Summer 2021

The Exact Factorization Equations for One- and Two-Level Systems

Bart Rosenzweig
CUNY Hunter College

The Exact Factorization Equations for One- and Two-Level Systems

Bart Rosenzweig

Submitted in partial fulfillment of the requirements for the degree of

Master of Arts in Applied Mathematics

Hunter College of the the City University of New York

2021

August 5, 2021
Date

Vincent Martinez
Thesis Sponsor

August 5, 2021
Date

Emmanuel Asante-Asamani
Second Reader

August 5, 2021
Date

Neepa Maitra
Third Advisor

Department of Mathematics and Statistics

Hunter College of the City University of New York

New York, New York

Acknowledgements

First and foremost, I would like to thank my advisors Vincent Martinez, Emmanuel Asante-Asamani, and Neepa Maitra for their guidance, support, and enthusiasm throughout the completion of this work, and for their mentorship throughout my time at Hunter. I am grateful to have been afforded the opportunity to work on such a fascinating, multifaceted problem, and look forward to our continuing collaboration.

I also must thank everyone whose paths I crossed as part of Neepa's research group, first at Hunter and now at Rutgers. Lionel, Norah, Phillip, Patricia, Davood, Evaristo, Steven, Tiffany: thank you for your encouragement, and thanks for all the laughs. Additionally, I am forever indebted to my phenomenal advisor Charlotte Glasser, as well as to Jonathan Rosenberg of the History Department, for our many compelling, thought-provoking conversations.

A last thank you to all my wonderful friends (you know who you are), and to Mom, Dad, and Milhouse, for your unconditional love and support, without which I would certainly not be where I am today.

Contents

List of Figures

Introduction

In this work we study the quantum dynamics for a system of n electrons and N nuclei whose full wavefunction has been decomposed into a *conditional* (electronic) and *marginal* (nuclear) probability amplitude, following the *exact factorization approach*.

Section 1 describes the molecular problem in some generality, introduces a historically important approach to this problem called the *Born-Oppenheimer approximation*, and describes a novel alternative method developed by Abedi, Maitra, and Gross, namely the *exact factorization of the molecular wavefunction*, and lays out the central results underpinning this method, namely the existence and uniqueness (up to a gauge transformation) of this factorization, and derives the differential equations governing the dynamics of these factors. The conditional factor is expanded in the basis of Born-Oppenheimer eigenfunctions, giving rise to the *local conditional equation*.

Section 2 spells out in some detail the derivations of the local conditional equation and the marginal equation under the assumption that the conditional (electronic) subsystem can be expressed as a linear combination of two Born-Oppenheimer eigenfunctions, taken to be the first two. A simple structure is observed in the two local conditional equations, namely the separate satisfaction of the important so-called *Partial Normalization Condition* by nine pairs of terms in the equations, to be used in developing approximations to the full system.

Section 3 describes the application of Newton's method to solving Schrödinger-like equations, looking ahead to future work in which the full two-level local conditional equations are simulated, with particular attention paid to a portion of that system we term the *dispersive part*. Python implementations using the NumPy library are given in Section A.

1 The Molecular Problem

1.1 The Born-Oppenheimer Approximation

In quantum mechanics, one wants to solve the *time-dependent Schrödinger equation*

$$i\hbar\partial_t\Psi = \hat{H}\Psi \tag{1}$$

for the *wavefunction* $\Psi : \mathbb{R}^d \times \mathbb{R}^{\geq 0} \to \mathbb{C}$, where $d \in \mathbb{N}$ is the number of spatial degrees of freedom, $\hbar$ is the 'reduced Planck constant' (equal to 1 in the system of Hartree atomic units) and $\hat{H}$ is the *Hamiltonian* operator, which varies from system to system. $|\Psi|^2$ is interpreted as giving the probability of the system under investigation being found in configuration $\vec{X} \in \mathbb{R}^d$ at time t, so we impose the following important normalization constraint

$$\int_{\mathbb{R}^d} |\Psi|^2 = 1$$

Generally, as will be the case for the remainder of this work, $\hat{H}$ takes on the form

$$\hat{H} = \hat{T} + V(\vec{x}_1, ..., \vec{x}_n) = \sum_{i=1}^{d} \frac{-\hbar^2\Delta_i}{2M_i} + V(\vec{x}_1, ..., \vec{x}_n)$$

where $\{\vec{x}_i\}$ are the spatial coordinates and $\{M_i\}$ are the associated masses. $\hat{T}$ is the *kinetic energy* operator and V is the *potential energy*. Here on out, we will concern ourselves with a central problem in chemistry: how to solve equation (1) for a system of n electrons and N nuclei which interact with one another. We conventionally denote by $\{\vec{r}_i\}_{i=1}^n$ the positions of the n electrons and $\{\vec{R}_I\}_{I=1}^N$ the positions of the N nuclei[1], and we note that while the nuclei will in general have different masses M_I and charges $Z_I > 0$, the electrons all have the electron mass m_e and (minus) the elementary charge e which in atomic units are both set to unity: $m_e = e = 1$. In the absence of external forces, the form for this Hamiltonian in

[1]Frequently we will abuse notation and drop the set brackets, letting $\vec{r}$ refer to all of the electronic coordinates $\{x_1, y_1, z_1, ..., z_n\}$ and $\vec{R}$ refer to the nuclear coordinates $\{X_1, Y_1, Z_1, ..., Z_N\}$.

atomic units (hereon referred to as the *molecular Hamiltonian* $\hat{H}_{\text{mol}}$) is then

$$\begin{aligned}\hat{H}_{\text{mol}} &:= \hat{T}_n + \hat{T}_e + \hat{V}_{nn} + \hat{V}_{ee} + \hat{V}_{en} \\ &= -\sum_{I=1}^{N} \frac{\Delta_I}{2M_I} - \sum_{i=1}^{n} \frac{\Delta_i}{2} + \sum_{I \neq J} \sum_{J=1}^{N} \frac{Z_I Z_J}{|\vec{R}_I - \vec{R}_J|} + \sum_{i \neq j} \sum_{j=1}^{n} \frac{1}{|\vec{r}_i - \vec{r}_j|} - \sum_{I=1}^{N} \sum_{i=1}^{n} \frac{Z_I}{|\vec{R}_I - \vec{r}_i|}\end{aligned}$$

In reality, each coordinate $\vec{r}_i$ and $\vec{R}_I$ has 3 components, so in particular $\Delta_i = \partial^2_{x_i} + \partial^2_{y_i} + \partial^2_{z_i}$ and so on, but in some of the simplified models considered later in the work, we consider idealized toy models of molecules in which either the nuclear or electronic degrees of freedom are assumed to be 1- or 2-dimensional.

One approach to solving this problem approximately, known as the *Born-Oppenheimer approximation*, exploits the smallness of the ratio of the electronic and nuclear masses, and will be an important reference point throughout. The approximation proceeds as follows. First one solves, for fixed nuclear configuration $\vec{R}_I$, the eigenvalue problem for the *Born-Oppenheimer Hamiltonian* $\hat{H}_{\text{BO}} := \hat{H}_{\text{mol}} - \hat{T}_n$

$$\hat{H}_{\text{BO}} \Phi^j_{\text{BO}} = \epsilon^j_{\text{BO}}(\vec{R}) \Phi^j_{\text{BO}}(\vec{r}; \vec{R}) \tag{2}$$

where we note that since $\hat{H}_{\text{BO}}$ does not depend on time, neither do the *Born-Oppenheimer wavefunctions* Φ^j_{BO}; and the *Born-Oppenheimer potential energy surfaces* ϵ^j_{BO} are assumed to depend only on $\vec{R}$, while Φ^j_{BO} are assumed to depend on $\vec{r}$, and on $\vec{R}$ parametrically. The wavefunctions Φ^j_{BO} furnish a complete (orthonormal) basis for the space of electronic wavefunctions, and we therefore make the ansatz

$$\Psi = \sum_{j=1}^{\infty} C_j(\vec{R}, t) \Phi^i_{\text{BO}}(\vec{r}; \vec{R}) \tag{3}$$

which we can in principle substitute into the full Schrödinger equation to obtain the following

equations for the coefficients $C_j(\vec{R}, t)$:

$$i\partial_t C_j = \Big[\sum_{I=1}^{N} \frac{(-i\nabla_I + i\int \Phi^*_{\mathrm{BO}^j} \nabla_I \Phi^j_{\mathrm{BO}} d\vec{r})^2}{2M_I} + \epsilon^j_{\mathrm{BO}}\Big] C_j$$
$$- \sum_{k\neq j} \sum_I \frac{1}{2M_I} \Big[\int \Phi^*_{\mathrm{BO}^j} \nabla_I \Phi^j_{\mathrm{BO}} d\vec{r} + 2\int \Phi^*_{\mathrm{BO}^j} \nabla_I \Phi^j_{\mathrm{BO}} d\vec{r}\Big] C_k \tag{4}$$

This procedure is sometimes called the *Born-Huang expansion*. In practice, the Born-Oppenheimer approximation instead poses the ansatz

$$\Psi \approx C_j(\vec{R}, t)\Phi^j_{\mathrm{BO}}(\vec{r}; \vec{R}) \tag{5}$$

for some j. This relies on the smallness of the ratio of the electronic and nuclear masses: informally, the idea is that if the electronic subsystem is initially prepared in an eigenstate of the Born-Oppenheimer Hamiltonian, it will *adiabiatically* (i.e. 'smoothly') adjust to any movement on the part of the nuclei over a time-scale smaller than that of the nuclei.

1.2 Exact Factorization

As discussed in Tully (2000), the Born-Oppenheimer approximation affords computational chemists a helpful and intuitive way to visualize chemical reactions in the form of the Born-Oppenheimer potential energy surface ϵ^j_{BO}. One might therefore ask: is there a way to have a picture like this, in which the solution Ψ to (1) is factored as in (5), and to obtain a single potential energy surface ϵ which contributes to the dynamics in a straightforward way as in (4), but without making any approximations? Such a decomposition is the goal of the *exact factorization approach* to the molecular Schrödinger equation developed by A. Abedi, N. Maitra, and E.K.U. Gross in [1] and [2], which we now describe.

1.2.1 The Equations

Theorem 1. *The solution* Ψ *to* (1) *can be written as a product of the form*

$$\Psi = \Phi_{\vec{R}}(\vec{r}, t)\chi(\vec{R}, t) \tag{6}$$

where $\Phi_{\vec{R}}(\vec{r}, t)$ *is called the* conditional wavefunction *and* $\chi(\vec{R}, t)$ *the* marginal wavefunction*, and where the conditional wavefunction satisfies the* Partial Normalization Condition

$$\int |\Phi_{\vec{R}}(\vec{r}, t)|^2 d\vec{r} = 1 \tag{7}$$

Moreover, in the absence of external potentials, these wavefunctions satisfy the following equations

$$i\partial_t \Phi_{\vec{R}} = \big(\hat{H}_{BO} + U_{en}[\Phi_{\vec{R}}, \chi] - \epsilon[\Phi_{\vec{R}}, \chi]\big)\Phi_{\vec{R}} \tag{8}$$

$$i\partial_t \chi = \sum_{I=1}^{N} \Big[\frac{(-i\nabla_I + A_I[\Phi_{\vec{R}}])^2}{2M_I} + \epsilon[\Phi_{\vec{R}}, \chi]\Big]\chi \tag{9}$$

where the terms A_I*,* U_{en} *and* ϵ *— respectively referred to as the* time-dependent vector potential*, the* electron-nuclear coupling operator*, and the* time-dependent potential energy surface *— are given by*

$$A_I[\Phi_{\vec{R}}] := -i \int \Phi^*_{\vec{R}} \nabla_I \Phi_{\vec{R}} d\vec{r} \tag{10}$$

$$U_{en}[\Phi_{\vec{R}}, \chi] := \sum_{I=1}^{N} \frac{1}{M_I} \Big[\frac{(-i\nabla_I - A_I)^2}{2} + \Big(\frac{-i\nabla_I \chi}{\chi} + A_I\Big) \cdot (-i\nabla_I - A_I)\Big] \tag{11}$$

$$\epsilon[\Phi_{\vec{R}}, \chi] := \int \Phi^*_{\vec{R}} (\hat{H}_{BO} + U_{en} - i\partial_t)\Phi_{\vec{R}} d\vec{r} \tag{12}$$

Proof. Following the treatment in Abedi, et al., one can show existence of such a factorization given a normalized wavefunction Ψ by merely setting $\chi(\vec{R}, t) \equiv e^{iS(\vec{R},t)}\sqrt{\int |\Psi(\vec{r}, \vec{R}, t)|^2 d\vec{r}}$ and $\Phi_{\vec{R}}(\vec{r}, t) \equiv \frac{\Psi(\vec{r},\vec{R},t)}{\chi(\vec{R},t)}$ for some arbitrary $S(\vec{R}, t)$. Then certainly $\int |\Phi_{\vec{R}}|^2 d\vec{r} = \frac{1}{|\chi|^2} \int |\Psi|^2 d\vec{r} =$

1. The equations of motion are obtained by simply substituting the ansatz (6) into the Schrödinger equation (1) and using the partial normalization condition (7). We have

$$i\chi\partial_t\Phi_{\vec{R}} + i\Phi_{\vec{R}}\partial_t\chi = \chi\hat{H}_{\mathrm{BO}}\Phi_{\vec{R}} - \sum_I \frac{1}{2M_I}\Big(\chi\Delta_I\Phi_{\vec{R}} + 2\nabla_I\chi\cdot\nabla_I\Phi_{\vec{R}} + \Phi_{\vec{R}}\Delta_I\chi\Big)$$

which gives, upon multiplying by $\Phi^*_{\vec{R}}$ and integrating over $\vec{r}$:

$$i\partial_t\chi = \Big[\int \Phi^*_{\vec{R}}\Big(\hat{H}_{\mathrm{BO}} - i\partial_t - \sum_I \frac{1}{2M_I}\Delta_I\Big)\Phi_{\vec{R}}d\vec{r} - \sum_I \frac{i}{M_I}A_I\cdot\nabla_I - \sum_I \frac{\Delta_I}{2M}\Big]\chi$$

