Accelerating Electron Dynamics Simulations through Machine Learned Time Propagators

Karan Shah [1 2] **Attila Cangi** [1 2]

Abstract

Time-dependent density functional theory (TDDFT) is a widely used method to investigate electron dynamics under various external perturbations such as laser fields. In this work, we present a novel approach to accelerate real time TDDFT based electron dynamics simulations using autoregressive neural operators as time-propagators for the electron density. By leveraging physics-informed constraints and high-resolution training data, our model achieves superior accuracy and computational speed compared to traditional numerical solvers. We demonstrate the effectiveness of our model on a class of one-dimensional diatomic molecules. This method has potential in enabling real-time, on-the-fly modeling of laser-irradiated molecules and materials with varying experimental parameters.

1. Introduction

Time-Dependent Density Functional Theory (TDDFT) (Runge & Gross, 1984) is a widely used method to study the evolution of electronic structure under the influence of time-dependent potentials. It is used to calculate various excited state properties such as excitation energies (Adamo & Jacquemin, 2013), charge transfer (Maitra, 2017), stopping power (Yost et al., 2017), optical absorption spectra (Jacquemin et al., 2011) and non-linear optical properties (Goncharov, 2014). Due to its favorable balance between accuracy and computational cost, TDDFT has been applied in various domains including photocatalysis, biochemistry, nanoscale devices and the study of light-matter interactions in general.

For weak perturbations, the linear response (LR) formalism

[1]Center for Advanced Systems Understanding, Görlitz, Germany [2]Helmholtz-Zentrum Dresden-Rossendorf, Dresden, Germany. Correspondence to: Attila Cangi <a.cangi@hzdr.de>.

of TDDFT is used to calculate the excitation spectrum of a system. It is calculated using the Casida equation (Casida, 1995) as an eigenvalue problem that describes the first order response of the density. In contrast, the electron density is directly propagated in time under the real time (RT) formalism. RTTDFT can be used to calculate the nonlinear response of the density under strong perturbations such as ultrafast electron dynamics with strong laser fields.

There are multiple components that must be decided for RTTDDFT calculations. These include preparation of the initial density and orbitals, choice of the exchange-correlation functional, the form of the time-dependent potential and the choice of the time-propagation scheme. The time evolution is a significant fraction of the computation required (Castro et al., 2004; Gómez Pueyo et al., 2018).

Machine learning (ML) for accelerating scientific simulations is a rapidly growing area of research (Carleo et al., 2019). A variety of unsupervised and data-driven models have been developed to solve differential equations across a wide range of domains (Karniadakis et al., 2021).

Neural operators (NOs) (Kovachki et al., 2023) are a class of models that map function-to-function spaces, as opposed to the finite-dimensional vector space mappings of neural networks. This is especially useful for partial differential equation (PDE) problems where experimental or simulation data is available. Fourier neural operators (FNOs) (Li et al., 2020) are a type of NO which represent operator weights in Fourier space. The main advantages of FNOs are that they generalize well across function spaces, and being resolution-invariant, they can be used for inference on higher resolution grids than the training set grids. FNOs have also been used for forward and inverse PDE problems in various domains (Azizzadenesheli et al., 2024).

While many applications of ML have been developed for ground state DFT (Snyder et al., 2012; Brockherde et al., 2017; Fiedler et al., 2022), there have been relatively fewer efforts focused on TDDFT. Some applications of ML for TDDFT include development of exchange-correlation potentials (Yang & Whitfield, 2023; Suzuki et al., 2020) and predicting properties such as spectra and stopping power (Ward et al., 2023). In this work, we demonstrate the ef-

fectiveness of FNOs in propagating the electron density in time under the TDDFT framework. Instead of propagating orbitals in time as done conventionally in terms of the time-dependent Kohn-Sham equations, we use the FNO propagator to directly evolve the density. This has two advantages: the computational cost does not scale with the number of orbitals and larger propagation time steps can be used, thus using fewer iterations. Section 2 contains a brief description of TDDFT, FNOs and our proposed autoregressive model. Section 3 contains results obtained with this model for the time evolution of one-dimensional diatomic molecules under the influence of an oscillating laser pulse. We show that the model can be generalized across ionic configurations, is faster and more accurate than comparable numerical simulations and can also be used for higher resolution grids. In Section 4, we discuss the physical viability of the predicted densities, and then offer conclusion in Section 5.

We use atomic units a.u. ($\hbar = m_e = e = 1$) unless specified otherwise.

