustrated in fig. 5.2 (a), this force generates a displacement along the axial degree of freedom towards the open end of the taper. During the action of the force the radial state expands in an isentropic manner. The dynamics in the axial direction are determined by integrating the equations of motion (section 5.1) in this direction. After evolving freely for τ_z , the ion reaches the position z_1 with a velocity v_{z1} . Here the ion interacts with the cold reservoir and thermalizes to a temperature of $\beta_c = 1/k_B T_c$. The force after the isochoric cooling the force is lowered to

$$F_c = \gamma \hbar \omega_{x0} R_c, \quad (5.13)$$

where R_c is given by section 5.1.1, where the temperature is substituted $\beta_h \rightarrow \beta_c$. The diminished force yields a shifted minimum of the potential, yielding an effective restoring force towards the narrow end of the trap. Again, the ion evolves freely in time, till a time $t = 2\tau$. This process is again isentropic. When the ion couples again with the hot thermal bath, the flywheel does not restore the original (z_0, v_{z0}) point but ends at (z_2, v_{z2}) due to the work done by the forces F_h and F_c . This open engine cycle is illustrated in fig. 5.2 (b).

To verify the analytical results of the accumulation of work, numerical simulations were performed. The results are shown in fig. 5.2 (c) and (d) with temperatures of $T_h = 1.2$ mK and $T_c = 1.0$ mK. The procedure to perform the simulations will be elaborated on in the next section. The blue lines in fig. 5.2 (c) depict a zoom in on the axial motion of the ion, to make the increase in elongation visible. The inset depicts the full trajectory in axial direction, where the blue solid lines are the numerical results and the red dashed line represent the analytically approximated solution discussed in this section. Both description are in agreement with each other. In fig. 5.2 (d) the energy in the radial (blue) and axial (red) degrees of freedom is plotted. The energy in the flywheel grows quadratically due to the elongation increasing linearly in each cycle.

5.2. Numerical simulation

In this section we will describe the numerical model used to simulate the dynamics of the system. For this we use a combination of classical and quantum simulations. The classical dynamic in axial direction is here simulated by the Störmer-Verlet method (section 4.1) and the quantum trajectory in the radial direction is propagated by the Newton propagator section 4.3.

Especially for harmonic oscillators the simulation of thermal states is challenging. There is an infinite number of eigenstates and the dimensionality of the Hilbert space is thus not limited. However, to perform computations on an infinite set of states is not feasible. Therefore, a truncation is necessary [Feh09]. But this truncation must be chosen carefully. A too tight truncation can lead to numerical artifacts, such as reflections at the truncation. Especially for thermal states, which can be expressed as

$$\hat{\rho}_{\text{th}} = \exp(-\beta\hat{H}). \quad (5.14)$$

For high temperatures the distribution has an exponentially decaying tail, yielding reflections quite easily at the truncation of the Liouville space. To ensure that the truncation is sufficient, we generated a thermal state with temperature higher than the simulations for this work. We examined at which entry the population became negligible. The criterion for this is when the population became lower than the precision of the floating point number. From there we took a factor of two to truncate the Hilbert space for good measure.

5.2.1. Algorithm for the simulation of classical and quantum trajectories

In this section the numerical algorithm for propagating the axial and radial degrees of freedom will be elaborated on. To achieve this propagation, two different propagators are required. The classical axial degree of freedom is simulated using the Störmer Verlet method and the quantum radial degree of freedom is propagated by the Newton propagator. To ensure that at every step the according dynamic is properly described, the density matrix and the Hamiltonian of the radial degree of freedom and the position z , velocity v and the force \mathcal{F} in axial direction needs to be updated.

The simulation is initialized at an initial axial position z_0 and velocity v_0 . The radial state is then set to the desired state. The value $R(t_i)$ is then computed and used to propagate the axial degree of freedom by one time step Δt . The newly obtained axial position is then used to update the Hamiltonian in accordance with eq. (5.3). From the Hamiltonian the Liouville superoperator is computed and the density matrix is propagated by Δt . This procedure is repeated until $t = \frac{\pi}{\omega_z}$. At this time the radial degree of freedom is coupled to a reservoir. The density matrix is thermalized to it and the procedure is started over again. This scheme is illustrated in fig. 5.3.

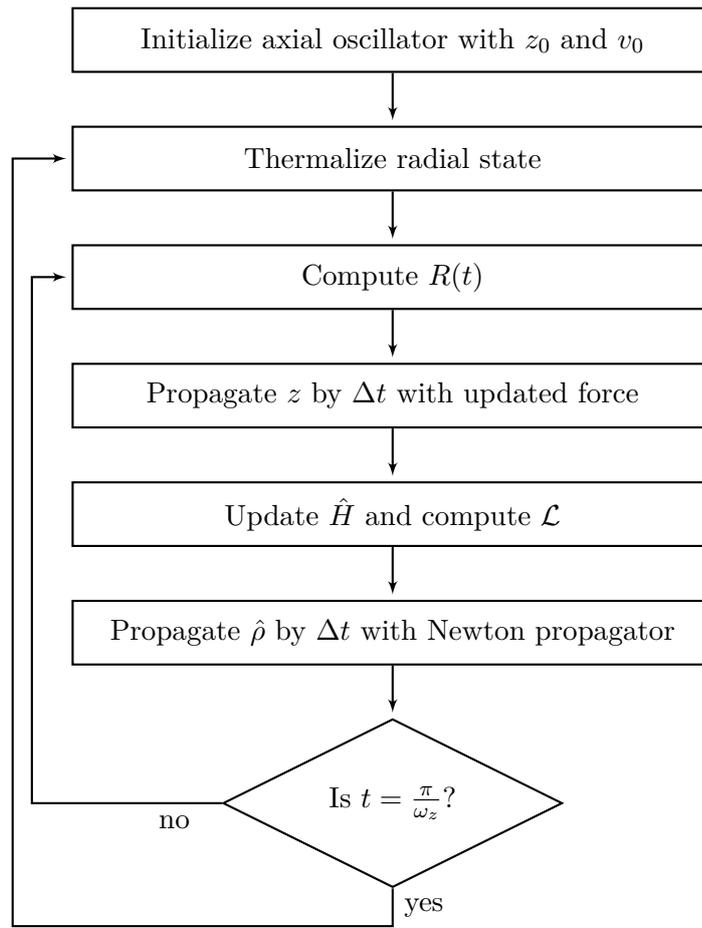


Figure 5.3.: Flowchart of the implemented algorithm.

The simulation is coded in Python [Van09]. For handling array operations, the NumPy [Har20] module is utilized. Visualization of the results are done using Matplotlib [Hun07]. The implementation of constructing operators and density matrices is left to the "quantum toolbox in python" (QuTiP [Joh13]). The implementation of the in section 4.3 described newton propagator for the simulation of the radial degree of freedom is imported from the newton propagator package [Goe18]. To optimize the performance of the simulation we used numba [Lam15] to compile the python code into machine code.

```

1 import os
2 import glob
3 import string
4 import pickle
5 import numpy as np
6 from numpy import pi
7 import matplotlib
8 import matplotlib.pyplot as plt
9 import newtonprop
10 from scipy.special import factorial
11 import qutip
12 from qutip.visualization import matrix_histogram, hinton

```

```

13 from qutip import position, momentum, expect, create, destroy,
    coherent_dm, thermal_dm, fock_dm, displace, squeeze, Qobj,
    operator_to_vector, wigner
14 from numba import jit

```

The function for updating the Hamiltonian implements eq. (5.3). The necessary parameters for this are the axial position z_pos , the parameter of the taper $\alpha = \tan(\theta)/r_0$, the axial trapping frequency w_x0 and the dimensions of the Hilbert space N . For sake of numerical precision, the constant $\hbar = 1$ which is denoted in the variable `hbar_1`.

```

1 def get_hamiltonian(z_pos, alpha, w_x0, N=20):
2     """function returns the Hamiltonian of the radial state."""
3     # input: z_pos, alpha, w_x0, N the system dimension
4     # return: Hamiltonian as a Qobject
5
6     taper_ratio = 1+z_pos*alpha
7     # w_x = w_x0/taper_ratio**2
8     hbar_1 = 1
9
10    # unperturbed Hamiltonian
11    H0 = np.array(np.zeros(shape=(N, N), dtype=np.complex128))
12    for i in range(N):
13        H0[i, i] = (0.5+i)*w_x0*hbar_1
14
15    # including the interaction term
16    H = Qobj(H0) + hbar_1/4*w_x0*((destroy(N)+create(N))**2)*(1/
17    taper_ratio**4-1)
18
19    return H

```

For achieving better precision, it is of advantage to compute in the dimensions intrinsic to the system. The trapping frequency in radial direction in the experiment is in the order of GHz and have typical displacements in axial direction of μm . Therefore, we adjust the constants to fit this chosen units.

```

1 two_pi = 2.0 * pi
2 k_B = 1.38064852e-23 #μm^2kg/K/μs^2
3 hbar = 6.62607004e-34/two_pi*1e6 # Planck constant in μm^2kg/μs
4 w_x0 = two_pi * 1.0 # radial trapping frequency in MHz at z=0
5 alpha = np.tan(30/180*pi)/(1e3)

```

The by QuTiP implemented initialization of a thermal state takes as argument the average phonon number `nbar`. The function `getNbar_thermal(temp,z_pos)` is calculating this value by a simple energy consideration. The total energy, which is the sum of the kinetic and potential energy is equivalent to the energy of the harmonic oscillator. Here the kinetic and potential term summed are equal to the

total thermal energy.

$$\begin{aligned} \left\langle \hbar\omega_x(z) \left(\hat{n} + \frac{1}{2} \right) \right\rangle &= T + V = \frac{1}{2}k_B T + \frac{1}{2}k_B T & (5.15) \\ \Leftrightarrow \hbar\omega_x(z) \left(\bar{n} + \frac{1}{2} \right) &= k_B T \\ \Leftrightarrow \bar{n} &= \frac{k_B T}{\hbar\omega_x(z)} - \frac{1}{2} \end{aligned}$$

```

1 def getNbar_thermal(temp=0.0,z_pos=0.0):
2     taper_ratio = 1+z_pos*alpha
3     #compute the energy
4     w_x_z=w_x0*(1/(taper_ratio**4))
5     thermalEnergy = k_B*temp
6     nbar = thermalEnergy/hbar/w_x_z-0.5
7     print(nbar)
8     return nbar