Now notice that

$$\begin{aligned}\sum_I \frac{(-i\nabla_I + A_I)^2}{2M_I} + \int \Phi^*_{\vec{R}}U_{\mathrm{en}}\Phi_{\vec{R}}d\vec{r} =& \sum_I \frac{1}{2M_I}\Big[-\Delta_I - i\nabla_I\cdot A_I - iA_I\cdot\nabla_I + A_I\cdot A_I\Big] \\ &+ \sum_I \frac{1}{M_I}\Big[\frac{1}{2}\int \Phi^*_{\vec{R}}\Big(-\Delta_I + i\nabla_I\cdot A_I + iA_I\cdot\nabla_I \\ &+ A_I\cdot A_I + \int \Phi^*_{\vec{R}}\Big(-\frac{\nabla_I\chi}{\chi}\cdot\nabla_I - iA_I\cdot\nabla_I \\ &+ i\frac{\nabla_I\chi}{\chi}\cdot A_I - A_I\cdot A_I\Big)\Phi_{\vec{R}}d\vec{r}\Big] \\ =& \sum_I \frac{1}{2M_I}\Big(-\Delta_I - 2A_I\cdot\nabla_I - \int \Phi^*_{\vec{R}}\Delta_I\Phi_{\vec{R}}d\vec{r}\Big)\end{aligned}$$

since we have the following cancellations:

$$\begin{aligned}\int \Phi^*_{\vec{R}}\Big(-\frac{\nabla_I\chi}{\chi}\cdot\nabla_I + i\frac{\nabla_I\chi}{\chi}\cdot A_I\Big)\Phi_{\vec{R}}d\vec{r} =& -\frac{\nabla_I\chi}{\chi}\cdot\int \Phi^*_{\vec{R}}\nabla_I\Phi_{\vec{R}}d\vec{r} \\ &+ \Big(\int |\Phi_{\vec{R}}|^2 d\vec{r}\Big)\frac{\nabla_I\chi}{\chi}\cdot\int \Phi^*_{\vec{R}}\nabla_I\Phi_{\vec{R}}d\vec{r} \\ =& 0\end{aligned}$$

and the remaining eight terms (i.e. all terms besides the two which do not cancel and the

two whose cancellation we just demonstrated) reduce to

$$\begin{aligned}
&\frac{1}{2}\Big[-i\nabla_I \cdot A_I - iA_I \cdot \nabla_I + A_I \cdot A_I + i\int \Phi^*_{\vec{R}}(\nabla_I \cdot A_I)\Phi_{\vec{R}} d\vec{r} \\
&- i\int \Phi^*_{\vec{R}}(A_I \cdot \nabla_I)\Phi_{\vec{R}} d\vec{r} - \int \Phi^*_{\vec{R}}(A_I \cdot A_I)\Phi_{\vec{R}} d\vec{r}\Big] \\
&= \frac{1}{2}\Big[-i(\nabla_I \cdot A_I) - i(A_I \cdot \nabla_I) + i(\nabla_I \cdot A_I) + (A_I \cdot A_I) - i(A_I \cdot \nabla_I) - (A_I \cdot A_I)\Big] \\
&= -iA_I \cdot \nabla_I
\end{aligned}$$

and we're done. The conditional equation is now obtained using this equation for $\partial_t \chi$ as well as the fact that $\Phi_{\vec{R}} = \frac{\Psi}{\chi}$. We have

$$\begin{aligned}
i\partial_t \Phi_{\vec{R}} =& \frac{i\chi\partial_t\Psi - i\Psi\partial_t\chi}{\chi^2} \\
=& \frac{(\hat{H}_{\mathrm{BO}} - \sum_I \frac{\Delta_I}{2M_I})(\Phi_{\vec{R}}\chi)}{\chi} - \frac{\big(\sum_I \frac{(-i\nabla_I + A_I)^2}{2M_I} + \epsilon\big)\chi}{\chi}\Phi_{\vec{R}} \\
=& \Big(\hat{H}_{\mathrm{BO}} - \epsilon\Big)\Phi_{\vec{R}} - \frac{1}{\chi}\Big[\sum_I \frac{\chi\Delta_I + 2(\nabla_I\chi \cdot \nabla_I) + \Delta_I\chi + (-i\nabla_I + A_I)^2\chi}{2M_I}\Big]\Phi_{\vec{R}} \\
=& \Big(\hat{H}_{\mathrm{BO}} - \epsilon\Big)\Phi_{\vec{R}} - \Big[\sum_I \frac{1}{2M_I}\Big(\Delta_I + 2\Big(\frac{\nabla_I\chi}{\chi} \cdot \nabla_I\Big) - \frac{i(\nabla_I \cdot \chi A_I)}{\chi} - \frac{i(A_I \cdot \nabla_I)\chi}{\chi} \\
&+ (A_I \cdot A_I)\Big)\Big]\Phi_{\vec{R}} \\
=& \Big(\hat{H}_{\mathrm{BO}} - \epsilon\Big)\Phi_{\vec{R}} + \Big[\sum_I \frac{1}{M_I}\Big[\frac{(-i\nabla_I - A_I)^2}{2} + \Big(\frac{-i\nabla_I\chi}{\chi} + A_I\Big) \cdot (-i\nabla_I - A_I)\Big]\Big]\Phi_{\vec{R}}
\end{aligned}$$

□

1.2.2 Gauge Invariance

Moreover, we have the following important uniqueness result:

Theorem 2. *Given Ψ a solution to (1), the factorization $(\Phi_{\vec{R}}, \chi)$ given in Theorem 1 is*

unique up to the following so-called 'gauge transformation'

$$\tilde{\Phi}_{\vec{R}} := \Phi_{\vec{R}} e^{i\theta(t,\vec{R})} \tag{13}$$

$$\tilde{\chi} := \chi e^{-i\theta(t,\vec{R})} \tag{14}$$

where $\theta(t,\vec{R})$ is arbitrary. The equations these gauge-transformed wavefunctions satisfy is identical in form to (8)–(12), under the following transformation of the potentials:

$$\tilde{A}_I := A_I + \nabla_I \theta \tag{15}$$

$$\tilde{U}_{en} := U_{en} \tag{16}$$

$$\tilde{\epsilon} := \epsilon + \partial_t \theta \tag{17}$$

Proof. Given a wavefuntion Ψ, suppose one has two distinct factorizations $(\Phi_{\vec{R}}, \chi)$ and $(\tilde{\Phi}_{\vec{R}}, \tilde{\chi})$. Then certainly $|\Phi_{\vec{R}}|^2 = \frac{|\tilde{\chi}|^2}{|\chi|^2}|\tilde{\Phi}_{\vec{R}}|^2$, so $\int |\Phi_{\vec{R}}|^2 d\vec{r} = \frac{|\tilde{\chi}|^2}{|\chi|^2}\int |\tilde{\Phi}_{\vec{R}}|^2 d\vec{r}$. Since both conditional wavefunctions must satisfy the partial normalization constraint, this means $\tilde{\chi} = \chi e^{i\phi(\vec{R},t)}$ for some ϕ, and therefore $\tilde{\Phi}_{\vec{R}} = \Phi_{\vec{R}} e^{-i\phi}$.

Deriving the transformations (15)–(17) amounts to substituting the transformations (13) and (14) into the equations for the potentials (10)–(12). We have

$$\begin{aligned}
\tilde{A}_I =& -i \int \tilde{\Phi}^*_{\vec{R}} \nabla_I \tilde{\Phi}_{\vec{R}} d\vec{r} \\
=& -i \int \Phi^*_{\vec{R}} e^{-i\theta} \left[e^{i\theta} (\nabla_I \Phi_{\vec{R}}) + i(\nabla_I \theta) e^{i\theta} \Phi_{\vec{R}} \right] d\vec{r} \\
=& -i \int \Phi^*_{\vec{R}} \nabla_I \Phi_{\vec{R}} d\vec{r} + (\nabla_I \theta) \int \Phi^*_{\vec{R}} \Phi_{\vec{R}} d\vec{r} \\
=& A_I + \nabla_I \theta
\end{aligned}$$

and

$$\begin{aligned}
\tilde{U}_{\text{en}} =& \sum_{I=1}^{N} \frac{1}{M_I}\Big[\frac{(-i\nabla_I - \tilde{A}_I)^2}{2} + \Big(\frac{-i\nabla_I\tilde{\chi}}{\tilde{\chi}} + \tilde{A}_I\Big)\cdot(-i\nabla_I - \tilde{A}_I)\Big] \\
=& \sum_{I=1}^{N} \frac{1}{M_I}\Big[\frac{-\Delta_I + (i\nabla_I\cdot A_I) + i\Delta_I\theta + (iA_I\cdot\nabla_I) + (A_I\cdot A_I) + (A_I\cdot\nabla_I\theta)}{2} \\
&+ \frac{(i\nabla_I\theta\cdot\nabla_I) + (\nabla_I\theta\cdot A_I) + (\nabla_I\theta\cdot\nabla_I\theta)}{2} - i\Big[\Big(\frac{\nabla_I\chi}{\chi} - i\nabla_I\theta\Big)\cdot\nabla_I\Big] \\
&+ i\Big[\Big(\frac{\nabla_I\chi}{\chi} - i\nabla_I\theta\Big)\cdot A_I\Big] + i\Big[\Big(\frac{\nabla_I\chi}{\chi} - i\nabla_I\theta\Big)\cdot\nabla_I\theta\Big] - (iA_I\cdot\nabla_I) \\
&- (A_I\cdot A_I) - (A_I\cdot\nabla_I\theta) - (i\nabla_I\theta\cdot\nabla_I) - (\nabla_I\theta\cdot A_I) - (\nabla_I\theta\cdot\nabla_I\theta)\Big] \\
=& U_{\text{en}} + \sum_{I=1}^{N} \frac{1}{M_I}\Big[\frac{i\Delta_I\theta + (A_I\cdot\nabla_I\theta) + (i\nabla_I\theta\cdot\nabla_I) + (\nabla_I\theta\cdot A_I) + (\nabla_I\theta\cdot\nabla_I\theta)}{2} \\
&- (\nabla_I\theta\cdot\nabla_I) + (\nabla_I\theta\cdot A_I) + (\nabla_I\theta\cdot\nabla_I\theta) - (A_I\cdot\nabla_I\theta) - (i\nabla_I\theta\cdot\nabla_I) \\
&- (\nabla_I\theta\cdot A_I) - (\nabla_I\theta\cdot\nabla_I\theta)\Big] \\
=& U_{\text{en}}
\end{aligned}$$

and finally

$$\begin{aligned}
\tilde{\epsilon} =& \int \tilde{\Phi}^*_{\vec{R}}(\hat{H}_{\text{BO}} + \tilde{U}_{\text{en}} - i\partial_t)\tilde{\Phi}_{\vec{R}}d\vec{r} \\
=& \int \Phi^*_{\vec{R}}(\hat{H}_{\text{BO}} + U_{\text{en}})\Phi_{\vec{R}}d\vec{r} - i\int \Phi^*_{\vec{R}}e^{-i\theta}\big(e^{i\theta}\partial_t\Phi_{\vec{R}} + ie^{i\theta}(\partial_t\theta)\Phi_{\vec{R}}\big)d\vec{r} \\
=& \epsilon + \partial_t\theta
\end{aligned}$$

□

1.2.3 The Local Conditional Equation

Remark 1. *To avoid computing the integral in (12) (and to retain the 'Born-Oppenheimer perspective' on the molecular problem) we will expand the conditional wavefunction $\Phi_{\vec{R}}$ in the Born-Oppenheimer basis — i.e. we write $\Phi_{\vec{R}}(\vec{r},t) = \sum_{j=1}^{\infty} C_j(\vec{R},t)\Phi^j_{BO}(\vec{r};\vec{R})$, reminiscent of*

the Born-Huang expansion given by (3) *— and we rewrite the conditional equation* (8) *as an evolution equation for the coefficients* $C \equiv \{C_j(\vec{R},t)\}_{j=1}^{\infty}$*, which takes the following form*

$$i\partial_t C_j = (\epsilon^j_{BO} - \epsilon[\chi, C])C_j + U^j_{en}[\chi, C] \tag{18}$$

which we will refer to as the local conditional equation. *Note that the Partial Normalization Condition reduces to*

$$\sum_{j=1}^{\infty} |C_j(\vec{R},t)|^2 = 1 \tag{19}$$

and that we may write the coupling potentials as follows, in terms of C*:*

$$A_I[C] = -i\sum_k \Big[C_k^* \nabla_I C_k + \sum_l C_k^* C_l d_{kl}\Big] \tag{20}$$

$$\begin{aligned} U^k_{en}[\chi, C] = &\sum_{I=1}^{N} \frac{1}{M_I}\Big[\Big(\frac{i\nabla_I \cdot A_I - A_I \cdot A_I - \Delta_I}{2} + \frac{\nabla_I \chi}{\chi} \cdot (iA_I - \nabla_I)\Big)C_k \\ &- \sum_l \Big(\frac{1}{2}\delta_{kl} + d_{kl} \cdot \nabla_I + \frac{\nabla_I \chi}{\chi} \cdot d_{kl}\Big)C_l\Big] \end{aligned} \tag{21}$$

$$\epsilon[\chi, C] = \sum_k \big(C_k^* U^k_{en} + |C_k|^2 \epsilon^k_{BO} - iC_k^* \partial_t C_k\big) \tag{22}$$

where $d_{kl}(\vec{R}) := \int (\Phi^k_{BO})^* \nabla_I \Phi^l_{BO} d\vec{r}$ *and* $\delta_{kl}(\vec{R}) := \int (\Phi^k_{BO})^* \Delta_I \Phi^l_{BO} d\vec{r}$ *are the* non-adiabatic couplings. *Note that while the quantities in equations* (20) *and* (22) *are the same as those in* (10) *and* (12)*, respectively, the quantity* U^j_{en} *in* (21) *is new. Sometimes called the* jth-projected electron-nuclear coupling *in the literature, this quantity is related to the potential in equation* (11) *via the following:* $U^j_{en} := \sum_k \int (\Phi^j_{BO})^* U_{en}\big(C_k \Phi^k_{BO}\big) d\vec{r}$*. Note that, as a consequence, though* (20) *and* (22) *certainly transform like* (15) *and* (17)*, respectively, under the gauge transformation* (13)–(14)*, we have* $\tilde{U}^j_{en} := U^j_{en} e^{i\theta}$ *under the same transformation, as we will show below. Also note that* (13) *can (and will) be equivalently expressed as* $\tilde{C}_k := C_k e^{i\theta(t,\vec{R})}$*.*