2. Methods

2.1. Time-Dependent Density Functional Theory

The time-dependent many-body Schrödinger equation describes the dynamics of systems composed of multiple interacting particles. A common problem is interacting electrons in atoms and molecules, for which the Schrödinger equation is

$$i\frac{\partial}{\partial t}\Psi(\underline{r}, t) = \hat{H}(t)\Psi(\underline{r}, t), \tag{1}$$

where $\underline{r}$ represents the collective coordinates of the N electrons, $\Psi(\underline{r}, t)$ is the many-body wavefunction and $\hat{H}$ is the Hamiltonian operator, representing the total energy of they system. For this problem the form of $\hat{H}$ is $\hat{H}(t) = \hat{T} + \hat{W} + \hat{V}(t)$, where $\hat{T} = \sum_{j=1}^{N} -\nabla_j^2/2$ is the kinetic energy operator, $\hat{W} = \frac{1}{2}\sum_{\substack{j,k \\ j \neq k}}^{N} 1/|\mathbf{r}_j - \mathbf{r}_k|$ represents the electron-electron interaction, and $\hat{V}(t) = v_{\text{ext}}(\mathbf{r}, t)$ is the time-dependent external potential operator which includes the static external potential due to the ions and an external time-dependent potential that drives the system. Note that in writing this Schrödinger equation we are assuming the Born-Oppenheimer approximation, i.e. we are treating the dynamics of the ions as occurring on a much longer time scale than the dynamics of the electrons, and can therefore treat the ions as classical point-like particles. The initial state $\Psi(\underline{r}) = \Psi(\underline{r}, t_0)$ is obtained by solving the time-independent Schrödinger equation

$$\hat{H}\Psi(\underline{r}) = E\Psi(\underline{r}) \tag{2}$$

which is an eigenvalue problem where E denotes the eigenvalues and corresponds to the total energy of the system.

The many-body wavefunction Ψ contains all the information about the system and can be used to calculate its properties. However for 3 spatial dimensions, we need to solve a system of $3N$ variables and the computational cost scales exponentially with N, rendering it intractable for all but the simplest systems. DFT and TDDFT allow us to solve Eq. 2 and Eq. 1 respectively by reformulating the problem in terms of electron density rather than the wavefunction, dramatically reducing the number of variables from $3N$ to 3.

The Runge-Gross theorem (Runge & Gross, 1984) states that for a system with a given ground state many-body wavefunction $\Psi_0 = \Psi_0(\mathbf{r}, t_0)$, there exists a unique mapping between the potential and the time-dependent density. The density can be obtained by solving a system of fictitious non-interacting particles governed by the time-dependent Kohn-Sham (TDKS) equations (van Leeuwen, 1998)

$$\left[-\frac{1}{2}\nabla^2 + v_s[n](\mathbf{r}, t)\right]\phi_j(\mathbf{r}, t) = i\frac{\partial\phi_j(\mathbf{r}, t)}{\partial t}, j = 1, ..., N \tag{3}$$

where the electron density

$$n(\mathbf{r}, t) = \sum_j |\phi_j(\mathbf{r}, t)|^2, j = 1, ..., N \tag{4}$$

is the quantity of interest. The Kohn-Sham potential $v_s[n](\mathbf{r}, t)$ is a functional of the density n which is a function of position $\mathbf{r}$ and time t. The correspondence between density and potential is established through the Kohn-Sham potential

$$v_{\text{S}}[n](\mathbf{r}, t) = v_{\text{ext}}(\mathbf{r}, t) + v_{\text{H}}[n](\mathbf{r}, t) + v_{\text{XC}}[n](\mathbf{r}, t). \tag{5}$$

where $v_{\text{ext}}(\mathbf{r}, t)$ is the external potential, $v_{\text{H}}(\mathbf{r}, t)$ is the Hartree potential and $v_{\text{XC}}(\mathbf{r}, t)$ is the exchange-correlation potential. The external potential is composed of the potential of the ions and any time-dependent external perturbation such as a laser field.

While in theory, $v_{\text{XC}}(\mathbf{r}, t)$ depends on the density at all previous time steps, the adiabatic approximation is often used in practice (Ullrich, 2012). We use the adiabatic local density approximation (ALDA). Under ALDA, $v_{\text{XC}}(\mathbf{r}, t)$ is approximated as:

$$v_{\text{XC}}^{\text{ALDA}}[n, \Psi_0, \Phi_0](\mathbf{r}, t) = v_{\text{XC}}^{\text{LDA}}[n](\mathbf{r}, t), \tag{6}$$

where $v_{\text{XC}}^{\text{LDA}}$ is the LDA exchange-correlation potential in ground-state DFT.

These coupled equations are solved through iterative numerical algorithms which represent a significant computational cost in the TDDFT workflow (Castro et al., 2004).
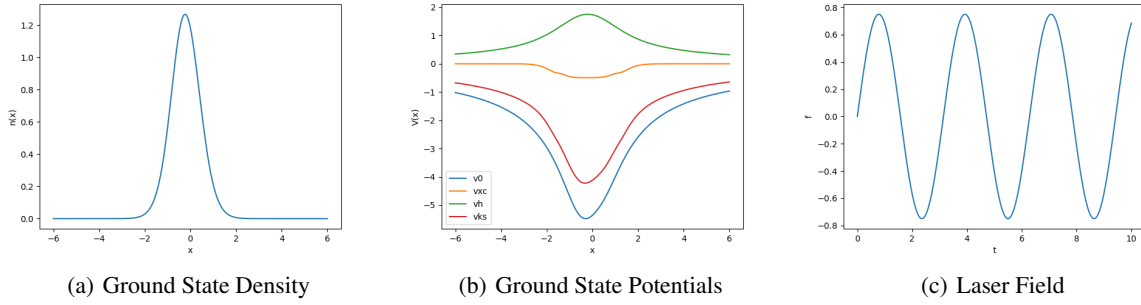
(a) Ground State Density   (b) Ground State Potentials   (c) Laser Field