```

To generate the density matrices for different kind of states, the function `ini_rho` is used. This function returns the corresponding density matrix `rho` with dimension $N \times N$. The distribution is chosen by the string `distribution` which goes through a comparison to call the corresponding routine of the QuTiP library. The necessary parameters depending on the type of distribution is passed as further arguments of the function.

```

1 def ini_rho(N=20, mean_phonon_number=0, distribution = 'thermal',
2     squeeze_para = 0.25, displace_para = 0.25,temp=0.0,zpos=0.0):
3     """function generates the density matrix of the initial radial
4     state."""
5
6     if distribution == 'coherent':
7         rho = coherent_dm(N, np.sqrt(mean_phonon_number))
8         # coherent state generated with qutip.coherent_dm
9     elif distribution == 'fock':
10        rho = fock_dm(N, mean_phonon_number)
11        # Fock state generated with qutip.fock_dm
12    elif distribution == 'thermal':
13        nbar = getNbar_thermal(temp,zpos)
14        rho = thermal_dm(N, nbar, 'analytic')
15        rho = rho/rho.norm()
16        # Thermal state generated with qutip.thermal_dm
17    elif distribution == 'squeezed':
18        rho = coherent_dm(N, np.sqrt(mean_phonon_number))
19        s = squeeze(N, squeeze_para)
20        rho = s*rho*s.dag()
21    elif distribution == 'squeezedisplaced':
22        rho = coherent_dm(N, np.sqrt(0))
23        s = squeeze(N, squeeze_para)
24        d = displace(N, displace_para)
25        rho = d*s*rho*s.dag()*d.dag()
26    elif distribution == 'squeezedvacuum':
27        rho = fock_dm(N, 0)
28        s = squeeze(N, squeeze_para)
29        rho = s*rho*s.dag()

```

```
28     return rho
```

To update the Force, driving the ion along the axial degree of freedom, the expectation value of the squared position operator $\langle \hat{x}^2 \rangle$ needs to be evaluated. Therefore, we construct the position operator `x_op` in the Fock basis and compute the product with itself. The result is stored in `x_sq_op`. The same is done for the momentum operator `x_mom_op` and `x_sq_mom_op`. The expectation value of these operators are computed by the built-in function of QuTiP. The results are returned.

```

1 def get_x_var(rho, N=20):
2     """function returns the measurables (x,x^2,p,p^2) from the input
3     density matrix rho"""
4
5     x_op = destroy(N) + create(N)
6     x_sq_op = x_op*x_op
7
8     x_mom_op = -1j*(destroy(N) - create(N))
9     x_sq_mom_op = x_mom_op*x_mom_op
10
11    x_pos = expect(rho, x_op)
12    x_pos_var = expect(rho, x_sq_op)
13    x_mom = expect(rho, x_mom_op)
14    x_mom_var = expect(rho, x_sq_mom_op)
15
16    return x_pos, x_pos_var, x_mom, x_mom_var

```

The class `Result` is handling storing and saving the data of the simulation. The member `states` is an array of `qutip.Qobj`, where the density matrices are stored. `t_series` is an array tracking the time t . The remaining arrays are handling the variables required for performing simulation as denoted by the the comments in the code. The member variable `stroke` denotes the current stroke number. Due to the higher dimensions especially it is often not feasible to run the simulation for the whole duration due to limitation of allocatable memory for it. This is why the simulation is run till the next bath is reached. Because we assume for our system that the thermalization destroys all coherences, meaning that the radial degree of freedom is fully reset to the state with which it will evolve further. After a stroke, the obtained data is written to disk serialized in a binary format. This allows for fast writing as well as fast read in.

```

1 class Result():
2     """Dummy class for propagation result"""
3
4     states = [Qobj()]
5     t_series = np.array([])
6     z_positions = np.array([]) # position along z-axis
7     velocities = np.array([]) # velocity in z direction
8     x_positions = np.array([]) # expectation value of x-operator in
9     the radial degree of freedom
10    x_pos_var = np.array([]) # expectation value of x-operator^2 in
11    the radial degree of freedom
12    x_momenta = np.array([]) # expectation value of p-operator in the
13    radial degree of freedom
14    x_momenta_var = np.array([]) # expectation value of p-operator^2 in
15    the radial degree of freedom
16    axial_energy = np.array([]) # energy in axial direction
17    radial_energy = np.array([]) # energy in radial direction
18    z_acc_potential = np.array([]) # acceleration due to harmonic
19    potential
20    z_acc_radial = np.array([]) # acceleration due to force from x-z
21    coupling
22    stroke = 0 # current stroke number

```

```

19     def __init__(self):
20         self.states = [Qobj()]
21         self.t_series = np.linspace(0, two_pi/w_z/2, 1000)
22         self.z_positions = 0*self.t_series
23         self.velocities = 0*self.t_series
24         self.x_positions = 0*self.t_series
25         self.x_pos_var = 0*self.t_series
26         self.x_momenta = 0*self.t_series
27         self.x_momenta_var = 0*self.t_series
28         self.axial_energy = 0*self.t_series
29         self.radial_energy = 0*self.t_series
30         self.z_acc_potential = 0*self.t_series
31         self.z_acc_radial = 0*self.t_series
32         self.stroke = 0
33
34     def plot_pop_dynamics(self):
35         """function to visualize the simulation results"""
36
37         #fig1, ax_1 = hinton(self.states[0])
38         #fig2 = plt.figure(1)
39         #fig2, ax_2 = hinton(self.states[-1])
40
41         fig2 = plt.figure(2)
42         fig2.set_size_inches(18.5, 10.5, forward=True)
43
44         ax1 = plt.subplot(231)
45         ax2 = plt.subplot(232)
46         ax3 = plt.subplot(233)
47         ax4 = plt.subplot(234)
48         ax5 = plt.subplot(235)
49         ax6 = plt.subplot(236)
50
51         ax1.plot(self.t_series, self.z_positions)
52         ax2.plot(self.t_series, self.x_positions, self.t_series, self.
x_momenta)
53         ax3.plot(self.t_series, self.x_pos_var, self.t_series, self.
x_momenta_var)
54         ax4.plot(self.t_series, self.axial_energy-np.mean(self.
axial_energy), self.t_series, self.radial_energy-np.mean(self.
radial_energy))
55         ax5.plot(self.t_series, self.z_acc_radial+self.z_acc_potential)
56
57         ax1.set_xlabel(r'Time ( $\mu$  s)')
58         ax1.set_ylabel(r'$z$ position ( $\mu$  m)')
59         ax1.legend(['z position'])
60
61         ax2.set_xlabel(r'Time ( $\mu$  s)')
62         ax2.set_ylabel(r'$x$ position ( $\mu$  m)')
63         ax2.legend(['x position', 'x momentum'])
64
65         ax3.set_xlabel(r'Time ( $\mu$  s)')
66         ax3.set_ylabel(r'Variances ( $\mu$  m2)')
67         ax3.legend(['x pos. var.', 'x mom. var.'])
68
69         ax4.set_xlabel(r'Time ( $\mu$  s)')
70         ax4.set_ylabel(r'Energy change')

```

```

71     ax4.legend(['axial energy', 'radial energy'])
72
73     ax5.set_xlabel(r'Time ( $\mu$  s)')
74     ax5.set_ylabel(r'z accerlation')
75     ax5.legend(['potential force', 'radial force'])
76
77     ax1.set_xlim([self.t_series[0],self.t_series[-1]])
78     ax2.set_xlim([self.t_series[0],self.t_series[-1]])
79     ax3.set_xlim([self.t_series[0],self.t_series[-1]])
80     ax4.set_xlim([self.t_series[0],self.t_series[-1]])
81     ax5.set_xlim([self.t_series[0],self.t_series[-1]])
82
83     plt.show()
84
85     #fig1.savefig(os.getcwd()+"\\output\\1stroke"+str(self.stroke)
+'fig1.png')
86     fig2.savefig(os.getcwd()+"\\output\\2stroke"+str(self.stroke)+'
fig2.png')
87
88     plt.close("all")
89
90     def save(self, folder=".\\"):
91         directory = folder
92         if not os.path.exists(directory):
93             os.makedirs(directory)
94         filename = directory+'stroke'+str(self.stroke)+'.pkl'
95         with open(filename, 'wb') as output: # Overwrites any existing
file.
96             pickle.dump(self, output, pickle.HIGHEST_PROTOCOL)

```

The function `read_results` loads each stroke to extract the simulation results from the raw data. It iterates over all strokes and the relevant data is stored in the variables at the top of the body of the function. The relevant parameters are named analogues to the members of the class `Result`. The relevant parameters for evaluating further performance are stored in ASCII text files. The function returns the array storing the positions in axial direction and their respective times.

```

1 def read_results(folder):
2     path = os.getcwd()+"\\output\\"
3     filenameumber = len(glob.glob1(path+folder, '*.pkl'))
4     z_positions = np.array([])
5     times = np.array([])
6     z_acc_potential = np.array([])
7     z_acc_radial = np.array([])
8     z_acc = np.array([])
9     t = 0
10
11     for a in range(filenameumber):
12         filename = path+folder+'stroke'+str(a)+'.pkl'
13         print(filename)
14         with open(path+folder+'stroke'+str(a)+'.pkl', 'rb') as f:
15             data = pickle.load(f)
16             z_positions = np.append(z_positions, data.z_positions)
17             times = np.append(times, t+data.t_series)
18             z_acc_potential = np.append(z_acc_potential, data.
z_acc_potential)

```

```

19         z_acc_radial = np.append(z_acc_radial, data.z_acc_radial)
20         t = np.amax(times)
21
22
23     z_acc = z_acc_potential + z_acc_radial
24     #plt.plot(times, z_acc_potential, 'ro', label=' z_acc_potential')
25     #plt.plot(times, z_acc_radial, 'ro', label='z_acc_radial')
26     #plt.plot(times, z_acc, 'ro', label=' z_acc_potential+ z_acc_radial')
27     np.savetxt(path+folder+'accel.txt', z_acc)
28     np.savetxt(path+folder+'accel_pot.txt', z_acc_potential)
29     np.savetxt(path+folder+'accel_rad.txt', z_acc_radial)
30     np.savetxt(path+folder+"times.txt", times)
31     np.savetxt(path+folder+"pos.txt", z_positions)
32     return times, z_positions

