Optimal Control of Electron Transfer

by

Bradley Thomas Friesen

Undergraduate Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of
Hons. Bachelor of Science

in the
Department of Physics
Faculty of Science

SIMON FRASER UNIVERSITY
Spring 2020

Abstract

We investigate a model that allows us to look at electron transfer in the fast-hopping regime. Using recent developments in the study of non-equilibrium processes, we compute optimal protocols which minimize the excess work required to drive the system from one control parameter value to another. Using these protocols, we evolve the system using Fokker-Planck dynamics to calculate how successful these protocols are over a variety of parameter values. We find that in using these protocols there is a trade-off between reducing the dissipation and successfully transferring the electron.

Keywords: Electron Transfer; Optimal Control; Fokker-Planck Dynamics

Dedication

For my parents Don and Karen Friesen.

Acknowledgements

I'd like to thank my supervisor, David Sivak (SFU Physics) for giving me the opportunity to work on this project. I'd like to thank David again, as well as the rest of the Sivak group for all the help and guidance they've offered me throughout this year. I'd like to thank Joseph Lucero in particular for being an absolutely massive help with the Fokker-Planck dynamics.

Table of Contents

List of Figures

Chapter 1

Introduction

Our goal is to explore and develop theory on how to minimize the excess work done when trying to transfer an electron from one surface to another in non-equilibrium conditions. We examine the non-equilibrium thermodynamics of a simple model system that reflects the potential an electron may feel.

This topic may be of interest in a variety of fields. For example, in many fuel cells, electron-transfer efficiency plays an important role. Efficiency is also important when designing solar cells or artificial photosynthesis. It may also have further application in biophysics in explaining how molecular machines achieve very high efficiency, despite operating out of equilibrium.

A theoretical background will be described in Chapter 2 to flesh out the necessary details on optimal control of an out-of-equilibrium stochastic system.

1.1 The Model System

This section will cover the specifics of the model used [1]. We are interested in the transfer of an electron from one surface to another, such as from a metal to a molecule. We want to transfer the electron by ramping the potential voltage of one of the surfaces up or down appropriately in order to make it energetically favourable for the electron to be on one surface or the other. For the model, the two surfaces are described by two offset quadratic traps. $V_0(x, \lambda)$ describes the system when the molecule is uncharged, and $V_1(x)$ describes the system when an electron has been transferred from the metal to the molecule:

$$V_0(x, \lambda) = \frac{1}{2} k_0 x^2 + \lambda \tag{1.1}$$

$$V_1(x) = \frac{1}{2} k_1 (x - x_0)^2, \tag{1.2}$$

where λ is our time-dependent control parameter (the voltage) which we optimize, x_0 is some fixed distance between the surfaces, and the k's are the force constants of the traps.

We will make the approximation that we are in the fast-hopping regime. Then our electron is quickly switching between the distinct potentials $V_0(x, \lambda)$ and $V_1(x)$, and effectively sees a continuous potential of mean force (PMF) [2] defined by

$$E(x, \lambda) = -\frac{1}{\beta} \ln \left(e^{-\beta V_0} + e^{-\beta V_1} \right), \tag{1.3}$$

which is the Helmholtz free energy of the partition function summing over the two distinct potentials. Here, $\beta \equiv (k_B T)^{-1}$ is the inverse temperature. The control parameter λ can take any value, but in the limit of large $|\lambda|$ the PMF approaches one of the distinct potentials (V_0 for negative λ and V_1 for positive λ). k_0 and k_1 can take any positive values, and we set x_0 and β to unity.

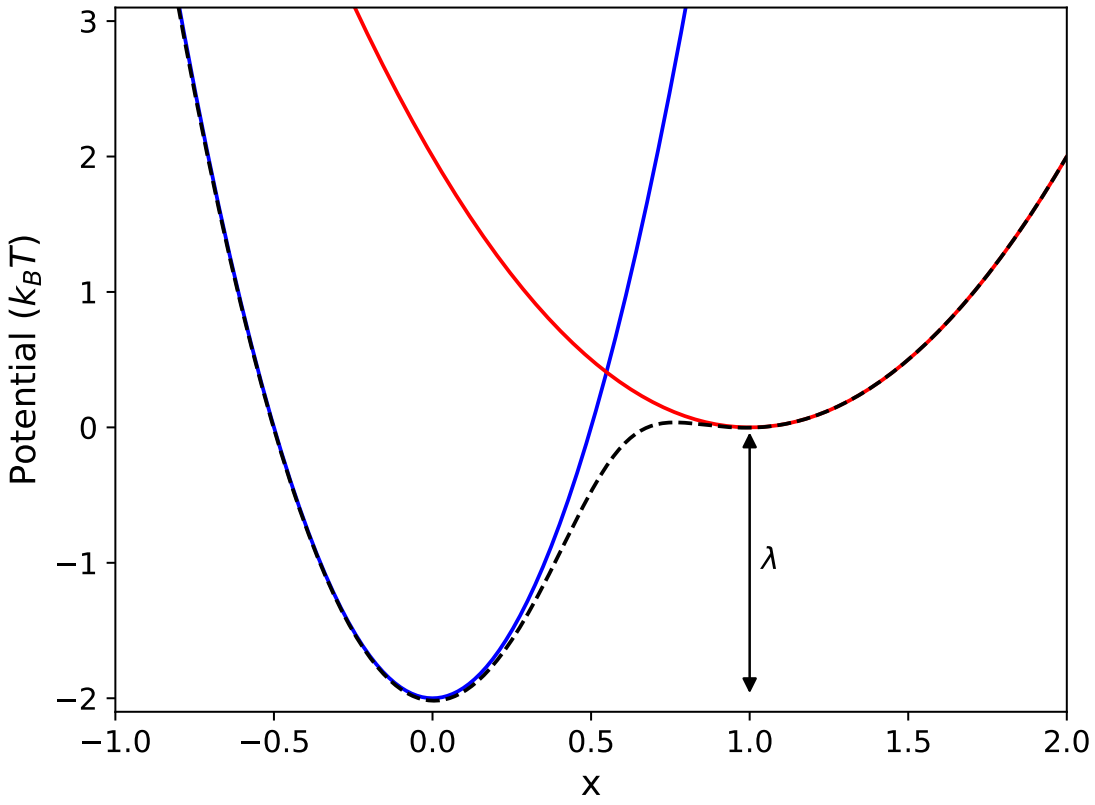

Figure 1.1: PMF $E(x, \lambda)$ (black dashed curve) and distinct potentials $V_0(x, \lambda)$ (blue curve) and $V_1(x)$ (red curve), for $k_0 = 16.0$, $k_1 = 4.0$, and $\lambda = -2.0$.

Chapter 2

Theoretical Background

Fokker-Planck dynamics [3] describe a particle under the influence of drag and random forces, such as in Brownian motion. For any system that obeys Fokker-Planck dynamics, the excess work which is the extra work we have to do because we are operating out of equilibrium, is guaranteed to be non-negative for any path through control-parameter space [4]. We want to find a way to dynamically change our control parameter in order to drive electron transfer. Ideally, we would like to find an optimal path through parameter space that will transfer the electron, while minimizing the energy dissipation of the system. In order to minimize the excess work we need some framework that will approximate the near-equilibrium excess work [5]. We will later compare the optimal path to a naive path that goes linearly from λ_i to λ_f.