Figure 1. Example system with $d = 1.0$, $Z_1 = 2.0$, $Z_2 = 4.0$. (a) Ground State Density, (b) Ground State Potentials, (c) Laser Field

2.1.1. TIME PROPAGATORS

The general form of the time evolution operator $\hat{U}$ for a time domain T is given by:

$$\phi_i(\mathbf{r}, T) = \hat{U}(T, 0)\phi_i(\mathbf{r}, 0). \tag{7}$$

In practice, it is applied multiple times for shorter time periods over the domain

$$\hat{U}(T, 0) = \prod_{i=0}^{N-1} \hat{U}(t_i + \Delta t_i, t_i) \tag{8}$$

with $t_0 = 0$, $t_{i+1} = t_i + \Delta t_i$, and $t_N = T$. Additionally, the evolution operator is unitary $\hat{U}^\dagger(t+\Delta t, t) = \hat{U}^{-1}(t+\Delta t, t)$, which is required for conserving the electron density, and follows time-reversal symmetry $\hat{U}(t+\Delta t, t) = \hat{U}^{-1}(t, t+\Delta t)$. A time propagation algorithm should account for these properties. Multiple time propagation methods have been developed for this evolution (Castro et al., 2004; Gómez Pueyo et al., 2018). For this work, we use the Crank-Nicholson method (Crank & Nicolson, 1947) to generate the reference data. We then train a FNO to directly propagate the density in time as opposed to propagating single particle orbitals.

2.1.2. MODEL SYSTEMS

We consider an external potential

$$v_{\text{ext}}(\mathbf{r}, t) = v_{\text{ion}}(\mathbf{r}) + v_{\text{las}}(t), \tag{9}$$

where we simulate a class of one-dimensional diatomic molecules that serves as a model system with two interacting electrons under the static ionic potential:

$$v_{\text{ion}}(\mathbf{r}) = -\frac{Z_1}{\sqrt{(\mathbf{r} - \frac{d}{2})^2 + a^2}} - \frac{Z_2}{\sqrt{(\mathbf{r} + \frac{d}{2})^2 + a^2}}, \tag{10}$$

where Z_1 and Z_2 denote the charge of the atomic wells, d denotes the bond length, and a is a softening parameter for numerical stability.

The system is excited with a laser given by a time-dependent external perturbing potential:

$$v_{\text{las}}(t) = A \sin \omega t. \tag{11}$$

With the dipole approximation, we can treat the laser as spatially constant because the size of our molecule is much small than the wavelength of the laser. Given the ground state prepared with $v_{\text{ion}}(\mathbf{r})$, we propagate the density under the influence of the laser $v_{\text{las}}(t)$.

The domain is defined by [-L, L] for time [0, T] with discretization Δx and Δt. We use fixed boundary conditions

$$\phi_i(-L, t) = \phi_i(L, t) = 0, \quad \forall t \in [0, T]. \tag{12}$$

The ground state density, ground state potential terms and shape of the laser pulse are shown in in Figure 1. We use the LDA functional due to its simplicity and accuracy for such simple one-dimensional systems. The density evolution is calculated by solving the time-dependent Kohn-Sham equations with the external potential $v_{\text{ext}}(r, t)$ using the Crank-Nicholson scheme. The numerical simulation is performed using the Octopus RT-TDDFT code (Tancogne-Dejean et al., 2020).

2.2. Fourier neural operators

Neural operators (NOs) (Lu et al., 2019; Anandkumar et al., 2020) extend neural networks by mapping functions to functions instead of finite-dimensional vectors. While a neural network maps $\mathbb{R}^n$ to $\mathbb{R}^m$, a neural operator maps function spaces, $G : \mathcal{A} \to \mathcal{U}$. Our goal is to approximate the nonlinear map $\mathcal{G}^\dagger : \mathcal{A} \to \mathcal{U}$ with a neural operator $\mathcal{G}_\theta$, parameterized by $\theta \in \mathbb{R}^p$.

Training involves observations $\{(a_i, u_i)\}_{i=1}^N$ where $u_i = \mathcal{G}^\dagger(a_i)$. The objective is to find parameters θ^* minimizing the loss:

$$\theta^* = \min_{\theta \in \mathbb{R}^p} \frac{1}{N} \sum_{i=1}^N \|u_i - \mathcal{G}_\theta(a_i)\|_{\mathcal{U}}^2. \tag{13}$$
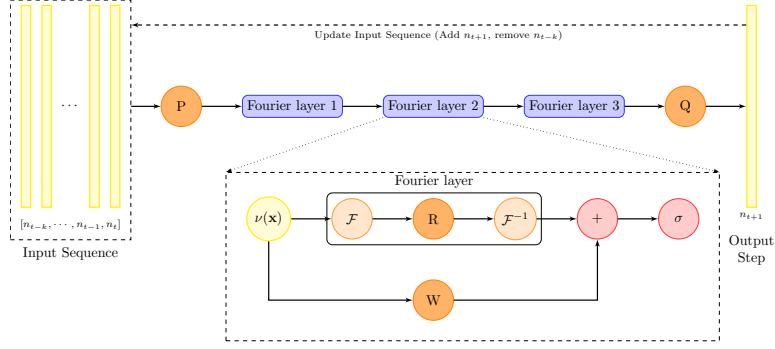
Figure 2. Autoregressive FNO architecture for predicting the density at time $t + 1$ based on k previous time steps.

