Quasi-direct quantum molecular dynamics: The time-dependent adaptive density-guided approach for potential energy surface construction

Nicolai Machholdt Høyer* and Ove Christiansen

Department of Chemistry, Aarhus University, DK-8000 Aarhus C, Denmark

E-mail: hoyer@chem.au.dk

Abstract

We present a new quasi-direct quantum molecular dynamics computational method which offer a compromise between quantum dynamics using a pre-computed potential energy surface (PES) and fully direct quantum dynamics. This method is termed the time-dependent adaptive density-guided approach (TD-ADGA) and is a method for constructing a PES on the fly during a dynamics simulation. This is achieved by acquisition of new single point (SP) calculations and refitting of the PES depending on the need of the dynamics. The TD-ADGA is a further development of the adaptive density-guided approach (ADGA) for PES construction where the placement of SPs is guided by the density of the nuclear wave function. In TD-ADGA, the ADGA framework has been integrated into the time-propagation of the time-dependent nuclear wave function and we use the reduced one-mode density of this wave function to guide when and where new SPs are placed. The PES is thus extended or updated if the wave function moves into new areas or if a certain area becomes more important. We

here derive equations for the reduced one-mode density for the time-dependent Hartree (TDH) method and for multi configuration time-dependent Hartree (MCTDH) methods, but the TD-ADGA can be used with any time-dependent wave function method as long as a density is available. The TD-ADGA method has been investigated on molecular systems containing single- and double-minimum potentials and on single- and multi-mode systems. We explore different approaches to handle the fact that the TD-ADGA involves a PES that changes during the computation and show how results can be obtained that are in very good agreement with results obtained by using an accurate reference PES. Dynamics with TD-ADGA is essentially a black box procedure, where only the initialization of the system and how to compute SPs must be provided. The TD-ADGA thus makes it easier to carry out quantum molecular dynamics and the quasi-direct framework opens up the possibility to compute quantum dynamics accurately for larger molecular systems.

1 Introduction

Simulating the dynamics of molecular systems is key to understanding many processes of nature, as such simulations can give insights into the fundamental molecular motions and chemical reactions involved. Full quantum mechanical simulations of molecules have the promise of accurately revealing molecular dynamics from first principles, but high computational costs and complex workflows often make such computations prohibitive to apply in practice. There are two major challenges inherent to any quantum molecular dynamics computation i) how to represent and compute the potential energy surface (PES) that the dynamics takes place on and ii) how to represent and propagate accurate in time, the wave function for nuclear motion on the PES. These two problems are tightly connected as on one hand, the PES dictates how the wave function for nuclear motion evolves over time while on the other hand, it is difficult to know quantitatively which domains of the PES the wave function visits before carrying out an accurate dynamics simulation. Two opposite strategies

have been prevailing for dealing with the PES issue in time-dependent simulations.

In the first approach, the PES is constructed prior to the dynamics computation. This PES is then used to propagate the dynamics and no further electronic structure calculations need to be carried out. In this strategy the PES is often fitted to a set of electronic structure single points (SPs) for different molecular geometries of the system under investigation. The problem then becomes to determine which domains of the PES are relevant for the dynamics, in order to ensure sufficient accuracy of the PES representation for computing the dynamics at all necessary times. The computational cost of an electronic structure SP for a molecular system is non-trivial and in a fitted PES approach, the goal is thus to obtain a PES that is sufficiently accurate while requesting a minimal number of SPs. The strength of the fitted PES approach is that one obtains an analytical expression for the potential energy operator that can be expediently used in quantum simulations. Further, by fitting (or refitting) the PES it can be cast as a sum-over-products form which is optimal for the integral evaluations of the potential that is often required in quantum dynamics.[1–3] The drawback of this method is that one might end up spending a lot of time computing expensive electronic structure SPs for domains of the PES that are not relevant for the dynamics or that the dynamics turn out to run outside the domains properly covered by the SPs and the fitted PES. In the latter case the results of the dynamics will be non-physical or at least inaccurate. Also, to obtain an actual high quality fit representation is by no means trivial.

In the other approach, the required potential energy evaluations are carried out by calls to electronic structure calculations during the dynamics, thereby completely removing the need for a fitted PES. The big advantage of this *direct* dynamics approach, is that one is not limited by the availability of a fitted PES and the limitations in form of accuracy and domain it comes with. In classical molecular dynamics (MD) simulations, such an *ab initio* MD approach has long existed but is also known to be comparatively much more expensive than MD computations using fitted PESs or classical force fields. In the quantum dynamical context the problem is even more severe. In many quantum dynamical wave

function computations integrals of the potential energy times the wave function or the basis set are needed. The numerical computation of such integrals would in a naive implementation lead to an extreme amount of SP evaluations and be intractable. This can be somewhat mitigated by using a wave function with basis sets consisting of evolving Gaussians that uses either classical-like equations of motion and/or approximations for integral evaluations, requiring only low-order derivative computations. This allows the individual Gaussians to be propagated forward in time using only local PES information obtained from single point calculations Variants of direct quantum dynamical methods has been developed and applied in different contexts.[4–12] Still, the direct approach is inherently expensive.

In this work we follow an intermediate strategy which we may denote as quasi-direct. In this approach we construct a fitted PES, but this is done during the dynamics computation instead of before. The approach is direct in the sense that the PES is extended on the fly during the dynamics simulation with calls for new electronic structure SPs when required by the dynamics. We will utilize a PES fitted to SPs but we will only request new SPs and do refittings of the PES according to need. Compared to the approach with a pre-fitted PES, this will circumvent the unnecessary computation of SPs in domains that the wave function do not actually visit and if the molecular wave function moves towards domains not covered by the fitted PES, new SPs will be acquired so the dynamics can continue in a reliable manner. This will lead to a PES that is constructed for the specific problem and propagation of consideration. The PES is a product of the whole computation and can be reused in other computations, but the core idea is that the PES is constructed for the purpose of the dynamics computation. The quasi-direct framework is similar in spirit to some direct dynamics works on growing a PES or employing interpolation or machine learning (ML) methods such as Gaussian process regression (GPR) during the direct dynamics.[13,14] In future work we return to acceleration of the present methodology with ML. In this work we wish, however, to build up a physical motivated algorithm for constructing and updating the PES during time-dependent molecular quantum dynamics computations without noise and errors from ML.

The concrete realization of the general idea pursued here is what we term the time-dependent adaptive density-guided approach (TD-ADGA) for PES construction. The TD-ADGA is a time-dependent generalization of the adaptive density-guided approach (ADGA)[15–18] that has been applied for more than a decade for constructing PESs for anharmonic vibrational computations. Essential to the ADGA approach is the concept of adaptively constructing the PES. In ADGA SPs are added to a grid until certain convergence criteria are satisfied. These criteria involve the PES itself, but also involve the density of the wave function and targets the PES domains that are critical for the accuracy of the final predictions. The grid domains and grid mesh for the SPs are in the ADGA iteratively determined based on convergence of energy-like integrals over the PES and wave function densities. Furthermore, the *time-independent* ADGA method (denoted here TI-ADGA to spell out the difference to TD-ADGA) has been based on the mode-coupling expansion of the potential,[2] where the PES is expanded in a set mode-combinations (MCs) and by truncating the set of included MCs an approximate PES is obtained with potential for significant reductions in computational time. Truncation to include only terms coupling less than a given number n of modes gives rise to a standard approach known under many names, including the n-mode representation,[19–23] high-dimensional model representations (HDMR),[24] many-body expansion[25] and pair approximation.[26] It is the sub-PES for the different MCs that in the TI-ADGA PES are adaptively constructed using reduced one-mode densities from a vibrational self consistent field (VSCF) wave function[27–30] to guide the placement of SP calculations. These sub-PES are typically of sufficient low dimensionality to be comparatively easy to fit, and methods for using arbitrary one-mode fit functions in a sum of products approach are available.[16] In this manner the TI-ADGA PES is constructed in accordance with the need of the time-independent wave function, and the computational cost and accuracy can be controlled by a few logical variables and convergence thresholds.

In the TD-ADGA developed in the present work, the main idea is to grow the PES according to the needs of a wavefunction evolving in time. This means that the PES obtained using

the TD-ADGA method must be updated during the dynamics simulation to ensure similar accuracy at all times. We use similar convergence criteria as in the original TI-ADGA method but generalized and augmented with new convergence conditions tailored the time-dependent context. To generalize the TI-ADGA to the time-dependent case, a time-dependent wave function density is used in order to obtain the ADGA integrals involved in identifying needed SPs. We thus define the reduced one-mode densities for a general time-dependent nuclear wave function. We then use densities obtained with the time-dependent Hartree (TDH)[31] and multi configuration time-dependent Hartree (MCTDH) wave functions.[32,33] The TD-ADGA procedure can in general be used with any time-dependent wave function method as long as a wave function density can be formulated. In future woork, we expect to combine the methods with time-dependent vibrational coupled cluster methods.[34–36] In this work we exclusively consider single surface computations. The wave function may be initiated from a wave function obtained on another surface but the simulated time-dependent dynamics and PES construction is restricted to a single surface. Extension to multiple adiabatic or diabatic surfaces and non-adiabatic events are anticipated to be possible along similar lines but require significant extensions and are subjects of future research.

In Section 2 we will first introduce the mode-coupling expansion and define the reduced one-mode density for time-dependent wave function. We will then introduce the ADGA method and generalize this to the TD-ADGA method. In Section 3 the implementation of the TD-ADGA method in the Molecular Interactions, Dynamics And Simulations Chemistry Program Package (MidasCpp)[37] will be discussed in detail and in Section 4, we will present and discuss test results obtained using the TD-ADGA method. Finally, we will in Section 5 summarize our findings.

2 Theory

In quantum molecular dynamics we seek to solve the time-dependent Schrödinger equation for the nuclear wave function. Under the assumption of the Born-Oppenheimer approximation, the Hamiltonian describing the nuclear motion for a single electronic state is described by the kinetic energy of the nuclei and the potential in which they move

$$H = T + V. \tag{1}$$

Here T is the kinetic energy operator for the nuclei and V is the potential energy surface which is dependent on the nuclear geometry. The form of the kinetic energy operator depends on the choice of coordinates and is not the focus of this work. We will assume some set of coordinates has been chosen and the kinetic energy operator can be obtained exactly or at least sufficiently accurate within these coordinates. We will instead focus on the construction of the PES by fitting to electronic structure single point calculations.

In the following subsections, we will describe the theoretical background for the TD-ADGA. First, we will set up some preliminaries on the form of the PES representation and on the description of the time-dependent wave functions for nuclear motion and their reduced one-mode densities. We will then describe the time-independent ADGA method and generalize this to the time-dependent ADGA method, including how automatic decisions for PES updates are made and different options for updating the dynamics with an updated PES.

2.1 Representing the potential energy surface

2.1.1 Mode-coupling expansion of the potential energy surface

In order to reduce the prohibitive dimensionality of the PES, the PES is described using an expansion in mode-couplings including the by now standard n-mode expansion. In the

n-mode expansion, $V(\boldsymbol{q})$ is approximated by a series of potential energy terms of increasing dimensionality

$$V^{(1)}(\boldsymbol{q}), V^{(2)}(\boldsymbol{q}), \ldots, V^{(M)}(\boldsymbol{q}), \tag{2}$$

where the $V^{(1)}(\boldsymbol{q})$ potential is the sum of all one-mode uncoupled potentials in the system, while the $V^{(2)}(\boldsymbol{q})$ potential includes also the pairwise coupling of the modes. For the higher order potentials $V^{(k)}(\boldsymbol{q})$ up to k modes are coupled.[2] Each potential to a given order k consist of potential energy terms coupling a subset of the M modes in the system, where k coordinates are different from the reference value which we here set to zero

$$V^{m_1}(q_{m_1}) = V(0, \ldots, 0, q_{m_1}, 0, \ldots, 0), \tag{3}$$

$$V^{m_1,m_2}(q_{m_1}, q_{m_2}) = V(0, \ldots, 0, q_{m_1}, 0, \ldots, 0, q_{m_2}, 0, \ldots, 0), \tag{4}$$

and so on for higher order potentials. We denote a set of modes in a potential term a mode combination (MC) and the n-mode set of n unique modes are identified with the mode indexing vector $\boldsymbol{m}_n = (m_1, m_2, \ldots, m_n)$. The potential terms of a MC is symmetric under permutation of the mode indexing vector[38,39] so that $V^{m_1,m_2}(q_{m_1}, q_{m_2}) = V^{m_2,m_1}(q_{m_2}, q_{m_1})$. The complete potential surface is regained in the limit where all mode couplings are included $V(\boldsymbol{q}) = V^{\boldsymbol{m}_M}(\boldsymbol{q})$.