2.1 Optimal Paths

This section summarizes results from several papers [4–7]. A physical system at thermal equilibrium with a heat reservoir of temperature T is described by the canonical ensemble

$$\pi(x|\lambda) \equiv \exp \beta[F(\lambda) - E(x, \lambda)], \tag{2.1}$$

where $E(x, \lambda)$ is the system energy (for our purposes, this is just the PMF) as a function of microstate x and control parameter λ. $F(\lambda)$ is the free energy integrated over all the possible microstates of the system,

$$F(\lambda) \equiv -k_{\mathrm{B}}T \ln \int_{-\infty}^{\infty} dx \exp[-\beta E(x, \lambda)] \ . \tag{2.2}$$

For twice-differentiable protocols, applying linear response theory gives the average excess power (above the average equilibrium process power), exerted at time t' by the controller on the system as

$$P_{ex}(t') = \left[\frac{d\lambda^T}{dt}\right]_{t'} \zeta \left[\lambda(t')\right] \left[\frac{d\lambda}{dt}\right]_{t'} \tag{2.3}$$

for the generalized friction tensor

$$\zeta_{ij}\left[\lambda(t')\right] \equiv \beta \int_0^\infty dt'' \langle \delta f_j(0) \delta f_i(t'') \rangle_{\lambda(t')}. \tag{2.4}$$

$\langle \delta f_j(0) \delta f_i(t'') \rangle_{\lambda(t')}$ is the force autocorrelation function, in terms of the equilibrium force fluctuations. The friction coefficient is indicative of the increase in energy cost of driving through parameter space quickly.

For potentials which satisfy $E(x,\lambda) \to \infty$ as $|x| \to \infty$ it is possible [7], and much more convenient, to express the friction tensor in the form

$$\zeta_{ij}(\lambda) = \frac{1}{D} \int_{-\infty}^{\infty} dx \frac{\partial_{\lambda^i} \Pi_{eq}(x,\lambda) \partial_{\lambda^j} \Pi_{eq}(x,\lambda)}{\rho_{eq}(x,\lambda)}. \tag{2.5}$$

Here the components of the generalized friction are written entirely in terms of the equilibrium probability distribution ρ_{eq} and the equilibrium cumulative distribution function Π_{eq} (both of which are analytic for our system), and a diffusion coefficient, which will be set to unity.

Our system only has a single control parameter λ, so the excess power and friction simplify to

$$P_{ex}(t') = \zeta(\lambda(t')) \left(\frac{d\lambda}{dt}\right)^2, \tag{2.6}$$

$$\zeta(\lambda) = \frac{1}{D} \int_{-\infty}^{\infty} \frac{\left[\partial_\lambda \Pi_{eq}(x,\lambda)\right]^2}{\rho_{eq}(x,\lambda)}. \tag{2.7}$$

The average excess work is just the time integral of the average excess power

$$W_{ex} = \int_0^\tau dt P_{ex}(t), \tag{2.8}$$

where τ is the duration of the protocol. We also have a generalized thermodynamic length

$$L \equiv \int_0^\tau dt \sqrt{P_{ex}(t)}. \tag{2.9}$$

We can place a lower bound on the excess work:

$$W_{ex} \geq \frac{L^2}{\tau}. \tag{2.10}$$

For the linear response approximation, by the Cauchy-Schwarz inequality the bound is only achieved by a protocol such that the excess power is constant over the protocol duration. We can solve the Euler-Lagrange equation, where the cost function $f(\lambda(t), \dot{\lambda}) = \zeta(\lambda)\dot{\lambda}^2$ is

the excess power. This gives the solution

$$\dot{\lambda}^{opt}(t) = \frac{(\lambda_f - \lambda_i)\zeta(\lambda(t))^{-1/2}}{\int_0^\tau dt \zeta(\lambda(t))^{-1/2}} \propto \zeta(\lambda(t))^{-1/2} \ . \tag{2.11}$$

Put more intuitively, this shows that we would like to choose a path through parameter space where we are driving slowly when the friction coefficient is large, and quickly when the friction coefficient is small. Using the above expression, we can compute an optimal path through protocol space which should minimize the excess work.

Furthermore, it can be shown that

$$\frac{W_{ex}^{naive}}{W_{ex}^{opt}} = \frac{\overline{\zeta}}{\overline{\zeta^{1/2}}^2}, \tag{2.12}$$

where the overbars represent the averages over the protocol. W_{ex}^{naive} is the excess work done for a naive (linear) protocol, and W_{ex}^{opt} is the excess work for the optimal protocol. This expression for the work ratio is the asymptotic limit we expect our work ratio to reach as we go to sufficiently long protocol duration.

Chapter 3

Results

3.1 Friction Coefficient

The model system is fairly simple and tractable, because of the quadratic form of the potential. Despite this, the friction coefficient in Eq. (2.5) must be solved numerically. The free energy of the system can be written

$$F(\lambda) = k_{\mathrm{B}}T \ln \int_{-\infty}^{\infty} dx e^{-\beta E(x,\lambda)} \tag{3.1}$$

$$= -k_{\mathrm{B}}T \ln \int_{-\infty}^{\infty} dx \left[e^{-\beta V_0(x,\lambda)} + e^{-\beta V_1(x)} \right] , \tag{3.2}$$

which is just the natural logarithm of the sum of two Gaussian integrals. Performing the integration gives

$$F(\lambda) = -k_{\mathrm{B}}T \ln \left[\sqrt{\frac{2\pi}{\beta}} \left(\frac{e^{-\beta\lambda}}{\sqrt{k_0}} + \frac{1}{\sqrt{k_1}} \right) \right]. \tag{3.3}$$

This and the PMF can then be plugged into Eq. (2.1) for the equilibrium probability. Because of the form of the probability distribution, we can also get an analytic expression for the cumulative probability distribution. Skipping the ugly expression for the probability distribution, the cumulative probability distribution is

$$\Pi(x) = \frac{\sqrt{k_0} \left[e^{\beta\lambda} + \mathrm{erf}\left(\sqrt{\frac{\beta k_1}{2}} \right) (x - x_0) \right] + \sqrt{k_1} \left[\mathrm{erf}\left(1 + \sqrt{\frac{\beta k_0}{2}} x \right) \right]}{2\sqrt{k_0} e^{\beta\lambda} + \sqrt{k_1}}. \tag{3.4}$$

We can easily take $\frac{\partial \Pi}{\partial \lambda}$. We now have every component of the integrand for the friction coefficient. The friction coefficient does *not* have an closed-form expression, so we turn to numerical integration.