A neural operator $\mathcal{G}_\theta(a)$ is defined by:

$$\mathcal{G}_\theta(a) = Q(v_L(v_{L-1}(\dots v_1(P(a))))), \qquad (14)$$

with layers v_{l+1} as:

$$v_{l+1}(x) = \sigma_{(l+1)}\left(W_l v_l(x) + \left(\mathcal{K}_l(a; \lambda)v_l\right)(x)\right), \quad (15)$$

where $\mathcal{K}_l$ is a non-local kernel integral operator:

$$\left(\mathcal{K}_l(a; \lambda)v_l\right)(x) = \int_{\Omega_l} \kappa_l(x, y, a(x), a(y); \lambda)v_l(y)dy. \tag{16}$$

The kernel function $\kappa_l(x, y, a(x), a(y))$ depends on $x, y, a(x)$, and $a(y)$, and is parameterized by λ. W_l is a learnable weights matrix which corresponds to a linear transformation and σ_{l+1} is a component-wise non-linear activation function. $P(a)$ preprocesses a into higher dimensional space v_0, and $Q(v_L)$ post-processes v_L back into $\mathcal{U}$.

FNOs (Li et al., 2020) use the Fourier transform for efficient computation. With $\kappa(x - y)$ as a convolution operator, we use the convolution theorem:

$$\left(\mathcal{K}_l(a; \lambda)v_l\right)(x) = \mathcal{F}^{-1}\left(\mathcal{F}\left(\kappa_l\right) \cdot \mathcal{F}\left(v_l\right)\right)(x), \quad (17)$$

where $\mathcal{F}$ and $\mathcal{F}^{-1}$ are the Fourier transform and its inverse, respectively. Parameterization in Fourier space is determined by the number of modes k_{max} and width of the convolutional layers, allowing efficient kernel computation and effective capture of global patterns using Fast Fourier Transform.

2.3. Modeling the time-dependent electron density with Fourier neural operators

An autoregressive model is used to predict future values based on past observations. The model takes a slice of T_{in} time steps and predicts the next time step. The input sequence is then updated to include the new predicted value while discarding the oldest time step.

Let $\mathbf{n}_t$ be the density grid at time step t. The input to the model at time step t is a sequence of T_{in} previous observations, denoted as:

$$\mathbf{N}_t = \left[\mathbf{n}_{t-T_{in}+1}, \mathbf{n}_{t-T_{in}+2}, \dots, \mathbf{n}_t\right], \qquad (18)$$

with $\mathbf{N}_0$ consisting of the initial T_{in} time steps beginning with the ground state density $\mathbf{n}_0$.

The model predicts the next time step $\hat{\mathbf{n}}_{t+1}$ as:

$$\hat{\mathbf{n}}_{t+1} = \mathcal{G}_\theta(\mathbf{N}_t) \qquad (19)$$

where $\mathcal{G}_\theta(\cdot)$ represents the autoregressive FNO model parameterized by θ.

After predicting $\hat{\mathbf{n}}_{t+1}$, the input sequence is updated for the next prediction. The new input sequence $\mathbf{N}_{t+1}$ is formed by appending $\hat{\mathbf{n}}_{t+1}$ and removing the oldest time step $\mathbf{n}_{t-T_{in}+1}$:

$$\mathbf{N}_{t+1} = \left[\mathbf{n}_{t-T_{in}+2}, \mathbf{n}_{t-T_{in}+3}, \dots, \mathbf{n}_t, \hat{\mathbf{n}}_{t+1}\right] \qquad (20)$$

This architecture is shown in Figure 2. This process is repeated iteratively to generate further predictions.

The loss function for this model is the mean squared error (MSE) between the predicted and true densities, averaged over all time steps and all systems. Let $\mathcal{D}$ denote the set of all systems, and let $\mathcal{T}$ denote the set of all predicted time steps for a given system. The loss function $\mathcal{L}(\theta)$ is defined as:

$$\mathcal{L}(\theta) = \frac{1}{|\mathcal{D}|} \sum_{d \in \mathcal{D}} \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} \left\|\mathbf{n}_t^{(d)} - \hat{\mathbf{n}}_t^{(d)}\right\|_2^2, \quad (21)$$

where $\mathbf{n}_t^{(d)}$ is the true density and $\hat{\mathbf{n}}_t^{(d)}$ is the predicted density at time step t for system d.