If all potential terms of different dimensionality were summed, the final potential energy would contain over-counting as terms of higher dimensionality, by construction, contain the potential energy terms of lower dimensionality. To avoid this over-counting, we introduce the bar-potentials[2] as the relevant sub-potentials

$$\bar{V}^{\boldsymbol{m}_n}(\boldsymbol{q}^{\boldsymbol{m}_n}) = S^{\boldsymbol{m}_n} \sum_{n'=1}^{n} (-1)^{(n-n')} \binom{n}{n'} V^{\boldsymbol{m}_{n'}}(\boldsymbol{q}^{\boldsymbol{m}_{n'}}). \tag{5}$$

Here $\binom{n}{n'}$ is a binomial coefficient and $S^{\boldsymbol{m}_n}$ is an operator that symmetrizes with respect to

the n indices in the mode indexing vector, for example

$$S^{m_1,m_2}V^{m_1,m_2}(q_{m_1},q_{m_2}) = \frac{1}{2}(V^{m_1,m_2}(q_{m_1},q_{m_2}) + V^{m_2,m_1}(q_{m_2},q_{m_1})). \tag{6}$$

Furthermore, the notation $\boldsymbol{q}^{\boldsymbol{m}_n}$ denotes simply the set of coordinates given by the mode indices in the MC $\boldsymbol{m}_n$, i.e. $(\boldsymbol{q}^{\boldsymbol{m}_n}) = (q_{m_1}, q_{m_2}, \cdots, q_{m_n})$.

The bar-potentials are constructed such that if any of the coordinates are equal to the reference value q_m^{ref}, the function vanishes

$$\bar{V}^{\boldsymbol{m}_n}(\dots, q_m = q_m^{\text{ref}}, \dots) = 0. \tag{7}$$

This feature of Eq. 7 becomes apparent if we as an example use Eq. 5 to determine the $\bar{V}^{m_1,m_2}(q_{m_1},q_{m_2})$ potential

$$\bar{V}^{m_1,m_2}(q_{m_1},q_{m_2}) = V^{m_1,m_2}(q_{m_1},q_{m_2}) - V^{m_1}(q_{m_1}) - V^{m_2}(q_{m_2}). \tag{8}$$

From Eq. 8 it is easily seen that the $\bar{V}^{m_1,m_2}(q_{m_1},q_{m_2})$ potential is simply the $V^{m_1,m_2}(q_{m_1},q_{m_2})$ potential with the uncoupled one-mode potentials $V^{m_1}(q_{m_1})$ and $V^{m_2}(q_{m_2})$ subtracted.

The complete potential is now obtained as a sum of all bar-potentials in the mode combination range (MCR)

$$V(\boldsymbol{q}) = \sum_{\boldsymbol{m}_n \in \text{MCR}[V(\boldsymbol{q})]} \bar{V}^{\boldsymbol{m}_n}(\boldsymbol{q}^{\boldsymbol{m}_n}). \tag{9}$$

The MCR is the set of MCs that are included in the potential surface and the complete potential can systematically be approximated by restricting how many MCs are included in the MCR.[2] The particular approximation may follow the logic of the n-mode representation but can also deviate from it by selecting the MCR in other ways. For example, one can in the PES follow the logic of the excitation spaces used in MR-MCTDH[n][40] and define a

vector of weights w_m for each mode and include MCs in the MCR if

$$\sum_m w_m \leq n. \tag{10}$$

The case where one particular $w_m = 0$ while $w_{m'} = 1$ for the remaining modes, define for $n = 2$ a set of MCs where all one- and two-mode couplings are included, as well as all three-mode couplings including mode m but excluding all other three-mode couplings.

2.1.2 Sum-over-product representation

In general, the analytical form of the PES is unknown and we must resort to a numerical fitting. It is from a computational point of view highly desirable to obtain the Hamiltonian in a sum-over-product form. This is not directly the case in the n-mode expansion, but a sum-over-product form can be obtained by fitting all the individual bar-potentials of the PES to a direct product of one-mode functions. This will give a PES of the general form

$$V(\boldsymbol{q}) = \sum_{t=1}^{T} c_t \prod_{m=1}^{M} h^{m,t}, \tag{11}$$

where each term t is a simple product of one-mode operators $h^{m,t}$ with an associated coefficient c_t. Using this representation of the PES, the Hamiltonian in Eq. 1 can be written in a sum-over-products form. The computational advantage of this representation is that instead of performing multi-dimensional integrals and large contractions, we do an intermediate fit to obtain a sum-over-products form with one-dimensional integrals and simple contractions. The fits are carried out for the individual MCs and are thus in practical cases of limited dimensionality.

The notation $h^{m,t}$ suggest that there is a unique one-mode operator for each mode in each term, but we can make a mode specific notation that match the situation in practice better. The one-mode operators, $h^m_{o^m}$, for the potential come from a set of one-mode functions,

$\bar{f}_k^m(q_m)$, that constitute the fit basis used,

$$h_{o^m}^m \in \{\bar{f}_1^m(q_m), \bar{f}_2^m(q_m), \ldots, \bar{f}_{N_{\text{func}}}^m(q_m)\}. \tag{12}$$

The fit basis can be any general one-mode functions[16] with polynomials and Morse functions as common choices. The bar denotes, as before, that $\bar{f}_k^m(q_m = 0) = 0$ for example achieved by using $\bar{f}_k^m(q_m) = f_k^m(q_m) - f_k^m(0)$ for a desired function $\bar{f}_k^m$. The sum-over-product potential in Eq. 11 can be thereby be rewritten into

$$V(\boldsymbol{q}) = \sum_{\boldsymbol{m}_n \in \text{MCR}[V(\boldsymbol{q})]} \sum_{\boldsymbol{o}_n^{\boldsymbol{m}_n}} c_{\boldsymbol{o}_n^{\boldsymbol{m}_n}}^{\boldsymbol{m}_n} \prod_{m \in \boldsymbol{m}_n} h_{o^m}^m, \tag{13}$$

where the sum over terms has been divided into the outer sum of MCs, i.e. all MCs in the MCR following Eq. (9), and the inner sum over terms coming from the fit of $\bar{V}^{\boldsymbol{m}_n}$ for each MC included . The summation over $\boldsymbol{o}_n^{\boldsymbol{m}_n}$ means an n-fold summation over the different indexes in $\boldsymbol{o}_n^{\boldsymbol{m}_n} = (o_{m_1}, o_{m_2}, \cdots o_{m_n})$. Thus, for each mode there is only a very limited set of one-mode operators. The full Hamiltonian is thus represented using a limited set of one-mode integrals together with the fitting coefficients $c_{\boldsymbol{o}_n^{\boldsymbol{m}_n}}^{\boldsymbol{m}_n}$.

2.2 One-mode densities of time-dependent wave functions

We want to use our knowledge of the time-dependent wave function to guide the dynamic construction of the PES. From the Born interpretation, the wave function density is an excellent measure of which parts of the PES are important to describe. In case of a one-mode system the density can be the obtained directly from the norm square of the wave function. Generally, we will for multi-dimensional PESs use the reduced one-mode densities of the wave function, $\rho^m(q_m, t)$, which describes the density of a mode m in the coordinate representation. The one-mode density is a function of the coordinate of the mth mode, q_m, and of the time, t, and is obtained as an expectation value of the operator probing the

presence of a certain mode at a point in space, $\delta^m_{q_m}$. In first quantization this operator is defined from the Dirac-delta function[41,42]

$$\delta^m_{q_m} = \delta^m(q'_m - q_m) \tag{14}$$

and the one-mode density then becomes

$$\rho^m(q_m, t) = \langle \Psi(t)| \, \delta^m_{q_m} \, |\Psi(t)\rangle \, . \tag{15}$$

We will consider wave functions that are constructed as Hartree products of one-mode basis functions. These one-mode basis functions are denoted modals in analogy to orbitals of electronic structure theory. The modals can be either time-dependent or time-independent in the present context, and we will write it up in the time-dependent case with the time-independent case following trivially. Typically, the time-dependent modals are constructed as a linear combination of primitive time-independent modals

$$\tilde{\phi}^m_{r^m}(q_m, t) = \sum_{\alpha^m} \phi^m_{\alpha^m}(q_m) U^m_{\alpha^m r^m}(t), \tag{16}$$

where the α index belongs to the primitive basis and $U^m_{\alpha^m r^m}(t)$ is an entry in the time-dependent coefficient matrix $\boldsymbol{U}^m(t)$. We use tilde to denote the time-dependent basis.

In a second quantization formulation,[43] the one-mode density operator, when expressed in the time-dependent basis, becomes

$$\delta^m_{q_m} = \sum_{r^m s^m} [\tilde{\delta}^m_{q_m}]_{r^m s^m} \tilde{E}^m_{r^m s^m} = \sum_{r^m s^m} \left(\tilde{\phi}^m_{r^m}(q_m, t)\right)^* \tilde{\phi}^m_{s^m}(q_m, t) \tilde{E}^m_{r^m s^m}, \tag{17}$$

since the integrals of the $\delta^m_{q_m}$ operator simplify as

$$[\tilde{\delta}^m_{q_m}]_{r^m s^m} = \int \left(\tilde{\phi}^m_{r^m}(q'_m, t)\right)^* \delta^m(q'_m - q_m) \tilde{\phi}^m_{s^m}(q'_m, t) dq'_m = \left(\tilde{\phi}^m_{r^m}(q_m, t)\right)^* \tilde{\phi}^m_{s^m}(q_m, t). \tag{18}$$

The operator $\tilde{E}^m_{r^m s^m}$ is the usual second quantization one-mode shift-operator given in terms of creation and annihilation operators as $\tilde{E}^m_{r^m s^m} = \tilde{a}^{m,\dagger}_{r_m} \tilde{a}^m_{s_m}$.

Evaluating the expectation value of the one-mode density operator in Eq. 17 for a general many-mode time-dependent state $\Psi(t)$ yields

$$\rho^m(q_m, t) = \langle \Psi(t) | \delta^m_{q_m} | \Psi(t) \rangle = \sum_{s^m r^m} \tilde{D}^m_{r^m s^m}(t) \left(\tilde{\phi}^m_{r^m}(q_m, t) \right)^* \tilde{\phi}^m_{s^m}(q_m, t), \tag{19}$$

where the one-mode density matrix has been introduced

$$\tilde{D}^m_{r^m s^m}(t) = \langle \Psi(t) | \tilde{E}^m_{r^m s^m} | \Psi(t) \rangle . \tag{20}$$

Since we compute this one-mode density matrix during the solution of the equations of motion for many wave functions anyway, it is simple to use this to compute the one-mode densities. If the one-mode density matrix has been diagonalized, the one-mode density of mode m is simply obtained as the norm square of each modal, weighted by the occupation number $\omega^m_{r^m}(t) = D^m_{r^m r^m}(t)$. However, the density matrix is in general not diagonal and the modal density is obtained by squaring all combinations of modals in m and weighing them with the corresponding entry of the density matrix.

When multi-dimensional densities are required we approximate these as products of one-mode densities

$$\rho^{\boldsymbol{m}_n}(\boldsymbol{q}, t) = \prod_{m \in \boldsymbol{m}_n} \rho^m(q_m, t). \tag{21}$$

It is in principle possible to compute the actual multi-dimensional densities. However, these would be time- and space-consuming to compute, store, and use. On the other hand, the densities obtained from a direct product need not be stored as they are implicitly defined from the one-mode densities. A further advantage of using this approximation to obtain multi-dimensional densities, is that multi-dimensional integrals of such densities with a sum-

of-product potential representations can be easily factorized.

2.2.1 One-mode densities of the TDH and MCTDH wave functions

In the TDH[31] method, each mode in the system is represented by a single time-dependent modal function and the total wave function is a simple Hartree product of these time-dependent modals and a phase factor containing the real phase F

$$\left|\bar{\Psi}_{\mathrm{TDH}}\right\rangle = \exp(iF) \prod_{m=1}^{M} \left|\tilde{\phi}^{m}\right\rangle = \exp(iF) \left|\tilde{\Phi}_{\mathrm{TDH}}\right\rangle . \tag{22}$$

During the time-evolution of the system, the time-dependent modals of the Hartree product are changed, carrying the time-dependency of the TDH wave function. The one-mode density matrix in Eq. 20 reduces to unity, meaning that the TDH one-mode densities for each mode are equivalent to the norm square of the time-dependent modal function

$$\rho^{m}_{\mathrm{TDH}}(q_m, t) = |\tilde{\phi}^{m}(q_m)|^2. \tag{23}$$

Obtaining the TDH one-mode density during a calculation is thus very simple.

In the MCTDH method,[32,33] the wave function is parametrized by using a set of time-dependent Hartree products

$$\left|\tilde{\Psi}_{\mathrm{MCTDH}}\right\rangle = \sum_{\boldsymbol{u}} C_{\boldsymbol{u}}(t) \left|\tilde{\Phi}_{\boldsymbol{u}}\right\rangle . \tag{24}$$

In each mode a number of *active* modals (indexed u^m, v^m, ...) are used and all possible Hartree products using combinations of active modals of different modes are used in the wave function. The index vector $\boldsymbol{u}$, indexes the specific Hartree product $\left|\tilde{\Phi}_{\boldsymbol{u}}(t)\right\rangle$ and the corresponding coefficient $C_{\boldsymbol{u}}(t)$ in the wave function. When the time-evolution is carried out, both the Hartree products and their coefficients change opening for using a much more compact wave function than if it was simply expanded in time-independent modal functions.