As may be expected by looking at its constituent parts, the friction coefficient, shown in Fig. 3.1 is somewhat sharply peaked at $\lambda \approx 0$. Intuitively, the force variance is largest when λ is close to zero due to the form of the potential. When λ is small, a change in λ causes a large change in the potential, but as λ gets larger the exponentials cause the potential to

approach the form of one of the distinct potentials, and changes in λ have little effect. In the large and equal k limit, the force variance approaches

$$\langle \delta f^2 \rangle_\lambda = \frac{1}{4} \operatorname{sech}^2 \frac{\lambda}{2} \ . \tag{3.5}$$

For $k_0 = k_1$, the friction peaks at exactly $\lambda = 0$ and is symmetric with respect to λ. For $k_0 \neq k_1$ the friction's peak is shifted to a positive λ when $k_0 < k_1$, and a negative λ for $k_0 > k_1$, and the friction is no longer symmetric about its maximum. In general, the friction peaks at larger values the larger the force constants are. This is also intuitive: as k increases, the force variance reaches some limit, but one would expect the relaxation time to continue increasing as the barrier between the wells continues to increase with k. Additionally, small values of λ do not significantly change the large barrier height, so we expect the relaxation time to be relatively constant near $\lambda = 0$, where our force variance is largest.

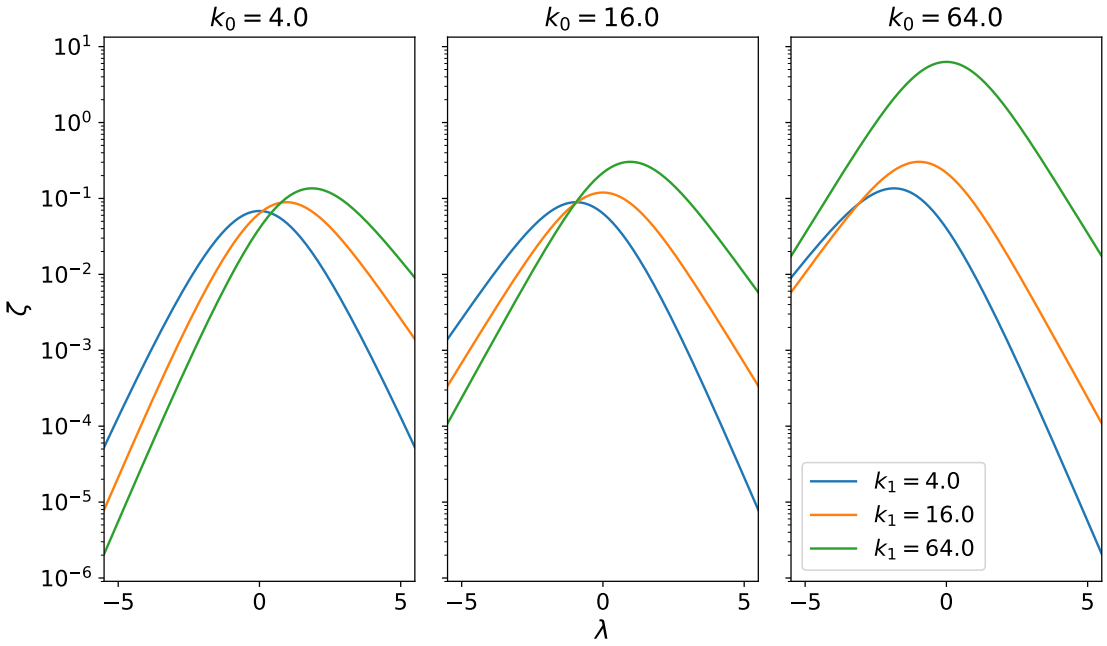

Figure 3.1: The friction coefficient (Eq. (2.5)) as a function of λ for several values of k_0 and k_1.

3.2 Optimal Path

As shown in Sec. 2.1, for the designed protocols the time derivative of the control parameter is proportional to the inverse square root of the friction coefficient. Again, there is no analytic solution to the path as a function of time, or as a function of λ, so we must use numerical

methods. We can rearrange Eq. (2.11) to get

$$t(\lambda) = \frac{\int_{\lambda_i}^{\lambda} d\lambda \sqrt{\zeta(\lambda)}}{\frac{1}{\tau}\int_{\lambda_i}^{\lambda_f} d\lambda \sqrt{\zeta(\lambda)}}, \tag{3.6}$$

which is not a particularly enlightening form, but gives us expected behaviour as we drive quickly where the friction is small, and slowly where the friction is large.

As we expect from looking at our friction coefficient in Fig. 3.1 and considering the velocity's dependence on it (Eq. (2.11)), the designed protocol ramps slowly when λ is close to zero, where the friction is large, and quickly when $|\lambda|$ is large, where the friction is small, as shown in Fig. 3.2. All three paths are quite similar, this is because the velocity depends on the shape of the friction coefficient with respect to the control parameter λ, not necessarily its magnitude. For example, the optimal path is insensitive to multiplying the friction by a constant. This is especially noticeable for the $k = 64$ and $k = 16$ curves.

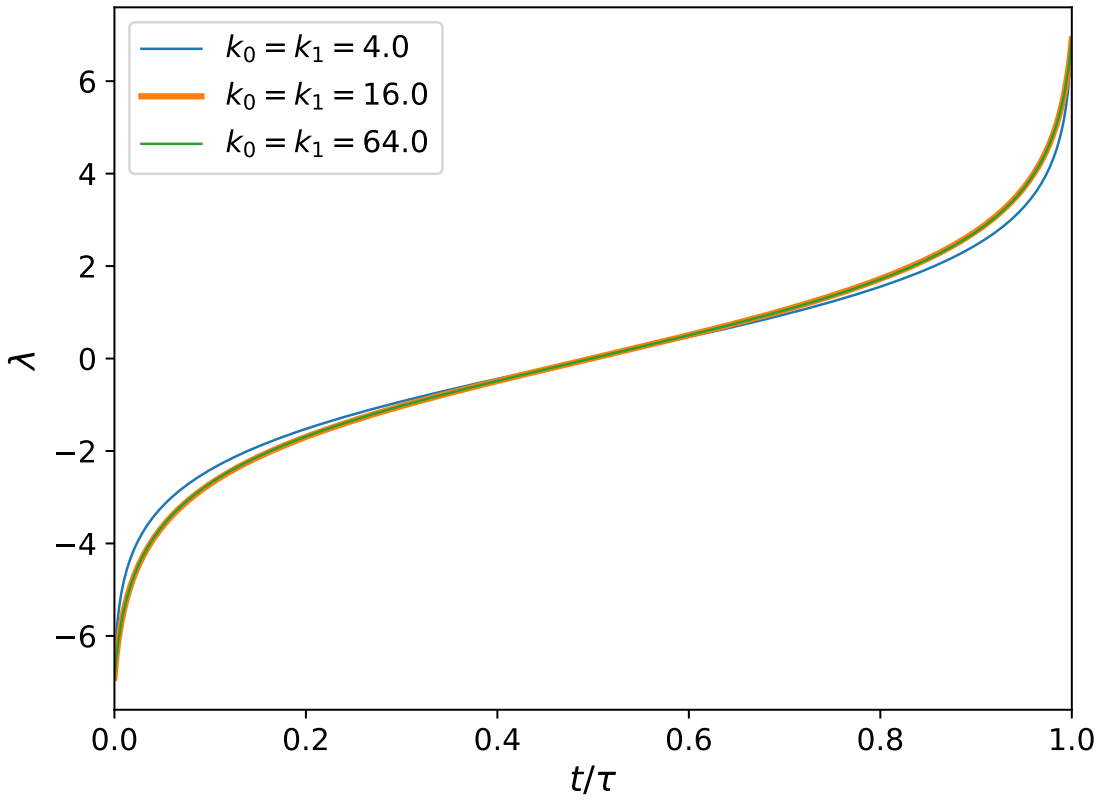

Figure 3.2: Calculated optimal path as a function of time (normalized by the total protocol duration τ), for several pairs of force constants. The $k_0 = k_1 = 16$ curve is thicker to make it more legible.