To ensure that the density is conserved through time, we calculate the norm of density

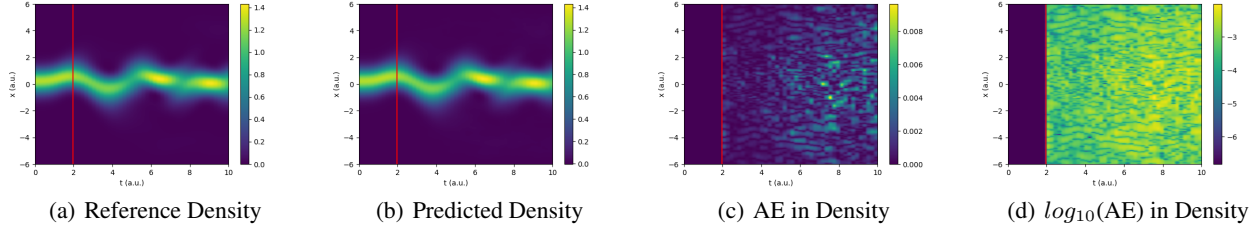$$N = \int_{-\infty}^{\infty} n(x, t)dx \approx \sum_i \hat{n}_{t,i}^{(d)} \Delta x \qquad (22)$$

4

(a) Reference Density   (b) Predicted Density   (c) AE in Density   (d) log_{10}(AE) in Density

Figure 3. Density evolution of the representative system. The red line denotes the initial $T_{in} = 10$ input time slices. The errors are calculated on the predicted density slices after the red line.

and add a loss term

$$\mathcal{L}_{norm}(\theta) = \lambda \left(\sum_i \hat{n}_{t,i}^{(d)} \Delta x - 2 \right)^2, \qquad (23)$$

where i denotes index over grid points x_i and λ is weight term. N = 2 for the reference system.

To evaluate model performance, we use the mean absolute error (MAE) and mean squared error (MSE) between the reference and predicted densities.

3. Results

As baseline, we generate a set of 729 systems with varying Z_1, Z_2, d parameters with fixed softening parameter $a = 1.0$. The domain parameters are $L = 6.0$ a.u. and $T = 10$ a.u. (0.241 fs). The discretization used is $\Delta x = 0.05$ a.u. and $\Delta t = 0.025$ a.u. (0.6 as). The laser parameters are $A = 0.75$ a.u. (corresponding to peak intensity $I = 1.97 \times 10^{16}$ W cm^{-2}) and $\omega = 2.0$ Ha (corresponding to $\lambda = 22.78$ nm). Using this reference set, we generate datasets with discretization $\Delta t = 0.2$ a.u. (4.83 as), resulting in $T_n = 51$ time slices, each on a grid of $X_n = 241$ spatial points. The training dataset consists of 600 systems, the validation dataset consists of 10 systems and performance is evaluated on a test dataset of 100 systems. The networks consist of 3 Fourier layers, trained using the Adam optimizer with learning rate = 0.001, halving every 200 epochs.

3.1. Effect of varying input time slice

We first show that by incorporating more information in the input time slices, the performance of the network increases. In this case, the network consists of layers with width 64 and 16 modes. Prediction metrics are evaluated on $T_n - T_{in}$ time slices. As the number of input time slices T_{in} increases, the errors decrease at the expense of the initial data required. We show the error metrics for four models with $T_{in} = 5, 10, 15, 20$ in Table 1. As a balance between the accuracy and data requirements, a larger model is created with $T_{in} = 10$, consisting of layers with 16 modes and width 128, trained for 2000 epochs. This model serves as

the baseline for further comparisons. The reference and predicted densities along with errors for a representative system are shown in Figure 3.

3.2. Comparison with coarse solver

We compare the accuracy and speed of the baseline model with a coarse solver. The baseline model is trained on a grid with $\Delta t = 0.2$ a.u., using data generated with a finer grid $\Delta t = 0.025$ a.u. Numerical results are calculated on a comparable coarse grid $\Delta t = 0.2$ a.u. in Octopus. Both these results are compared with reference fine grid numerical calculations. As show in Table 1, the baseline model is roughly $2\times$ faster than the numerical solver with error reductions of about $4\times$ MAE and $10\times$ MSE.

3.3. Time offset

We show that the model generalizes well when used to predict the density evolution on a time grid offset by $\frac{\Delta T}{2} = 0.1$ a.u. The domain in this case becomes $[0.1, T - 0.1]$. As show in Table 1, the MAE=5.779×10^{-3} and prediction time per step $t = 1.75$ ms is comparable to the performance of the baseline model, which was evaluated on the same time grid as the training time grid.

3.4. Spatial super-resolution

One key advantage of FNOs is discretization invariance. We show the performance of the model trained on a dataset with $\Delta x = 0.05$ a.u. ($X_n = 241$) on a higher resolution grid with $\Delta x = 0.025$ a.u. ($X_n = 481$). We get accurate predictions with a reasonable increase in errors but with the same inference speed, as shown in Table 1. This is especially useful because the time taken per step increases to 9 ms for the numerical solver on this grid while the model inference time stays constant at 1.75 ms.