Proof. Following the treatment in Gossel, et al., we plug in the ansatz $\sum_k C_k \Phi^k_{\text{BO}}$ into the conditional equation (18) and project onto the Born-Oppenheimer state Φ^j_{BO} (i.e. we multiply on the left by $(\Phi^j_{\text{BO}})^*$ and integrate over $\vec{r}$). By (2) and orthonormality of the Born-Oppenheimer states, we have

$$i\int (\Phi^j_{\text{BO}})^* \partial_t \Big(\sum_k C_k \Phi^k_{\text{BO}}\Big) d\vec{r} = \int (\Phi^j_{\text{BO}})^* \big(\hat{H}_{BO} + U_{en} - \epsilon\big)\Big(\sum_k C_k \Phi^k_{\text{BO}}\Big) d\vec{r}$$
$$\longrightarrow i\partial_t C_j = (\epsilon^j_{\text{BO}} - \epsilon) C_j + U^j_{\text{en}}$$

by the definition of U^j_{en}. All that's left to verify is that the integral $\sum_k \int (\Phi^j_{\text{BO}})^* U_{\text{en}}\big(C_k \Phi^k_{\text{BO}}\big) d\vec{r}$ takes on the form given in (21). We have that

$$\begin{aligned} U_{\text{en}}\Big(C_k \Phi^k_{\text{BO}}\Big) &= \sum_{I=1}^{N} \frac{1}{M_I}\Big[\frac{(-i\nabla_I - A_I)^2}{2} + \Big(\frac{-i\nabla_I \chi}{\chi} + A_I\Big)\cdot(-i\nabla_I - A_I)\Big] C_k \Phi^k_{\text{BO}} \\ &= \sum_{I=1}^{N} \frac{1}{M_I}\Bigg[\frac{-C_k(\Delta_I \Phi^k_{\text{BO}}) - \Phi^k_{\text{BO}}(\Delta_I C_k) - 2(\nabla_I C_k)\cdot(\nabla_I \Phi^k_{\text{BO}})}{2} \\ &\quad + \frac{i(\nabla_I \cdot A_I - A_I \cdot A_I) C_k \Phi^k_{\text{BO}}}{2} + \frac{\nabla_I \chi}{\chi}\cdot\Big(iA_I C_k \Phi^k_{\text{BO}} - (\nabla_I C_k)\Phi^k_{\text{BO}} \\ &\quad - C_k(\nabla_I \Phi^k_{\text{BO}})\Big)\Bigg] \end{aligned}$$

and that therefore, summing over k and projecting onto state j, we have

$$\begin{aligned} U^j_{\text{en}} = \sum_{I=1}^{N} \frac{1}{M_I}\Bigg[&- \sum_k \frac{\delta_{jk}}{2} C_k - \frac{\Delta_I C_j}{2} - \sum_k \frac{d_{jk}}{2}\cdot\nabla_I C_k + \frac{i(\nabla_I \cdot A_I - A_I \cdot A_I) C_j}{2} \\ &+ \frac{\nabla_I \chi}{\chi}\cdot\Big(iA_I C_j - (\nabla_I C_j) - \sum_k C_k(d_{jk})\Big)\Bigg] \end{aligned}$$

where we have used our definitions for the non-adiabatic coupling vectors d_{jk} and δ_{jk}. Now, our new expressions for the other potentials (20) and (22) are obtained by substituting our

ansatz into (10) and (12), respectively. We have

$$\begin{aligned}
A_I =& -i\int\Big(\sum_k C_k^*(\Phi_{\rm BO}^k)^*\Big)\nabla_I\Big(\sum_l C_l\Phi_{\rm BO}^l\Big)d\vec{r}\\
=& -i\sum_k\int C_k^*(\Phi_{\rm BO}^k)^*\sum_l\Big[(\nabla_I C_l)\Phi_{\rm BO}^l + C_l(\nabla_I\Phi_{\rm BO}^l)\Big]d\vec{r}\\
=& -i\sum_k\Big[C_k^*\nabla_I C_k + \sum_l C_k^* C_l d_{kl}\Big]
\end{aligned}$$

and

$$\begin{aligned}
\epsilon =& \int\Big(\sum_k C_k^*(\Phi_{\rm BO}^k)^*\Big)(\hat{H}_{\rm BO}+U_{en}-i\partial_t)\Big(\sum_l C_l\Phi_{\rm BO}^l\Big)d\vec{r}\\
=&\sum_{k,l}\Big[\int\Big(C_k^*C_l(\Phi_{\rm BO}^k)^*\Phi_{\rm BO}^l\epsilon_{\rm BO}^k\Big)d\vec{r} + C_k^*U_{\rm en}^k - i\int\Big(C_k^*(\partial_t C_l)(\Phi_{\rm BO}^k)^*\Phi_{\rm BO}^l\Big)d\vec{r}\Big]\\
=&\sum_k\Big[|C_k|^2\epsilon_{\rm BO}^k + C_k^*U_{\rm en}^k - iC_k^*\partial_t C_k\Big]
\end{aligned}$$

Finally, we would like to show how the jth projected electron-nuclear coupling potential $U_{\rm en}^k$ transforms under the gauge transformation (13)–(14). We have

$$\begin{aligned}
\tilde{U}_{\rm en}^k =&\sum_{I=1}^N\frac{1}{M_I}\Big[\Big(\frac{i\nabla_I\cdot\tilde{A}_I-\tilde{A}_I\cdot\tilde{A}_I-\Delta_I}{2}+\frac{\nabla_I\tilde{\chi}}{\tilde{\chi}}\cdot(i\tilde{A}_I-\nabla_I)\Big)\tilde{C}_k\\
&-\sum_l\Big(\frac{1}{2}\delta_{kl}+d_{kl}\cdot\nabla_I+\frac{\nabla_I\tilde{\chi}}{\tilde{\chi}}\cdot d_{kl}\Big)\tilde{C}_l\Big]\\
=&\sum_{I=1}^N\frac{1}{M_I}\Big[\Big(\frac{(i\nabla_I\cdot A_I)+i\Delta_I\theta-(A_I\cdot A_I)-(A_I\cdot\nabla_I\theta)-(\nabla_I\theta\cdot A_I)-(\nabla_I\theta\cdot\nabla_I\theta)}{2}\\
&+\Big(\frac{\nabla_I\chi}{\chi}-i\nabla_I\theta\Big)\cdot(iA_I+i\nabla_I\theta-\nabla_I)\Big)C_ke^{i\theta}-\sum_l\Big(\frac{1}{2}\delta_{kl}+d_{kl}\cdot\nabla_I\\
&+\Big(\frac{\nabla_I\chi}{\chi}-i\nabla_I\theta\Big)\cdot d_{kl}\Big)C_le^{i\theta}\Big]\\
=&U_{\rm en}^k e^{i\theta}
\end{aligned}$$

□

The purview of this work will be to understand the coupled equations (18) and (9).

2 Deriving the Finite-Level Equations

In this work we will exploit the gauge freedom as described by equations (13) and (14) to understand how we might modify equations (9) and (18) to make them more tractable. The goal will be to choose a θ such that the term $-i\sum_j C_j^* \partial_t C_j$ in ϵ — in the literature this term is often referred to as the *gauge-dependent term* ϵ_{GD} of the time-dependent potential energy surface ϵ — is zero. We note that, by equation (17), this amounts to choosing

$$\theta(\vec{R}, t) = g(\vec{R}) + \int_0^t \sum_j C_j^*(\vec{R}, s)\partial_s C_j(\vec{R}, s) ds$$

where g is a function purely of the nuclear coordinates $\vec{R}$. If one chooses to modify the form of A_I, that would depend on one's choice of g in the above; though in our analysis we do not alter the initial form of A_I, which may be interpreted as choosing $g \equiv 0$. Notice that in this gauge equation (18) becomes

$$i\partial_t C_j = \left[\epsilon^j_{\text{BO}} - \sum_k (C_k^* U^k_{\text{en}} + |C_k|^2 \epsilon^j_{\text{BO}})\right] C_j + U^j_{\text{en}} \tag{23}$$

which, just by glancing at equations (20) and (21) for U^k_{en} and A_I, is evidently extremely complicated: in particular, it is an infinite system of fully nonlinear equations, each of which is coupled to *all* of the (infinitely many) other equations. In this work, we will make two important simplifying assumptions: first, we will take the nuclear subsystem to be 1-dimensional (so $\vec{R} \equiv R \in \mathbb{R}$), and we will assume that the conditional electronic subsystem is sufficiently described by finitely many Born-Oppenheimer levels, i.e. $\Phi_R(\vec{r}, t) \equiv \sum_{j=1}^k C_j(R, t)\Phi^j_{\text{BO}}(\vec{r}; R)$ for some finite k. In particular, we will focus on the case of $k = 1, 2$. Our approach to studying the numerical approximation of the exact factorization system of equations differs from those developed in (4) and (8), in that we will explicitly write out the

form of (23) for a molecular system with a two-level conditional electronic subsystem and examine the effects of all the various terms.

2.1 The Case of a Single Born-Oppenheimer Level

Before proceeding to the two-level case, let's write out equation (23) for the case of $k = 1$. We have

$$i\partial_t C_1 = \epsilon^1_{\mathrm{BO}} C_1 - |C_1|^2 U^1_{\mathrm{en}} - |C_1|^2 \epsilon^1_{\mathrm{BO}} + U^1_{\mathrm{en}} = 0$$

since by partial normalization, $|C_1|^2 = 1$. Therefore, for the single-level case, the conditional wavefunction $\Phi_R = C_1 \Phi^1_{\mathrm{BO}}$, where $C_1 = e^{i\theta(R)}$ — significantly, the phase θ is independent of time, so its R-dependence is fully specified by the initial data. Therefore, in some sense, the one-level case is a degenerate situation which reduces to studying just the marginal equation. Turning to it now, first notice that the vector potential A is given by

$$A = -i\Big[\sum_{k,l} C^*_k C_l d_{kl} + \sum_k C^*_k \partial_R C_k\Big] = -i d_{11} + \partial_R \theta(R,t)$$

and the electron-nuclear coupling U^1_{en} is given by

$$\begin{aligned} U^1_{\mathrm{en}} =& \frac{1}{M}\Big[\Big(\frac{i\partial_R A - A^2 - \partial_R}{2} + \frac{\partial_R \chi}{\chi}(iA - \partial_R)\Big) C_1 - \sum_k \Big(\frac{1}{2}\delta_{jk} + d_{jk}\partial_R + \frac{\partial_R \chi}{\chi} d_{jk}\Big) C_k\Big] \\ =& -\frac{ie^{i\theta}}{M}\Big[\frac{1}{2} + \frac{\partial_R \chi}{\chi} + d_{11}\Big]\partial_R \theta + \frac{e^{i\theta}}{M}\Big[\frac{\partial_R d_{11} + i\partial_R^2 \theta + (d_{11})^2 + 2i d_{11}\partial_R \theta - (\partial_R \theta)^2 - \delta_{11}}{2} \\ & + i\frac{\partial_R \chi}{\chi}(\partial_R \theta)\Big] \end{aligned}$$

so equation (9) reduces to

$$\begin{aligned} i\partial_t \chi =& \Big[\frac{(-i\partial_R + A)^2}{2M} + C^*_1 U^1_{\mathrm{en}} + \epsilon^1_{\mathrm{BO}}\Big]\chi \\ =& -\frac{1}{2M}\partial_R^2 \chi - \frac{(d_{11} + i\partial_R \theta)}{2M}\partial_R \chi + \frac{1}{M}\Big[-i\partial_R \theta\Big(\frac{1}{2} + d_{11}\Big) - \frac{\delta_{11}}{2} + \epsilon^1_{\mathrm{BO}}\Big]\chi \end{aligned}$$

and, if initial data are chosen such that $\partial_R \theta \equiv 0$, the one-level marginal equation is

$$i\partial_t \chi = -\frac{1}{2M}\partial_R^2 \chi - \frac{d_{11}}{2M}\partial_R \chi + \frac{2\epsilon^1_{\mathrm{BO}} - \delta_{11}}{2M}\chi \tag{24}$$