Compared to a naive (straight-line) path, we want to drive our system quickly at the start and end of our protocol but more slowly in the middle.

3.3 Excess Work

Now that we know what our optimal path looks like, we can calculate the approximate excess work we do over the duration of our protocol, and compare it to the excess work we would do following a naive path. Although our previous expression for the path in Eq. (3.6) gives us the correct answer, we would prefer to have $\lambda(t)$, rather than $t(\lambda)$. We achieve this by switching from an integral to a discrete step,

$$\lambda(t + \Delta t) = \lambda(t) + \frac{\Delta t}{\tau} \frac{\int_{\lambda_i}^{\lambda_f} d\lambda \sqrt{\zeta(\lambda)}}{\sqrt{\zeta(\lambda)}}, \tag{3.7}$$

which depends on our chosen protocol duration and endpoints. For sufficiently small Δt steps, we get the same paths we did previously using Eq. (3.6), and in a much more useful form.

We can now begin calculating the approximate average excess work, as well as run simulations to find the actual expected average excess work. In the limit of long protocol duration, it is expected that the results of the approximation and the simulation will converge. This is because the approximation of the average excess work given in Eq.(3.8) assumes that we are near equilibrium over the course of the entire protocol. No matter how fast our protocol is, we assume that we are still near equilibrium. This obviously holds better if we can go slower. In the simulations, we instead solve the Fokker-Planck equation to find how our probability distribution changes over the protocol, while not necessarily assuming that the system remains near equilibrium.

In general, we expect that for short protocol duration the average excess work will be similar for both naive and optimal paths as there isn't enough difference between the protocols to significantly affect the excess work. In the extremely short-duration limit, both protocols converge to the same path, i.e., an instantaneous jump between the two endpoints λ_i and λ_f. We also expect the approximation results to be higher than the simulated results. The approximation of the excess work scales as τ^{-1}, which diverges as $\tau \to 0$. However, there is an upper bound on the excess work as protocol time goes to zero: the true work can not be any larger than the energy difference between the final and initial states, weighted by the initial equilibrium probability distribution. In the long-duration limit, we expect the optimal protocol to have a reduced average excess work compared to the naive protocol, and that the ratio of the average excess works should approach Eq. (2.12). We also expect that in the long duration limit, the average excess work will reach its lower bound and go as $W_{ex} \propto \frac{1}{\tau}$.

The excess work is just the time integral of the excess power. Then the excess work is

$$\langle W_{ex} \rangle = \int_0^\tau dt\, \zeta(\lambda(t)) \left(\frac{d\lambda}{dt}\right)^2, \tag{3.8}$$

which is approximated numerically by the sum

$$\langle W_{ex} \rangle = \sum_i \delta t\, \zeta(\lambda(t_i)) \left(\frac{d\lambda}{dt}\right)_i^2, \tag{3.9}$$

where $\left(\frac{d\lambda}{dt}\right)_i^2$ indicates the time derivative of the control parameter at a particular time t_i. There is some symmetry in the approximation of the excess work. Conjugate pairs of k, where the force constants are swapped (for example having parameters $k_0 = 4$ and $k_1 = 16$, then $k_0 = 16$ and $k_0 = 4$), will have the same excess work. This is due to our system having no preference (in the near-equilibrium limit) between going forward (driving from the left well to right well) and backwards (driving from the right well to left well). They are the same thing energetically, we just choose to go from left to right. Having $k_0 = a$ and $k_1 = b$ and going from left to right is the same situation as having $k_0 = b$ and $k_1 = a$ and going from right to left.

For the Fokker-Planck simulations, the work is

$$\langle W \rangle_{\lambda_i \to \lambda_{i+1}} = \int_{-\infty}^{\infty} dx\, [E(x, \lambda_{i+1} - E(x, \lambda_i)]\, \rho(x), \tag{3.10}$$

where $\rho(x)$ is the current probability distribution, which is generally not the same as the equilibrium distribution. Again, numerically we compute this as the sum

$$\langle W \rangle_{\lambda_i \to \lambda_{i+1}} = \sum_j [E(x_j, \lambda_{i+1}) - E(x_j, \lambda_i)]\, \rho(x_j). \tag{3.11}$$

To go to the average excess work we sum over all steps in the protocol, minus the free energy, to get

$$\langle W_{ex} \rangle = \langle W \rangle - \Delta F. \tag{3.12}$$

The simulations act as expected in the long- and short-duration regimes. Generally, as the force constant of the traps increases, the excess work is increased. An interesting point is that although we start with our electron in the left well, where it is affected mostly by k_0, the excess work for short durations depends more on k_1, shown by the grouping of the curves in Fig. 3.3. This is likely due to the probability distribution lagging far behind its equilibrium value for a particular λ along the protocol for short protocol durations. Because of the form of the PMF, as λ is increased, the PMF looks more and more like $V_1(x)$, and our electron, which has fallen behind due to the fast protocol, is farther from its equilibrium probability value. As our potential looks more like $V_1(x)$, it looks more like a quadratic, so