Thus, the summation limits for the active modal index u^m can be kept much smaller than what would be the case for the general index r^m.

The MCTDH one-mode density is defined from its expectation value

$$\tilde{\rho}^m_{\mathrm{MCTDH}}(q_m,t) = \left\langle \tilde{\Psi}(t) \middle| \tilde{\delta}^m_{q_m} \middle| \tilde{\Psi}(t) \right\rangle = \sum_{u^m v^m} \left(\tilde{\phi}^m_{u^m}(q_m,t)\right)^* \tilde{D}^m_{u^m v^m}(t) \tilde{\phi}^m_{v^m}(q_m,t), \tag{25}$$

where we note that only the active one-mode density matrix in the time-dependent modals basis contributes

$$\tilde{D}^m_{u^m v^m}(t) = \left\langle \tilde{\Psi}_{\mathrm{MCTDH}}(t) \middle| \tilde{E}^m_{u^m v^m} \middle| \tilde{\Psi}_{\mathrm{MCTDH}}(t) \right\rangle. \tag{26}$$

In MCTDH theory the one-mode density matrix is used in the propagation of the equations of motion and is thus readily available during the calculation and evaluating the one-mode densities adds only the cost of the contraction in Eq. 25 for each mode.

The above definition of the MCTDH density also applies for the truncated multi configuration time-dependent Hartree (MCTDH[n])[44] and multi-reference truncated multi configuration time-dependent Hartree (MR-MCTDH[n])[40] wave functions, as these wave functions are parametrized equivalently to the MCTDH wave function but differ in the amount of mode couplings which are included.

2.2.2 Using wave function densities

The one-mode density of individual modes can vary strongly both along the coordinate and over time. For the purpose of our PES construction algorithms it is anticipated that smoothened densities will be much more useful. We will thus employ both a space- and a time-smoothening in order to get a more uniform density over space and time.

Consider mode m with coordinate q_m and assume it has the one-mode density $\rho^m(q_m,t)$

at time t. The space-smoothened one-mode density is obtained as

$$^{\mathrm{ss}}\rho^m(q_m,t) = \frac{1}{2\Delta_q}\int_{-\Delta_q}^{+\Delta_q} \rho^m(q_m+s,t)ds, \tag{27}$$

where the space-smoothening parameter Δ_q determine the half-width of space that we average over. The time-smoothened one-mode density is similarly obtained as

$$^{\mathrm{ts}}\rho^m(q_m,t) = \frac{1}{\Delta_t}\int_{-\Delta_t}^{0} \rho^m(q_m,t+s)ds, \tag{28}$$

where the time-smoothening parameter Δ_t is used. We also note that we do not do smoothing forward in time as this would be rather inconvenient. Combining these into a space- and time-smoothened density gives the averaged probability distribution over a past interval of time up to the current time t and with spatial smoothing to remove strong oscillations in the distribution

$$^{\mathrm{sts}}\rho^m(q_m,t) = \frac{1}{\Delta_t}\int_{-\Delta_t}^{0} {}^{\mathrm{ss}}\rho^m(q_m,t+s)ds. \tag{29}$$

In practice, we may discretize in time over past N_t densities obtained at time t and indexing these times as t_i with $i = 0, -1, \cdots, -(N_t-1)$, and t_0 being t at the current time

$$^{\mathrm{sts}}\rho^m(q_m,t) = \frac{1}{N_t}\sum_i {}^{\mathrm{ss}}\rho^m(q_m,t_i). \tag{30}$$

The densities back in time may be any that are convenient from solving the equations of motion but does require storing past one-mode densities. Similarly, if we discretize $\rho^m(q_m,t)$ we may for each point $q_{m,j}$ perform the average over $2N_q+1$ density values

$$^{\mathrm{ss}}\rho^m(q_{m,j},t) = \frac{1}{2N_q+1}\sum_{i=j-N_q}^{j+N_q} \rho^m(q_{m,i},t), \tag{31}$$

where it is to be understood that $\rho^m(q_{m,i},t)$ is zero by definition for any i that fall outside

the domain of the wave function computation and therefore will not be used in the averaging of the density.

We note that the space smoothening extend the density artificially to a slightly wider interval than the actual wave function, but of course only for meaningful values of the coordinates. As will be discussed in Section 2.3.1, mildly expanding the density with low weight into the area where it "might go" soon is actually an advantage as it improves safety and stability. A hard limit for coordinate minimum and maximum may be relevant in some coordinates and can easily be introduced. The smoothening of densities backwards in time may be only over a small previous time interval or all the way from the beginning of the computation. This can be used to tune the PES construction for what is important right now at time t for small N_t values or being focused on the full domain of where the wave function has propagated in its history for large N_t values.

The above smoothenings are directly carried over to the multi-dimensional densities obtained as direct products. Thus the multi-dimensional densities still do not need to be constructed, but instead computed when needed. Altogether, all required densities can in this manner be very efficiently evaluated from the one-mode density matrix which is in almost all cases anyway needed during the time-propagation of the wave function.

2.3 The adaptive density-guided approach for potential energy surface construction

The adaptive density-guided approach (ADGA)[15–18] for constructing PESs is an iterative method where a wave function density is used to guide the placement of SPs on a grid of molecular displacements until a convergence is reached. The overall goal of ADGA is to automatically map out an accurate PES with as few SPs as possible. This is done by creating an initial grid for each mode and then iteratively subdivide and extend these grids until all one-dimensional grids have obtained convergence. When the one-dimensional grids are converged higher-dimensional grids for the MCs included are created from the boundaries of

the one-dimensional grids. These MCs are then also iteratively subdivided until convergence is achieved.

The procedure for generating, subdividing, and extending grids have been described in detail previously and we thus refer to the previous work in our group for more in-depth details.[15,16] In this subsection we will describe only the basic elements of the ADGA as it has been developed so far as a background for a time-dependent analogous method.

At the heart of the ADGA is the ability to rigorously define a set of convergence criteria for the molecular grid. This is obtained by letting SPs define the boundaries of a set of integration boxes and then for each box determine the energy-like ADGA-integral

$$T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}} = \int_{I^{m_1}_{\ell^{m_1}}} \int_{I^{m_2}_{\ell^{m_2}}} \cdots \int_{I^{m_n}_{\ell^{m_n}}} \rho^{\boldsymbol{m}_n}_{\text{ave}}(\boldsymbol{q}) V^{\boldsymbol{m}_n}(\boldsymbol{q}) dq_{m_1} dq_{m_2} \cdots dq_{m_n}, \tag{32}$$

where the label on the integration bounds, $I^{m_k}_{\ell^{m_k}}$, is the integration bounds of interval ℓ^{m_k} of mode m_k. The ADGA integral is in general a multidimensional integral over all dimensions included in a MC. $T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}}$ is obtained from the PES fitted to the calculated SPs in the MC in the current iteration, $V^{\boldsymbol{m}_n}(\boldsymbol{q})$, and the multi-dimensional wave function density of the MC obtained by using the potential in the current iteration, $\rho^{\boldsymbol{m}_n}_{\text{ave}}(\boldsymbol{q})$. The ADGA integral is used by the ADGA to determine whether an integration box is converged and the size of the integral can also be interpreted as how "important" an area of the PES is, which will be discussed further below.

In previous works on ADGA, the wave function density has been obtained from intermediate vibrational self-consistent field (VSCF)[27–30] calculations. The VSCF method provides an approximate solution to the time-independent Schrödinger equation where the ansatz is a single Hartree product of modals

$$\Phi^{\text{VSCF}}_{\boldsymbol{s}}(\boldsymbol{q}) = \prod_{m=1}^{M} \phi^{m}_{s^m}(q_m), \tag{33}$$

where $\boldsymbol{s}$ is a M-dimensional index vector indexing which modal, $\phi^{m}_{s^m}$, to use for each mode.

The VSCF modals are functions of a single coordinate and the density for a mode m is simply obtained as the norm square of the modal function, $|\phi^m_{s^m}(q_m)|^2$.

The densities of the VSCF states vary over space and the eigenstates of higher energy have a wider distribution of the wave function as a function of the coordinate. In the ADGA algorithm, the default is to use an average density determined as an average over the N_{modals} lowest-energy eigenstates

$$\rho^m_{\text{ave}}(q_m) = \frac{1}{N_{\text{modals}}} \sum_{s^m=1}^{N_{\text{modals}}} |\phi^m_{s^m}(q_m)|^2. \tag{34}$$

By increasing the number of eigenstates used in the average, a larger domain of the PES is explored as the higher energy of the higher lying states leads to a wider distribution of the wavefunction. The averaging with the lower energy states ensures that the averaged density does not contain any nodes within the potential boundaries. Similar to Eq. 21, multi-dimensional averaged densities, $\rho^{\boldsymbol{m}_n}_{\text{ave}}(\boldsymbol{q})$, are approximated as a simple product of the VSCF densities for the individual modes.

When the one-dimensional grids are constructed in the ADGA procedure, a VSCF calculation is carried out in each ADGA iteration using the current fitted potential in the Hamiltonian. From this VSCF calculation the averaged densities for each mode are created and used to calculate the ADGA integrals. For the higher dimensional grids the multi-dimensional averaged density is supplied as a product of the one-dimensional average densities obtained in the final iteration of the one-dimensional grids.

In the following, we term this formulation of ADGA as the time-independent adaptive density-guided approach (TI-ADGA) to make it distinct from the time-dependent method introduced later.

To determine whether an integration box is converged, and thus not sub-divide it by adding a SP, the relative and absolute difference in the integral in the current ($T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}}$ (curr. it.)) and previous ($T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}}$ (prev. it.)) iteration is compared to a set of thresholds. A TI-ADGA

box is thus considered converged and not updated if

$$\left(\left|\frac{T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}}(\text{curr. it.}) - T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}}(\text{prev. it.})}{T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}}(\text{curr. it.})}\right| < \epsilon_{\text{rel}}\right)$$
$$\vee\left(\left|T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}}(\text{curr. it.}) - T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}}(\text{prev. it.})\right| < \epsilon_{\text{abs}} \wedge \left|T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}}(\text{curr. it.})\right| < \epsilon_{\text{abs}}\right), \tag{35}$$

where ϵ_{rel} and ϵ_{abs} are suitably chosen threshold parameters. Eq. 35 shows how the TI-ADGA algorithm interprets the value of the TI-ADGA integral. The *relative* condition compares the relative change in $T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}}$ between two iterations and if this change is larger than the relative threshold, ϵ_{rel}, the integration box should be sub-divided. In this way areas of the PES where the density times the energy varies a lot between iterations are subdivided until a convergence is obtained. The TI-ADGA integral can be viewed as a measure for how much a given integration box contributes to the energy and thus if the current value of $T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}}$ is very small it can be considered unimportant. This is enforced by the *absolute* condition, where an integration box is considered converged if both the absolute difference between two iterations and the absolute size of $T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}}$ (curr. it.) are smaller than ϵ_{abs}.

To determine whether a one-dimensional grid should be extended, it is used that the density should integrate to one if integrated over the entire space. The grid boundaries of a one-dimensional cut are thus extended if the integrated one-mode density is smaller than one within a specified threshold. That is, no extension of the grid is needed if

$$\sum_i \int_i \rho^m_{\text{ave}}(q_m) dq_m > 1 - \epsilon_\rho, \tag{36}$$

where i defines the ith one-dimensional integration box and ϵ_ρ is the amount of density that is allowed to be outside the domain of the grid. In order to determine the amount of the density that is outside the grid boundaries, the basis set used in the VSCF calculation is expanded a small amount outside the grid bounds.

The TI-ADGA algorithm continues to add SPs to subdivide and extend the one-mode

grids until convergences is obtained. Afterwards higher order MCs are converged until all MC in the MCR are converged and a n-mode PES have been obtained.

2.3.1 Time-dependent adaptive density-guided approach

We now generalize the ADGA procedure for use in quasi-direct quantum dynamics simulations, where the goal is to construct the PES during the integration of the time-dependent Schrödinger equation. As this formulation of ADGA is based on the time-dependent wave function and the corresponding evolving densities, we will term it the time-dependent adaptive density-guided approach (TD-ADGA). The construction of the TD-ADGA grid is similar to the TI-ADGA grid, where independent grids for each MC are iteratively refined by automatic placement of new SPs. However, we introduce some extra convergence criteria, since there is a fundamental difference between the TI-ADGA and TD-ADGA algorithms. In TI-ADGA, the iterations were purely algorithmic iterations until convergence (requiring of order 1-15 iterations). In the TD-ADGA, the iterations continues for the entire dynamics simulation due to the evolving wave function density. I.e. convergence criteria will be revisited many times and it is therefore important each visit does not lead to addition of SPs unless really needed.