4. Discussion

In this section, we summarize the numerical results and discuss the performance of the FNO time propagator for

Model	MAE ($\times 10^{-3}$)	MSE ($\times 10^{-4}$)	Time (ms)
FNO $T_{in} = 05$	8.595	4.18	1.58
FNO $T_{in} = 10$	7.087	4.10	1.62
FNO $T_{in} = 15$	6.101	3.16	1.60
FNO $T_{in} = 20$	5.652	3.32	1.60
Baseline	5.118	2.56	1.75
Octopus (Coarse)	19.616	24.69	4.00
Time Offset	5.779	2.78	1.75
Super-resolution	7.335	6.56	1.75

Table 1. Performance metrics of various models. The time column indicates the calculation/inference time to obtain the next density time slice.

calculating physically viable densities. For the baseline case, the MAE in density is 5.118×10^{-3} with a prediction time of 1.75 ms per time step. This is in excellent agreement to the reference data with a fraction of the computational cost. This model can be used for offset time grids and for higher-resolution space grids as well without additional training. To compare the nature of calculated densities, we evaluate the model on the properties of time evolution operators and use the density to calculate an observable.

4.1. Properties of time evolution operators

As discussed in 2.1.1, time propagators for the Kohn-Sham equations follow certain properties.

- Density conservation: The norm of the density $N = \int_{-\infty}^{\infty} n(x,t)dx$ must be conserved in the model system. For our system with two electrons, $N = 2$. For the baseline system, the FNO predicted densities have a mean norm of 1.99992 ± 0.00039, averaged over all predicted time steps over all systems. The deviation of the predicted norm increases with time, as shown in Figure 4.

- Time reversal symmetry: A stable time-propagator must follow time-reversal symmetry. As a qualitative check, densities are predicted using the baseline model in reverse time order. However, the MAE jumps to 0.0751, with even the MAE on the training set jumping to 0.0741.

The FNO model follows density conservation, which is also constrained in the loss function during training. However, time reversal symmetry is broken as shown by the increase in MAE for reverse time order predictions.

4.2. Dipole Moment Calculation

$$\mu(t) = \int_{-\infty}^{\infty} x\,n(x,t)\,dx \approx \sum_i x_i\,n_{t,i}\,\Delta x \quad (24)$$

To demonstrate that the model predicted densities can be used for calculating observables, we also compare the dipole moments calculated using predicted and reference densities. For the baseline, the dipole moment MSE is 0.02976 with a mean average percentage error of 8.34 %. The error in dipole moment accumulates across time, as shown in Figure 5.
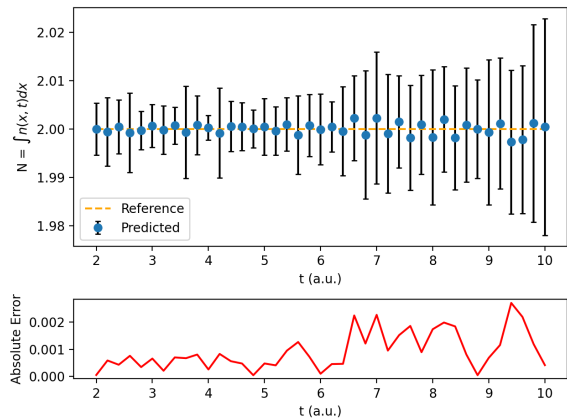

Figure 4. Evolution of predicted density norm over time. The blue points represent the norms of predicted density, averaged across all systems, at varying time steps with the error bars representing the stand deviation. The orange line represents the true norm. The subplot shows the absolute error between reference and predicted norm in time.
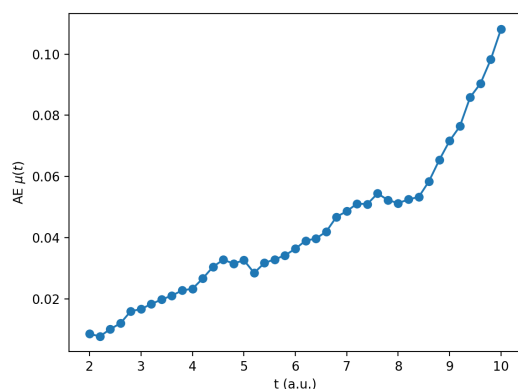
Figure 5. Absolute error between the reference and predicted dipole moment, averaged across all systems, at different time steps.

The performance of the FNO time propagator in calculating these physically relevant quantities shows that while density predictions are accurate, the model can be improved further by incorporating more physics-based constraints such as time reversal symmetry and stronger density conservation. Since the typical time scale for electron dynamics is in the range of a few hundred femtoseconds at most, achieving small error accumulation on this time scale seems feasible based on our results.

4.3. Conclusion

We show that machine learned time propagators have the potential to accelerate TDDFT calculations. A promising direction would be the development of ML time propagators that can predict the time evolution given just the ground state density and the shape of a laser pulse. Such extensions would include having the laser as a separate input for better generalization across laser parameters and to encode time reversal symmetry in the model structure. Extending this work to physics-informed three-dimensional models would enable on-the-fly modeling of the electronic response properties of laser-excited molecules and materials in various scattering experiments that are conducted at photon sources around the globe. This would enable fast simulations that generalize well over the input parameters of the experimental setup. Rapid modeling would also enable the design of laser pulses to precisely control quantum dynamics under quantum optimal control theory (Werschnik & Gross, 2007).