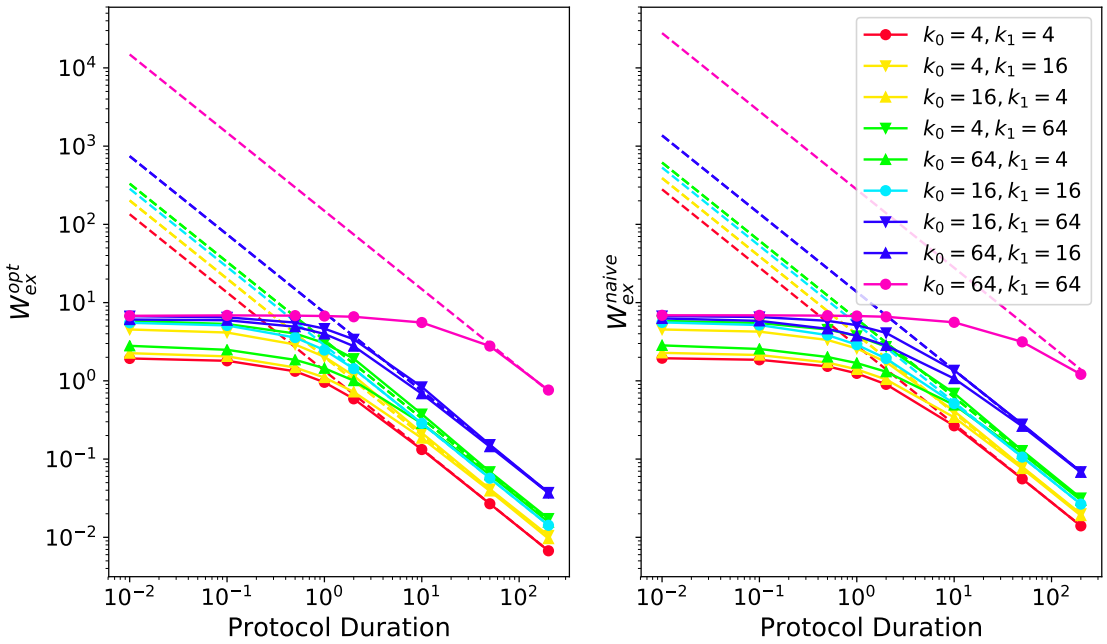

Figure 3.3: Optimal (left) and naive (right) excess works as a function of protocol duration, for the linear-response approximation given in Eq. (3.8), and Eq. (3.11). The dashed lines are the approximate excess work, and the markers are the results of simulation. Circle markers indicate $k_0 = k_1$, up triangle markers indicate $k_0 > k_1$, and down triangle markers indicate $k_0 < k_1$. For conjugate pairs of k (where one pair has $k_0 = a$, $k_1 = b$ and the conjugate pair has $k_0 = b$, $k_1 = a$), the approximate excess works are equal.

when the electron is farther from equilibrium it is pushed on harder. Then for short protocol durations we push our particle the hardest when our PMF looks more like $V_1(x)$. Going from a stronger to a weaker trap strength we approach the approximation of the excess work from below, and going from weaker to stronger, we approach the approximation of the excess work from above. Similar patterns also follow for the naive protocol.

Figure 3.4 shows the ratio of the excess works. For almost all the protocols, the work ratio shows significant improvement at $\tau \approx 1.0$. This aligns reasonably well with one of the time scales in our model. There is the characteristic timescale of the particle bouncing around inside the well, which goes as $\tau_{\text{trap}} \approx \frac{L^2}{D}$, where L is the characteristic length scale $L = \frac{1}{\sqrt{\beta k}}$, and D is the diffusion coefficient. This timescale shrinks as the force constant is increased, which is opposite to what we observe. The other characteristic timescale is roughly for the particle crossing between the traps, and goes as $\tau_{cross} \approx \frac{x_0^2 e^{\beta \Delta E}}{D}$, where x_0 is the separation of the traps, and ΔE is the energy difference between the bottom of the left well and the barrier, which will depend on k. Because the work ratio for each k combination (aside from $k_0 = k_1 = 64$) begins increasing significantly at roughly the same protocol duration, it seems that the timescale that matters the most is τ_{cross}. One complication is

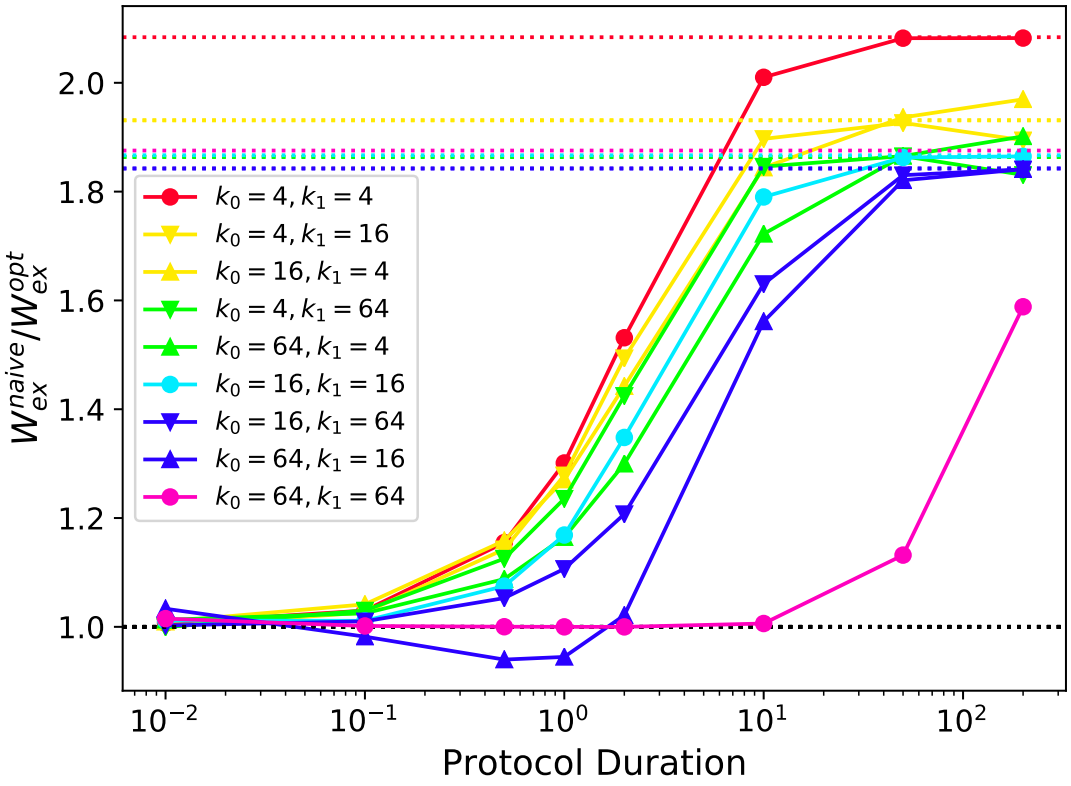

Figure 3.4: Ratio of excess works as a function of protocol duration, for linear-response approximation. The dashed lines are the expected asymptotic excess work, and the markers are the results of simulation. Circle markers indicate $k_0 = k_1$, up triangle markers indicate $k_0 > k_1$, and down triangle markers indicate $k_0 < k_1$. For conjugate pairs of k, the asymptotic excess works are equal.

that the barrier height is changing as we advance the protocol. To get a feel for the numbers, $k_0 = 4$ and $k_1 = 16$ give $\tau_{\text{cross}} \approx 1.3$, $k_0 = 16$ and $k_1 = 4$ give $\tau_{\text{cross}} \approx 1.5$, and $k_0 = 16$ and $k_1 = 16$ give $\tau_{\text{cross}} \approx 3.7$. Though certainly rough, this timescale does follow the correct observable pattern of increased time before significant improvement, as k increases.

For short protocol times ($\tau \lessapprox 1.0$), as expected, the ratio is close to 1. As the protocol time goes to zero, it is expected to go to exactly 1. For $k_0 = 64$ and $k_1 = 16$, the optimal protocol actually does worse for short protocol times. Although not expected, we do not necessarily expect the optimal protocol to always do better: recall that the derivation of the optimal path depends on the linear-response approximation, which does not necessarily hold if we are driving our system too quickly. For long protocol duration, the work ratios for the most part approach their asymptotic limit and go flat. Noticeably, for mixed k values, the last points at $\tau = 200$ diverge to either side of the asymptotic limit. This is due to numerical error in solving the Fokker-Planck equation. If we were to increasing the precision in solving

it, the points would converge. Unfortunately, this is difficult to continue due to increasing computational cost, both in calculating the optimal path on a finer time grid, and in solving the Fokker-Planck equation on a finer time grid and finer spatial grid.