In TD-ADGA we consider an energy-like quantity very similar to the one in Eq. 32, but we replace the VSCF densities with the time-dependent densities described in Section 2.2 and instead of using the average density from a number of states for each mode, we use the space- and time-smoothened one-mode densities in Eq. 29. This gives the TD-ADGA integral

$$T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}}(t) = \int_{I^{m_1}_{\ell^{m_1}}} \int_{I^{m_2}_{\ell^{m_2}}} \cdots \int_{I^{m_n}_{\ell^{m_n}}} {}^{\mathrm{sts}}\rho^{\boldsymbol{m}_n}(\boldsymbol{q},t) V^{\boldsymbol{m}_n}(\boldsymbol{q}) dq_{m_1} dq_{m_2} \cdots dq_{m_n}, \tag{37}$$

where the multi-dimensional averaged densities are obtained according to Eq. 21.

The space- and time-smoothening of the one-mode density serve similar purposes as the

average density of TI-ADGA. The smoothening over time and space ensures that fast oscillations and changes that might occur in the one-mode density are smoothened out, so that the TD-ADGA integral does not change wildly between iterations. The space-smoothening further expands the density to a slightly wider interval than what the wavefunction itself suggests. This forces the TD-ADGA algorithm to take into account a slightly wider interval of the PES and this may precisely be the domains where the wave function propagates within the next short time.

Similar to the TI-ADGA, we define a set of convergence thresholds for determining when the TD-ADGA grid is converged. When the convergence is checked in TD-ADGA we compare the TD-ADGA integral at the current time $T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}}(t_{\text{curr}})$, with the TD-ADGA integrals obtained at a previous time $T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}}(t_{\text{prev}})$. Thus, an integration box in the TD-ADGA grid is considered converged and should not be sub-divided if

$$
\begin{aligned}
&\left(\left|\frac{T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}}(t_{\text{curr}}) - T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}}(t_{\text{prev}})}{T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}}(t_{\text{curr}})}\right| < \epsilon_{\text{rel}}\right) \\
\vee&\left(\left|T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}}(t_{\text{curr}}) - T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}}(t_{\text{prev}})\right| < \epsilon_{\text{abs}} \wedge \left|T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}}(t_{\text{curr.}})\right| < \epsilon_{\text{abs}}\right) \\
\vee&\left(\left|T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}}(t_{\text{curr}}) - T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}}(\text{lin.})\right| < \epsilon_{\text{lin}}\right) \\
\vee&\left(\left|T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}}(t_{\text{curr}})\right| < \left|T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell^{\boldsymbol{m}_n}}}(\text{max.})\right|\right).
\end{aligned}
\tag{38}
$$

As is apparent from Eq. 38, we have kept the relative and absolute conditions from the TI-ADGA algorithm but we have also added two new conditions which can be fulfilled in order for an integration box to be converged.

The first new condition for the integration box under consideration is the *linear* condition. This condition tests whether the current TD-ADGA integral is well approximated by assuming that the PES is well described by a multi-linear interpolation between the SPs that constitute the integration box boundaries. For the one-dimensional grids, this multi-linear interpolation of the PES is simply a linear interpolation between the two grid points constituting the box boundaries. For a two-mode grid with $\boldsymbol{m} = (m_1, m_2)$ and a two-dimensional

box consisting of the four grid points $\{(q_{m_1,1}, q_{m_2,1}), (q_{m_1,2}, q_{m_2,1}), (q_{m_1,1}, q_{m_2,2}), (q_{m_1,2}, q_{m_2,2})\}$, the multi-linear interpolation function inside the integration box is

$$V_{\text{lin}}^{m_1,m_2}(q_{m_1}, q_{m_2}) = a_{00} + a_{01}q_{m_1} + a_{10}q_{m_2} + a_{11}q_{m_1}q_{m_2} \tag{39}$$

and the coefficients are determined from the linear system of equations

$$\begin{bmatrix} 1 & q_{m_1,1} & q_{m_2,1} & q_{m_1,1}q_{m_2,1} \\ 1 & q_{m_1,2} & q_{m_2,1} & q_{m_1,2}q_{m_2,1} \\ 1 & q_{m_1,1} & q_{m_2,2} & q_{m_1,1}q_{m_2,2} \\ 1 & q_{m_1,2} & q_{m_2,2} & q_{m_1,2}q_{m_2,2} \end{bmatrix} \begin{bmatrix} a_{00} \\ a_{01} \\ a_{10} \\ a_{11} \end{bmatrix} = \begin{bmatrix} V^{m_1,m_2}(q_{m_1,1}, q_{m_2,1}) \\ V^{m_1,m_2}(q_{m_1,2}, q_{m_2,1}) \\ V^{m_1,m_2}(q_{m_1,1}, q_{m_2,2}) \\ V^{m_1,m_2}(q_{m_1,2}, q_{m_2,2}) \end{bmatrix}. \tag{40}$$

This approach is easily generalized to higher dimensions. The *linearized* TD-ADGA integral $T_{I_{\ell}^{\boldsymbol{m}_n}}^{\boldsymbol{m}_n}$ (lin.) is then calculated similar to Eq. 37 but where the multi-linear interpolation of the PES in the integration box is used instead of the global fitted PES for the MC. To test whether the linear approximation is good, the absolute difference between the TD-ADGA integral of the current iteration and the linearized TD-ADGA integral is compared to the linear threshold value ϵ_{lin}. In effect, this condition ensures a minimum distance between the SPs that the TD-ADGA algorithm can place, because a linear approximation will be good for any potential PES if the grid becomes fine enough. We emphasize that the multi-linear interpolation is never used in the final PES representation and is solely used to identify domains of the PES where the representation is already good as interpolation between interval end points appears to be trivial.

The second new condition we denote as the *maximum* condition, which state that an integration box is converged if the current value of the TD-ADGA integral is smaller than the maximum historic value of the TD-ADGA integral. This condition is applied because we do not want to add new single points on parts of the PES that have previously been considered converged unless the importance of it is actually increasing. In particular, we

do not want to add SPs in domains of the PES where the wave function density is leaving (which changes the TD-ADGA integral to smaller values) or if the density is re-entering to these parts but with smaller magnitude than before. Note, that if the density is increased the TD-ADGA integral can increase and SPs can still be added to the grid.

We also want the TD-ADGA procedure to be able to expand the one-mode grids so that the wave function is always contained within the grid boundaries even as the wave function moves around on the PES. We use an equivalent condition as in the TI-ADGA for when a one-mode grid should be expanded dependent on the integral of the space- and time-smoothened one-mode density. The grid is not extended if

$$\sum_i \int_i^{\text{sts}} \rho^m(q_m, t) dq_m > 1 - \epsilon_\rho. \tag{41}$$

Similar to the TI-ADGA, this condition arises from the fact that the density must integrate to one.

The TD-ADGA iterations are carried out at given times in the dynamics. In a TD-ADGA iteration the convergence of the one-mode grids are initially checked and if one or more one-mode grids are not converged, they are extended or subdivided iteratively until convergence is obtained. The TD-ADGA thus contains sub-iterations for each TD-ADGA iteration. After the one-mode grids are converged the two-modes grids are checked for convergence and this continues until all MCs are converged.

2.4 Updating the potential energy surface during the dynamics

We now assume we have reached a point in time t_{upd} during the dynamics, where the TD-ADGA analysis finds that an update of the set of SPs are required. The question is then how to update the PES. Even though the change in the PES should primarily be significant in regions where the density has not yet grown large, simply replacing the old PES with the new PES at t_{upd} can lead to non-physical behaviour as the propagation of the density will

have been on account on the old PES. Furthermore, changing the PES will result in loss of the exact conservation of energy. To mitigate these challenges we propose two different schemes for updating the PES in a dynamics simulation.

The first scheme is the *full restart scheme*, where we simply restart the dynamics from the beginning each time an update is carried out and in each restart we use the newest PES. This scheme is the safest option as a constant PES is used in the propagation of the dynamics meaning that the dynamics are solely dependent on the most accurate PES and we have no issues with the conservation of energy. However, this scheme will also be costly as a lot of time will be used in the wave function propagation. We therefore also propose a *partial restart scheme*. In the partial restart scheme we restart the calculation at a previous time before the PES was updated, t_{bef}. The PES is then continuously updated from the old PES at t_{bef} to the new PES at t_{upd}. The procedure for this will be described below. This partial restart scheme will be more efficient than the full restart as much less time will be spent in the wave function propagation and if t_{bef} is sufficiently long back in time the simulation of the dynamics is expected to be largely unaffected. We note that the partial restart scheme formally does not solve the problem of losing exact energy conservation but is expected to strongly reduce the numerical consequence.

A composite strategy can be suggested, where the partial restart method is used to map out the relevant domain of the PES and then when this propagation is finished a final wave function propagation, with the final PES, is carried out and analysed as the result of the simulation. We suspect that this strategy will be the most attractive with regards to balancing accuracy, computational cost, and avoiding artefacts.

When the PES is updated in the partial restart scheme we want to continuously update the potential from before the update to the potential after the update in the time interval t_{bef} to t_{upd}. This is achieved by formulating the potential in this interval as

$$V(\boldsymbol{q},t) = f_{\mathrm{bef}}(t)V_{\mathrm{bef}}(\boldsymbol{q}) + f_{\mathrm{upd}}(t)V_{\mathrm{upd}}(\boldsymbol{q}), \tag{42}$$

where $V_{\mathrm{bef}}(\boldsymbol{q})$ and $V_{\mathrm{upd}}(\boldsymbol{q})$ are the potentials before and after the update, respectively. The weight functions $f_{\mathrm{bef}}(t)$ and $f_{\mathrm{upd}}(t)$ must satisfy that

$$f_{\mathrm{bef}}(t) = 1 \ \wedge \ f_{\mathrm{upd}}(t) = 0, \qquad t \leq t_{\mathrm{bef}}, \tag{43}$$

$$f_{\mathrm{bef}}(t) = 0 \ \wedge \ f_{\mathrm{upd}}(t) = 1, \qquad t \geq t_{\mathrm{upd}}, \tag{44}$$

and

$$f_{\mathrm{bef}}(t) + f_{\mathrm{upd}}(t) = 1. \tag{45}$$

This can be realized in numerous ways, but we shall here explore only one simple choice by defining $f_{\mathrm{upd}}(t)$ as the following C^1 function

$$f_{\mathrm{upd}}(t) = \begin{cases} 0 & t < t_{\mathrm{bef}} \\ \left(\cos\left(\frac{t_{\mathrm{upd}} - t}{t_{\mathrm{upd}} - t_{\mathrm{bef}}}\pi\right) + 1\right)/2 & t_{\mathrm{bef}} \leq t \leq t_{\mathrm{upd}} \\ 1 & t_{\mathrm{upd}} < t \end{cases}. \tag{46}$$

and define $f_{\mathrm{bef}}(t) = 1 - f_{\mathrm{upd}}(t)$. Any quantity y^H that is linear in the Hamiltonian $H = T + V$ will then be subject to a similar transformation

$$y^H = y^T + f_{\mathrm{bef}}(t) y^{V_{\mathrm{bef}}} + f_{\mathrm{upd}}(t) y^{V_{\mathrm{upd}}}. \tag{47}$$

Explicitly using this formulation, means that the y quantity must be evaluated for two different potentials and this could increase the computational cost for the equations of motion in a specific time-domain significantly. However, in the case where the two PESs are fitted to the same fit basis, and no non-linear optimization of fit functions are required, the update of the potential is obtained at essentially no cost by a continuous update of the sum-over-

products coefficients in Eq. 13

$$c^{\boldsymbol{m}_n}_{\boldsymbol{o}_n^{\boldsymbol{m}_n}}(t) \leftarrow f_{\mathrm{bef}}(t) c^{\boldsymbol{m}_n,\mathrm{bef}}_{\boldsymbol{o}_n^{\boldsymbol{m}_n}} + f_{\mathrm{upd}}(t) c^{\boldsymbol{m}_n,\mathrm{upd}}_{\boldsymbol{o}_n^{\boldsymbol{m}_n}}. \tag{48}$$

How the times t_{bef} and t_{upd} are chosen will be discussed in detail in Section 3.2.