2.2 The Two-Level Equations

2.2.1 The Local Conditional Equation

The local conditional equation (23) in the $k = 2$ case (i.e. the conditional wavefunction is given by $\Phi_R(\vec{r}, t) \equiv \sum_{j=1}^{2} C_j(R, t)\Phi^j_{\mathrm{BO}}(\vec{r}; R)$) is given by (note that here we select $j = 1$ in (23))

$$i\partial_t C_1 = \left[\epsilon^1_{\mathrm{BO}} - C_1^* U^1_{en} - C_2^* U^2_{en} - |C_1|^2\epsilon^1_{\mathrm{BO}} - |C_2|^2\epsilon^2_{\mathrm{BO}}\right]C_1 + U^1_{en} \tag{25}$$

In this section we will carefully unpack this equation by grouping terms by degree of homogeneity in C_1 (referred to below simply as 'degree'). Recall that a function f is homogeneous in x of degree k if $f(ax) = a^k f(x)$ for any constant a. Looking at equation (20), we see that the vector potential $A = -i\left[C_1^*\partial_R C_1 + |C_1|^2 d_{11} + C_1^* C_2 d_{12} + C_2^*\partial_R C_2 + C_2^* C_1 d_{21} + |C_2|^2 d_{22}\right]$; so it contains terms of degree 0, 1 and 2 (ignoring the fact that $|C_2|^2 = 1 - |C_1|^2$ for now). For convenience, we will separate these groups, defining

$$A_0 = C_2^*\partial_R C_2 + |C_2|^2 d_{22}$$
$$A_1 = C_1^* C_2 d_{12} + C_2^* C_1 d_{21}$$
$$A_2 = C_1^*\partial_R C_1 + |C_1|^2 d_{11}$$

so that $A = -i[A_0 + A_1 + A_2]$, and A_k is homogeneous in C_1 of degree k. This means that A^2, which appears in the coupling potentials U^1_{en} and U^2_{en} given by (21), has terms of degree 0-4, so the coupling potentials respectively have terms of degree 0-5 and 0-4. In particular,

we have (separating these groups into curly brackets)

$$
\begin{aligned}
U^1_{en} =& \Big\{ -\frac{\delta_{12}}{2M}C_2 - \frac{d_{12}}{M}\partial_R C_2 - \frac{d_{12}}{M}\frac{\partial_R\chi}{\chi}C_2 \Big\} + \Big\{ \frac{1}{2M}(\partial_R A_0)C_1 + \frac{1}{2M}(A_0)^2 C_1 - \frac{1}{2M}\partial_R^2 C_1 \\
&+ \frac{1}{M}\frac{\partial_R\chi}{\chi}A_0 C_1 - \frac{1}{M}\frac{\partial_R\chi}{\chi}\partial_R C_1 - \frac{\delta_{11}}{2M}C_1 - \frac{d_{11}}{M}\partial_R C_1 - \frac{d_{11}}{M}\frac{\partial_R\chi}{\chi}C_1 \Big\} + \Big\{ \frac{1}{2M}(\partial_R A_1)C_1 \\
&+ \frac{1}{M}A_0 A_1 C_1 + \frac{1}{M}\frac{\partial_R\chi}{\chi}A_1 C_1 \Big\} + \Big\{ \frac{1}{2M}(\partial_R A_2)C_1 + \frac{1}{M}A_0 A_2 C_1 \\
&+ \frac{1}{2M}(A_1)^2 C_1 + \frac{1}{M}\frac{\partial_R\chi}{\chi}A_2 C_1 \Big\} + \Big\{ \frac{1}{M}A_1 A_2 C_1 \Big\} \ + \Big\{ \frac{1}{2M}(A_2)^2 C_1 \Big\}
\end{aligned}
$$

$$
\begin{aligned}
U^2_{en} =& \Big\{ \frac{1}{2M}(\partial_R A_0)C_2 + \frac{1}{2M}(A_0)^2 C_2 - \frac{1}{2M}\partial_R^2 C_2 + \frac{1}{M}\frac{\partial_R\chi}{\chi}A_0 C_2 - \frac{1}{M}\frac{\partial_R\chi}{\chi}\partial_R C_2 \\
&- \frac{\delta_{22}}{2M}C_2 - \frac{d_{22}}{M}\partial_R C_2 - \frac{d_{22}}{M}\frac{\partial_R\chi}{\chi}C_2 \Big\} + \Big\{ \frac{1}{2M}(\partial_R A_1)C_2 + \frac{1}{M}A_0 A_1 C_2 + \frac{1}{M}\frac{\partial_R\chi}{\chi}A_1 C_2 \\
&- \frac{\delta_{21}}{2M}C_1 - \frac{d_{21}}{M}\partial_R C_1 - \frac{d_{21}}{M}\frac{\partial_R\chi}{\chi}C_1 \Big\} + \Big\{ \frac{1}{2M}(\partial_R A_2)C_2 + \frac{1}{M}A_0 A_2 C_2 + \frac{1}{2M}(A_1)^2 C_2 \\
&+ \frac{1}{M}\frac{\partial_R\chi}{\chi}A_2 C_2 \Big\} + \Big\{ \frac{1}{M}A_1 A_2 C_2 \Big\} + \Big\{ \frac{1}{2M}(A_2)^2 C_2 \Big\}
\end{aligned}
$$

This means, reading off the terms in (25) one by one, the equation has terms of degree 1, 2-7, 1-5, 3, 1, and 0-5. Writing the equation as $i\partial_t C_1 = \sum_{k=0}^{7} f_k$ where f_k is the sum of all terms of degree k, we have

$$
f_0 = -\frac{\delta_{12}}{2M}C_2 - \frac{d_{12}}{M}\partial_R C_2 - \frac{d_{12}}{M}\frac{\partial_R\chi}{\chi}C_2
$$

$$
\begin{aligned}
f_1 =&\epsilon^1_{\rm BO}C_1 - C_2^* C_1\Big[\frac{1}{2M}(\partial_R A_0)C_2 + \frac{1}{2M}(A_0)^2 C_2 - \frac{1}{2M}\partial_R^2 C_2 \\
&+ \frac{1}{M}\frac{\partial_R\chi}{\chi}A_0 C_2 - \frac{1}{M}\frac{\partial_R\chi}{\chi}\partial_R C_2 - \frac{\delta_{22}}{2M}C_2 - \frac{d_{22}}{M}\partial_R C_2 - \frac{d_{22}}{M}\frac{\partial_R\chi}{\chi}C_2\Big] \\
&- |C_2|^2\epsilon^2_{\rm BO}C_1 + \frac{1}{2M}(\partial_R A_0)C_1 + \frac{1}{2M}(A_0)^2 C_1 - \frac{1}{2M}\partial_R^2 C_1 + \frac{1}{M}\frac{\partial_R\chi}{\chi}A_0 C_1 \\
&- \frac{1}{M}\frac{\partial_R\chi}{\chi}\partial_R C_1 - \frac{\delta_{11}}{2}C_1 - d_{11}\partial_R C_1 - \frac{\partial_R\chi}{\chi}d_{11}C_1 \\
=&\epsilon^1_{\rm BO}C_1 + \frac{C_1}{2M}(1-|C_2|^2)\Big\{\partial_R A_0 + (A_0)^2 + 2\frac{\partial_R\chi}{\chi}A_0\Big\} + \frac{C_2^* C_1}{2M}\partial_R^2 C_2 \\
&+ \frac{C_2^* C_1}{M}\frac{\partial_R\chi}{\chi}\partial_R C_2 + \frac{|C_2|^2 C_1}{2M}\delta_{22} + \frac{C_2^* C_1 d_{22}}{M}\partial_R C_2 + \frac{|C_2|^2 C_1 d_{22}}{M}\frac{\partial_R\chi}{\chi} \\
&- |C_2|^2\epsilon^2_{\rm BO}C_1 - \frac{1}{2M}\partial_R^2 C_1 - \frac{1}{M}\frac{\partial_R\chi}{\chi}\partial_R C_1 - \frac{\delta_{11}}{2}C_1 - d_{11}\partial_R C_1 - \frac{\partial_R\chi}{\chi}d_{11}C_1
\end{aligned}
$$

$$
\begin{aligned}
f_2 =& - |C_1|^2\Big(- \frac{\delta_{12}}{2M}C_2 - \frac{d_{12}}{M}\partial_R C_2 - \frac{d_{12}}{M}\frac{\partial_R\chi}{\chi}C_2\Big) - C_2^* C_1\Big(\frac{1}{2M}(\partial_R A_1)C_2 + \frac{1}{M}A_0 A_1 C_2 \\
&+ \frac{1}{M}\frac{\partial_R\chi}{\chi}A_1 C_2 - \frac{\delta_{21}}{2M}C_1 - \frac{d_{21}}{M}\partial_R C_1 - \frac{d_{21}}{M}\frac{\partial_R\chi}{\chi}C_1\Big) + \frac{1}{2M}(\partial_R A_1)C_1 + \frac{1}{M}A_0 A_1 C_1 \\
&+ \frac{1}{M}\frac{\partial_R\chi}{\chi}A_1 C_1 \\
=& - |C_1|^2\Big\{ - \frac{\delta_{12}}{2M}C_2 - \frac{d_{12}}{M}\partial_R C_2 - \frac{d_{12}}{M}\frac{\partial_R\chi}{\chi}C_2\Big\} + \frac{C_1}{2M}(1-|C_2|^2)\Big\{\partial_R A_1 \\
&+ 2A_0 A_1 + 2\frac{\partial_R\chi}{\chi}A_1\Big\} + \frac{C_2^*\delta_{21}}{2M}(C_1)^2 + \frac{C_2^* d_{21}}{2M}C_1\partial_R C_1 + \frac{C_2^* d_{21}}{M}\frac{\partial_R\chi}{\chi}(C_1)^2
\end{aligned}
$$

$$
\begin{aligned}
f_3 =& -|C_1|^2\Big(\frac{1}{2M}(\partial_R A_0)C_1+\frac{1}{2M}(A_0)^2C_1-\frac{1}{2M}\partial_R^2 C_1+\frac{1}{M}\frac{\partial_R\chi}{\chi}A_0C_1-\frac{1}{M}\frac{\partial_R\chi}{\chi}\partial_R C_1\\
&-\frac{\delta_{11}}{2M}C_1-\frac{d_{11}}{M}\partial_R C_1-\frac{d_{11}}{M}\frac{\partial_R\chi}{\chi}C_1\Big)-C_2^*C_1\Big(\frac{1}{2M}(\partial_R A_2)C_2+\frac{1}{M}A_0A_2C_2\\
&+\frac{1}{2M}(A_1)^2C_2+\frac{1}{M}\frac{\partial_R\chi}{\chi}A_2C_2\Big)-\epsilon^1_{\rm BO}|C_1|^2C_1+\frac{1}{2M}(\partial_R A_2)C_1+\frac{1}{M}A_0A_2C_1\\
&+\frac{1}{2M}(A_1)^2C_1+\frac{1}{M}\frac{\partial_R\chi}{\chi}A_2C_1\\
=& -|C_1|^2\Big\{\frac{1}{2M}(\partial_R A_0C_1+\frac{1}{2M}(A_0)^2C_1-\frac{1}{2M}\partial_R^2 C_1+\frac{1}{M}\frac{\partial_R\chi}{\chi}A_0C_1-\frac{1}{M}\frac{\partial_R\chi}{\chi}\partial_R C_1\\
&-\frac{\delta_{11}}{2M}C_1-\frac{d_{11}}{M}\partial_R C_1-\frac{d_{11}}{M}\frac{\partial_R\chi}{\chi}C_1\Big\}+\frac{C_1}{2M}(1-|C_2|^2)\Big\{\partial_R A_2+2A_0A_2\\
&+(A_1)^2+2\frac{\partial_R\chi}{\chi}A_2\Big\}-\epsilon^1_{\rm BO}|C_1|^2C_1
\end{aligned}
$$

$$
\begin{aligned}
f_4 =& -|C_1|^2\Big(\frac{1}{2M}(\partial_R A_1)C_1+\frac{1}{M}A_0A_1C_1+\frac{1}{M}\frac{\partial_R\chi}{\chi}A_1C_1\Big)-C_2^*C_1\Big(\frac{1}{M}A_1A_2C_2\Big)\\
&+\frac{1}{M}A_1A_2C_1\\
=& -|C_1|^2\Big\{\frac{1}{2M}(\partial_R A_1)C_1+\frac{1}{M}A_0A_1C_1+\frac{1}{M}\frac{\partial_R\chi}{\chi}A_1C_1\Big\}\\
&+\frac{C_1}{M}(1-|C_2|^2)\{A_1A_2\}
\end{aligned}
$$

$$
\begin{aligned}
f_5 =& -|C_1|^2\Big(\frac{1}{2M}(\partial_R A_2)C_1+\frac{1}{M}A_0A_2C_1+\frac{1}{2M}(A_1)^2C_1+\frac{1}{M}\frac{\partial_R\chi}{\chi}A_2C_1\Big)\\
&-C_2^*C_1\Big(\frac{1}{2M}(A_2)^2C_2\Big)+\frac{1}{2M}(A_2)^2C_1\\
=& -|C_1|^2\Big\{\frac{1}{2M}(\partial_R A_2)C_1+\frac{1}{M}A_0A_2C_1+\frac{1}{2M}(A_1)^2C_1+\frac{1}{M}\frac{\partial_R\chi}{\chi}A_2C_1\Big\}\\
&+\frac{C_1}{2M}(1-|C_2|^2)\{(A_2)^2\}
\end{aligned}
$$

$$
f_6 = -|C_1|^2\Big\{\frac{1}{M}A_1A_2C_1\Big\}
$$

$$f_7 = -\frac{1}{2M}|C_1|^2\{(A_2)^2 C_1\}$$

And we notice that the terms in curly brackets cancel due to partial normalization, since we have:

$$\begin{aligned}
i\partial_t C_1 =& \frac{C_1}{2M}(1-|C_2|^2-|C_1|^2)\Big[\partial_R A_0 + (A_0)^2 + 2\frac{\partial_R\chi}{\chi}A_0 + \partial_R A_1 + 2A_0A_1 \\
&+ 2\frac{\partial_R\chi}{\chi}A_1 + \partial_R A_2 + 2A_0A_2 + (A_1)^2 + 2\frac{\partial_R\chi}{\chi}A_2 + A_1A_2 + (A_2)^2\Big] \\
&+ (1-|C_1|^2)\Big(\frac{\delta_{12}}{2M}C_2 + \frac{d_{12}}{2M}\partial_R C_2 + \frac{d_{12}}{M}\frac{\partial_R\chi}{\chi}C_2\Big) + (1-|C_1|^2)\epsilon^1_{\mathrm{BO}}C_1 \\
&+ \frac{C_2^*\partial_R^2 C_2}{2M}C_1 + \frac{C_2^*\partial_R C_2}{M}\frac{\partial_R\chi}{\chi}C_1 + \frac{|C_2|^2\delta_{22}}{2M}C_1 + \frac{C_2^*\partial_R C_2 d_{22}}{M}C_1 + \frac{|C_2|^2 d_{22}}{M}\frac{\partial_R\chi}{\chi}C_1 \\
&- |C_2|^2\epsilon^2_{\mathrm{BO}}C_1 - \frac{(1-|C_1|^2)}{M}\partial_R^2 C_1 - \frac{(1-|C_1|^2)}{M}\frac{\partial_R\chi}{\chi}\partial_R C_1 - \frac{(1-|C_1|^2)\delta_{11}}{2M}C_1 \\
&- \frac{(1-|C_1|^2)d_{11}}{M}\partial_R C_1 - \frac{(1-|C_1|^2)d_{11}}{M}\frac{\partial_R\chi}{\chi}C_1 + \frac{C_2^*\delta_{21}}{2M}(C_1)^2 + \frac{C_2^* d_{21}}{2M}C_1\partial_R C_1 \\
&+ \frac{C_2^* d_{21}}{M}\frac{\partial_R\chi}{\chi}(C_1)^2 \\
=& -\frac{|C_2|^2}{2M}\partial_R^2 C_1 - \frac{|C_2|^2}{M}\Big(\frac{\partial_R\chi}{\chi} + d_{11}\Big)\partial_R C_1 + \frac{C_2^* d_{21}}{2M}C_1\partial_R C_1 + \frac{C_2^*}{M}\Big(\frac{\delta_{21}}{2} + d_{21}\frac{\partial_R\chi}{\chi}\Big)(C_1)^2 \\
&+ \frac{C_2^*}{M}\Big(\frac{\partial_R^2 C_2}{2} + \partial_R C_2\frac{\partial_R\chi}{\chi} + \partial_R C_2 d_{22}\Big)C_1 + \frac{|C_2|^2}{2M}\Big(2M\epsilon^1_{\mathrm{BO}} + \delta_{22} + 2d_{22}\frac{\partial_R\chi}{\chi} - 2M\epsilon^2_{\mathrm{BO}} \\
&- \delta_{11} - 2d_{11}\frac{\partial_R\chi}{\chi}\Big)C_1 - |C_2|^2\Big(\frac{\delta_{12}}{2M}C_2 + \frac{d_{12}}{2M}\partial_R C_2 + \frac{d_{12}}{M}\frac{\partial_R\chi}{\chi}C_2\Big)
\end{aligned} \tag{26}$$

So (26) is the local conditional equation giving the dynamics of the ground-state population C_1: it is apparent that the equation for C_2 can be obtained in an identical fashion, or simply by switching all the indices in the above:

$$\begin{aligned}
i\partial_t C_2 =& -\frac{|C_1|^2}{2M}\partial_R^2 C_2 - \frac{|C_1|^2}{M}\Big(\frac{\partial_R\chi}{\chi} + d_{22}\Big)\partial_R C_2 + \frac{C_1^* d_{12}}{2M}C_2\partial_R C_2 + \frac{C_1^*}{M}\Big(\frac{\delta_{12}}{2} + d_{12}\frac{\partial_R\chi}{\chi}\Big)(C_2)^2 \\
&+ \frac{C_1^*}{M}\Big(\frac{\partial_R^2 C_1}{2} + \partial_R C_1\frac{\partial_R\chi}{\chi} + \partial_R C_1 d_{11}\Big)C_2 + \frac{|C_1|^2}{2M}\Big(2M\epsilon^2_{\mathrm{BO}} + \delta_{11} + 2d_{11}\frac{\partial_R\chi}{\chi} - 2M\epsilon^1_{\mathrm{BO}} \\
&- \delta_{22} - 2d_{22}\frac{\partial_R\chi}{\chi}\Big)C_2 - |C_1|^2\Big(\frac{\delta_{21}}{2M}C_1 + \frac{d_{21}}{2M}\partial_R C_1 + \frac{d_{21}}{M}\frac{\partial_R\chi}{\chi}C_1\Big)
\end{aligned} \tag{27}$$

Remark 2. *The quantity* $|C_1|^2+|C_2|^2$ *is conserved by equations* (26) *and* (27)*; in particular, if the partial normalization condition* $|C_1|^2 + |C_2|^2 = 1$ *holds at time* $t = 0$*, it will hold for any corresponding solution of* (26)–(27)*.*

Proof. Consider first the following system which we will call the 'dispersive part' of (26) – (27), since it isolates the terms involving second derivatives:

$$\begin{cases} i\partial_t C_1 = -\frac{|C_2|^2}{2M}\partial_R^2 C_1 + \frac{C_2^*\partial_R^2 C_2}{2M} C_1 \\ i\partial_t C_2 = -\frac{|C_1|^2}{2M}\partial_R^2 C_2 + \frac{C_1^*\partial_R^2 C_1}{2M} C_2 \end{cases} \tag{28}$$

And notice that

$$\begin{aligned} &\partial_t\big(|C_1|^2 + |C_2|^2\big) \\ &= \partial_t(C_1^* C_1 + C_2^* C_2) \\ &= C_1^*\partial_t C_1 + (C_1^*\partial_t C_1)^* + C_2^*\partial_t C_2 + (C_2^*\partial_t C_2)^* \\ &= 2\mathrm{Re}(C_1^*\partial_t C_1 + C_2^*\partial_t C_2) \\ &= 2\mathrm{Re}\Big[-iC_1^*\Big(-\frac{|C_2|^2}{2M}\partial_R^2 C_1 + \frac{C_2^*\partial_R^2 C_2}{2M} C_1\Big) - iC_2^*\Big(-\frac{|C_1|^2}{2M}\partial_R^2 C_2 + \frac{C_1^*\partial_R^2 C_1}{2M} C_2\Big)\Big] \\ &= 0 \end{aligned}$$

There are eight other pairings of terms with which one can perform the identical computation. Writing them as isolated systems as in (28), these 'parts' of the full system are

$$\begin{cases} i\partial_t C_1 = -\frac{|C_2|^2}{M}\frac{\partial_R \chi}{\chi}\partial_R C_1 + \frac{C_2^*\partial_R C_2}{M}\frac{\partial_R \chi}{\chi} C_1 \\ i\partial_t C_2 = -\frac{|C_1|^2}{M}\frac{\partial_R \chi}{\chi}\partial_R C_2 + \frac{C_1^*\partial_R C_1}{M}\frac{\partial_R \chi}{\chi} C_2 \end{cases}$$

$$\begin{cases} i\partial_t C_1 = -\frac{d_{11}|C_2|^2}{M}\partial_R C_1 + \frac{d_{22}C_2^*\partial_R C_2}{M} C_1 \\ i\partial_t C_2 = -\frac{d_{22}|C_1|^2}{M}\partial_R C_2 + \frac{d_{11}C_1^*\partial_R C_1}{M} C_2 \end{cases}$$

$$\begin{cases} i\partial_t C_1 = \frac{d_{21}C_2^*}{2M} C_1\partial_R C_1 - \frac{d_{12}|C_2|^2\partial_R C_2}{2M} \\ i\partial_t C_2 = \frac{d_{12}C_1^*}{2M} C_2\partial_R C_2 - \frac{d_{21}|C_1|^2\partial_R C_1}{2M} \end{cases}$$

$$\begin{cases} i\partial_t C_1 = \frac{\delta_{21}C_2^*}{2M}(C_1)^2 - \frac{\delta_{12}|C_2|^2 C_2}{2M} \\ i\partial_t C_2 = \frac{\delta_{12}C_1^*}{2M}(C_2)^2 - \frac{\delta_{21}|C_1|^2 C_1}{2M} \end{cases}$$

$$\begin{cases} i\partial_t C_1 = \frac{d_{21}C_2^*}{M}\frac{\partial_R\chi}{\chi}(C_1)^2 - \frac{d_{12}C_2|C_2|^2}{2M}\frac{\partial_R\chi}{\chi} \\ i\partial_t C_2 = \frac{d_{12}C_1^*}{M}\frac{\partial_R\chi}{\chi}(C_2)^2 - \frac{d_{21}C_1|C_1|^2}{2M}\frac{\partial_R\chi}{\chi} \end{cases}$$

$$\begin{cases} i\partial_t C_1 = (\epsilon^2_{\mathrm{BO}} - \epsilon^1_{\mathrm{BO}})C_2|C_1|^2 \\ i\partial_t C_2 = (\epsilon^1_{\mathrm{BO}} - \epsilon^2_{\mathrm{BO}})C_1|C_2|^2 \end{cases}$$

$$\begin{cases} i\partial_t C_1 = \frac{\delta_{11}-\delta_{22}}{2M}C_2|C_1|^2 \\ i\partial_t C_2 = \frac{\delta_{22}-\delta_{11}}{2M}C_1|C_2|^2 \end{cases}$$

$$\begin{cases} i\partial_t C_1 = \frac{d_{11}-d_{22}}{M}\frac{\partial_R\chi}{\chi}C_2|C_1|^2 \\ i\partial_t C_2 = \frac{d_{22}-d_{11}}{M}\frac{\partial_R\chi}{\chi}C_1|C_2|^2 \end{cases}$$

□

2.2.2 The Marginal Equation

Now we will write out the explicit form of the marginal equation (9) for the two-level case. Recall that we have

$$i\partial_t\chi = \Big[\frac{-\partial_R^2 - iA\partial_R - i\partial_R A + A^2}{2M} + C_1^* U_{en}^1 + |C_1|^2\epsilon_{\mathrm{BO}}^1 + C_2^* U_{en}^2 + |C_2|^2\epsilon_{\mathrm{BO}}^2\Big]\chi$$

Notice that, organizing once again by homogeneity in C_1, these terms are respectively of degree 0, 0-2, 0-2, 0-4, 1-6, 2, 0-4, 0. As above, we will write $i\partial_t\chi = \sum_{k=0}^{6} g_k$, where g_k is the sum of all the terms of degree k. We have

$$\begin{aligned} g_0 = &-\frac{1}{2M}\partial_R^2\chi - \frac{A_0}{2M}\partial_R\chi - \frac{\partial_R A_0}{2M}\chi - \frac{A_0^2}{2M}\chi + C_2^*\Big[\frac{1}{2M}(\partial_R A_0)C_2 + \frac{1}{2M}(A_0)^2 C_2 \\ &- \frac{1}{2M}\partial_R^2 C_2 + \frac{1}{M}\frac{\partial_R\chi}{\chi}A_0 C_2 - \frac{1}{M}\frac{\partial_R\chi}{\chi}\partial_R C_2 - \frac{\delta_{22}}{2M}C_2 - \frac{d_{22}}{M}\partial_R C_2 - \frac{d_{22}}{M}\frac{\partial_R\chi}{\chi}C_2\Big]\chi \\ &+ |C_2|^2\epsilon_{\mathrm{BO}}^2\chi \end{aligned}$$

$$\begin{aligned} g_1 = &-\frac{A_1}{2M}\partial_R\chi - \frac{\partial_R A_1}{2M}\chi - \frac{A_0 A_1}{M}\chi + C_1^*\Big[-\frac{\delta_{12}}{2M}C_2 - \frac{d_{12}}{2M}\partial_R C_2 - \frac{d_{12}}{M}\frac{\partial_R\chi}{\chi}C_2\Big]\chi \\ &+ C_2^*\Big[\frac{1}{2M}(\partial_R A_1)C_2 + \frac{1}{M}A_0 A_1 C_2 + \frac{1}{M}\frac{\partial_R\chi}{\chi}A_1 C_2 - \frac{\delta_{21}}{2M}C_1 - \frac{d_{21}}{M}\partial_R C_1 - \frac{d_{21}}{M}\frac{\partial_R\chi}{\chi}C_1\Big]\chi \end{aligned}$$

$$\begin{aligned} g_2 = &-\frac{A_2}{2M}\partial_R\chi - \frac{\partial_R A_2}{2M}\chi - \frac{A_0 A_2}{2M}\chi - \frac{A_1^2}{M}\chi + C_1^*\Big[\frac{1}{2M}(\partial_R A_0)C_1 + \frac{1}{2M}(A_0)^2 C_1 - \frac{1}{2M}\partial_R^2 C_1 \\ &+ \frac{1}{M}\frac{\partial_R\chi}{\chi}A_0 C_1 - \frac{1}{M}\frac{\partial_R\chi}{\chi}\partial_R C_1 - \frac{\delta_{11}}{2M}C_1 - \frac{d_{11}}{M}\partial_R C_1 - \frac{d_{11}}{M}\frac{\partial_R\chi}{\chi}C_1\Big]\chi + |C_1|^2\epsilon_{\mathrm{BO}}^1\chi \\ &+ C_2^*\Big[\frac{1}{2M}(\partial_R A_2)C_2 + \frac{1}{M}A_0 A_2 C_2 + \frac{1}{2M}(A_1)^2 C_2 + \frac{1}{M}\frac{\partial_R\chi}{\chi}A_2 C_2\Big]\chi \end{aligned}$$