```

The function `visualize` is used to display the full trajectory in z direction. The function `read_result` is called to obtain the result of the simulation. The full trajectory is saved as an image as well.

```

1 def visualize(folder):
2     times, position = read_results(folder)
3     plt.plot(times, position)
4     plt.xlabel('t /  $\mu$ s')
5     plt.ylabel('z position /  $\mu$ m')
6     plt.savefig('\output\'+folder+"dynamic.png")
7     plt.close("all")

```

The following functions are some auxiliary methods for making the code more readable.

```

1 def zero_vectorized(v):
2     return np.zeros(shape=v.shape, dtype=v.dtype)
3
4 def norm_vectorized(v):
5     return np.linalg.norm(v)
6
7 def inner_vectorized(a, b):
8     return np.vdot(a, b)

```

The function `velVerletStep` implements the updating of the axial position in dependence of the radial variance using the Störmer-Verlet algorithm. This algorithm is elaborated on in section 4.1. The force for this is computed in dependence of the expectation value of $\langle \hat{x}^2 \rangle$, which is passed in the variable `x_var`. To have a more thorough evaluation the acceleration due to the harmonic confinement in axial direction as well as the acceleration in axial direction due to the force generated by the variance in radial direction. Those values are returned.

```

1 @jit(nopython=True)
2 def velVerletStep(position_old, velocity_old, x_var, dt):
3     """function returns the new position, momentum of in the axial
4     direction"""
5
6     v_half = 0*velocity_old
7     position_new = 0*velocity_old
8     velocity_new = 0*velocity_old

```

```

9     v_half = velocity_old + dt/2.0*(-(w_z**2)*position_old + hbar*w_x0*
x_var*alpha/(1+position_old*alpha)**5/mass)
10
11     position_new = position_old+dt*v_half
12
13     velocity_new = v_half+ dt/2.0*(-(w_z**2)*position_new + hbar*w_x0*
x_var*alpha/(1+position_new*alpha)**5/mass)
14
15     acc_potential = -(w_z**2)*position_new
16     acc_radial = hbar*w_x0*x_var*alpha/(1+position_new*alpha)**5/mass
17
18     return position_new, velocity_new, acc_potential, acc_radial

```

To run the simulation for one stroke is implemented in `single_stroke`. The input parameters are the density matrix `rho`, the dimension of the Hilbert space `N`, the taper parameter `alpha`, initial position `z_pos` and velocity `z_vel`, the radial and axial trapping frequencies `w_x0` and `w_z` as well as a result object `result` to save the data generated by the simulation. The parameter `result` will be returned. As discussed above, first the trajectory of the ion in axial direction is propagated. The updated values are then used to compute Hamiltonian `H`. From the Hamiltonian, the Liouvillian is constructed and the density matrix `rho` is propagated by a time step `dt`.

```

1 def single_stroke(rho, N, alpha, z_pos, z_vel, w_x0, w_z, result):
2
3     dt = result.t_series[1]-result.t_series[0]
4     for step in range(len(result.t_series)):
5         result.x_positions[step], result.x_pos_var[step], result.
x_momenta[step], result.x_momenta_var[step] = np.real(get_x_var(
Qobj(rho.reshape(N,N)), N))
6
7         z_pos, z_vel, z_acc_p, z_acc_r = velVerlStep(z_pos,z_vel,result
.x_pos_var[step],dt)
8
9         result.z_positions[step] = z_pos
10        result.velocities[step] = z_vel
11        result.z_acc_potential[step] = z_acc_p
12        result.z_acc_radial[step] = z_acc_r
13        # Hamiltonian as a function of z position
14        H = get_hamiltonian(z_pos, alpha, w_x0, N)
15        # Liouvillian derived from the Hamiltonian
16        L = qutip.liouvillian(H)
17        Lmatrix = L.full()
18
19        # Propagating density matrix with newton propagator
20        rho = newtonprop.newton(lambda v: Lmatrix @ v, rho, dt, func=np
.exp, zero=zero_vectorized, norm=norm_vectorized, inner=
inner_vectorized, tol=1e-12)
21        result.states.append(Qobj(rho.reshape(N,N)))
22
23        if not (step % 100): print("step: ", step, "/", len(result.
t_series), "; z position", z_pos)
24
25    return result

```

Some further constants which are required for performing simulations are defined.

```

1 N = 80      # dimension of the simulation
2 two_pi = 2.0 * pi
3 mass=40*1.66053886e-27      # mass of Ca+ ion in kg

```

To run a simulation the number of strokes `n_strokes` needs to be defined. The simulation is started with the ion thermalized to the hot reservoir with temperature `T_hot`. The initial elongation `z_pos` is computed by simple consideration of the energy in the system. The ion is then propagated stroke by stroke, in which the ion is thermalized alternating to the hot and cold reservoirs `T_hot` and `T_cold` respectively. To piece the stroke wise simulation together, the initial position and velocity of the next stroke is obtained by the last position and velocity of the current stroke.

```

1 n_strokes = 20# number of strokes
2
3 result = Result()
4 z_vel = 0e-2
5 T_cold=2e-4 #K
6 T_hot=2.1e-4 #K
7 z_pos = - np.sqrt(k_B*T_hot*2/(w_z**2)/mass)
8
9 for stroke in range(n_strokes):
10     result.stroke = stroke
11     print("Stroke: ",stroke)
12     if (stroke % 2) == 0:
13
14         rho0 = ini_rho(N, 0, 'thermal',squeeze_para = 0.0,
15         displace_para = 0.0,temp=T_hot,zpos=z_pos)#hot bath: initial
16         density matrix
17
18         rho0_vectorized = operator_to_vector(rho0).full().flatten() #
19         convert qutip object to numpy array
20         rho = rho0_vectorized
21
22     else:
23         rho0 = ini_rho(N, 0, 'thermal',squeeze_para = 0.0,
24         displace_para = np.sqrt(0),temp=T_cold,zpos=z_pos) #cold bath:
25         initial density matrix
26         rho0_vectorized = operator_to_vector(rho0).full().flatten() #
27         convert qutip object to numpy array
28         rho = rho0_vectorized
29
30     result.states = [Qobj()]
31     result.states[0] = rho0
32     result = single_stroke(rho, N, alpha, z_pos, z_vel, w_x0, w_z,
33     result)
34
35     #With out giving the maximum amplitude as turning point
36     z_pos = result.z_positions[-1]
37     z_vel = result.velocities[-1]
38     result.plot_pop_dynamics()
39     #save data after every stroke
40     result.save('0_21mkto0_2withCorrection2\\')

```

5.3. Single-ion heat engine as a sensitive thermal probe

Compared to previous work [Roß16b], in which laser cooling was applied in axial direction to close the working cycle of the single-atom heat engine, we will consider the operation of the heat engine without dissipation in this direction. When the energy transferred to the axial flywheel is not dissipated, each time the engine performs a cycle energy will be added. This accumulative nature of the open cycle engine allows for a sensitive probe for especially small temperature differences. An estimate for the sensitivity for this device will be derived in section 5.3.1. Furthermore, in section 5.3.2 it will be demonstrated that exploiting quantum resources yields an even higher sensitivity.

In fig. 5.4 (a) the measurement protocol is depicted. The ion is prepared in a thermal state at Doppler temperature using Doppler cooling. Letting the engine run for N cycles, work is stored in the axial flywheel. To obtain the information about the energy, one does solely need to perform position measurements after the ion has interacted with each bath respectively. The necessary position measurement is done by collecting photons with an objective and using a camera for recording.

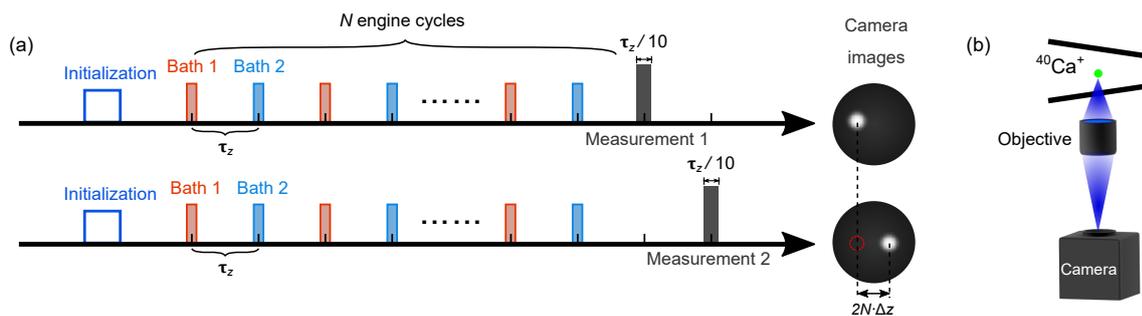


Figure 5.4.: (a) Measurement protocol. After initialization with Doppler cooling, the engine is set to operate under the driving of the two baths. After a number of N engine cycles, measurements are performed to determine the extreme of the axial location. The measurements are realized by illuminating the ion with a short laser pulse of duration $\tau_z/10$, at times τ_z and $2\tau_z$ after the $2N$ -th bath interaction, respectively. The physical separation of the ion's image on the camera directly translates to the axial oscillation amplitude $2N \cdot \Delta z$. It is necessary to repeat the protocol many times in order to precisely determine the amplitude. (b) The photons emitted by the $^{40}\text{Ca}^+$ ion at 397 nm are collected by an objective and focused on a camera.