3 Implementation

In this section we will describe the implementation of the TD-ADGA algorithm as it has been implemented in the Molecular Interactions, Dynamics And Simulations Chemistry Program Package (MidasCpp).[37] The implementation relies on the previous implementations in MidasCpp for the integration and propagation of ordinary differential equations for time-dependent nuclear wave functions as well as the ADGA framework for handling SP calculations, grid placement, and evaluation of integrals.[15–18]

A high level description of how the TD-ADGA procedure is implemented as a part the ordinary differential equation integrator is shown in Algorithm 1. The ordinary differential equation integrator is on input given the initial and end times (t_0 and t_{end}), the initial wave funtion parameters ($\boldsymbol{y}_0$), and the initial PES (V_0). Further TD-ADGA inputs are also given, these include a boolean indicating if a full restarts should be utilized, the time interval between running TD-ADGA convergence checks ($t_{\text{TD-ADGA}}$), the time interval at which the one-mode densities are calculated (t_{dens}), and the time interval between reset points (t_{reset}). These intervals will be explained further below. In the initial 5 lines of Algorithm 1 the integration is prepared and in the **while** loop from line 6 to line 38 the actual time-propagation takes place until the dynamics have been propagated to the end time.

The first **if** block in the **while** loop (line 7) is only executed if the TD-ADGA algorithm has found that the PES was no longer converged after the previous time-step. If this is the case the current time (t), wave function ($\boldsymbol{y}$), wave function derivatives ($\frac{\mathrm{d}\boldsymbol{y}}{\mathrm{d}t}$), and PES ($V$)

Input : t_0, t_{end}, $\boldsymbol{y}_0$, V_0
TD-ADGA input: full_replay, $t_{\text{TD-ADGA}}$, t_{dens}, t_{reset}

Calculate $\left(\frac{\mathrm{d}\boldsymbol{y}}{\mathrm{d}t}\right)_{t=0}$;
Save t_0, $\boldsymbol{y}_0$, $\left(\frac{\mathrm{d}\boldsymbol{y}}{\mathrm{d}t}\right)_{t=0}$, and V_0;
Set $t = t_0$, $\boldsymbol{y} = \boldsymbol{y}_0$, $\frac{\mathrm{d}\boldsymbol{y}}{\mathrm{d}t} = \left(\frac{\mathrm{d}\boldsymbol{y}}{\mathrm{d}t}\right)_{t=0}$, and $V = V_0$;
Set $t_{\text{check}} = t_0 + t_{\text{TD-ADGA}}$, $t_{\text{calc\_dens}} = t_0 + t_{\text{dens}}$, $t_{\text{save}} = t_0 + t_{\text{reset}}$, and $t_{\text{upd}} = t_0$;
Set reset_dynamics = false and update_potential = false;
while $(t \leq t_{\text{end}})$ **do**
    if (reset_dynamics) **then**
        $t_{\text{upd}} = t_{\text{save}} - t_{\text{reset}}$;
        Set t_{bef}, $\boldsymbol{y}_{\text{bef}}$, $\left(\frac{\mathrm{d}\boldsymbol{y}}{\mathrm{d}t}\right)_{t=t_{\text{bef}}}$, and V_{bef} to values saved at $t_{\text{upd}} - t_{\text{reset}}$;
        if (full_replay || $t_{\text{bef}} \leq t_0$) **then**
            Set $t = t_0$, $\boldsymbol{y} = \boldsymbol{y}_0$, $\frac{\mathrm{d}\boldsymbol{y}}{\mathrm{d}t} = \left(\frac{\mathrm{d}\boldsymbol{y}}{\mathrm{d}t}\right)_{t=0}$, and $V = V_{\text{upd}}$;
            $t_{\text{check}} = t_0 + t_{\text{TD-ADGA}}$;
        else
            Set $t = t_{\text{bef}}$, $\boldsymbol{y} = \boldsymbol{y}_{\text{bef}}$, $\frac{\mathrm{d}\boldsymbol{y}}{\mathrm{d}t} = \left(\frac{\mathrm{d}\boldsymbol{y}}{\mathrm{d}t}\right)_{t=t_{\text{bef}}}$, and $V = V_{\text{bef}}$;
            $t_{\text{check}} = t_{\text{bef}} + t_{\text{TD-ADGA}}$;
            update_potential = true;
        end
        Clear all saved data for times greater than t; `// Also resets densities`
        Recalculate TD-ADGA integrals; `// Using` V_{upd} `fit`
        reset_dynamics = false;
    end
    Take time-step; `// Updates` t, $\boldsymbol{y}$, `and` $\frac{\mathrm{d}\boldsymbol{y}}{\mathrm{d}t}$
    if $(t \geq t_{\text{calc\_dens}})$ **then**
        Calculate and update one-mode densities;
        $t_{\text{calc\_dens}} = t_{\text{calc\_dens}} + t_{\text{dens}}$
    end
    if $(t \geq t_{\text{check}})$ **then**
        Do TD-ADGA iteration;
        `// If new SPs were added, fits new` V_{upd}`, set reset_dynamics true,`
        `// and continue to next iteration in while loop`
        $t_{\text{check}} = t_{\text{check}} + t_{\text{TD-ADGA}}$;
    end
    if (update_potential) **then**
        Update PES; `// Sets update_potential false if` $t \geq t_{\text{upd}}$
    end
    if $(t \geq t_{\text{save}}$ && !full_replay) **then**
        Save t, $\boldsymbol{y}$, $\frac{\mathrm{d}\boldsymbol{y}}{\mathrm{d}t}$, and V;
        $t_{\text{save}} = t_{\text{save}} + t_{\text{reset}}$;
    end
end

Algorithm 1: The overall scheme for how the TD-ADGA method is implemented as part of the ordinary differential equation integrator.

is reset to a previous time. If the full restart scheme is used, the wave function parameters are all set to their initial values at t_0 but the PES is set to the updated potential (V_{upd}). If the partial restart scheme is used, an update interval of length t_{reset} is determined where the two times t_{bef} and t_{upd} are the start and end times of this interval. The wave function parameters and the PES are then reset to the values at t_{bef}, which have been saved in a previous integration step. If t_{bef} is smaller than or equal to t_0 then the wave function parameters and PES are reset as if we were in a full restart run. After the parameters have been reset, all saved data for times that are larger than t are cleared. This includes the one-mode densities, so the newest densities are from the new t. The TD-ADGA integrals are then recalculated with the "new" one-mode densities without checking for grid convergence. As the final part of this **if** block the reset_dynamics boolean is set to false.

In line 22 the actual integration step is taken. The integration is done using the Dormand-Prince 8(5,3) explicit Runge-Kutta method[45] as implemented in MidasCpp. After the integration step is completed, the current time has been updated and the wave function parameters and the time-derivative of these have also been updated. In line 24 the one-mode densities for the specific wave function method are calculated using the updated wave function parameters and the relevant space- and time-smoothening is carried out. In order to not evaluate the wave function densities on a grid in each time step, the one-mode densities are only calculated at intervals spaced by t_{dens}. Note that t_{dens} should be chosen smaller than or equal to $t_{\text{TD-ADGA}}$ since it makes no sense to check the TD-ADGA convergence if the one-mode densities have not been updated.

If the current time exceeds the TD-ADGA check time (t_{check}) a TD-ADGA iteration is carried out in line 28. This iteration checks the convergence of the TD-ADGA grid and if the grid is not converged additional SP calculations are requested and the boolean reset_dynamics is set to true. When the TD-ADGA grid is again converged, the algorithm continues to the next iteration in the **while** loop. If instead the grid is still converged after the TD-ADGA iteration, the reset_dynamics boolean remains false and t_{check} is updated to

the next check time.

If the update_potential boolean is true, then the PES of the dynamics currently being continuously updated and this is done in line 32, using the continuous update discussed in Subsection 2.4. If t is greater than or equal to the end of the current update interval (t_{upd}), update_potential is set false since $V = V_{\text{upd}}$ and the PES now is fully updated.

The final part of the while loop saves the current wave function parameters, the PES, and updates the save time (t_{save}) if $t \geq t_{\text{save}}$ and we are not in the full restart scheme.

In the following two subsections we will describe in more detail two of the steps in Algorithm 1 that are directly related to the TD-ADGA implementation. In Section 3.1 we will discuss the how the TD-ADGA iterations are carried out at different time-steps and in Section 3.2 the update of the PES is considered in more detail with a specific example to illustrate the procedure.

3.1 The TD-ADGA iterations

We will now describe how the TD-ADGA algorithm is implemented as a part of the ordinary differential equation integrator in MidasCpp. Before the time-propagation is initialized, an initial iteration of the TD-ADGA procedure is carried out to create a PES to start both the dynamics and TD-ADGA procedure from. In this *zeroth* iteration of the TD-ADGA algorithm, the initial PES V_0 is created using the TI-ADGA procedure using a density either provided by a VSCF calculation on some other potential or by the user on input. The placement of the initial SPs are determined by the initial density. A SP is always placed at the reference structure and two further SPs are placed such that the entire density (within the residual threshold determined by Eq. 41) is contained within the SP boundaries. A further SP is also placed at the maximum density value if this is not close to the reference structure. Using these initial SPs and the fixed initial density, a TI-ADGA calculation is carried out. This creates the grids for all MCs with initial values for all the TD-ADGA integrals and it provides a PES to the ordinary differential equation integrator which then

initialize the time-dependent wave function $\boldsymbol{y}_0$.

After the wavefunction has been initialized, the *first* TD-ADGA iteration is carried out before the time-propagation starts. This "extra" iteration is carried out since the density used to converge the zeroth iteration might vary from the density of $\boldsymbol{y}_0$. This is because the initial density of the zeroth iteration can be any density supplied by the user or an average density obtained from a VSCF calculation, while the current density has been calculated using the space-smoothened one-mode densities of the initial state (there is no time-smoothening since no time has passed at which the one-mode density can be smoothened over). Using the new density, new TD-ADGA integrals are calculated and then each integration box is checked for convergence by checking against the conditions in Eq. 38 and Eq. 41. If one or more integration boxes are no longer converged or if the one-mode grids should be extended, new SPs are requested by the TD-ADGA algorithm and after these have been calculated a new PES is fitted. The TD-ADGA integrals are then re-calculated and convergence of the integration boxes checked again. This procedure is repeated until all integration boxes are converged at which point the time-integration can begin.

After the zeroth and first iterations have been carried out Algorithm 1 can be started. As briefly discussed above, the next TD-ADGA iteration is carried out after the integrator has taken a time step so that the current time exceeds the check time t_{check}. The TD-ADGA iteration check times are evenly distributed in time with the time interval $t_{\text{TD-ADGA}}$. When the current time is larger than the TD-ADGA check time, we enter step 28 of Algorithm 1. The TD-ADGA integrals at the current time t are calculated ($T^{\boldsymbol{m}_n}_{I^{\boldsymbol{m}_n}_{\ell}}(t_{\text{curr}})$) using the current space- and time-smoothened density and the convergence criteria for each box are checked. We note here that we choose the time interval of the calculation of the one-mode densities t_{dens} such that new one-mode densities are always calculated after the same time-step as the TD-ADGA iterations are also carried out. This is not a necessity for the algorithm but it makes the most sense to use information from the wave function at the current time in the TD-ADGA iterations.

If all boxes are converged and no extension of the one-mode grids are required, the time-propagation simply continues until the next check time is reached. If, on the other hand, the TD-ADGA grid is no longer converged, the same procedure as for obtaining convergence for the first iteration is carried out until the TD-ADGA grid is converged. If new SPs have been calculated a new PES have also been obtained and the time-propagation is reset to a previous time-step in order for the PES of the dynamics to be continuously updated. How we handle the update of the potential will be described in the following subsection.

3.2 Update of the potential energy surface

In this subsection we will describe the specific approach for updating the PES and resetting the dynamics after a TD-ADGA iteration has calculated new SPs.

We now assume we have reached a time where the TD-ADGA grid is no longer converged and thus the TD-ADGA procedure will calculate a number of SPs to obtain convergence of the grid. After these SPs have been calculated and a new PES have been fitted, we need to determine how we want to continue the dynamics simulation. As discussed in Section 2.4, we suggest a partial restart scheme and a full restart scheme for introducing the new PES. In the partial restart scheme we want the PES of the dynamics to be slowly and continuously updated from the previous PES (V_{bef}) to the updated PES (V_{upd}) in order to avoid non-physical behavior in the wave function propagation. In the full restart scheme, we simply restart the dynamics from scratch using the new PES. To update the PES during the wave function propagation in the partial restart scheme, the wave function is reset to a time before the PES was updated (t_{bef}) by overwriting the current wave function and its derivatives with the wave function and its derivatives at time t_{bef}. In order not to save all wave function data during the propagation, the update of the potential is done between set time-intervals. These reset intervals are spaced with the time t_{reset} and when the current time t of the simulation enters a new reset interval, t and all wave function information relevant for the time-propagation are saved. When new SPs have been placed by the TD-ADGA

routine t_{bef} is thus chosen to be the beginning of the previous reset interval and the time where the PES is fully updated (t_{upd}), is set to be the end of the previous update interval ($t_{\text{upd}} = t_{\text{bef}} + t_{\text{reset}}$). With t_{bef} and t_{upd} defined, the PES is then continuously updated in the time-interval t_{bef} to t_{upd} as the integrator propagates through time, updating the PES used in the dynamics after each time-step by using Eq. 42 to update from V_{bef} to V_{upd}.