Overall, for nearly any time, smaller k values allow a larger improvement in the amount of excess work done. The protocols with smaller k values approach their asymptotic limit more quickly (at a shorter protocol duration), and that limit is usually larger. In all cases, barring the case when $k_0 = 64$ and $k_1 = 16$, the optimal protocol successfully reduces the excess work done on the system. If the excess work is the primary concern in transferring our electron, the optimal path should be used in nearly all cases.

3.4 Distance from Equilibrium

The Kullback-Leibler divergence, also known as the relative entropy, is a measure of how one probability distribution is different from another reference probability distribution. We can use it as a measure of whether or not our protocol's succeed in getting the electron from the left well to the right well by comparing the final out-of-equilibrium distribution from the simulation to the equilibrium distribution expected for λ_f.

The relative entropy is defined as

$$D_{KL}(P \mid Q) \equiv \sum_{x \in \mathcal{X}} P(x) \ln \frac{P(x)}{Q(x)}, \tag{3.13}$$

where, in our usage, $P(x) = \rho^{neq}_{\lambda_f}(x)$ is the final probability distribution from the Fokker-Planck simulation, $Q(x) = \rho^{eq}_{\lambda_f}(x)$ is the equilibrium probability distribution for the system at the final control-parameter value of λ, and $\mathcal{X}$ is the system state space our distributions go over, which is the same for both of them. For our purposes, a smaller relative entropy is better, as it indicates the two distributions are more similar, and hence the protocol more successfully transfers the electron.

The relative entropy for several simulations are shown in Fig. 3.5, and the actual probability distributions are shown in Fig. 3.6. Again, at short protocol times, the results seem to be most strongly grouped by the k_1 value, rather than by k_0, even though the particle does not have a high probability of reaching the right well. Presumably, this is because k_1 has a dominant effect on the final equilibrium distribution we are comparing to. Similar to the ratio of the excess work, the relative entropy is best (lowest) when the force constants of the wells are lowest. Looking at the results for long protocol time, most of the simulations give well-grouped results, with the exception of simulations where $k_1 = 64$. There are likely two reasons for this. The first, is that it is difficult to get into the well; with such a strong force constant, there is a large energy barrier between the two wells, so even with long protocols it is by no means guaranteed that the electron makes it over the barrier. The second, and more dominant, reason is that with such a strong force constant, the probability

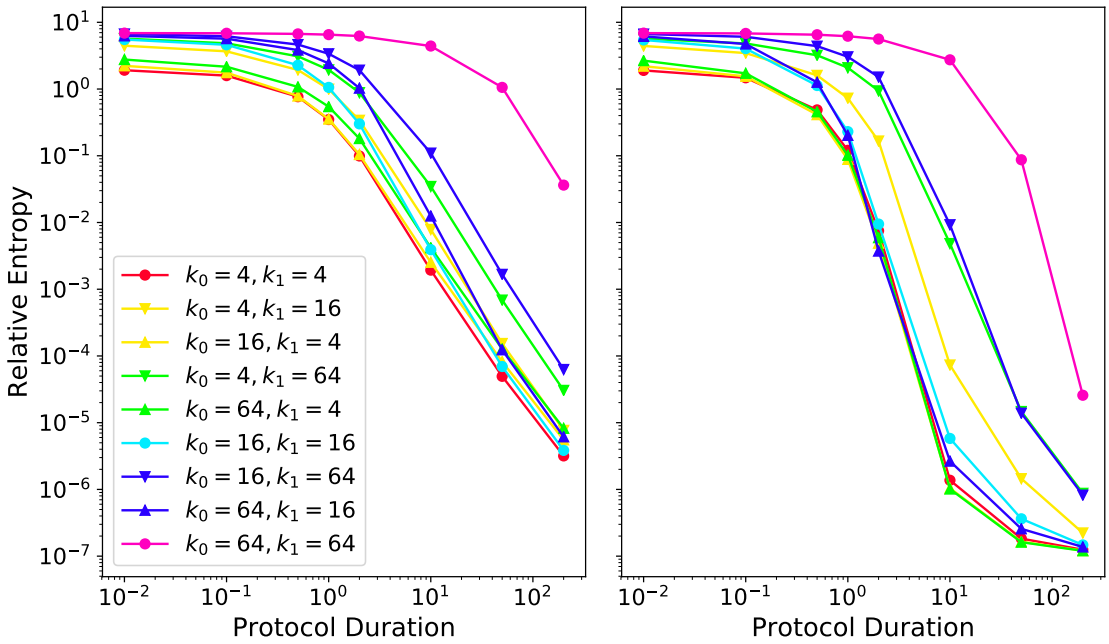

Figure 3.5: Relative entropy at the conclusion of optimal (left) and naive (right) protocols, as a function of protocol duration. Circle markers indicate $k_0 = k_1$, up triangle markers indicate $k_0 > k_1$, and down triangle markers indicate $k_0 < k_1$.

distribution at equilibrium is *very* narrow. Because of how narrow the distribution is, the initial distribution is relatively farther out of equilibrium than for a case with a broader final equilibrium distribution. Because of how the relative entropy is defined, when we have a very narrow final equilibrium distribution, differences from it have larger contributions to the relative entropy than differences from other broader final equilibrium distributions.

Similar to the results for the excess work, the relative entropy begins to decrease strongly for $\tau > 1.0$ for all simulations except $k_0 = k_1 = 64$. This again indicates that the important timescale is τ_{cross}, the timescale for transitions between the two wells. For the naive relative entropies the long-duration protocol points group strongly and start to flatten out. Again, similar to the excess work case, this is due to numerical error when solving the Fokker-Planck equation. The lower bound of the relative entropy is zero, and there is nothing physically preventing the distribution from continuing to relax further toward equilibrium as duration increases.

Overall, relative entropy decreases monotonically with longer protocol duration, and is also generally smaller for smaller k values. In all cases the naive protocol results in a lower relative entropy than the optimal protocol. This is not *necessarily* expected. Intuitively one could imagine the optimal path being more successful in transferring because it drives slowly when the friction is high, and it's possible for thermal fluctuations to help push our particle, without our input. In simulation this does not seem to be the case for this system.