5.3.1. Temperature difference estimation

To estimate the temperature difference of two different baths, the ion needs to be prepared in an initial state. This can be achieved by employing a Doppler cooling laser with a variable detuning. This allows to have control over the initial tempera-

ture T_0 . The initial position x_0 and velocity v_0 in axial direction is described by the Maxwell-Boltzmann distribution of the initial temperature.

$$f(z_0, v_{z0}) \propto \exp[-m(v_{z0}^2 + \omega_z^2 z_0^2)/2k_B T_0] \quad (5.16)$$

From here on, the atom is set to operate as an Otto engine. The two baths to be measured are thermalizing the ion to T_1 or T_2 . Each cycle performs work, which is stored in the axial degree of freedom. This increase in elongation at the turning points can be measured directly by directly observing the position of the ion on camera (fig. 5.4 b). This parameter can be determined by the following equations for the odd and even turning points.

$$\begin{aligned} z_n &= z_0 + \frac{n\hbar\omega_{x0}\gamma(R_2 - R_1)}{m\omega_z^2} \quad \text{for } n = 0, 2, 4, \dots \\ z_n &= -z_0 + \frac{2\hbar\omega_{x0}\gamma R_1 - (n-1)\hbar\omega_{x0}\gamma(R_2 - R_1)}{m\omega_z^2} \quad \text{for } n = 1, 3, 5, \dots, \end{aligned} \quad (5.17)$$

where R_1 and R_2 are given by section 5.1.1 with the corresponding temperatures T_1 and T_2 . Thus, the difference in the Temperature yields a difference in the elongation in axial direction Δz after a full cycle (see fig. 5.2 c). This increase reads

$$\Delta z = z_{n+2} - z_n = \frac{2\hbar\omega_{x0}\gamma}{m\omega_z^2}(R_2 - R_1). \quad (5.18)$$

In the case that $\hbar\omega_{x0} \ll k_B T$ can be approximated as

$$\Delta z = \frac{4k_B\gamma}{m\omega_z^2}\Delta T, \quad (5.19)$$

where $\Delta T = T_2 - T_1$. The sign of δZ is dependent of the temperature difference ΔT . The engine is run for N cycles, where $N \gg z_0/\Delta Z$. Then the elongation due to the work accumulated in the flywheel from driving the engine cycles becomes dominant. Regardless of the initial condition, the motion of the ion along the axial degree of freedom will be brought in phase of the bath interaction. After thermalizing to the hot/cold bath the ion moves towards the wide/narrow side of the tapered trap. To measure the temperature difference, we are proposing the following two measurements are performed. First, the position is illuminated by a pulse of $\tau = \tau_z/10$ at the turning point after $2N$ bath interactions. The position here can be directly obtained from the camera, which is observing the light emitted from the ion after it passed through a high numerical aperture (NA) objective. This measurement has to be done M times with restarting the engine, to obtain a better signal-to-noise ratio. From the acquired data, the ions position is determined by single-particle localization analysis [Für13].

$$\bar{z}_{2N} = \frac{2N\hbar\omega_{x0}\gamma(R_2 - R_1)}{m\omega_z^2}. \quad (5.20)$$

The second set of measurements is performed following the same procedure but at a time $2\tau_z$ after the $2N$ -th bath interaction, while the $(2N + 1)$ -th bath interaction is skipped (see lower panel of fig. 5.4a). The measured mean axial position is then

$$\bar{z}'_{2N} = -\frac{2N\hbar\omega_{x0}\gamma(R_2 - R_1)}{m\omega_z^2}, \quad (5.21)$$

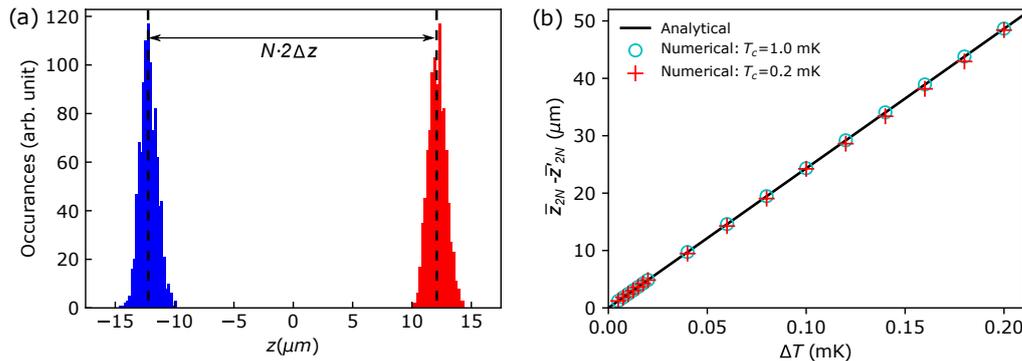


Figure 5.5.: (a) Distribution of the axial positions of the ion after $N = 10^5$ number of cycles for the two sets of measurements. $M = 2 \times 10^5$ initial positions were drafted randomly from the Boltzmann distribution with temperature $T_0 = 1$ mK. (b) The oscillation amplitude $\bar{z}_{2N} - \bar{z}'_{2N}$ after $N = 10^5$ engine cycles versus temperature difference ΔT between the two baths. Green circles and red crosses represent the results obtained from numerical simulations with $T_c = 1.0$ mK, 0.2 mK, respectively. The solid black line shows the prediction of section 5.3.1.

leading to the difference between the two positions

$$\bar{z}_{2N} - \bar{z}'_{2N} = 2N\Delta z \quad (5.22)$$

a linear function of the temperature differences ΔT . The distribution of the ions axial positions obtained by the numerical simulation with $\Delta T = 0.1$ mK, $T_0 = 1.0$ mK, $N = 10^5$ and $M = 2 \times 10^5$ is depicted in fig. 5.5 (a). The difference between the two measurement points $\bar{z}_{2N} - \bar{z}'_{2N}$ is shown in fig. 5.5 (b). The simulations are done with a base temperature of $T_c = 1.0$ mK and 0.2 mK which are plotted with green circles and red crosses, respectively. The number of the bath interactions is $N = 10^5$. The temperature difference ΔT is varied for the different simulations. The analytical solution is plotted as a black line. eq. (5.19) is in good agreement with the simulated data.

Because this method relies on localizing the ion, the major uncertainties stem from measuring the amplitude $2N\Delta z$. The precision with which a single particle can be localized is limited mainly by the signal-to-noise ratio of the camera. There is a multitude of factors to consider. Mainly the illumination time of the laser, the efficiency of photon collection, the quantum efficiency, and the background noise of the camera are having an influence on the signal-to-noise ratio. In the experiment an objective with a numerical aperture of 0.26 [Roß16b]. To obtain a precision of precision of ± 250 nm, an acquisition of 2×10^5 repeated measurements were necessary [Roß16a]. This resolution limitation can be translated to a temperature using eq. (5.19). The temperature difference uncertainty is ± 2 μ K.

To note is, that this measurement scheme is only dependent of the temperature difference between the two thermal reservoir and is not sensitive towards the base line temperature in the limit of $\hbar\omega_x \ll k_B T_{1,2}$. This yields the protocol to be used

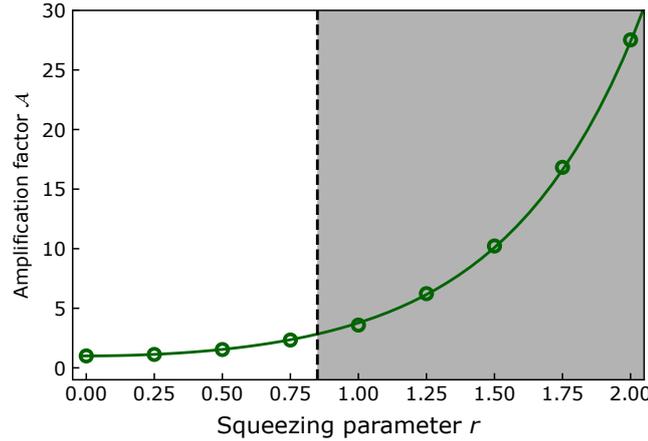


Figure 5.6.: Amplification factor \mathcal{A} as a function of the amplitude of squeezing r . Green circles show the results obtained from the numerical simulations. The solid line represents the outcome of eq. (5.26). The dashed black line indicates the value $r = 0.77$, and the shaded area denote the region where the squeezing operations brings the working medium into the quantum regime.

to detect small temperature differences at high base temperatures. However, if one of the baths is well known, the obtained temperature difference can then be used to characterize the absolute temperature of the unknown bath.

5.3.2. The squeezed engine: Enhancing sensitivity using quantum resources

To enhance the sensitivity of the single-atom heat probe, one can consider using quantum resources to enhance the performance. Here, it is demonstrated how the usage of squeezing after the isochoric strokes yields an amplification of the oscillation amplitude. This enhancement might be used in experiment to be able to detect even smaller temperature differences than $\pm 2 \mu\text{K}$. The squeezing applied to the motional radial state is described by the squeezing operator

$$\hat{S}(\xi) = \exp\left(\frac{1}{2}(\xi^* \hat{a}^2 - \xi \hat{a}^{\dagger 2})\right). \quad (5.23)$$

Because the squeezing is applied after the ion is thermalized to a thermal state, the end state is described by a squeezed thermal state $\hat{S}(\xi)\rho_{th}\hat{S}^\dagger(\xi)$. In this case, as $X_1(0) \neq 0, X_2(0) \neq 0$, $R(t)$ becomes time-dependent (see Eq. eq. (5.10)). The axial oscillation grows after one engine cycle by

$$\Delta z' = \frac{4\kappa\gamma\hbar}{m\omega_z} (\cosh(2r) + \sinh(2r) \cos(\alpha)/(4\kappa^2 - 1)) (n_{th}^1 - n_{th}^2), \quad (5.24)$$

with $\kappa = \omega_{x0}/\omega_z$, $n_{th}^{1,2} = (e^{\beta_{1,2}\hbar\omega_{x0}} - 1)^{-1}$, r and α the amplitude and phase of the squeezing parameter following $\xi = r e^{i\alpha}$. To obtain the maximal growth the