$$g_3 = -\frac{A_1 A_2}{M}\chi + C_1^*\Big[\frac{1}{2M}(\partial_R A_1)C_1 + \frac{1}{M}A_0 A_1 C_1 + \frac{1}{M}\frac{\partial_R \chi}{\chi}A_1 C_1\Big]\chi + C_2^*\Big[\frac{1}{M}A_1 A_2 C_2\Big]\chi$$

$$\begin{aligned} g_4 = &-\frac{A_2^2}{2M}\chi + C_1^*\Big[\frac{1}{2M}(\partial_R A_2)C_1 + \frac{1}{M}A_0 A_2 C_1 + \frac{1}{2M}(A_1)^2 C_1 + \frac{1}{M}\frac{\partial_R \chi}{\chi}A_2 C_1\Big]\chi \\ &+ C_2^*\Big[\frac{1}{2M}(A_2)^2 C_2\Big]\chi \end{aligned}$$

$$g_5 = C_1^*\Big[\frac{1}{M}A_1 A_2 C_1\Big]\chi$$

$$g_6 = C_1^*\Big[\frac{1}{2M}(A_2)^2 C_1\Big]\chi$$

Giving us a similar amount of canceling as in the conditional equation:

$$\begin{aligned} i\partial_t \chi = &-\frac{1}{2M}\partial_R^2\chi \\ &+ \frac{\partial_R\chi}{M}\Big[\frac{A_0}{2} + A_0|C_2|^2 - C_2^*\partial_R C_2 - d_{22}|C_2|^2 + \frac{A_1}{2} - d_{12}C_1^* C_2 + A_1|C_2|^2 - d_{21}C_2^* C_1 \\ &+ \frac{A_2}{2} + A_0|C_1|^2 - C_1^*\partial_R C_1 - d_{11}|C_1|^2 + A_2|C_2|^2 + A_1|C_1|^2 + A_2|C_1|^2\Big] \\ &+ \frac{\chi}{M}\Big[- C_2^*\partial_R^2 C_2 - \frac{\delta_{22}}{M}|C_2|^2 - d_{22}C_2^*\partial_R C_2 + |C_2|^2\epsilon_{\mathrm{BO}}^2 \\ &\quad - \frac{\delta_{12}}{2}C_1^* C_2 - \frac{d_{12}}{2}C_1^*\partial_R C_2 - \frac{\delta_{21}}{2}C_2^* C_1 \\ &\quad - d_{21}C_2^*\partial_R C_1 - \frac{\partial_R^2 C_1}{2} - \frac{\delta_{11}}{2}|C_1|^2 - d_{11}C_1^*\partial_R C_1 + |C_1|^2\epsilon_{\mathrm{BO}}^1\Big] \\ = &-\frac{1}{2M}\partial_R^2\chi + \frac{\partial_R\chi}{2M}\Big[C_1^*\partial_R C_1 + |C_1|^2 d_{11} + C_1^* C_2 d_{12} + C_2^*\partial_R C_2 + C_2^* C_1 d_{21} + |C_2|^2 d_{22}\Big] \\ &+ \frac{\chi}{M}\Big[- C_2^*\partial_R^2 C_2 - \frac{\delta_{22}}{M}|C_2|^2 - d_{22}C_2^*\partial_R C_2 + |C_2|^2\epsilon_{\mathrm{BO}}^2 \\ &\quad - \frac{\delta_{12}}{2}C_1^* C_2 - \frac{d_{12}}{2}C_1^*\partial_R C_2 - \frac{\delta_{21}}{2}C_2^* C_1 \\ &\quad - d_{21}C_2^*\partial_R C_1 - \frac{\partial_R^2 C_1}{2} - \frac{\delta_{11}}{2}|C_1|^2 - d_{11}C_1^*\partial_R C_1 + |C_1|^2\epsilon_{\mathrm{BO}}^1\Big] \end{aligned} \tag{29}$$

So the two-level dynamics are fully specified by the system (26), (27) and (29). Notice that unlike the local conditional equations (26) and (27), the marginal equation (29) is linear.

3 Numerical Simulations

3.1 Simulating Schrödinger-Like Equations with Newton's Method

3.1.1 Free Schrödinger Equation

Before beginning to examine the toy models derived above, we recall some techniques in numerically simulating elementary Schrödinger-like equations. In doing so, we will also be building toward a study of the one-level marginal equation (24), which is a 1-dimensional Schrödinger equation with two additional lower-order terms (an advective one and a damping one) with, in general, complex coefficients depending on R. Consider the free Schrödinger equation

$$i\partial_t \psi = \partial_R^2 \psi$$

Setting $\psi(R,t) := a(R,t) + ib(R,t)$, we obtain the system

$$\partial_t \begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \partial_R^2 \begin{pmatrix} a \\ b \end{pmatrix}$$

which is nothing but a 'generalized' two-dimensional heat equation. To simulate this, writing the solution vector as $\vec{w} := \left(\begin{smallmatrix} a \\ b \end{smallmatrix}\right)$, we employ the trapezoidal method with a centered difference for the second derivative. Recall that for an equation of the form $\partial_t y = F$, the discretization given by the trapezoidal rule is

$$\frac{y_i^k - y_i^{k-1}}{\Delta t} = \frac{1}{2}(F_i^k + F_i^{k-1}) \tag{30}$$

where subscripts denote spatial gridpoints, superscripts denote time gridpoints, and where the respective grid spacings are ΔR and Δt. It is proven in [6] that this method is second-order, as opposed to the first-order backward Euler method

$$\frac{y_i^k - y_i^{k-1}}{\Delta t} = F_i^k$$

which as we will see, will make a difference in preserving the integral of $|\psi|^2$. Implementing this method for the free Schrödinger equation gives us

$$\frac{\vec{w}_i^k - \vec{w}_i^{k-1}}{\Delta t} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \frac{1}{2}\left[\frac{\vec{w}_{i+1}^k - 2\vec{w}_i^k + \vec{w}_{i-1}^k}{(\Delta R)^2} + \frac{\vec{w}_{i+1}^{k-1} - 2\vec{w}_i^{k-1} + \vec{w}_{i-1}^{k-1}}{(\Delta R)^2}\right]$$

Setting $A := \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$, we can rewrite this as

$$\Big[I + \lambda A\Big]\vec{w}_i^k + \Big[-I + \lambda A\Big]\vec{w}_i^{k-1} - \frac{\lambda}{2}A(\vec{w}_{i+1}^k + \vec{w}_{i-1}^k + \vec{w}_{i+1}^{k-1} + \vec{w}_{i-1}^{k-1}) = 0 \tag{31}$$

where $\lambda := \frac{\Delta t}{(\Delta R)^2}$ and I is the identity matrix. So, our problem has been reduced to iteratively solving an equation of the form $F\big(\vec{w}_0^k, ..., \vec{w}_N^k; \vec{w}_0^{k-1}, ..., \vec{w}_N^{k-1}\big) = 0$, where there are N spatial grid points, starting at $k = 1$. Although in this instance, F is linear, so in particular this problem could be easily solved via an LU decomposition, we will employ Newton's method, since this is what would be needed if F were nonlinear, which is certainly the case if we were to solve any portion of the two-level local conditional system, including the dispersive part (28). Now, we recall that for a function f of one variable, this amounts to, given some initial guess x_0, choosing as one's next guess wherever the line tangent to the graph of f at x_0 intersects the x-axis, which is nothing but

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

One then iterates this process via the recursion

$$x_n = x_{n-1} - \frac{f(x_{n-1})}{f'(x_{n-1})}$$

Likewise, to determine the zeros of a (generally nonlinear) function $F : \mathbb{R}^k \to \mathbb{R}^k$ (i.e. to solve a possibly nonlinear system of k equations in k variables), given some initial guess, one iterates

$$\vec{X}_n = \vec{X}_{n-1} - J_F^{-1}(\vec{X}_{n-1})F(\vec{X}_{n-1})$$

where J_F is the Jacobian matrix of F, or equivalently (multiplying both sides by J_F)

$$J_F(\vec{X}_{n-1})(\vec{X}_n - \vec{X}_{n-1}) = -F(\vec{X}_{n-1}) \tag{32}$$

So, the discretized problem (31) has been reduced to solving the system of linear equations (32) involving the following Jacobian J_F (letting N be the number of spatial gridpoints and ordering the unknowns $a_0^k, b_0^k, a_1^k, ...$)

$$\begin{aligned}
(J_F)_{2i,2i-1} &= -\frac{\lambda}{2} \\
(J_F)_{2i,2i} &= 1 \\
(J_F)_{2i,2i+1} &= \lambda \\
(J_F)_{2i,2i+3} &= -\frac{\lambda}{2} \\
(J_F)_{2i+1,2i-2} &= \frac{\lambda}{2} \\
(J_F)_{2i+1,2i} &= -\lambda \\
(J_F)_{2i+1,2i+1} &= 1 \\
(J_F)_{2i+1,2i+2} &= \frac{\lambda}{2}
\end{aligned}$$

for all $0 < i < N$. We impose Neumann (i.e. zero-derivative) boundary conditions at $i = 0$ and N by introducing the fictitious points $(a^k_{-1}, b^k_{-1}, a^k_{N+1}, b^k_{N+1}) := (a^k_1, b^k_1, a^k_{N-1}, b^k_{N-1})$, ensuring that the following approximations to the first derivative evaluated at the boundaries

$$\left(\frac{da}{dR}\right)^k_0 \approx \frac{a^k_1 - a^k_{-1}}{2\Delta R}$$
$$\left(\frac{da}{dR}\right)^k_N \approx \frac{a^k_{N+1} - a^k_{N-1}}{2\Delta R}$$

(and likewise for b) are zero. This gives us for the 'corners' of the Jacobian:

$$(J_F)_{0,0} = 1$$
$$(J_F)_{0,1} = \lambda$$
$$(J_F)_{0,3} = -\lambda$$
$$(J_F)_{1,0} = -\lambda$$
$$(J_F)_{1,1} = 1$$
$$(J_F)_{1,2} = \lambda$$
$$(J_F)_{N,N-1} = -\lambda$$
$$(J_F)_{N,N} = 1$$
$$(J_F)_{N,N+1} = \lambda$$
$$(J_F)_{N+1,N-2} = \lambda$$
$$(J_F)_{N+1,N} = -\lambda$$
$$(J_F)_{N+1,N+1} = 1$$

Note that our choice of boundary conditions is also a matter of looking ahead to dealing with the two-level local conditional system: in that case, Dirichlet boundary conditions would be inadequate, since that would require violation of partial normalization at the boundaries (periodic boundary conditions would, however, be admissible). Now, the Free Schrödinger

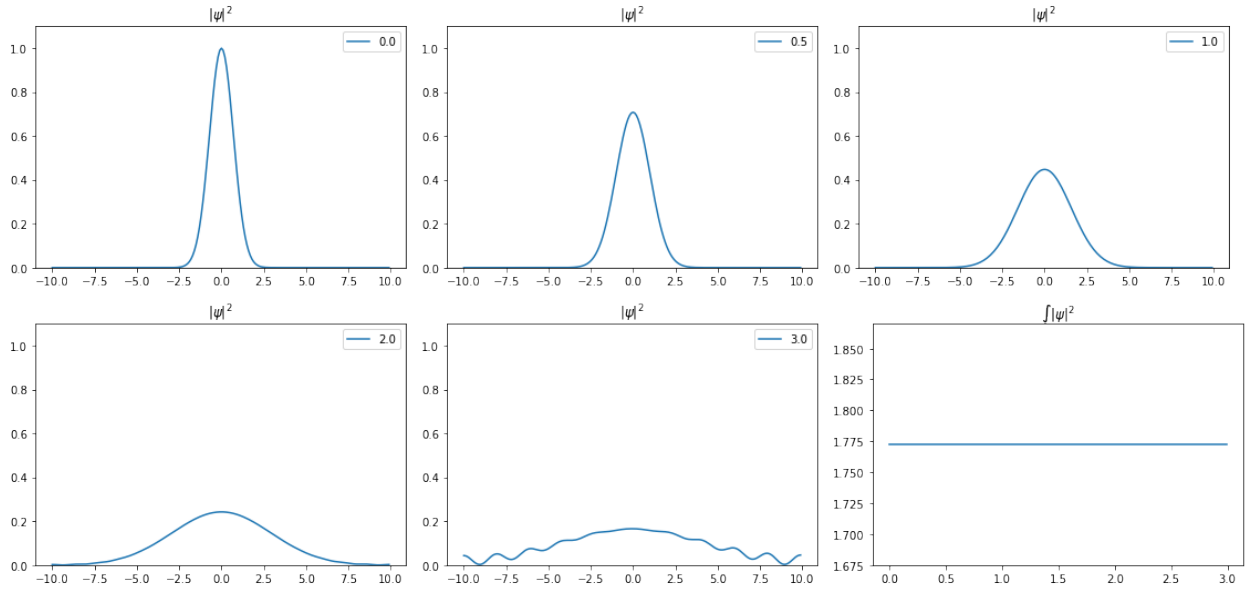

Figure 1: Free Schrödinger dynamics obtained using Newton's method and discretizing via the trapezoidal method ($\Delta t = 0.01$, $\Delta R = 0.1$, $R \in [-10, 10]$, 10 iterations of Newton per timestep, initial condition: $\psi = e^{-R^2/2}$). Top row: density $|\psi|^2$ at $t = 0, 0.5, 1$. Bottom row: density $|\psi|^2$ at $t = 2, 3$ and the energy $\int |\psi|^2 dR$ plotted over time.

dynamics obtained through Newton's method can be seen in *Figure* 1, and compared to results obtained through the backward Euler method in *Figure* 2. Though the results are superficially similar, we see that the trapezoidal method conserves the integral $\int |\psi|^2 dR$, while the backward Euler method does not.