To illustrate this procedure, we have in Fig. 1 shown two examples for how the updating of the PES in the TD-ADGA is handled. Fig. 1a shows the general case for the partial restart scheme where the TD-ADGA procedure at some point determines that the TD-ADGA grid is no longer converged and the PES must be updated in order to re-obtain convergence. In Fig. 1a the top horizontal line represents the time-propagation where the integrator takes small time-steps propagate the wave function. The propagation starts at t_0 and with time-intervals of $t_{\text{TD-ADGA}}$ (marked with crosses) the TD-ADGA convergence is checked. When the integration time exceeds $t_1 = t_{\text{reset}}$ the current wave function parameters $\boldsymbol{y}$ and $\frac{d\boldsymbol{y}}{dt}$ are saved for resetting purposes before the propagation continues. The time-propagation continues through the second reset interval and enters the third, where the wave function information is again saved. In this example the third TD-ADGA iteration of the third update interval (marked by a red cross) finds that now the wave function density has changed so much that the TD-ADGA grid is no longer converged. The TD-ADGA algorithm places new SPs until a new convergence have been reached, as discussed in Subsection 3.1.

After the new SPs have been calculated, the PES is refitted and the propagation is reset to the beginning of the second update interval at t_1. This is illustrated as the bottom horizontal line in Fig. 1a starting from t_1. From the wave function information and the potential saved at t_1, the time-propagation is restarted. In each integrator step between t_1 and t_2, the PES is incrementally updated from V_{bef} to V_{upd} until the PES is equal to V_{upd} at $t_{\text{upd}} = t_2$. When the integrator has passed the time t_2 the current wave function is again saved and from this point the time-propagation continues using the updated potential.

If the TD-ADGA procedure asks for new SPs early in the time-propagation, t_{bef} will be

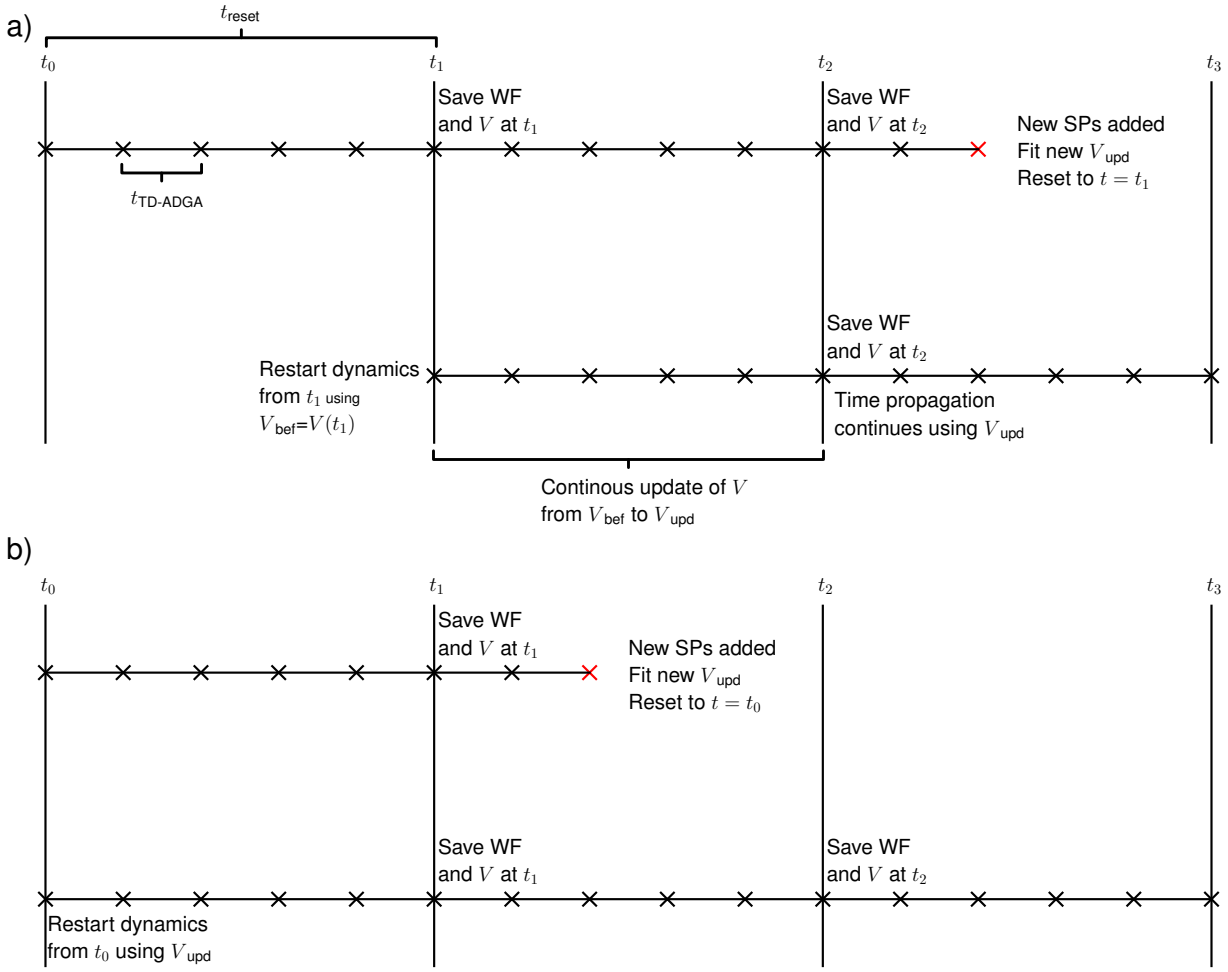


Figure 1: Sketch of the procedure for updating the PES in the TD-ADGA algorithm. The horizontal lines represents the time-evolution of the integrator, the vertical lines indicate the boundaries of the reset intervals, and the crosses indicate the times where the TD-ADGA convergence is checked. a) The wave function is propagated through time and in the third update interval new SPs are requested by the TD-ADGA routine (marked by a red cross). The propagation is reset to the time t_1 which is indicated by the horizontal line below. Initially the PES saved at time t_1 (V_{bef}) is used to propagate the wave function but during the propagation the PES is continuously updated until $t_{\text{upd}} = t_2$ is reached at which point the PES is equal to V_{upd}. b) The wave function is propagated through time and in the second update interval new SPs are requested by the TD-ADGA routine. The propagation is then reset to the initial time t_0. Since we reset to the initial time the PES is updated directly to V_{upd} and the propagation continues from the initial wave function.

equal to the initial time of the propagation. This is illustrated in Fig. 1b where new SPs are requested in the second update interval. In this case the dynamics are reset to the initial wave function and a new dynamics simulation is started where V_{upd} is used as the PES from the beginning without doing a continuous update. If the TD-ADGA is run with the full restart setting, then each time new SPs are added the dynamics will be reset to the initial time, similar to the case showed in Fig. 1b.

We note that at all times when the TD-ADGA integrals are computed, we use the fitted PES obtained by fitting to all calculated SPs. This means that the potential term in the TD-ADGA integrals (Eq. 37) are not continuously updated but instead the most accurate potential is used. This is done because a continuous update of the potential in the evaluation of the TD-ADGA integrals will lead to changes in the calculated integrals that are artificial in the sense that the change does not depend on the physical state of the system but only on how we chose to update the PES during the dynamics. Further, the extra change induced in the integrals by the change in the potential might lead to the TD-ADGA algorithm requesting more SPs than is actually required for converging the PES for the dynamics.

4 Results

We will in this section present a series of TD-ADGA calculations carried out to test and benchmark our TD-ADGA implementation. We have carried out calculations on a series of molecular systems with different complexity in both the nature of the PES, dynamics, and size of the system. These systems include the dynamics induced in the bromine dimer when moving the wave function from one excited state to another, the exploration of the double well potential in the inflection mode of ammonia, the small amplitude motion in water induced by a time-dependent perturbation, and finally the keto- enol tautomerization of salicylaldimine in a reduced dimensionality.

4.1 Computational Details

In our dynamics simulations we have used the TDH[31] and MCTDH[32,33] methods as implemented in MidasCpp in conjunction with the newly implemented TD-ADGA method. The dynamics obtained with the TD-ADGA method will be compared to reference calculations obtained with a pre-calculated PES obtained by using the TI-ADGA method. On that account the accuracy of the dynamics and the number of SPs used in obtaining the PES will be compared. We will denote the PES that is used in a reference calculation as the reference PES.

In all dynamics calculations a primitive B-spline basis has been employed using B-spline functions of order 10 and a basis density of 0.8.[46] The boundaries of the basis sets used were determined by the reference PESs for each system and used in all computations for consistency in comparisons. These boundaries can be found in the supporting information (SI).

The TD-ADGA settings and thresholds used in the simulations are listed in Tab. 1 and they have been described in Section 2. Unless otherwise stated, the epsilon thresholds used in the TD-ADGA calculations are the same as those which have been used in the TI-ADGA calculations to obtain the reference PESs, with the exception of the linear threshold which is not used in the TI-ADGA. The number of grid points which the density is smoothened over (N_q) is given as a fraction of the grid size (c_q) such that the number of grid points to smoothen over is given as $N_q = c_q N_{\text{pts}}$, where N_{pts} is the total number of grid points in a one-mode cut.

4.2 Exploring excited states of molecular bromine

In this subsection we will study the dynamics of the excited states of the bromine dimer. We want to study how well the TD-ADGA method can explore an unknown PES and the excited states. The bromine dimer provides a simple one-mode system, where a nuclear wave function is initialized on the PES of one electronic state and then moved to another

Table 1: TD-ADGA settings used in the calculations for each system.

	Bromine	Ammonia	Water	Salicylaldimine[a]
t_{reset}	100.0	100.0	100.0	100.0
$t_{\text{TD-ADGA}}$	100.0	100.0	100.0	100.0
t_{dens}	10.0	10.0	10.0	10.0
c_q	0.10	0.10	0.10	0.10
N_t	20	20	20	20
ϵ_{rel}	$1.0 \cdot 10^{-2}$	$1.0 \cdot 10^{-2}$	$1.0 \cdot 10^{-2}$	$1.0 \cdot 10^{-4}$
ϵ_{abs}	$1.0 \cdot 10^{-6}$	$1.0 \cdot 10^{-6}$	$1.0 \cdot 10^{-6}$	$1.0 \cdot 10^{-6}$
ϵ_{ρ}	$1.0 \cdot 10^{-3}$	$1.0 \cdot 10^{-3}$	$1.0 \cdot 10^{-3}$	$1.0 \cdot 10^{-4}$
ϵ_{lin}	$1.0 \cdot 10^{-4}$	$1.0 \cdot 10^{-4}$	$1.0 \cdot 10^{-4}$	$1.0 \cdot 10^{-4}$

[a]For the salicylaldimine system the convergence thresholds used in the TI-ADGA are not the same as for the TD-ADGA.

electronic state, which must be explored during the dynamics. This change of PES is a very simple simulation of a photo-excitation of bromine where we assume the excitation to be instantaneous and that the wave-function is simply moved to an excited state of higher energy.

Bromine has two isotopes of nearly equal abundance, ^{79}Br and ^{81}Br, but for simplicity we have chosen to use only the ^{79}Br isotope in our studies. The electronic structure of molecular bromine is very complicated since both relativistic effects and the crossings of multiple surfaces must be considered. We will use a set of potential energy curves for the lowest 23 covalent states of bromine that have been calculated by Gomes et al.[47] In the work by Gomes et al. a set of SPs, calclated at 47 different internuclear distances, have been fitted to an extended Ryberg function in order to obtain relevant spectroscopic constants for molecular bromine. The 47 SPs have been calculated by using complete open shell configuration interaction in a four-component relativistic framework using the aug-cc-pVTZ basis set.[48] However, the fits that they obtain are not satisfactory in the range of internuclear distances that we want to use and we have thus chosen to use the SPs provided by Gomes et al. and fit them to an extended Rydberg function ourselves. The Extended Rydberg

function takes the form[49]

$$V(R) = -D_e(1 + \sum_{k=1}^{n} C_k(R - R_e)^k)\mathrm{e}^{C_1(R-R_e)}, \tag{49}$$

where R_e is the equilibrium distance, and D_e is the dissociation energy, and C_k is a set of n fitting coefficients. One can choose the order n to fit the potential function. We have chosen n=10 similar to Gomes et al. as we found that at higher orders the function were prone to overfitting. Using the extended Rydberg function we fitted the ground state (X: 0_g^+) and the two lowest lying excited states (A′: 2_u and A: 1_u), which are all bounded. Details on the fitting procedure as well as the obtained fitting coefficients can be found in the SI.