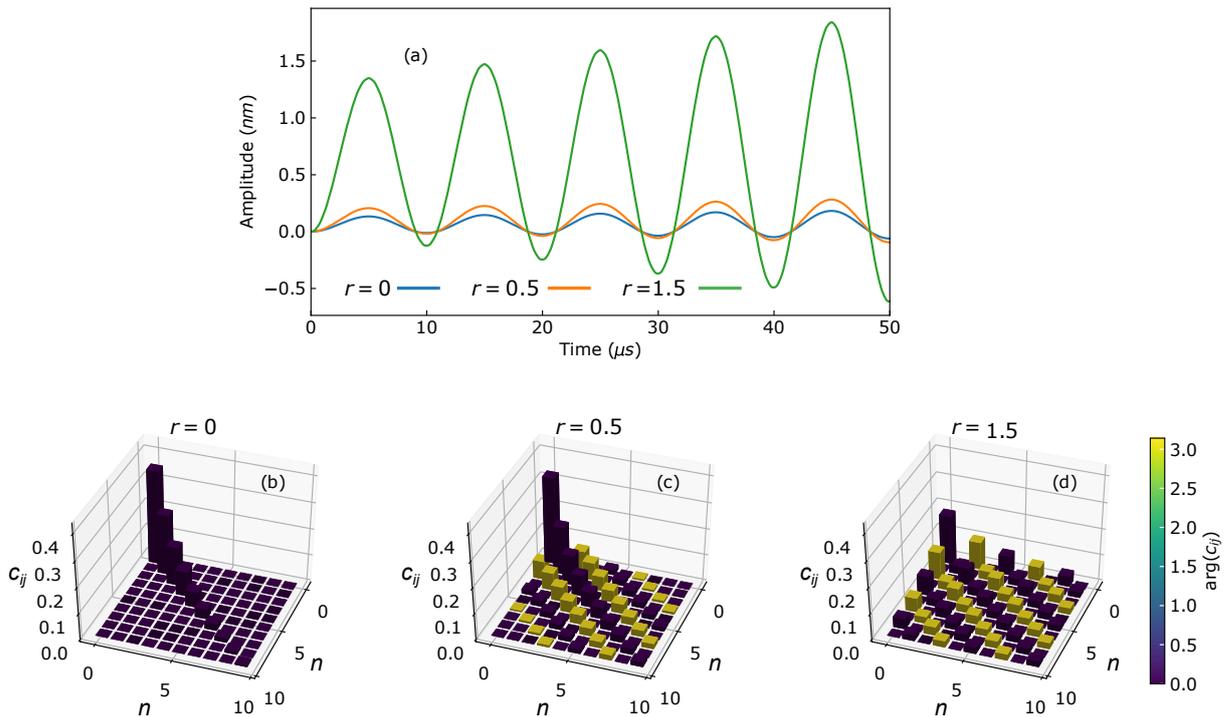


Figure 5.7.: (a) Amplification of the axial oscillation amplitude by squeezing the working medium after interaction with both baths. The simulations are performed with $T_h = 0.11$ mK and $T_c = 0.1$ mK. The blue, orange, and green curves represent the trajectories for $r = 0, 0.5$ and 1.5 , respectively. (b-d) Excerpt of the density matrices after interacting with the hot bath and the squeezing operation, where the color encodes the phase of the entries $\arg(c_{ij})$.

squeezing needs be applied in phase, which means that the phase $\alpha = 0$. In the hot limit $\hbar\omega_{x0} \ll k_B T$ eq. (5.24) can be written as

$$\Delta z' = \frac{4\gamma\hbar}{m\omega_z^2} (\cosh(2r) + \sinh(2r)/(4\kappa^2 - 1)) \Delta T. \quad (5.25)$$

To quantify the enhancement of the applied squeezing to the working medium, define the squeezing amplification factor

$$\mathcal{A} = \frac{\Delta z'}{\Delta z} = \cosh(2r) + \sinh(2r)/(4\kappa^2 - 1), \quad (5.26)$$

where Δz denotes the growth of the amplitude without squeezing. In fig. 5.6 this dependency is depicted as the green solid line. Furthermore, numerical simulations are shown as green circles. The simulations were done at $T_1 = 0.11$ mK, $T_2 = 0.1$ mK and $\kappa = 10$ with different squeezing amplitudes. This additional squeezing yields an

amplification of the axial oscillation by an order of magnitude, allowing for detection of much smaller temperature difference which in the scheme above would not be detectable due to experimental constraints.

To point out that quantum squeezing is applied, the squeezing operation must suppress fluctuations below the symmetric quantum limit in one of the quadrature components. From the uncertainty principle it follows that the canonical conjugate quadrature component will have increased fluctuations.

To satisfy this condition, the variance of one of the quadratures needs to be smaller than $1/4$. This corresponds to a phase in which the Glauber-Sudarshan distribution is negative [Sud63; Wal83], which yields the following condition:

$$\frac{1}{4}(2n_{th} + 1)e^{\pm 2r} < \frac{1}{4}. \quad (5.27)$$

This bound is illustrated in fig. 5.6. In the case for the simulated data the squeezing amplitude needs to be $r > 0.77$ in order to violate eq. (5.27). This value is depicted by the horizontal dashed line and the shaded region corresponds to the regime of quantum squeezing. In the white region where classical squeezing, the signal is still amplified. This amplification is considerable small. To get a significant enhancement of the oscillation one is required to enter the quantum regime. For further insight the simulated axial trajectories for $r = 0, 0.5$ and 1.5 are displayed in fig. 5.7a. The initial position and velocity is chosen to be zero for all simulations. The corresponding density matrices after interacting with the hot bath and the squeezing operation are depicted in fig. 5.7b-d. The phase of the entries of the density matrices $\arg(c_{ij})$ is indicated by the color scale. The applied squeezing yields a higher occupation number and excitation of the off-diagonal elements.

Squeezing in trapped ion experiments has been demonstrated by fast trap voltage control [Hei90; Alo13; Bur19] or dynamic optical forces [Mee96; Din14; Mas19]. The motional state is modulated at twice the trapping frequency. In a recent experimental implementation of a quantum absorption refrigerator [Mas19] squeezing operations of up to $r = 2$ was realized with two detuned lasers.

Even though the relation between temperature and axial oscillation growth is not as simple as the free running system, this scheme can be implemented to evaluate even smaller temperature differences.

6. Summary and Outlook

In this thesis the feasibility of a sensitive thermal probe with a single atom has been demonstrated by analytical and numerical methods. The system can detect temperature differences as small as $2\ \mu\text{K}$. Even higher sensitivity can be achieved by implementing squeezing operations on the working medium after the bath interactions. The presented scheme is independent of the base temperature and only requires Doppler cooling as initialization. To highlight is that the presented scheme is capable to measure small temperature differences even at high base temperatures, corresponding to the limit of $\hbar\omega_{x0} \ll k_B T$.

Furthermore, this scheme can be used to characterize non-thermal baths. The corresponding effective temperature of the working medium after thermalization can be estimated in a similar fashion. Because the dynamics in the axial degree of freedom depends on the radial degree of freedom, if the type of bath is known, one can qualitatively evaluate the corresponding operation on the ion. The properties of the radial degrees of freedom are only measured indirectly, which allows the engine to preserve its quantum features during its free evolution.

In the scheme presented here one has to manually couple the ion to the hot/cold bath. However, if one prepares the experiment, that two laser beams with different detunings are placed spatially separated, one can employ due to the intrinsic properties of the heat engine an autonomous heat engine. The different detunings of the lasers are corresponding here to different thermal reservoirs at different temperatures. This due to the Doppler temperature being dependent on the temperature. Once the ion is thermalizing to the "hotter" (i.e. more blue detuned) laser the ion is experience the force due to the higher temperature. The ion will be displaced by this force along the axis until it reaches the "cold" (i.e. more red detuned) laser. This laser cools the radial degree of freedom to its respective Doppler temperature. This will decrease the displacing force along the axis and the ion is experiencing the pondermotive force of the harmonic axial confinement. Due to this force the ion will propagate in the opposite direction as before until it reaches the interaction region of the "hot" laser.

A. Appendix

A.1. Laplace Transform

A commonly employed method for solving linear time-invariant differential equations is the Laplace transform. This transform is defined as

$$\tilde{f}(s) = \mathcal{L}[f(t)] = \int_0^{\infty} dt f(t) \exp(-st), \quad (\text{A.1})$$

where $s \in \mathbb{C}$. Here $f(t)$ denotes an arbitrary function which is continuous on the interval $[0, t], t \in \mathbb{R}_{>0}$. This transformation can be employed to convert differential equation, which have elaborate solutions, to form which can be solved by algebraic means. The found solution can be transformed back to the initial problem using the inverse Laplace transform.

$$f(t) = \mathcal{L}^{-1} [\tilde{f}(s)] \quad (\text{A.2})$$

The Laplace transform is also useful for finding the solution of derivatives, given an initial condition $x(0)$.

$$\mathcal{L} \left[\frac{dx(t)}{dt} \right] = s\mathcal{L}[x(t)] - x(0) \quad (\text{A.3})$$

For n -th order derivatives the transform can be obtain by applying eq. (A.3) n times.

$$\mathcal{L} \left[\frac{d^n x(t)}{dt^n} \right] = s^n \mathcal{L}[x(t)] - s^{n-1}x(0) - \dots - \frac{d^{n-1}x}{dt^{n-1}}(0) \quad (\text{A.4})$$

A useful relation is the convolution theorem, with which the reader might be familiar from Fourier transform. The Laplace transform of the convolution of two real valued functions $a(t)$ and $b(t)$ is equivalent to the multiplication of the Laplace transformed of each function.

$$\mathcal{L} [a(t) * b(t)] = \mathcal{L} [a(t)] \cdot \mathcal{L} [b(t)] \quad (\text{A.5})$$

A.1.1. Solving the driven harmonic oscillator using the Laplace transform method

To solve an damped and driven harmonic oscillator

$$s^2 \mathcal{L}[x(t)] - sx(0) - v(0) + \frac{\gamma}{m} s \mathcal{L}[x(t)] - \frac{\gamma}{m} x(0) + \omega^2 \mathcal{L}[x(t)] = \frac{1}{m} \mathcal{L}[f(t)] \quad (\text{A.6})$$

For sake of brevity we will introduce the notations $\tilde{x}(s) := \mathcal{L}[x(t)]$ and $\tilde{f}(s) := \mathcal{L}[f(t)]$. Furthermore, we denote $\gamma_0 = \gamma/(2m)$. This equation reads in the new notation:

$$s^2\tilde{x}(s) - sx(0) - v(0) + 2\gamma_0s\tilde{x}(s) - 2\gamma_0 + \omega^2\tilde{x}(s) = \frac{1}{m}\tilde{f}(s). \quad (\text{A.7})$$

With initial conditions $x(0) = x_0$ and $v(0) = v_0$ the equation becomes

$$[s^2 + 2\gamma_0s + \omega^2] \tilde{x}(s) - sx(0) - v(0) - 2\gamma_0x_0 = \frac{1}{m}\tilde{f}(s) \quad (\text{A.8})$$

$$[s^2 + 2\gamma_0s + \omega^2] \tilde{x}(s) = \frac{1}{m}\tilde{f}(s) + sx_0 + v_0 + 2\gamma_0x_0 \quad (\text{A.9})$$

$$\tilde{x}(s) = \frac{\frac{1}{m}\tilde{f}(s) + sx_0 + v_0 + 2\gamma_0x_0}{s^2 + 2\gamma_0s + \omega^2}. \quad (\text{A.10})$$

Let us now examine the denominator.