3.1.2 Schrödinger Equation with Advection

Having studied the most basic Schrödinger-like equation, we consider a scalar equation which more closely resembles the one-level marginal equation (24), for now ignoring the dependence of the coefficients on R. Note that, in practice, one would usually obtain those coefficients by studying a model Born-Oppenheimer Hamiltonian and finding its eigenfunctions: see Hagedorn, et al. (2005) and Martinez, P., et al. (2021) for examples of such model systems. For now, we look at

$$i\partial_t \psi = \partial_R^2 \psi + bi\partial_R \psi \tag{33}$$

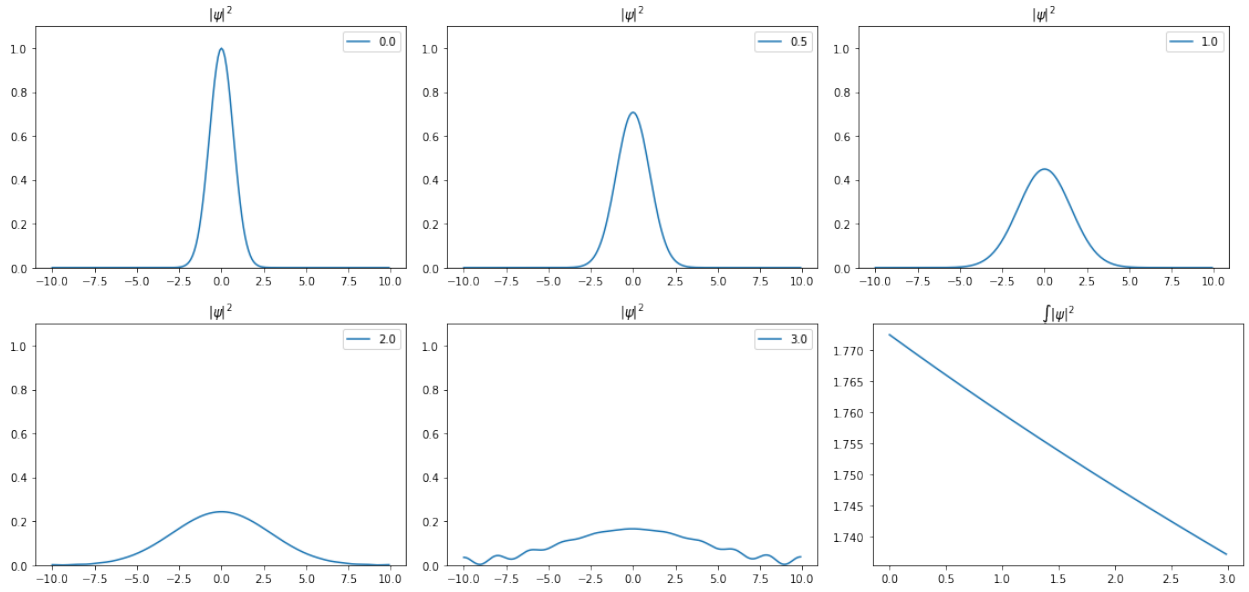

Figure 2: Free Schrödinger dynamics obtained using Newton's method and discretizing via the backward Euler method ($\Delta t = 0.01$, $\Delta R = 0.1$, $R \in [-10, 10]$, 10 iterations of Newton per timestep, initial condition: $\psi = e^{-R^2/2}$). Top row: density $|\psi|^2$ at $t = 0, 0.5, 1$. Bottom row: density $|\psi|^2$ at $t = 2, 3$ and the energy $\int |\psi|^2 dR$ plotted over time.

for $b \in \mathbb{R}$ constant. Since we are ignoring the R-dependent and highly context-dependent coefficients in (24), only looking at the advective term is sufficient modulo a simple integrating factor: letting ψ be a solution to the above and defining $\tilde{\psi} := e^{-iR}\psi$, we find that (33) implies

$$i\partial_t \tilde{\psi} = \partial_R^2 \tilde{\psi} + (2+b) i\partial_t \tilde{\psi} - (1+b)\tilde{\psi}$$

so it is sufficient for our purposes to look at (33). We discretize this equation via the same method as above, employing a centered-difference for the second derivative and the trapezoidal rule, and also using a centered-difference for the first derivative (i.e. $(\partial_R \psi)_i^k \approx \frac{\psi_{i+1}^k - \psi_{i-1}^k}{2\Delta R}$). This leads, setting $\lambda_1 := \frac{\Delta t}{\Delta x}$ and $\lambda_2 := \frac{\Delta t}{(\Delta R)^2}$, to

$$\Big[I + \lambda_2 A\Big]\vec{w}_i^k + \Big[-I + \lambda_2 A\Big]\vec{w}_i^{k-1} - \frac{\lambda}{2} A(\vec{w}_{i+1}^k + \vec{w}_{i-1}^k + \vec{w}_{i+1}^{k-1} + \vec{w}_{i-1}^{k-1}) - \frac{\lambda_1}{4} B(\vec{w}_{i+1}^k + \vec{w}_{i-1}^k - \vec{w}_{i+1}^{k-1} - \vec{w}_{i-1}^{k-1}) = 0$$

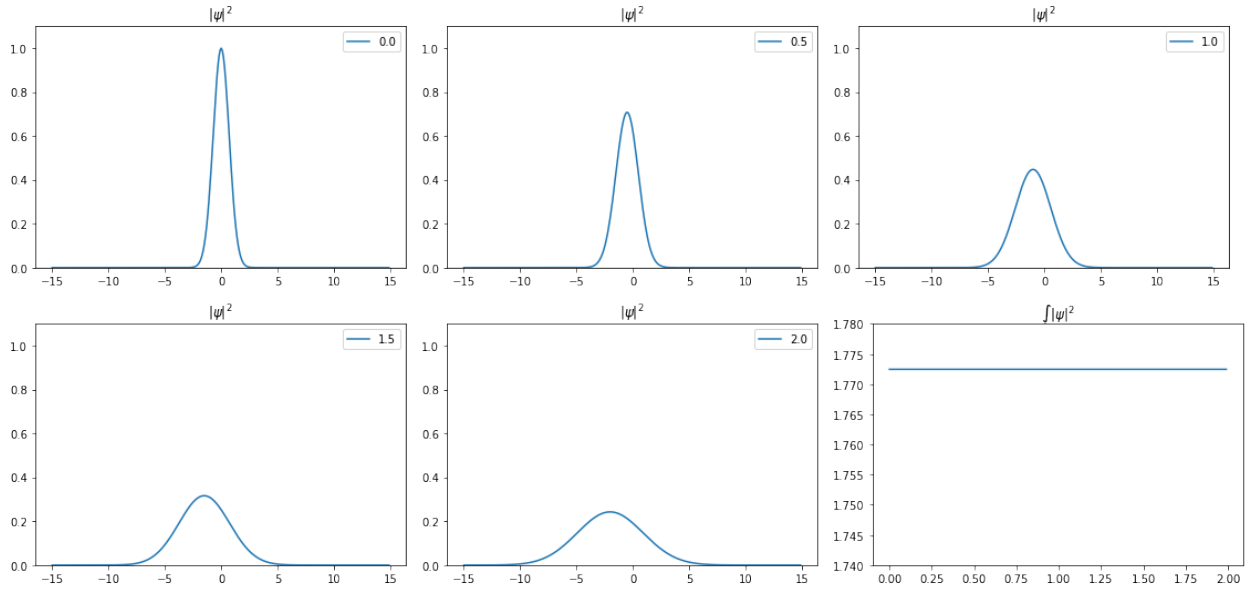

Figure 3: Schrödinger dynamics with pure imaginary advection obtained using Newton's method and discretizing via the trapezoid method ($\Delta t = 0.01$, $\Delta R = 0.1$, $R \in [-15, 15]$, $b = 1$, 10 iterations of Newton per timestep, initial condition: $\psi = e^{-R^2/2}$). Top row: density $|\psi|^2$ at $t = 0, 0.5, 1$. Bottom row: density $|\psi|^2$ at $t = 1.5, 2$ and the energy $\int |\psi|^2 dR$ plotted over time.

where, once again, $\vec{w}$ is the solution vector containing the real and imaginary parts of ψ as its components, $A = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$ and we set $B := \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$. Employing Newton's method once again, we obtain the dynamics found in Figure 3, noticing the broadening and leftward advection we expect, and energy conservation.

3.2 The Dispersive Part

Though our long-term goal is to simulate the system of three equations (26), (27) and (29), we expect that our simulations of the two equations (28), which we called the 'dispersive part' of the two-level local conditional equation, given by

$$\begin{cases} i\partial_t C_1 = -\frac{|C_2|^2}{2M}\partial_R^2 C_1 + \frac{C_2^* \partial_R^2 C_2}{2M} C_1 \\ i\partial_t C_2 = -\frac{|C_1|^2}{2M}\partial_R^2 C_2 + \frac{C_1^* \partial_R^2 C_1}{2M} C_2 \end{cases}$$

which was shown above to independently enforce the partial normalization condition (19), may in some sense build on the simulations discussed in the previous section. Naturally, the situation here is complicated by the solution-dependent coefficients on this system's dispersive and damping terms. Note that this piece of the equation contains no coupling to χ, so we need not concern ourselves with simulating the marginal equation as of yet. Defining, as above, $C_1 \equiv u_1 + iv_1$ and $C_2 \equiv u_2 + iv_2$, where u_i and v_i are real-valued for all i, and defining $\mu := \frac{1}{2M}$ we find that

$$i\partial_t(u_1 + iv_1) = -\mu(u_2^2 + v_2^2)\partial_R^2(u_1 + iv_1) + \mu(u_1u_1 + v_1v_2 - iu_1v_2 + iu_2v_1)\partial_R^2(u_2 + iv_2)$$

giving us the following system of four equations for u_1, v_1, u_2, and v_2 (note that the evolution equations for u_2 and v_2 may be obtained by switching the indices in the above):

$$\begin{cases} \partial_t u_1 = -\mu(u_1^2 + v_1^2)\partial_R^2 v_1 + (u_2v_1 - u_1v_2)\partial_R^2 u_2 + (v_1v_2 + u_1u_2)\partial_R^2 v_2 \\ \partial_t v_1 = \mu(u_1^2 + v_1^2)\partial_R^2 u_1 - (u_1u_2 + v_1v_2)\partial_R^2 u_2 + (u_2v_1 - u_1v_2)\partial_R^2 v_2 \\ \partial_t u_2 = -\mu(u_2^2 + v_2^2)\partial_R^2 v_2 + (u_1v_2 - u_2v_1)\partial_R^2 u_1 + (v_2v_1 + u_2u_1)\partial_R^2 v_1 \\ \partial_t v_2 = \mu(u_2^2 + v_2^2)\partial_R^2 u_2 - (u_2u_1 + v_2v_1)\partial_R^2 u_1 + (u_1v_2 - u_2v_1)\partial_R^2 v_1 \end{cases} \tag{34}$$

or, equivalently,

$$\partial_t \vec{C} = A(\vec{C};\mu)\partial_R^2 \vec{C}, \quad \text{where} \quad \vec{C} := \begin{pmatrix} u_1 & v_1 & u_2 & v_2 \end{pmatrix}^t,$$

and

$$A(\vec{C};\mu) := \begin{pmatrix} 0 & -\mu(u_1^2 + v_1^2) & u_2v_1 - u_1v_2 & v_2v_1 + u_1u_2 \\ \mu(u_1^2 + v_1^2) & 0 & -(u_1u_2 + v_1v_2) & v_1u_2 - u_1v_2 \\ u_1v_2 - v_1u_2 & v_1v_2 + u_1u_2 & 0 & \mu(u_2^2 + v_2^2) \\ -(u_1u_2 + v_1v_2) & u_1v_2 - v_1u_2 & \mu(u_2^2 + v_2^2) & 0 \end{pmatrix}.$$

In future work, we intend to lay out a procedure for studying the numerical approximation to (34), systematically adding terms from (26)–(27), careful to accompany each term by its partner term which enforces partial normalization.

4 References

[1] Abedi, A., Maitra., N., Gross, E.K.U. *Exact Factorization of the Time-Dependent Electron-Nuclear Wave Function.* Phys. Rev. Lett. 105, 123002 (2010); doi: 10.1103/PhysRevLett.105.123002

[2] Abedi, A., Maitra, N., Gross, E.K.U. *Correlated electron-nuclear dynamics: Exact factorization of the molecular wavefunction.* J. Chem. Phys. 137, 22A530 (2012); doi: 10.1063/1.4745836

[3] Deych, L. *Advanced Undergraduate Quantum Mechanics.* Cham, Switzerland: Springer, 2018.

[4] Gossel, G., Lacombe, L., Maitra., N. *On the Numerical Solution of the Exact Factorization Equations.* J. Chem. Phys. 150, 154112 (2019); doi: 10.1063/1.5090802

[5] Hagedorn, G., Toloza, J. *Exponentially Accurate Quasimodes for the Time-Independent Born–Oppenheimer Approximation on a One-Dimensional Molecular System.* Int. J. Quantum Chem. 105 (2005); doi: 10.1002/qua.20730

[6] Iserles, A. *A First Course in the Numerical Analysis of Differential Equations.* Cambridge University Press. 1996.