Acknowledgments

References

Adamo, C. and Jacquemin, D. The calculations of excited-state properties with Time-Dependent Density Functional Theory. *Chemical Society Reviews*, 42(3):845–856, January 2013. ISSN 1460-4744. doi: 10.1039/C2CS35394F. URL https://pubs.rsc.org/en/content/articlelanding/2013/cs/c2cs35394f. Publisher: The Royal Society of Chemistry.

Anandkumar, A., Azizzadenesheli, K., Bhattacharya, K., Kovachki, N., Li, Z., Liu, B., and Stuart, A. Neural Operator: Graph Kernel Network for Partial Differential Equations. In *ICLR 2020 Workshop on Integration of Deep Neural Models and Differential Equations*, February 2020.

Azizzadenesheli, K., Kovachki, N., Li, Z., Liu-Schiaffini, M., Kossaifi, J., and Anandkumar, A. Neural operators for accelerating scientific simulations and design. *Nature Reviews Physics*, 6(5):320–328, May 2024. ISSN 2522-5820. doi: 10.1038/s42254-024-00712-5. URL https://www.nature.com/articles/s42254-024-00712-5. Publisher: Nature Publishing Group.

Brockherde, F., Vogt, L., Li, L., Tuckerman, M. E., Burke, K., and Müller, K.-R. Bypassing the Kohn-Sham equations with machine learning. *Nature Communications*, 8 (1):872, October 2017. ISSN 2041-1723. doi: 10.1038/s41467-017-00839-3. URL https://www.nature.com/articles/s41467-017-00839-3. Publisher: Nature Publishing Group.

Carleo, G., Cirac, I., Cranmer, K., Daudet, L., Schuld, M., Tishby, N., Vogt-Maranto, L., and Zdeborová, L. Machine learning and the physical sciences. *Reviews of Modern Physics*, 91(4):045002, December 2019. doi: 10.1103/RevModPhys.91.045002. URL https://link.aps.org/doi/10.1103/RevModPhys.91.045002. Publisher: American Physical Society.

Casida, M. E. Time-Dependent Density Functional Response Theory for Molecules. In *Recent Advances in Computational Chemistry*, volume 1, pp. 155–192. WORLD SCIENTIFIC, November 1995. ISBN 978-981-02-2442-4 978-981-283-058-6. doi: 10.1142/9789812830586_0005. URL http://www.worldscientific.com/doi/abs/10.1142/9789812830586_0005.

Castro, A., Marques, M. A. L., and Rubio, A. Propagators for the time-dependent Kohn–Sham equations. *The Journal of Chemical Physics*, 121(8):3425–3433, August 2004. ISSN 0021-9606. doi: 10.1063/1.1774980. URL https://doi.org/10.1063/1.1774980.

Crank, J. and Nicolson, P. A practical method for numerical evaluation of solutions of partial differential equations of the heat-conduction type. *Mathematical Proceedings of the Cambridge Philosophical Society*, 43(1):50–67, 1947. doi: 10.1017/S0305004100023197.

Fiedler, L., Shah, K., Bussmann, M., and Cangi, A. Deep dive into machine learning density functional theory for materials science and chemistry. *Physical Review Materials*, 6(4):040301, April 2022. doi: 10.1103/PhysRevMaterials.6.040301. URL https://link.aps.org/doi/10.1103/PhysRevMaterials.6.040301. Publisher: American Physical Society.

Goncharov, V. *Non-Linear Optical Response in Atoms, Molecules and Clusters: An Explicit Time Dependent Density Functional Approach*. Springer, September 2014. ISBN 978-3-319-08320-9.

Gómez Pueyo, A., Marques, M. A. L., Rubio, A., and Castro, A. Propagators for the Time-Dependent Kohn–Sham Equations: Multistep, Runge–Kutta, Exponential Runge–Kutta, and Commutator Free Magnus Methods. *Journal of Chemical Theory and Computation*, 14(6):3040–3052, June 2018. ISSN 1549-9618. doi: 10.1021/acs.jctc.8b00197. URL https://doi.org/10.1021/acs.jctc.8b00197. Publisher: American Chemical Society.

Jacquemin, D., Mennucci, B., and Adamo, C. Excited-state calculations with TD-DFT: from benchmarks to simulations in complex environments. *Physical Chemistry Chemical Physics*, 13(38):16987–16998, 2011. doi: 10.1039/C1CP22144B. URL https://pubs.rsc.org/en/content/articlelanding/2011/cp/c1cp22144b. Publisher: Royal Society of Chemistry.

Karniadakis, G. E., Kevrekidis, I. G., Lu, L., Perdikaris, P., Wang, S., and Yang, L. Physics-informed machine learning. *Nature Reviews Physics*, 3(6):422–440, June 2021. ISSN 2522-5820. doi: 10.1038/s42254-021-00314-5.