$$s^2 + 2\gamma_0s + \omega^2 = s^2 + 2\gamma_0s + \gamma_0^2 - \gamma_0^2 + \omega^2 \quad (\text{A.11})$$

$$= (s + \gamma_0)^2 + \omega^2 - \gamma_0^2 \quad (\text{A.12})$$

$$= (s + \gamma_0)^2 + \Omega^2 \quad (\text{A.13})$$

Here, $\Omega := \omega^2 - \gamma_0^2$.

$$\tilde{x}(s) = \frac{\frac{1}{m}\tilde{f}(s) + sx_0 + v_0 + 2\gamma_0x_0}{(s + \gamma_0)^2 + \Omega^2} \quad (\text{A.14})$$

$$\tilde{x}(s) = \frac{\frac{1}{m}\tilde{f}(s)}{(s + \gamma_0)^2 + \Omega^2} + \frac{sx_0 + \gamma_0x_0}{(s + \gamma_0)^2 + \Omega^2} + \frac{v_0 + \gamma_0x_0}{(s + \gamma_0)^2 + \Omega^2} \quad (\text{A.15})$$

$$\tilde{x}(s) = \frac{1}{m\Omega} \frac{\Omega\tilde{f}(s)}{(s + \gamma_0)^2 + \Omega^2} + \frac{sx_0 + \gamma_0x_0}{(s + \gamma_0)^2 + \Omega^2} + \frac{v_0 + \gamma_0x_0}{(s + \gamma_0)^2 + \Omega^2} \frac{\Omega}{\Omega} \quad (\text{A.16})$$

With the help of the following Laplace inverse transforms, we can determine the trajectory in the time domain. [Obe73]

$$e^{\alpha t} \sin(\beta t) = \mathcal{L}^{-1} \left[\frac{\beta}{(s - \alpha)^2 + \beta^2} \right] \quad (\text{A.17})$$

$$e^{\alpha t} \cos(\beta t) = \mathcal{L}^{-1} \left[\frac{s - \alpha}{(s - \alpha)^2 + \beta^2} \right] \quad (\text{A.18})$$

Thus, the inverse Laplace transform eq. (A.16) reads.

$$x(t) = \mathcal{L}^{-1} [\tilde{x}(s)] \quad (\text{A.19})$$

$$= \mathcal{L} \left[\frac{1}{m\Omega} \frac{\Omega \tilde{f}(s)}{(s + \gamma_0)^2 + \Omega^2} + \frac{s x_0 + \gamma_0 x_0}{(s + \gamma_0)^2 + \Omega^2} + \frac{\Omega}{(s + \gamma_0)^2 + \Omega^2} \frac{v_0 + \gamma_0 x_0}{\Omega} \right] \quad (\text{A.20})$$

$$= \mathcal{L}^{-1} \left[\frac{1}{m\Omega} \frac{\Omega \tilde{f}(s)}{(s - (-\gamma_0))^2 + \Omega^2} \right] + \mathcal{L}^{-1} \left[x_0 \frac{s - (-\gamma_0)}{(s - (-\gamma_0))^2 + \Omega^2} \right] \quad (\text{A.21})$$

$$+ \mathcal{L}^{-1} \left[\frac{\Omega}{(s - (-\gamma_0))^2 + \Omega^2} \frac{v_0 + \gamma_0 x_0}{\Omega} \right] \quad (\text{A.22})$$

$$= \frac{1}{m\Omega} \exp(-\gamma_0 t) \sin(\Omega t) \cdot \mathcal{L}^{-1} [\tilde{f}(s)] + x_0 \exp(-\gamma_0 t) \cos(\Omega t) \quad (\text{A.23})$$

$$+ \frac{v_0 + \gamma_0 x_0}{\Omega} \exp(-\gamma_0 t) \sin(\Omega t) \quad (\text{A.24})$$

However, we are examining in this work thermally distributed ensembles. Therefore, the particles are Gaussian distributed with $x_0 = \langle x \rangle = 0$ and $v_0 = \langle v \rangle = 0$. This means that for such an ensemble the equation of motion reads

$$x(t) = \frac{1}{m\Omega} \exp(-\gamma_0 t) \sin(\Omega t) \cdot \mathcal{L}^{-1} [\tilde{f}(s)] \quad (\text{A.25})$$

We will denote the force independent term as impulse response function $H(t)$.

$$H(t) = \frac{1}{m\Omega} \exp(-\gamma_0 t) \sin(\Omega t) \quad (\text{A.26})$$

A.2. Implementation of the Strömer-Verlet method

In this chapter, an implementation of the Strömer-Verlet method is presented. As an example, we will use a harmonic potential which confines a particle in three dimension. The acceleration can be calculated by simply taking minus the derivative of the potential with respect to the coordinate and divide it by the mass.

```

1 #include <iostream>
2 #include <fstream>
3 #include <limits>
4 #define _USE_MATH_DEFINES
5 #include <math.h>
6
7 void stroemerVerlet(double * oldPos, double * oldVel, double * omega,
8     double dt, int dim, double * newPos, double * newVel) {
9     double * velHalf = new double[dim];
10    for (int i = 0; i < dim; ++i) {
11        velHalf[i] = oldVel[i] + dt / 2.0 * (-omega[i] * omega[i] * oldPos[i]);
12    }
13    for (int i = 0; i < dim; ++i) {
14        newPos[i] = oldPos[i] + dt * velHalf[i];
15    }
16    for (int i = 0; i < dim; ++i) {

```

```
16     newVel[i] = velHalf[i]+dt / 2.0*(-omega[i] * omega[i] * newPos[i]);
17 }
18 delete[] velHalf;
19 }
20
21
22 int main(int argc, char ** argv) {
23
24     double finalTime = 1e-5;
25     double dt = sqrt(std::numeric_limits<double>::epsilon()*finalTime);
26     int steps = static_cast<int>(finalTime / dt);
27     std::cout << steps << std::endl;
28
29     //allocate array for position and velocity
30     double ** position = new double*[steps];
31     double ** velocity = new double*[steps];
32     for (int i = 0; i < steps; ++i) {
33         position[i] = new double[3];
34         velocity[i] = new double[3];
35     }
36     double * velHalf = new double[3];
37
38     //set initial conditions
39     position[0][0] = 1.0e-6;
40     position[0][1] = -1.0e-6;
41     position[0][2] = 1.0e-6;
42     velocity[0][0] = 0.0e-6;
43     velocity[0][1] = 0.0e-6;
44     velocity[0][2] = 0.0e-6;
45
46     //set frequency
47     double omega[3];
48     omega[0] = 2*M_PI*1.0e6;
49     omega[1] = 2 * M_PI*1.0e6;
50     omega[2] = 2 * M_PI*0.1e6;
51
52     for (int i = 1; i < steps; ++i) {
53         stroemerVerlet(position[i - 1], velocity[i - 1], omega, dt,3,
54             position[i], velocity[i]);
55         if (!(i % 100)) std::cout << i << " / " << steps << std::endl;
56     }
57
58     std::ofstream results;
59     results.open("results.txt");
60     for (int i = 0; i < steps; ++i) {
61         results << i * dt << '\t' << position[i][0] << '\t' << position[i]
62             [1] << '\t' << position[i][2] << '\t' << velocity[i][0] << '\t' <<
63             velocity[i][1] << '\t' << velocity[i][2] << std::endl;
64     }
65
66     results.close();
67     std::cin.get();
68     return 0;
69 }
```

B. List of publications

The following article has been published during the preparation of this thesis.

Single-atom heat engine as a sensitive thermal probe

A. Levy, M. Göb, B. Deng, K. Singer and E Torrontegui and D. Wang
New Journal of Physics **22**, 093020 (2020).

We propose employing a quantum heat engine as a sensitive probe for thermal baths. In particular, we study a single-atom Otto engine operating in an open thermodynamic cycle. Owing to its cyclic nature, the engine is capable of translating small temperature differences between two baths into a macroscopic oscillation in a flywheel. We present analytical and numerical modeling of the quantum dynamics of the engine and estimate it to be capable of detecting temperature differences as small as $2 \mu\text{K}$. This sensitivity can be further improved by utilizing quantum resources such as squeezing of the ion motion. The proposed scheme does not require quantum state initialization and is able to detect small temperature differences in a wide range of base temperatures.

The following article has been published unrelated to this work.

Transient Non-Confining Potentials for Speeding Up a Single Ion Heat Pump

E. Torrontegui, S. T. Dawkins, M. Göb, K. Singer
New Journal of Physics **20**, 105001 (2018).

We propose speeding up a single ion heat pump based on a tapered ion trap. If a trapped ion is excited in an oscillatory motion axially the radial degrees of freedom are cyclically expanded and compressed such that heat can be pumped between two reservoirs coupled to the ion at the turning points of oscillation. Through the use of invariant-based inverse engineering, we can speed up the process without sacrificing the efficiency of each heat pump cycle. This additional control can be supplied with additional control electrodes or it can be encoded into the geometry of the radial trapping electrodes. We present novel insight how speed up only limited by the magnitude of the control voltage can be achieved through the use of inverted harmonic potentials. We have verified that stable trapping conditions can be achieved.