[7] Kelley, C.T. *Solving Nonlinear Equations with Newton's Method.* Philadelphia: SIAM. 2003.

[8] Lorin, E. *Numerical analysis of the exact factorization of molecular time-dependent Schrödinger wavefunctions.* Commun. Nonlinear Sci. Numer. Simul. 95 (2021); doi: 10.1016/j.cnsns.2020.105627

[9] Martinez, P., Rosenzweig, B., Hoffmann, N., Lacombe, L., Maitra, N. *Case studies of the time-dependent potential energy surface for dynamics in cavities.* J Chem Phys 154, 014102 (2021); doi: 10.1063/5.0033386

[10] Strikwerda, J.C. *Finite Difference Schemes and Partial Differential Equations.* Philadelphia: Society for Industrial and Applied Mathematics, 2004.

[11] Tully, J.C. *Perspective on "Zur Quantentheorie der Molekeln" Born M., Oppenheimer R. (1927) Ann. Phys. 84: 457.* Theor. Chem. Acc. 103: (2000); doi: 10.1007/s002149900049

A Python Codes

A.1 Newton's Method for Free Schrödinger Equation

A.1.1 Trapezoidal Method

```
import numpy as np
#parameters
delX = 0.1
delT = 0.01
Tmax = 1000
Xmin = -10
Xmax = 10
NewtonIters = 10
Tgrid = np.arange(0,Tmax+delT,delT)
Xgrid = np.arange(Xmin,Xmax,delX)

lam = delT / delX**2
A = [[0,1],[-1,0]]

def F(Cf,Ci):
    f = np.zeros(np.shape(Cf))
    for j in range(int(len(Cf)/2)):
        if j > 0 and j < int(len(Cf)/2)-1:
            f[2*j] += Cf[2*j]+(-1)*Ci[2*j]+0.5*((-lam)*Cf[2*j+3]+2*lam*Cf[2*j+1]
                +(-lam)*Cf[2*j-1]+(-lam)*Ci[2*j+3]+2*lam*Ci[2*j+1]+(-lam)*Ci[2*j-1])
            f[2*j+1] += Cf[2*j+1]+(-1)*Ci[2*j+1]+0.5*((lam)*Cf[2*j+2]-2*lam*Cf[2*j]
                +(lam)*Cf[2*j-2]+(lam)*Ci[2*j+2]-2*lam*Ci[2*j]+(lam)*Ci[2*j-2])
        elif j == 0:
```

```
            f[2*j] += Cf[2*j]+(-1)*Ci[2*j]+0.5*((-lam)*Cf[2*j+3]+2*lam*Cf[2*j+1]
                +(-lam)*Cf[2*j+3]+(-lam)*Ci[2*j+3]+2*lam*Ci[2*j+1]+(-lam)*Ci[2*j+3])
            f[2*j+1] += Cf[2*j+1]+(-1)*Ci[2*j+1]+0.5*((lam)*Cf[2*j+2]-2*lam*Cf[2*j]
                +(lam)*Cf[2*j+2]+(lam)*Ci[2*j+2]-2*lam*Ci[2*j]+(lam)*Ci[2*j+2])
        elif j == int(len(Cf)/2)-1:
            f[2*j] += Cf[2*j]+(-1)*Ci[2*j]+0.5*((-lam)*Cf[2*j-1]+2*lam*Cf[2*j+1]
                +(-lam)*Cf[2*j-1]+(-lam)*Ci[2*j-1]+2*lam*Ci[2*j+1]+(-lam)*Ci[2*j-1])
            f[2*j+1] += Cf[2*j+1]+(-1)*Ci[2*j+1]+0.5*((lam)*Cf[2*j-2]-2*lam*Cf[2*j]
                +(lam)*Cf[2*j-2]+(lam)*Ci[2*j-2]-2*lam*Ci[2*j]+(lam)*Ci[2*j-2])
    return f


def Newton(Cf,Ci):
    rhs = [-f for f in F(Cf,Ci)]
    Jacobian = np.zeros((len(Cf),len(Cf)))
    for j in range(int(len(Cf)/2)):
        if j > 0 and j < int(len(Cf)/2)-1:
            Jacobian[2*j,2*j-1] += -0.5*lam
            Jacobian[2*j,2*j] += 1
            Jacobian[2*j,2*j+1] += lam
            Jacobian[2*j,2*j+3] += -0.5*lam
            Jacobian[2*j+1,2*j-2] += 0.5*lam
            Jacobian[2*j+1,2*j] += -lam
            Jacobian[2*j+1,2*j+1] += 1
            Jacobian[2*j+1,2*j+2] += 0.5*lam
        elif j == 0:
            Jacobian[2*j,2*j] += 1
            Jacobian[2*j,2*j+1] += lam
```

```
            Jacobian[2*j,2*j+3] += -lam
            Jacobian[2*j+1,2*j] += -lam
            Jacobian[2*j+1,2*j+1] += 1
            Jacobian[2*j+1,2*j+2] += lam
        elif j == int(len(Cf)/2)-1:
            Jacobian[2*j,2*j-1] += -lam
            Jacobian[2*j,2*j] += 1
            Jacobian[2*j,2*j+1] += lam
            Jacobian[2*j+1,2*j-2] += lam
            Jacobian[2*j+1,2*j] += -lam
            Jacobian[2*j+1,2*j+1] += 1
    return np.linalg.solve(Jacobian,rhs)

#run dynamics
chi = np.zeros((len(Tgrid),len(Xgrid)*2))

for (i,x) in enumerate(Xgrid):
    chi[0,2*i] += np.exp(-x**2/2)
    chi[0,2*i+1] += 0

for (i,t) in enumerate(Tgrid[:-1]):
    chiIntermed = [chi[i,:]]
    for k in range(NewtonIters):
        stepGuess = Newton(chiIntermed[k],chi[i,:])
        chiIntermed.append(chiIntermed[k]+stepGuess)
    chi[i+1,:] += chiIntermed[-1]
```

A.1.2 Backward Euler Method

```
    def F(Cf,Ci):
    f = np.zeros(np.shape(Cf))
    for j in range(int(len(Cf)/2)):
        if j > 0 and j < int(len(Cf)/2)-1:
            f[2*j] += Cf[2*j]+(-1)*Ci[2*j]+(-lam)*Cf[2*j+3]+2*lam*Cf[2*j+1]
                +(-lam)*Cf[2*j-1]
            f[2*j+1] += Cf[2*j+1]+(-1)*Ci[2*j+1]
                +(lam)*Cf[2*j+2]-2*lam*Cf[2*j]+(lam)*Cf[2*j-2]
        elif j == 0:
            f[2*j] += Cf[2*j]+(-1)*Ci[2*j]+(-lam)*Cf[2*j+3]+2*lam*Cf[2*j+1]
                +(-lam)*Cf[2*j+3]
            f[2*j+1] += Cf[2*j+1]+(-1)*Ci[2*j+1]+(lam)*Cf[2*j+2]-2*lam*Cf[2*j]
                +(lam)*Cf[2*j+2]
        elif j == int(len(Cf)/2)-1:
            f[2*j] += Cf[2*j]+(-1)*Ci[2*j]+(-lam)*Cf[2*j-1]
                +2*lam*Cf[2*j+1]+(-lam)*Cf[2*j-1]
            f[2*j+1] += Cf[2*j+1]+(-1)*Ci[2*j+1]
                +(lam)*Cf[2*j-2]-2*lam*Cf[2*j]+(lam)*Cf[2*j-2]
    return f


def Newton(Cf,Ci):
    rhs = [-f for f in F(Cf,Ci)]
    Jacobian = np.zeros((len(Cf),len(Cf)))
    for j in range(int(len(Cf)/2)):
        if j > 0 and j < int(len(Cf)/2)-1:
            Jacobian[2*j,2*j-1] += -lam
```

```
            Jacobian[2*j,2*j] += 1
            Jacobian[2*j,2*j+1] += 2*lam
            Jacobian[2*j,2*j+3] += -lam
            Jacobian[2*j+1,2*j-2] += lam
            Jacobian[2*j+1,2*j] += -2*lam
            Jacobian[2*j+1,2*j+1] += 1
            Jacobian[2*j+1,2*j+2] += lam
        elif j == 0:
            Jacobian[2*j,2*j] += 1
            Jacobian[2*j,2*j+1] += 2*lam
            Jacobian[2*j,2*j+3] += -2*lam
            Jacobian[2*j+1,2*j] += -2*lam
            Jacobian[2*j+1,2*j+1] += 1
            Jacobian[2*j+1,2*j+2] += 2*lam
        elif j == int(len(Cf)/2)-1:
            Jacobian[2*j,2*j-1] += -2*lam
            Jacobian[2*j,2*j] += 1
            Jacobian[2*j,2*j+1] += 2*lam
            Jacobian[2*j+1,2*j-2] += 2*lam
            Jacobian[2*j+1,2*j] += -2*lam
            Jacobian[2*j+1,2*j+1] += 1
    return np.linalg.solve(Jacobian,rhs)

#same script to run dynamics as above
```

A.2 Newton's Method for Schrödinger Equation with Pure Imaginary Advection

```
lam = delT / delX**2

lam2 = delT / delX


b = 0


def F(Cf,Ci):
    f = np.zeros(np.shape(Cf))
    for j in range(int(len(Cf)/2)):
        if j > 0 and j < int(len(Cf)/2)-1:
            f[2*j] += Cf[2*j]+(-1)*Ci[2*j]+(-0.5)*lam*Cf[2*j+3]+(lam)*Cf[2*j+1]
                +(-0.5*lam)*Cf[2*j-1]+(-0.5*lam)*Ci[2*j+3]+(lam)*Ci[2*j+1]
                +(-0.5*lam)*Ci[2*j-1]+(-0.25*b*lam2)*Cf[2*j+2]+(0.25*b*lam2)*Cf[2*j-2]
                +(-0.25*b*lam2)*Ci[2*j+2]+(0.25*b*lam2)*Ci[2*j-2]
            f[2*j+1] += Cf[2*j+1]+(-1)*Ci[2*j+1]+(0.5*lam)*Cf[2*j+2]+(-lam)*Cf[2*j]
                +(0.5*lam)*Cf[2*j-2]+(0.5*lam)*Ci[2*j+2]+(-lam)*Ci[2*j]
                +(0.5*lam)*Ci[2*j-2]+(-0.25*b*lam2)*Cf[2*j+3]+(0.25*b*lam2)*Cf[2*j-1]
                +(-0.25*b*lam2)*Ci[2*j+3]+(0.25*b*lam2)*Ci[2*j-1]
        elif j == 0:
            f[2*j] += Cf[2*j]+(-1)*Ci[2*j]+ -0.5*lam*Cf[2*j+3]+(lam)*Cf[2*j+1]
                +(-0.5*lam)*Cf[2*j+3]+(-0.5*lam)*Ci[2*j+3]+(lam)*Ci[2*j+1]
                +(-0.5*lam)*Ci[2*j+3]
            f[2*j+1] += Cf[2*j+1]+(-1)*Ci[2*j+1]+(0.5*lam)*Cf[2*j+2]
                +(-lam)*Cf[2*j]+(0.5*lam)*Cf[2*j+2]+(0.5*lam)*Ci[2*j+2]
                +(-lam)*Ci[2*j]+(0.5*lam)*Ci[2*j+2]
        elif j == int(len(Cf)/2)-1:
```

```
            f[2*j] += Cf[2*j]+(-1)*Ci[2*j]-0.5*lam*Cf[2*j-1]+ (lam)*Cf[2*j+1]
                +(-0.5*lam)*Cf[2*j-1]+(-0.5*lam)*Ci[2*j-1]+(lam)*Ci[2*j+1]
                +(-0.5*lam)*Ci[2*j-1]
            f[2*j+1] += Cf[2*j+1]+(-1)*Ci[2*j+1]+(0.5*lam)*Cf[2*j-2]
                +(-lam)*Cf[2*j]+(0.5*lam)*Cf[2*j-2]+(0.5*lam)*Ci[2*j-2]+
                (-lam)*Ci[2*j]+(0.5*lam)*Ci[2*j-2]
    return f


def J(Cf):
    Jacobian = np.zeros((len(Cf),len(Cf)))
    for j in range(int(len(Cf)/2)):
        if j > 0 and j < int(len(Cf)/2)-1:
            Jacobian[2*j,2*j-2] += 0.25*b*lam2
            Jacobian[2*j,2*j] += 1
            Jacobian[2*j,2*j+1] += lam
            Jacobian[2*j,2*j+2] += -0.25*b*lam2
            Jacobian[2*j,2*j+3] += -0.5*lam
            Jacobian[2*j+1,2*j-2] += 0.5*(lam)
            Jacobian[2*j+1,2*j-1] += 0.25*b*lam2
            Jacobian[2*j+1,2*j] += -lam
            Jacobian[2*j+1,2*j+1] += 1
            Jacobian[2*j+1,2*j+2] += 0.5*lam
            Jacobian[2*j+1,2*j+3] += -0.25*b*lam2
        elif j == 0:
            Jacobian[2*j,2*j] += 1
            Jacobian[2*j,2*j+1] += lam
            Jacobian[2*j,2*j+3] += -lam
```

```
            Jacobian[2*j+1,2*j] += -lam
            Jacobian[2*j+1,2*j+1] += 1
            Jacobian[2*j+1,2*j+2] += lam
        elif j == int(len(Cf)/2)-1:
            Jacobian[2*j,2*j-1] += -lam
            Jacobian[2*j,2*j] += 1
            Jacobian[2*j,2*j+1] += lam
            Jacobian[2*j+1,2*j-2] += lam
            Jacobian[2*j+1,2*j] += -lam
            Jacobian[2*j+1,2*j+1] += 1
    return Jacobian
```