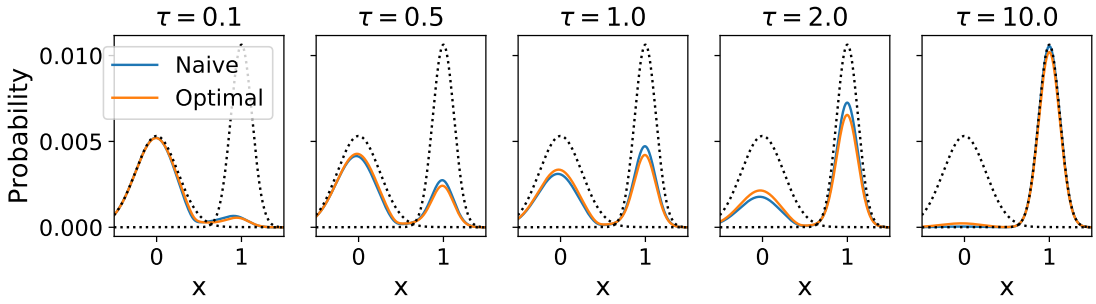

Figure 3.6: Probability mass function at end of protocols for simulations where $k_0 = 16, k_1 = 64$. The left dotted curve is the starting equilibrium distribution. The right dotted curve is the final equilibrium distribution.

Therefore, if the primary concern is having a high probability of transferring the electron, the naive protocol, though not necessarily optimal, does a better job than the optimal one.

Chapter 4

Methods

Most of the programming for this project was done in FORTRAN. The only exception was the code used for the Fokker-Planck equation, which was adapted from existing code written in Cython, which was developed by Joseph Lucero.

4.1 Integration

Because of the simple form of the model, many functions had a closed-form expression. For those that did not, numerical methods were used.

Many of the expressions for quantities we were interested in included integration from negative to positive infinity. If there was no analytic solution, in order to make the problem tractable, we selected an appropriate range over which to integrate. For integration over λ, the canonical ensemble for the system was used to find the probability of being on the potential V_0 or V_1, as appropriate. Taking $V_0(x, \lambda)$ and $V_1(x)$ as the energy states of the system, we can then integrate out x by taking

$$U_\alpha(\lambda) = \frac{-1}{\beta} \ln \int_{-\infty}^{\infty} dx e^{-\beta V_\alpha}, \tag{4.1}$$

where α just labels the potential we are currently working with. Because the individual potentials are simple quadratic wells, these are just Gaussian integrals with analytic solutions. Additionally, V_1 is independent of λ, so this expression will just give a constant. We can then find our probability of being on one potential or the other by taking

$$\rho_\alpha = \frac{e^{-\beta U_\alpha}}{e^{-\beta U_0} + e^{-\beta U_1}}. \tag{4.2}$$

We set a tolerance on the probability of 0.001, and can then solve for the minimum value of λ necessary to ensure, with probability 0.999, that our particle is in the right well, and the maximum value of λ necessary to ensure, with probability 0.999, that our particle is in the left well. The resulting λ range is a function of the force constants. For equal force constants, the start and end points are always $\pm \ln 0.001$. For mixed force constants, the

range is skewed towards the tighter trap. For example, for $k_0 < k_1$ the λ range will shift to more positive values.

We do something similar when trying to find a reasonable x range to integrate over. In this case, we look to our analytic solution for the probability distribution. We are interested in an x range where the probability of finding our particle is greater than some tolerance, and also contains both wells (we don't stop at the low probability at the energy barrier). We can find where the probability distribution gets sufficiently small by using bisection [8]. We bracket an interval (a, b) such that $f(a)$ has a value greater than our desired tolerance, and $f(b)$ has a value less than our desired tolerance. Using bisection we can find, to machine precision, where our probability distribution is equal to our desired tolerance. As long as our initial guess for a and b bracket our desired tolerance, the bisection method can not fail. Using this method we find out how far we must integrate over x before the probability of our particle being there is sufficiently small such that we can call that point infinity. As one may expect, this x range is significantly smaller for traps with higher force constants, where the probability distribution quickly becomes small away from the centres of the wells.

4.2 Simulations

The Fokker-Planck equation, was used to find the non-equilibrium distribution of the system. The Fokker-Planck equation is

$$\frac{\partial}{\partial t}\rho(x,t) = -\frac{\partial}{\partial x}\left[\mu(x,t)\rho(x,t)\right] + \frac{\partial^2}{\partial x^2}\left[D(x,t)\rho(x,t)\right], \tag{4.3}$$

where ρ is the probability distribution, $\mu \equiv -\frac{\partial E(x,\lambda)}{\partial x}$ is the convection or drift, and D is the diffusion constant, which is (the same diffusion coefficient as is used Eq.(2.5) for the friction). The Fokker-Planck equation is a partial differential equation that describes the time evolution of the probability density function ρ of the velocity of a particle under the influence of drag, random, and deterministic forces. It can also be applied to other observables, in our case position. When applied to the probability density function of the position of a particle it is often called the Smoluchowski equation.

Simulations are done by starting with the steady-state distribution. At each timestep we update λ and the PMF, and then integrate the discrete Fokker-Planck equation timestep until the protocol is over, and the final probability distribution is produced. The work is computed at each timestep using Eq. (3.11).

We solved the Fokker-Planck equation using a Forward-Time Central-Space (FTCS) algorithm, which is an explicit method of solving partial differential equations [8]. Explicit methods use information about the system's current state to calculate its future state. This is contrary to implicit methods, which are more stable but more complex, computing the future state of the system through solving a system of equations with chosen boundary

conditions and a guess of the solution. In order to apply the FTCS method, we must discretize the derivatives in Eq. (4.3). The FTCS scheme is first order in time, and second order in space. Then the obvious way to discretize the derivatives is to set

$$\frac{\partial}{\partial t}\rho(x,t) \approx \frac{\rho(x,t+\Delta t)-\rho(x,t)}{\Delta t}, \tag{4.4a}$$

$$\frac{\partial}{\partial x}[\mu(x,t)\rho(x,t)] \approx \frac{\mu(x+\Delta x,t)\rho(x+\Delta x,t)-\mu(x,t)\rho(x,t)}{2\Delta x}, \tag{4.4b}$$

$$\frac{\partial^2}{\partial x^2}[D(x,t)\rho(x,t)] \approx \frac{D(x+\Delta x,t)\rho(x+\Delta x,t)-2D(x,t)\rho(x,t)+D(x-\Delta x,t)\rho(x-\Delta x,t)}{(\Delta x)^2}, \tag{4.4c}$$

where Δt and Δx are the temporal and spatial spacing of the grid we are solving on. Plugging Eq. (4.4) in to Eq. (4.3), and solving for $\rho(x,t+\Delta t)$ gives the complete update as

$$\begin{aligned}\rho(x,t+\Delta t) =& \rho(x,t)\\ &-\Delta t\left[\frac{\mu(x+\Delta x,t)\rho(x+\Delta x,t)-\mu(x-\Delta x,t)\rho(x-\Delta x,t)}{2\Delta x}\right]\\ &+\Delta t\left[\frac{D(x+\Delta x,t)\rho(x+\Delta x,t)-2D(x,t)\rho(x,t)+D(x-\Delta x,t)\rho(x-\Delta x,t)}{(\Delta x)^2}\right].\end{aligned} \tag{4.5}$$

4.3 Boundary Conditions

We want no flux at the boundaries of the simulated domain. Therefore, we wish to impose von Neumann boundary conditions by setting the derivative of the probability flux to zero at the boundaries, thereby containing all of our probability in the box. This is surprisingly nontrivial. Using left- or right-handed finite difference schemes was not sufficient to keep the probability density from escaping. Instead, a ghost point method was used. The ghost point is calculated as if there were an additional point, outside of our grid. We can then solve for the conditions on this outside point in terms of points inside the grid in order to produce the desired outcome. We then plug these conditions in at the boundaries, where our space centered derivatives require knowledge of points outside our grid. By definition, this should be achievable by requiring that the fluxes of the convection (first term on RHS in Eq. (4.3)) be zero at the ends of the box, such that

$$\mu_{-1}\rho_{-1} = \mu_0\rho_0, \tag{4.6}$$

where the subscript indicates the ith point on the space grid, and -1 indicates a point outside the grid. In practice, this is not sufficient: the probability leaks out of the simulation box over time.