Bibliography

- [Aba12] O. Abah, J. Roßnagel, G. Jacob, S. Deffner, F. Schmidt-Kaler, K. Singer, E. Lutz, O. Abah, J. Roßnagel, G. Jacob, S. Deffner, F. Schmidt-Kaler, K. Singer, and E. Lutz, “Single-Ion Heat Engine at Maximum Power”, *Phys. Rev. Lett.* **109**, 203006 (2012).
- [Aba16] O. Abah and E. Lutz, “Optimal performance of a quantum Otto refrigerator”, *EPL (Europhysics Letters)* **113**, 60002 (2016).
- [Aba19] O. Abah and M. Paternostro, “Shortcut-to-adiabaticity Otto engine: A twist to finite-time thermodynamics”, *Physical Review E* **99**, 022110 (2019).
- [Alo13] J. Alonso, F. M. Leupold, B. C. Keitch, and J. P. Home, “Quantum control of the motional states of trapped ions through fast switching of trapping potentials”, *New J. Phys.* **15**, 023001 (2013).
- [Ash95] G. Ashkenazi, R. Kosloff, S. Ruhman, and H. Tal-Ezer, “Newtonian propagation methods applied to the photodissociation dynamics of I₃⁻”, *The Journal of Chemical Physics* **103**, 10005–10014 (1995).
- [Bas17] D. Basilewitsch, R. Schmidt, D. Sugny, S. Maniscalco, and C. P. Koch, “Beating the limits with initial correlations”, *New Journal of Physics* **19**, 113042 (2017).
- [Bas21] D. Basilewitsch, J. Fischer, D. M. Reich, D. Sugny, and C. P. Koch, “Fundamental bounds on qubit reset”, *Physical Review Research* **3**, 013110 (2021).
- [Ben00] C. M. Bender, D. C. Brody, and B. K. Meister, “Quantum mechanical Carnot engine”, *Journal of Physics A: Mathematical and General* **33**, 4427 (2000).
- [Ber98] D. J. Berkeland, J. D. Miller, J. C. Bergquist, W. M. Itano, and D. J. Wineland, “Minimization of ion micromotion in a Paul trap”, *Journal of Applied Physics* **83**, 5025 (1998).
- [Bre19] S. M. Brewer, J.-S. Chen, A. M. Hankin, E. R. Clements, C. W. Chou, D. J. Wineland, D. B. Hume, and D. R. Leibbrandt, “²⁷Al⁺ Quantum-Logic Clock with a Systematic Uncertainty below 10⁻¹⁸”, *Physical Review Letters* **123**, 033201 (2019).
- [Bur19] S. C. Burd, R. Srinivas, J. J. Bollinger, A. C. Wilson, D. J. Wineland, D. Leibfried, D. H. Slichter, and D. T. C. Allcock, “Quantum amplification of mechanical oscillator motion”, *Science* **364**, 1163–1165 (2019).
- [Çak19] B. Çakmak and Ö. E. Müstecaplı, “Spin quantum heat engines with shortcuts to adiabaticity”, *Physical Review E* **99**, 032108 (2019).
- [Cal85] H. B. Callen, “Thermodynamics and an Introduction to Thermostatistics”, John Wiley & Sons (1985).

- [Car72] S. Carnot, “Réflexions sur la puissance motrice du feu et sur les machines propres à développer cette puissance”, *Annales scientifiques de l’École normale supérieure* **1**, 393 (1872).
- [Cia19] S. Cialdi, C. Benedetti, D. Tamascelli, S. Olivares, M. G. A. Paris, and B. Vacchini, “Experimental investigation of the effect of classical noise on quantum non-Markovian dynamics”, *Physical Review A* **100**, 052104 (2019).
- [Cof12] W. T. Coffey and Y. P. Kalmykov, “The Langevin equation : with applications to stochastic problems in physics, chemistry, and electrical engineering”, World Scientific (2012).
- [Cur75] F. L. Curzon and B. Ahlborn, “Efficiency of a Carnot engine at maximum power output”, *American Journal of Physics* **43**, 22 (1975).
- [Dan19] R. Dann and R. Kosloff, “Quantum Signatures in the Quantum Carnot Cycle”, eprint: arXiv:1906.06946 (2019).
- [Den13] J. Deng, Q.-h. Wang, Z. Liu, P. Hänggi, J. Gong, J. Deng, Q.-h. Wang, Z. Liu, P. Hänggi, and J. Gong, “Boosting work characteristics and overall heat-engine performance via shortcuts to adiabaticity: Quantum and classical systems”, *Physical Review E* **88**, 062122 (2013).
- [Din14] S. Ding, H. Loh, R. Hablutzel, M. Gao, G. Maslennikov, and D. Matsukevich, “Microwave Control of Trapped-Ion Motion Assisted by a Running Optical Lattice”, *Phys. Rev. Lett.* **113**, 073002 (2014).
- [Dir27] P. A. M. Dirac, “The quantum theory of the emission and absorption of radiation”, *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character* **114**, 243–265 (1927).
- [Esp10a] M. Esposito, R. Kawai, K. Lindenberg, and C. V. den Broeck, “Efficiency at Maximum Power of Low-Dissipation Carnot Engines”, *Physical Review Letters* **105**, 150603 (2010).
- [Esp10b] M. Esposito, R. Kawai, K. Lindenberg, and C. V. den Broeck, “Quantum-dot Carnot engine at maximum power”, *Physical Review E* **81**, 041106 (2010).
- [Feh09] H. Fehske, J. Schleede, G. Schubert, G. Wellein, V. S. Filinov, and A. R. Bishop, “Numerical approaches to time evolution of complex quantum systems”, **373**, 2182 (2009).
- [Für13] A. Fürstenberg and M. Heilemann, “Single-molecule localization microscopy – near-molecular spatial resolution in light microscopy with photoswitchable fluorophores”, **15**, 14919 (2013).
- [Gal09] M. Galassi, “GNU scientific library : reference manual”, Network Theory Bristol(2009).
- [Gar05] J. J. García-Ripoll, P. Zoller, and J. I. Cirac, “Quantum information processing with cold atoms and trapped ions”, *Journal of Physics B: Atomic, Molecular and Optical Physics* **38**, S567–S578 (2005).

- [Gel15] D. Gelbwaser-Klimovsky, W. Niedenzu, P. Brumer, and G. Kurizki, “Power enhancement of heat engines via correlated thermalization in a three-level “working fluid””, *Scientific Reports* **5** (2015).
- [Gev92] E. Geva and R. Kosloff, “A quantum-mechanical heat engine operating in finite time. A model consisting of spin-1/2 systems as the working fluid”, *The Journal of Chemical Physics* **96**, 3054 (1992).
- [Gil96] D. T. Gillespie, “Exact numerical simulation of the Ornstein-Uhlenbeck process and its integral”, *Physical Review E* **54**, 2084–2091 (1996).
- [Göb19] M. Göb, “Speeding Up a Single Ion Thermal Machine”, Bachelor thesis, Universität Kassel (2019).
- [Goe15] M. H. Goerz, “Optimizing Robust Quantum Gates in Open Quantum Systems”, Universität Kassel (2015).
- [Goe18] M. H. Goerz, *The Newton prop python package*(, 2018), URL: <https://github.com/qucontrol/newtonprop>.
- [Hai03] E. Hairer, C. Lubich, and G. Wanner, “Geometric numerical integration illustrated by the Störmer–Verlet method”, *Acta Numerica* **12**, 399 (2003).
- [Har20] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, “Array programming with NumPy”, *Nature* **585**, 357–362 (2020).
- [Hei90] D. J. Heinzen and D. J. Wineland, “Quantum-limited cooling and detection of radio-frequency oscillations by laser-cooled ions”, *Phys. Rev. A* **42**, 2977–2994 (1990).
- [Het15] M. Hettrich, T. Ruster, H. Kaufmann, C. Roos, C. Schmiegelow, F. Schmidt-Kaler, and U. Poschinger, “Measurement of Dipole Matrix Elements with a Single Trapped Ion”, *Physical Review Letters* **115**, 143003 (2015).
- [Hun07] J. D. Hunter, “Matplotlib: A 2D Graphics Environment”, *Computing in Science and Engineering* **9**, 90–95 (2007).
- [Joh13] J. Johansson, P. Nation, and F. Nori, “QuTiP 2: A Python framework for the dynamics of open quantum systems”, *Computer Physics Communications* **184**, 1234–1240 (2013).
- [Kas95] N. Kasdin, “Discrete simulation of colored noise and stochastic processes and 1/f/sup α / power law noise generation”, *Proceedings of the IEEE* **83**, 802–827 (1995).
- [Kau17] H. Kaufmann, T. Ruster, C. Schmiegelow, M. Luda, V. Kaushal, J. Schulz, D. von Lindenfels, F. Schmidt-Kaler, and U. Poschinger, “Scalable Creation of Long-Lived Multipartite Entanglement”, *Physical Review Letters* **119**, 150503 (2017).

- [Kau20] V. Kaushal, B. Lekitsch, A. Stahl, J. Hilder, D. Pijn, C. Schmiegelow, A. Bermudez, M. Müller, F. Schmidt-Kaler, and U. Poschinger, “Shuttling-based trapped-ion quantum information processing”, *AVS Quantum Science* **2**, 014101 (2020).
- [Kon03] B. Kongtragool and S. Wongwises, “A review of solar-powered Stirling engines and low temperature differential Stirling engines”, *Renewable and Sustainable Energy Reviews* **7**, 131 (2003).
- [Kos13] R. Kosloff, “Quantum Thermodynamics: A Dynamical Viewpoint”, *Entropy* **15**, 2100 (2013).
- [Kos17] R. Kosloff and Y. Rezek, “The Quantum Harmonic Otto Cycle”, *Entropy* **19**, 136 (2017).
- [Kre05] A. Kreuter, C. Becher, G. P. T. Lancaster, A. B. Mundt, C. Russo, H. Häffner, C. Roos, W. Hänsel, F. Schmidt-Kaler, R. Blatt, and M. S. Safronova, “Experimental and theoretical study of the $3d^2D$ -level lifetimes of $^{40}\text{Ca}^+$ ”, *Physical Review A* **71**, 032504 (2005).
- [Lam15] S. K. Lam, A. Pitrou, and S. Seibert, “Numba: A llvm-based python jit compiler”, *Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC* (, 2015), 1–6.
- [Lei03] D. Leibfried, R. Blatt, C. Monroe, and D. Wineland, “Quantum dynamics of single trapped ions”, *Reviews of Modern Physics* **75**, 281 (2003).
- [Lem13] A. Lemmer, A. Bermudez, and M. B. Plenio, “Driven geometric phase gates with trapped ions”, *New Journal of Physics* **15**, 083001 (2013).
- [Lev17] A. Levy, E. Torrontegui, and R. Kosloff, “Action-noise-assisted quantum control”, *Physical Review A* **96**, 033417 (2017).
- [Lev18] A. Levy, A. Kiely, J. G. Muga, R. Kosloff, and E. Torrontegui, “Noise resistant quantum control using dynamical invariants”, *New Journal of Physics* **20**, 025006 (2018).
- [Lin13] Y. Lin, J. P. Gaebler, F. Reiter, T. R. Tan, R. Bowler, A. S. Sørensen, D. Leibfried, and D. J. Wineland, “Dissipative production of a maximally entangled steady state of two quantum bits”, *Nature* **504**, 415–418 (2013).
- [Lin19] D. von Lindenfels, O. Gräß, C. Schmiegelow, V. Kaushal, J. Schulz, M. T. Mitchison, J. Goold, F. Schmidt-Kaler, and U. Poschinger, “Spin Heat Engine Coupled to a Harmonic-Oscillator Flywheel”, *Physical Review Letters* **123**, 080602 (2019).
- [Luc20] I. Luchnikov, S. Vintskevich, D. Grigoriev, and S. Filippov, “Machine Learning Non-Markovian Quantum Dynamics”, *Physical Review Letters* **124**, 140502 (2020).
- [Ma17] Y.-H. Ma, S.-H. Su, and C.-P. Sun, “Quantum thermodynamic cycle with quantum phase transition”, *Physical Review E* **96**, 022143 (2017).