For the time-evolution we will use the TDH method to propagate the dynamics. The TDH state is initialized from a VSCF state obtained on an initial PES supplied beforehand and the dynamics are then propagated in time on a different PES. In order to test the accuracy of the simulation when using the TD-ADGA method to obtain the potential during the simulation, we will also carry out an identical TDH simulation but where the potential of the second curve has been supplied beforehand by a TI-ADGA calculation. We will use the Extended Rydberg fits to calculate SPs in both the TI-ADGA and TD-ADGA calculations and the PESs that we use in the TDH simulations will be all fitted using 12th order polynomial fits. In all calculations the dynamics were simulated for at total of $2.0 \cdot 10^5$ a.u. (4.838 ps).

In both the TD-ADGA calculation and the reference calculation, the initial TDH state is prepared as the VSCF ground state on the PES for the A′ state. The TDH time-evolution is then subsequently carried out on the PES for the A state. Since we change the PES, the TDH initial state is no longer a stationary state and the TDH wave function will start to move on the A state PES. We have chosen to use the A′ → A transition, since the transition from the ground state to either the A or A′ state dissociates the bromine dimer.

In both the TD-ADGA and the two TI-ADGA calculations the equilibrium geometry for the A′ state was used as the reference geometry for the PES fits. In the TI-ADGA

calculations to obtain the reference A′ and A states, a mean VSCF density consisting of the lowest eight eigenstates were used. This high number of eigenstates was used to ensure a good description of the PES in a broad domain of the PES. The settings were also modified to place the initial SPs at a displacement of ± 1.1180 Bohr from the used reference distance of 5.1386 Bohr. This was done in to reduce the number of SPs required by the TI-ADGA algorithm as many SPs were placed when extending the potential boundaries using the default initialization. In total 18 SPs were used to fit the A′ surface and 26 SPs were used to fit the A surface.

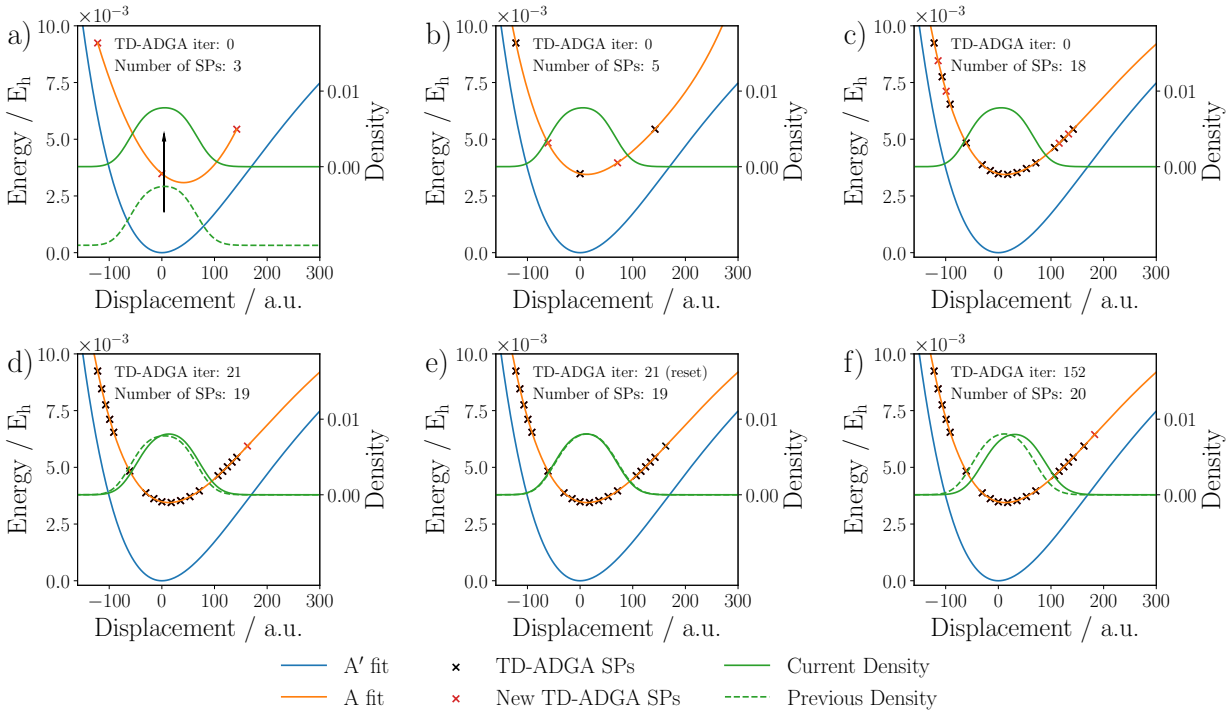


Figure 2: Construction of the PES for the A state in the bromine dimer using the TD-ADGA method. See text for a detailed description.

In Fig. 2, we show how the TD-ADGA algorithm constructs a PES for the A state during the dynamics. Figs. 2a, 2b, and 2c shows the zeroth, first, and fifth sub-iteration of the zeroth TD-ADGA iteration, respectively. In Fig. 2a, the zeroth sub-iteration of the zeroth TD-ADGA iteration is started by inputting the initial density. The initial density is space-smoothened (dashed green line) and then inputted to the TD-ADGA algorithm

(full green line). The TD-ADGA then places SPs at the reference geometry and at two displacements, such that the initial density is contained. A PES is then fitted to the initial SPs and the integration boxes defined by the initial SPs are evaluated. In Fig. 2b, the integration boxes are all sub-divided by adding new SPs and the convergence of each box is evaluated. Integration boxes that are not converged are further sub-divided in the subsequent sub-iterations until convergence is obtained. Fig. 2c shows the PES and SPs obtained after the fifth sub-iteration of the zeroth TD-ADGA iteration. After this sub-iteration, the PES is converged and the TD-ADGA algorithm continues. The values of the integration boxes at this step is saved and used to evaluate the change in all the subsequent TD-ADGA iterations until new SPs are evaluated. The time propagation is now started and the TD-ADGA convergence is checked at intervals of 100 a.u. (2.42 fs). Fig. 2d shows the 21st TD-ADGA iteration, where the density has moved enough for the TD-ADGA to request a new SP. After the new SP has been evaluated the integration boxes are again evaluated and checked for convergence compared to the boxes evaluated just before the SP was added. It is found in this iteration that the PES is converged after only adding a single SP and the TD-ADGA algorithm continues. This TD-ADGA iteration was carried out after 2000 a.u. had passed and thus the dynamics are reset to wave function saved at 1800 a.u. The reset is shown in Fig. 2e, where the current density (full green line) is moved slightly towards a more negative displacement compared to the density in iteration 21 (dashed greeen line). At this point the PES was considered converged and all integration boxes are evaluated with the reset density. Similar to the integrals evaluated in Fig. 2c, these integrals are saved and used to evaluate the change of the integration boxes in the subsequent convergence checks. As the dynamics restart from the reset wave function, the PES used in the propagation is slowly updated from the PES before the new SPs were added to the new PES obtained using the new SPs, in the interval [1800, 1900] a.u. Fig. 2f shows the final TD-ADGA iteration where new SPs are added. This iteration is carried out after a time-propagation of 14900 a.u. A restart similar to the one described for Fig. 2e is then carried out. After this restart,

the PES is considered converged for the rest of the dynamics simulation and no more SPs are added.

The TD-ADGA requests a total of 20 SPs to construct the PES for the A state. This is fewer than the 26 SPs requested by the TI-ADGA calculation in the construction of the reference potential for the A state. However, the TI-ADGA PES covers a much wider area of the PES than the TD-ADGA. The number of SPs placed by the TI-ADGA that are within the maximum and minimum displacements of the TD-ADGA SPs is 16, with two additional SPs placed at slightly larger displacements. The number of SPs placed in the domain relevant for the dynamics is thus comparable between the two PESs. A plot of the fitted A state surfaces and single points obtained using the TI-ADGA and the TD-ADGA can be found in Fig. S1 in the SI.

We now turn our attention to comparing the results of the two dynamics calculations. To compare the dynamics simulations, we compare the autocorrelation function of the two TDH simulations as well as the Fourier-transforms of these. The auto correlation function is defined as the expectation value of the time-evolution operator taking the wave function from time t' to t, $U(t,t')$,

$$S(t,t') = \langle\Psi(t')|U(t,t')|\Psi(t')\rangle = \langle\Psi(t')|\Psi(t)\rangle\,. \tag{50}$$

Determining the autocorrelation function with respect to the initial wave function at t_0 we obtain

$$S(t) = \langle\Psi(t_0)|\Psi(t)\rangle\,. \tag{51}$$

The autocorrelation function is a sensitive property and differences between the two simulations will thus show up in these results. We will also consider the Fourier transform of the

autocorrelations function

$$S(\omega) = \int_{-\infty}^{\infty} S(t) e^{-i\omega t} dt. \tag{52}$$

The Fourier transform yields a spectrum of the frequencies from the autocorrelation function and can show whether the PESs produces the same frequencies even though the autocorrelation function might differ between the two PESs.

Fig. 3 shows the autocorrelation functions and their Fourier transforms obtained in the TD-ADGA and reference calculation. The Fourier transforms have been normalized with respect to the maximum value of the reference calculation. The TD-ADGA results are superimposed on the reference results and no differences between the two can be observed. However the functions are not completely identical. The autocorrelation function differs on the order of 10^{-7} and the difference very slowly increases during the dynamics. The normalized Fourier transform also shows small deviations on the order of 10^{-6} this primarily arises from very slight displacements of the maximum frequency. The most intense peak is at around 2.8 E_h. These difference are to be expected as the PESs are not exactly identical, but obviously the numerical difference should be small for PESs converged in the two different manners. A difference plot between the TD-ADGA and reference results can be found in Fig. S2 in SI.

We have also recorded the mean value of the displacement coordinate during the two simulations. Similar to the results for the autocorrelation function, the mean displacements during the calculation is indistinguishable and the deviations are on the order of 10^{-5}. A figure of these results can be found in in Fig. S3 of the SI.

We finally conclude that the TD-ADGA method can quantitatively reproduce the reference calculation for this system with a similar number of required SP calculations.

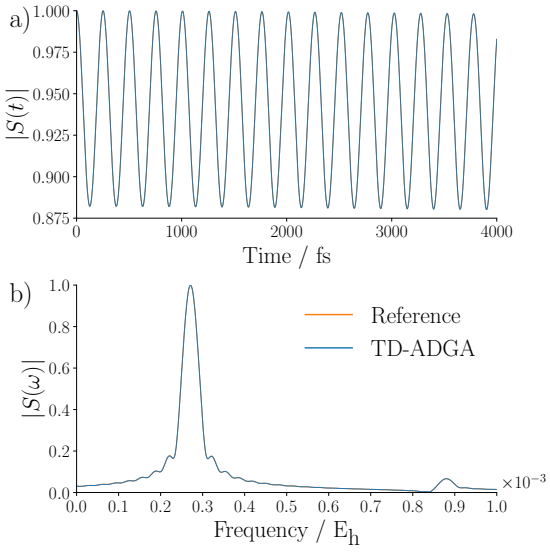


Figure 3: Autocorrelation function (a) and the Fourier transform of the autocorrelation function (b) for the bromine dimer. The Fourier transforms have been normalized with respect to the maximum peak of the reference calculation.

4.3 Dynamics in a one-mode double-well potential

In this subsection we study how the TD-ADGA algorithm fares when it must determine a more difficult potential energy surface. To do this we will investigate how well the TD-ADGA method describe a double-well potential that is discovered during a dynamics simulation. To do this, the time-dependent wave function is initialized from a VSCF state localized in one well of a double-well potential and only during the time-propagation will the wave function reach the other well of the system. The TD-ADGA algorithm must thus be able to detect that the wave function is located in a double-well potential and fit a potential energy surface effectively for the dynamics to be accurate.

In the previous calculation the wave function was initialized from an initial state obtained on a different PES than the one the dynamics was propagated on. In this subsection we will explore an initialization from a stationary VSCF ground state calculation on the potential surface and then initialize dynamics by introducing a time-dependent Hamiltonian. This time-dependent Hamiltonian can, as an example, model a short laser pulse interacting with the vibrational ground state. We note that we in the present case do not consider the

possibility of a electronic excitation as this is beyond the scope of the current investigation.

The laser pulse is simulated by adding a time-dependent perturbation operator to the time-independent Hamiltonian

$$H(t) = H_0 + c(t)H_1, \tag{53}$$

where H_0 is the time-independent Hamiltonian (Eq. 1) described by the PES and kinetic energy operators, H_1 is the perturbation operator, and $c(t)$ is the perturbation parameter which is modelled as a Gaussian pulse

$$c(t) = Ae^{-\frac{(t-t_p)^2}{2\sigma^2}} \cos(\omega(t - t_p) - \phi). \tag{54}$$

In the Gaussian pulse, A is the overall amplitude of the pulse, t_p is the time at which the pulse peaks, σ is the standard deviation of the pulse, ω is the angular frequency, and ϕ an overall phase. For a laser pulse, the physically relevant choice for H_1 would be the dipole operator, but we chose to simply use the displacement coordinate ($H_1 = q$) as the perturbation operator, as it gives a simple interpretation of the pulse and because the goal is simply to start the dynamics.