One may also consider discretizing the probability flux and setting it to zero, where the flux is obtained by solving

$$\frac{\partial \rho}{\partial t} = -\frac{\partial J}{\partial x}, \tag{4.7}$$

where J is the flux. This gives us

$$J = \mu\rho - \frac{\partial}{\partial x}(D\rho). \tag{4.8}$$

Discretizing this and setting it to zero at the boundary, we get

$$0 = \mu\rho_0 - D\left(\frac{\rho_0 - \rho_{-1}}{\Delta x}\right), \tag{4.9a}$$

$$\rho_{-1} = \rho_0\left(\frac{\mu\Delta x}{D} - 1\right), \tag{4.9b}$$

for the left boundary, which is still not good enough. The probability is still able to leak out of the simulation box. However, if we apply both conditions (Eq. (4.6) and Eq. (4.9)) together, we get

$$\begin{aligned}\rho_0(t+\Delta t) =& \rho_0(t) \\ &-\Delta t\left[\frac{\mu_1(t)\rho_1(t) - \mu_0(t)\rho_0(t)}{2\Delta x}\right] \\ &+\Delta t\left[\frac{D_0(t)\rho_0(t)\left(-\frac{1+\Delta x\mu_0(t)}{D_0(t)}\right) + D_0(t)\rho_1(t)}{(\Delta x)^2}\right],\end{aligned} \tag{4.10}$$

for the left boundary, and

$$\begin{aligned}\rho_{N-1}(t+\Delta t) =& \rho_{N-1}(t) \\ &-\Delta t\left[\frac{\mu_{N-1}(t)\rho_{N-1}(t) - \mu_{N-1}(t)\rho_{N-1}(t)}{2\Delta x}\right] \\ &+\Delta t\left[\frac{D_{N-1}(t)\rho_{N-1}(t)\left(\frac{1+\Delta x\mu_{N-1}(t)}{D_{N-1}(t)}\right) + D_{N-1}(t)\rho_{N-2}(t)}{(\Delta x)^2}\right],\end{aligned} \tag{4.11}$$

for the right boundary, where $N-1$ is the last point in the grid, as is the convention in Cython, which succeed in setting the probability flux to zero at the boundaries. Thus, it seems that we need two separate boundary conditions, one for the convection term, and one for the diffusion term. Using this form produces the same equilibrium results as the analytic form of the probability density.

Chapter 5

Conclusions

This research explored how to transfer an electron or other particle from one surface to another out of equilibrium while minimizing the amount of excess work done. To this end, a simple model was developed, using two quadratic potential wells to simulate the two surfaces. We made the assumption that we were operating in the fast-hopping regime, where we could treat the two distinct potentials as one continuous one. We calculated the general friction tensor, and used it to calculate an optimal path through parameter space that should require the least excess work. We then simulated the evolution of the probability distribution using the Fokker-Planck equation. From these simulations we were able to calculate the excess work, and we were able to calculate the relative entropy by comparing the final non-equilibrium distribution to the expected equilibrium distribution.

Our results suggest that there is a trade-off between optimizing for reduced excess work, and how fast one is able to have a high chance of successfully transferring the electron. In nearly all cases, excluding some for short protocol times, the calculated optimal protocol required less excess work than the naive straight line protocol. This can be seen most clearly in Fig. 3.4, where the ratios of the excess works for naive and optimal protocols are shown. As may be intuitive, more loosely bound electrons (represented by smaller force constants) allow for greater excess work savings, and we can reach the asymptotic limit of these savings in a shorter protocol duration. Conversely, the distance from equilibrium at the conclusion of the protocol (quantified by the relative entropy, shown in Fig. 3.5) was higher for the optimal protocol compared to the naive protocol for equal duration. This indicates that the final non-equilibrium probability density of the optimal protocol was not as similar to the desired equilibrium probability density as was its naive counterpart. Using the optimal protocol generally results in excess work savings, but has a reduced chance of successfully transferring the electron in a given protocol time.

5.1 Future Work

For future works continuing this research we would like to improve the precision of our simulations. This would be best done by changing our method of solving the Fokker-Planck equation, as well as how we input protocols. At the time of writing, the protocol was calculated and written to file so that they would be reusable. For increased precision, it is necessary to reduce the time step between points on the protocol. Gaining high precision means very small time steps (which results in very large files) and long run times. Moving forward, it may be beneficial to compute the path one step at a time, as the simulation runs, to keep the memory cost manageable. Additionally, we could move to more sophisticated methods, such as implicit methods. Although the FTCS algorithm works, it is not particularly stable. Implicit methods would allow better stability, putting less constraints on the timestep size we use when integrating.

Aside from programming improvements, we would also like to explore other aspects of the system. In this instance we operated in the fast-hopping regime, where we smoothed our two distinct potentials into one continuous one. Alternatively, we could leave the potentials as distinct, and describe the transfer of electrons through a hopping probability such as $\gamma_{0\to1} = \frac{\Gamma\Delta t}{\hbar} f(E(x,t))$ for transfer from the left well to the right well, and $\gamma_{1\to0} = \frac{\Gamma\Delta t}{\hbar}(1 - f(E(x,t)))$ for transfer from the right well to the left well, where $f(E(x,t))$ is the Fermi function, $E(x,t)$ is the energy difference between the distinct potentials, and Γ is a constant which describes the electron coupling, which is independent of the energy gap.

Bibliography

[1] G. Medders, D. A. Sivak, and J. Subotnik, "Energetically efficient driving of electron transfer at molecule-metal surfaces", Unpublished manuscript, 2019.

[2] D. Chandler, *Introduction to modern statistical mechanics* (Oxford University Press, Oxford, 1987).

[3] H. Risken, *The fokker-planck equation*, 3rd (Springer, 1996).

[4] P. R. Zulkowski, D. A. Sivak, G. E. Crooks, and M. R. DeWeese, "Geometry of thermodynamic control", Phys. Rev. E **86**, 041148 (2012).

[5] D. A. Sivak and G. E. Crooks, "Thermodynamic metrics and optimal paths", Phys. Rev. Lett. **108**, 190602 (2012).

[6] A. I. Brown and D. A. Sivak, "Toward the design principles of molecular machines", Physics in Canada **73**, 61–66 (2017).

[7] P. R. Zulkowski and M. R. DeWeese, "Optimal control of overdamped systems", Phys. Rev. E **92**, 032117–9 (2015).

[8] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical recipes in fortran 77*, 2nd (Cambridge University Press, 1992).