Kovachki, N., Li, Z., Liu, B., Azizzadenesheli, K., Bhattacharya, K., Stuart, A., and Anandkumar, A. Neural Operator: Learning Maps Between Function Spaces With Applications to PDEs. *Journal of Machine Learning Research*, 24(89):1–97, 2023. ISSN 1533-7928. URL http://jmlr.org/papers/v24/21-1524.html.

Li, Z., Kovachki, N. B., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., and Anandkumar, A. Fourier Neural Operator for Parametric Partial Differential Equations. In *International Conference on Learning Representations*, October 2020.

Lu, L., Jin, P., and Karniadakis, G. E. DeepONet: Learning nonlinear operators for identifying differential equations based on the universal approximation theorem of operators. *CoRR*, abs/1910.03193, 2019.

Maitra, N. T. Charge transfer in time-dependent density functional theory. *Journal of Physics: Condensed Matter*, 29(42):423001, September 2017. ISSN 0953-8984. doi: 10.1088/1361-648X/aa836e. URL https://dx.doi.org/10.1088/1361-648X/aa836e. Publisher: IOP Publishing.

Runge, E. and Gross, E. K. U. Density-Functional Theory for Time-Dependent Systems. *Physical Review Letters*, 52(12):997–1000, March 1984. doi: 10.1103/PhysRevLett.52.997. URL https://link.aps.org/doi/10.1103/PhysRevLett.52.997. Publisher: American Physical Society.

Snyder, J. C., Rupp, M., Hansen, K., Müller, K.-R., and Burke, K. Finding Density Functionals with Machine Learning. *Physical Review Letters*, 108(25):253002, June 2012. doi: 10.1103/PhysRevLett.108.253002. URL https://link.aps.org/doi/10.1103/PhysRevLett.108.253002. Publisher: American Physical Society.

Suzuki, Y., Nagai, R., and Haruyama, J. Machine learning exchange-correlation potential in time-dependent density-functional theory. *Physical Review A*, 101(5):050501, May 2020. doi: 10.1103/PhysRevA.101.050501. URL https://link.aps.org/doi/10.1103/PhysRevA.101.050501. Publisher: American Physical Society.

Tancogne-Dejean, N., Oliveira, M. J. T., Andrade, X., Appel, H., Borca, C. H., Le Breton, G., Buchholz, F., Castro, A., Corni, S., Correa, A. A., De Giovannini, U., Delgado, A., Eich, F. G., Flick, J., Gil, G., Gomez, A., Helbig, N., Hübener, H., Jestädt, R., Jornet-Somoza, J., Larsen, A. H., Lebedeva, I. V., Lüders, M., Marques, M. A. L., Ohlmann, S. T., Pipolo, S., Rampp, M., Rozzi, C. A., Strubbe, D. A., Sato, S. A., Schäfer, C., Theophilou, I., Welden, A., and Rubio, A. Octopus, a computational framework for exploring light-driven phenomena and quantum dynamics in extended and finite systems. *The Journal of Chemical Physics*, 152(12):124119, March 2020. ISSN 0021-9606. doi: 10.1063/1.5142502. URL https://doi.org/10.1063/1.5142502.

Ullrich, C. *Time-Dependent Density-Functional Theory: Concepts and Applications*. Oxford Graduate Texts. Oxford University Press, Oxford, New York, February 2012. ISBN 978-0-19-956302-9.

van Leeuwen, R. Causality and Symmetry in Time-Dependent Density-Functional Theory. *Physical Review Letters*, 80(6):1280–1283, February 1998. doi: 10.1103/PhysRevLett.80.1280. URL https://link.aps.org/doi/10.1103/PhysRevLett.80.1280. Publisher: American Physical Society.

Ward, L., Blaiszik, B., Lee, C.-W., Martin, T., Foster, I., and Schleife, A. Accelerating Electronic Stopping Power Predictions by 10 Million Times with a Combination of Time-Dependent Density Functional Theory and Machine Learning, November 2023. URL http://arxiv.org/abs/2311.00787. arXiv:2311.00787 [cond-mat].

Werschnik, J. and Gross, E. K. U. Quantum optimal control theory. *Journal of Physics B: Atomic, Molecular and Optical Physics*, 40(18):R175, September 2007. ISSN 0953-4075. doi: 10.1088/0953-4075/40/18/R01. URL https://dx.doi.org/10.1088/0953-4075/40/18/R01.

Yang, J. and Whitfield, J. Machine-learning Kohn–Sham potential from dynamics in time-dependent Kohn–Sham systems. *Machine Learning: Science and Technology*, 4 (3):035022, August 2023. ISSN 2632-2153. doi: 10.1088/2632-2153/ace8f0. URL https://dx.doi.org/10.1088/2632-2153/ace8f0. Publisher: IOP Publishing.

Yost, D. C., Yao, Y., and Kanai, Y. Examining real-time time-dependent density functional theory nonequilibrium simulations for the calculation of electronic stopping power. *Physical Review B*, 96(11): 115134, September 2017. doi: 10.1103/PhysRevB.96.115134. URL https://link.aps.org/doi/10.1103/PhysRevB.96.115134. Publisher: American Physical Society.