- [Ma18] Y.-H. Ma, D. Xu, H. Dong, and C.-P. Sun, “Optimal operating protocol to achieve efficiency at maximum power of heat engines”, *Physical Review E* **98**, 022133 (2018).
- [Mas19] G. Maslennikov, S. Ding, R. Hablützel, J. Gan, A. Roulet, S. Nimmrichter, J. Dai, V. Scarani, and D. Matsukevich, “Quantum absorption refrigerator with trapped ions”, *Nature Communications* **10**, 202 (2019).
- [Mas21] M. Masuhr, “Observing a quadrupole transition of calcium in an ion trap”, Bachelor thesis, Universität Kassel (2021).
- [Mas93] J. Masoliver and J. M. Porrà, “Harmonic oscillators driven by colored noise: Crossovers, resonances, and spectra”, *Phys. Rev. E* **48**, 4309–4319 (1993).
- [Mat98] M. Matsumoto and T. Nishimura, “Mersenne twister”, *ACM Transactions on Modeling and Computer Simulation* **8**, 3–30 (1998).
- [Mee96] D. M. Meekhof, C. Monroe, B. E. King, W. M. Itano, and D. J. Wineland, “Generation of Nonclassical Motional States of a Trapped Atom”, *Phys. Rev. Lett.* **76**, 1796–1799 (1996).
- [Mén13] V. ç. Méndez, D. Campos, and W. Horsthemke, “Stationary energy probability density of oscillators driven by a random external force”, *Phys. Rev. E* **87**, 062132 (2013).
- [Mic21] K. Micadei, J. P. Peterson, A. M. Souza, R. S. Sarthour, I. S. Oliveira, G. T. Landi, R. M. Serra, and E. Lutz, “Experimental Validation of Fully Quantum Fluctuation Theorems Using Dynamic Bayesian Networks”, *Physical Review Letters* **127**, 180603 (2021).
- [Møl99] K. Mølmer and A. Sørensen, “Multiparticle Entanglement of Hot Trapped Ions”, *Physical Review Letters* **82**, 1835 (1999).
- [Nie19] W. Niedenzu, I. Mazets, G. Kurizki, and F. Jendrzejewski, “Quantized refrigerator for an atomic cloud”, *Quantum* **3**, 155 (2019).
- [Nil88] H. Nilsson, “Submarine Power Systems Using the V4-275R Stirling Engine”, *Proceedings of the Institution of Mechanical Engineers, Part A: Power and Process Engineering* **202**, 257 (1988).
- [Nov58] I. Novikov, “The efficiency of atomic power stations (a review)”, *Journal of Nuclear Energy (1954)* **7**, 125–128 (1958).
- [Obe73] F. Oberhettinger and L. Badii, “Tables of Laplace Transforms”, Springer Berlin Heidelberg (1973).
- [Pau90] W. Paul, “Electromagnetic traps for charged and neutral particles”, *Reviews of Modern Physics* **62**, 531–540 (1990).
- [Pla01] M. Planck, “Ueber das Gesetz der Energieverteilung im Normalspectrum”, *Annalen der Physik* **309**, 553–563 (1901).
- [Ram13] M. Ramm, T. Pruttivarasin, M. Kokish, I. Talukdar, and H. Häffner, “Precision Measurement Method for Branching Fractions of Excited $P_{1/2}$ States Applied to $^{40}\text{Ca}^+$ ”, *Physical Review Letters* **111**, 023004 (2013).

- [Rei15] D. M. Reich, “Efficient Characterisation and Optimal Control of Open Quantum Systems - Mathematical Foundations and Physical Applications”, Universität Kassel (2015).
- [Rez06] Y. Rezek and R. Kosloff, “Irreversible performance of a quantum harmonic heat engine”, *New J. Phys.* **8**, 83 (2006).
- [Roo00] C. F. Roos, “Controlling the quantum state of trapped ions”, Dissertation, Leopold-Franzens-Universität Innsbruck (2000).
- [Roß14] J. Roßnagel, O. Abah, F. Schmidt-Kaler, K. Singer, and E. Lutz, “Nanoscale Heat Engine Beyond the Carnot Limit”, *Physical Review Letters* **112**, 030602 (2014).
- [Roß16a] J. Roßnagel, “A Single-Atom Heat Engine”, Dissertation, Johannes Gutenberg-Universität Mainz (2016).
- [Roß16b] J. Roßnagel, S. T. Dawkins, K. N. Tolazzi, O. Abah, E. Lutz, F. Schmidt-Kaler, and K. Singer, “A single-atom heat engine”, *Science* **352**, 325 (2016).
- [Sch03] F. Schmidt-Kaler, H. Häffner, M. Riebe, S. Gulde, G. P. T. Lancaster, T. Deuschle, C. Becher, C. F. Roos, J. Eschner, and R. Blatt, “Realization of the Cirac–Zoller controlled-NOT quantum gate”, *Nature* **422**, 408 (2003).
- [Sch52] E. Schrödinger, “Are there Quantum Jumps?”, *The British Journal for the Philosophy of Science* **3**, 109–123 (1952).
- [Sin10] K. Singer, U. Poschinger, M. Murphy, P. Ivanov, F. Ziesel, T. Calarco, and F. Schmidt-Kaler, “Colloquium: Trapped ions as quantum bits: Essential numerical tools”, *Reviews of Modern Physics* **82**, 2609 (2010).
- [Sud63] E. C. G. Sudarshan, “Equivalence of Semiclassical and Quantum Mechanical Descriptions of Statistical Light Beams”, *Phys. Rev. Lett.* **10**, 277–279 (1963).
- [Tam18] D. Tamascelli, A. Smirne, S. Huelga, and M. Plenio, “Nonperturbative Treatment of non-Markovian Dynamics of Open Quantum Systems”, *Physical Review Letters* **120**, 030402 (2018).
- [Tor13] E. Torrontegui and R. Kosloff, “Quest for absolute zero in the presence of external noise”, **88**, 032103 (2013).
- [Tor18] E. Torrontegui, S. T. Dawkins, M. Göb, and K. Singer, “Transient non-confining potentials for speeding up a single ion heat pump”, *New Journal of Physics* **20**, 105001 (2018).
- [Tur00] Q. A. Turchette, C. J. Myatt, B. E. King, C. A. Sackett, D. Kielpinski, W. M. Itano, C. Monroe, and D. J. Wineland, “Decoherence and decay of motional quantum states of a trapped atom coupled to engineered reservoirs”, *Physical Review A* **62**, 053807 (2000).
- [Uta21] S. Utagi, S. Banerjee, and R. Srikanth, “On the non-Markovianity of quantum semi-Markov processes”, *Quantum Information Processing* **20** (2021).

- [Uzd16] R. Uzdin, A. Levy, and R. Kosloff, “Quantum Heat Machines Equivalence, Work Extraction beyond Markovianity, and Strong Coupling via Heat Exchangers”, *Entropy* **18**, 124 (2016).
- [Van09] G. Van Rossum and F. L. Drake, “Python 3 Reference Manual”, CreateSpace (2009).
- [Veg17] I. de Vega and D. Alonso, “Dynamics of non-Markovian open quantum systems”, *Reviews of Modern Physics* **89**, 015001 (2017).
- [Wal83] D. F. Walls, “Squeezed states of light”, *Nature* **306**, 141–146 (1983).
- [Wan96] K.-G. Wang and J. Masoliver, “Linear oscillators driven by Gaussian colored noise: Crossovers and probability distributions”, *Physica A-statistical Mechanics and Its Applications - PHYSICA A* **231**, 615–630 (1996).
- [Wu20] K.-D. Wu, Z. Hou, G.-Y. Xiang, C.-F. Li, G.-C. Guo, D. Dong, and F. Nori, “Detecting non-Markovianity via quantified coherence: theory and experiments”, *npj Quantum Information* **6** (2020).
- [Wu98] F. Wu, L. Chen, F. Sun, C. Wu, and Y. Zhu, “Performance and optimization criteria for forward and reverse quantum Stirling cycles”, *Energy Conversion and Management* **39**, 733 (1998).
- [Yin17] Y. Yin, L. Chen, and F. Wu, “Optimal power and efficiency of quantum Stirling heat engines”, *The European Physical Journal Plus* **132**, 45 (2017).
- [Yin18] Y. Yin, L. Chen, and F. Wu, “Performance of quantum Stirling heat engine with numerous copies of extreme relativistic particles confined in 1D potential well”, *Physica A: Statistical Mechanics and its Applications* **503**, 58 (2018).

Acknowledgements

This work would have not been manageable without the support from colleagues, friends and family. To put my gratitude for those people into words, i want to thank them in this section.

I want to thank Prof. Dr. Kilian Singer for enabling this work. I am grateful for the opportunity to already work on publications before finishing my master's degree. I am also thankful for the opportunity to continue my work in the lab after the involuntary break, due to the pandemic.

I like to thank Dr. Daqing Wang for the invaluable guidance he gave during my master's program. In scientific discussion he gave insightful comments during discussions. Thanks to his excellent organizing the collaboration during the lock-downs was endurable.

Bo Deng and I have now been working together for close to three. I highly appreciate our regular discussions. Furthermore, I like to thank Dr. Amikam Levy and Dr. Erik Torrontegui for our collaboration on our publication. In discussion as well as in theoretical derivations they brought a lot of understanding of our system to us.

I like to thank my colleagues Stefan Aull, Manika Bhardwaj, Max Masuhr, Florian Elsen, Ricky-Joe Plate, Jan Thieme, Nick Vogeley and Dr. Peter Zahariev for many interesting and sometimes to heated discussions. Furthermore, I like to thank the administrative staff Andreas Arend, Cornelia Moritz, and Sandra Berner as well, for assisting me in handling various requests and forms.

I like to thank Dr. Daniel Reich and Daniel Basilewitsch, who showed an incredible amount of patience explaining theoretical and numerical principles to me.

However, this work would have not been doable without the support of my family. I want to express my deepest gratitude for the funding my mother, Sonja Range, granted during my studies. I also want to thank my brother Lukas Range for supporting me.