As a model system to investigate, we have chosen the inversion mode of ammonia. An analytical PES for the inversion in ammonia have been calculated by J. B. Coon et al.[50] and we use this analytical expression as implemented in MidasCpp to calculate SPs that are required by the ADGA algorithms. We use the transition state of the inversion mode (planar D_{3h} structure) as the reference structure and describe the dynamics using a single mode connecting the two degenerate minima (C_{3v} structures).

In order to localize the wave function in one of the double wells, we split the basis set up into two sub-basis sets with no overlap and a boundary basis that overlaps with both basis sets. A VSCF calculation is then carried out in this split basis in order to obtain a set of localized VSCF modals in each basis set. The initial wave function is initialized from the

lowest-energy state which is localized in a single well. Details on how to carry out a VSCF calculation in a split basis can be found in Appendix A.

We carry out three simulations of the ammonia system. We have performed a reference dynamics simulation using a pre-calculated reference TI-ADGA PES, a TD-ADGA dynamics simulation where the PES is determined on the fly, and a rerun dynamics simulation where the dynamics are simulated using the final PES obtained in the TD-ADGA simulation throughout the entire dynamics simulation. For all simulations, the initial state is a localized VSCF state obtained from the pre-calculated TI-ADGA PES.

In the TI-ADGA calculation, an average over the ten lowest energy VSCF densities was used in the PES construction. This is a high number of eigenstates to include but since the VSCF states are pairwise degenerate a high number is required in order to ensure that the PES is described in a broad domain. The TI-ADGA calculation required 25 SPs to converge. Fig. 4a shows the fitted PES and SPs obtained from the TI-ADGA calculation. Fig. 4a also shows the density of the localized VSCF state used as the initial state in all the simulations.

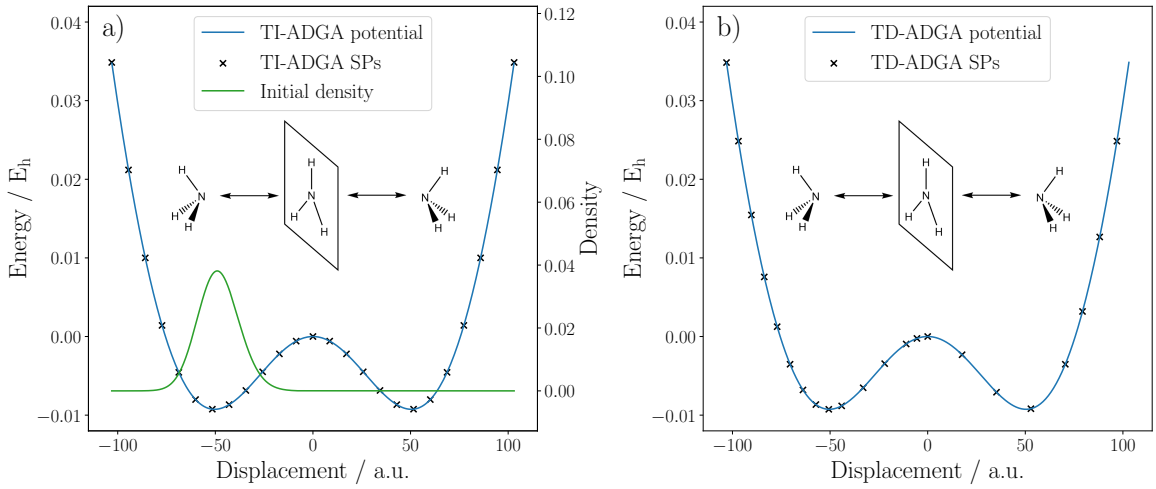


Figure 4: The setup for initializing the dynamics simulations for the ammonia inversion mode. (a) The reference potential for the fitted potential and the SPs are obtained from the reference TI-ADGA calculation. The initial density is the (non-smoothened) density obtained from the localized VSCF state in the left well. (b) The final potential and all SPs obtained from the TD-ADGA calculation.

The dynamics are propagated at the TDH level of theory but by using the MCTDH

module in MidasCpp. This is done in order to add the time-dependent pulse. For all simulations, the initial state is the localized VSCF state obtained from the pre-calculated TI-ADGA PES and the simulation time is $2.0 \cdot 10^5$ a.u. (4.838 ps). The amplitude of the pulse in the simulations was $A = 1.0 \cdot 10^{-5}$ a.u., the peak position of the pulse was at $t_p = 3.5 \cdot 10^4$ a.u. (846.61 fs), and the phase was $\phi = 0$. The angular frequency of the pulse have been chosen to match the excitation energy from the lowest-energy state to the first excited state within the same well obtained from the split basis VSCF calculation, $\omega = 4.4619 \cdot 10^{-3}$ a.u. The standard deviation of the pulse was chosen such that the full width half maximum of the pulse was ten times the oscillation period $\tau_{\text{per}} = 2\pi/\omega$, thus $\sigma = 10\frac{\pi}{\omega\sqrt{2\ln 2}} = 5.9900 \cdot 10^3$ a.u.

We will give a short summary of the dynamics observed during the TD-ADGA time-propagation. A video of the propagation showing the density and potential at selected TD-ADGA iterations can be found in Fig. S4 of the SI. The initial TD-ADGA PES is generated using the localized VSCF density obtained using the reference PES and the split basis. The dynamics is then initialized and since the localized VSCF state is not a stationary state, the wave function moves a little around in the left well. After 16.93 fs (the eighth TD-ADGA iteration) this leads the TD-ADGA to place another SP in the left well. The density is localized in the right well until the amplitude of the perturbation increases enough to perturb the wave function. After 788.56 fs, the density is no longer contained only in the left well and a new SP is added in the right well. At this point more and more density slowly moves to the right well and the TD-ADGA places mores SPs in the right well (and one in the left well) during the next 220 fs. After 1001.42 fs (TD-ADGA iteration 427) no more SPs are placed in the TD-ADGA calculation. For the rest of the simulation, the density moves back and forth between the two wells of the double well system, but always with the largest weight of the wave function being localized in the initial well. Fig. 4b shows the SPs and fitted PES from the TD-ADGA calculation at the end of the simulation. A total of 22 SPs are requested during the TD-ADGA simulation, which is slightly less than the 25 SPs

requested in the TI-ADGA calculation.

To compare the results of the simulations, Fig. 5 shows the expectation of the displacement coordinate, $\langle q(t) \rangle$, for the three simulations. Fig. 5a shows the expectation value of the displacement coordinate in the reference simulation while Fig. 5b shows the value of the perturbation parameter $c(t)$ used to model the Gaussian pulse. The initial value of $\langle q(t) \rangle$ is -48.298 a.u. which corresponds to the initial density shown in Fig. 4a. In the initial 600 fs of the simulation, $\langle q(t) \rangle$ exhibits very small oscillations. This is before the perturbation parameter has become large enough to affect the wavefunction and these oscillations arise from the initial state not being an exact stationary state. As the perturbation parameter increases in amplitude $\langle q(t) \rangle$ begins to oscillate. It is observed that two overall oscillations are induced, one with a short period and one with a longer period.. During the remainder of the simulation, the oscillations in $\langle q(t) \rangle$ continues. The value of $\langle q(t) \rangle$ remain negative for the entire simulation, in agreement with the the simulation video in Fig. S4, where density oscillates back and fourth but is at all times primarily localized in the left well.

The same overall behaviour is seen for the TD-ADGA and the TD-ADGA rerun simulations. For at better comparison of the differences between these two simulations and the reference simulation, the TD-ADGA and TD-ADGA rerun expectation values of the displacement coordinate have been plotted on top of the reference results in Figs. 5c and 5d, respectively. In Fig. 5c it is observed that the values of $\langle q(t) \rangle$ in the TD-ADGA simulation differs from the values of the reference simulation. Small discrepancies can be observed as the oscillations $\langle q(t) \rangle$ are induced and overall it is seen that the displacements of the two simulations deviates a little with the TD-ADGA simulation predicting a smaller mean displacement that the reference simulation. Looking at the TD-ADGA rerun calculation in Fig. 5d a much better agreement with the reference is observed. There is small differences in the smaller oscillations but the TD-ADGA rerun calculation reproduces the reference results very accurately.

The observed difference between the TD-ADGA and TD-ADGA rerun calculations shows

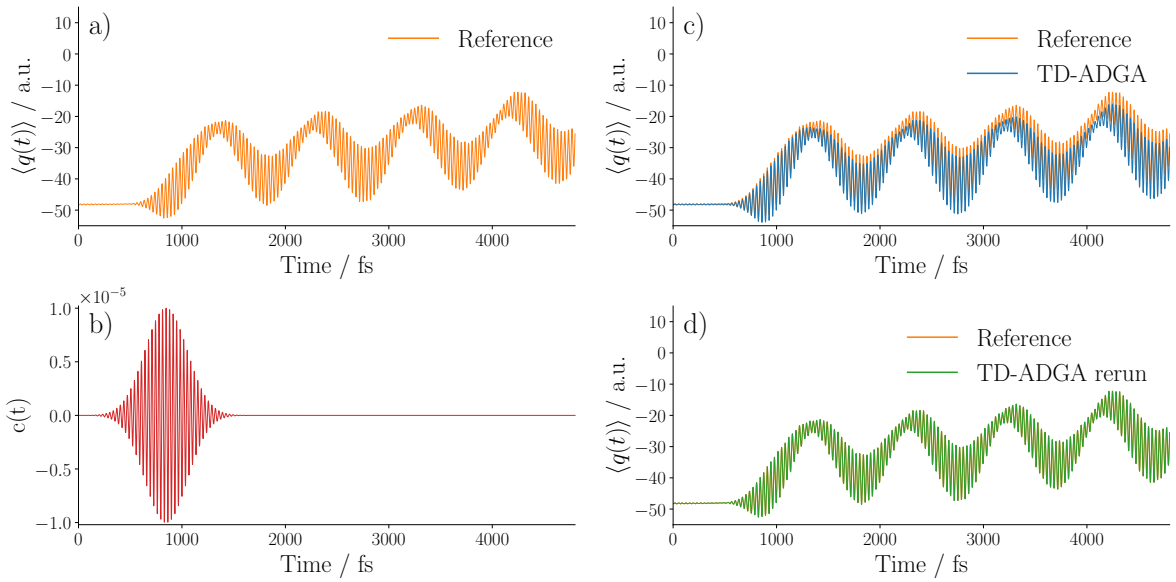


Figure 5: Expectation value of the displacement coordinate in the ammonia inversion mode and the value of the perturbation parameter during the simulation. (a) The expectation value in the reference simulation. (b) Perturbation parameter during all simulations. (c) The expectation value in the TD-ADGA and reference simulations. (d) The expectation value in the TD-ADGA rerun and reference simulations.

that the used restart time is not long enough to eliminate all artefacts from partial restarts with potential updates during the dynamics. However, the similarity of the TD-ADGA rerun calculation with the reference calculation shows that the PES fitted by the TD-ADGA calculation is very similar to the reference PES. To confirm this we carried out an additional TD-ADGA simulation using the full restart scheme. Indeed we find that using the full restart scheme, the reference results can be obtained from a single TD-ADGA simulation. The results from this simulation can be found in Fig. S5 of the SI. Since using longer restart times is not computationally attractive for larger systems, the TD-ADGA rerun approach may very well be an attractive compromise where the initial TD-ADGA calculation map out the PES in all areas of relevance and then the rerun simulation with a fixed PES provides a description of the time-evolution completely free of any artefacts. The extra computational cost added in this extra careful approach depends on both the wave function method and total simulation time, however, from the point of uncovering the physical content of the simulation this extra step might be unnecessary.

4.4 Full mode coupled dynamics of water

We now turn our attention to a multi-dimensional system. Water is an attractive system to benchmark against, as water only has three internal degrees of freedom which makes it possible to easily describe the PES in the full coupling limit and also carry out the dynamics at the full coupling level. The three internal degrees of freedom are described using normal coordinates and we enumerate the normal coordinates as q_0, q_1, and q_2 for the symmetric bend, symmetric stretch, and asymmetric stretch, respectively.

We use the analytical potential for water determined by Partridge and Schwenke[51] to obtain SP energies as required by the TD-ADGA algorithm and then we fit these SPs to obtain the PES. We will use the full three-mode representation of the PES and in each of the seven MCs, the PES is fitted to a polynomial function in the displacement coordinates.

As a reference potential, we initially determine a TI-ADGA PES using the four lowest energy VSCF states in the average density and coupling all vibrational modes. To obtain this PES the TI-ADGA requested 2052 SPs.

We propagate the dynamics using MCTDH with six active modals for each mode and as an initial wave function we use the full vibrational configuration interaction (FVCI) wave function obtained from the reference three-mode PES and using the six lowest-energy VSCF modals from each mode as the basis. We will propagate the wave function through a simulation time of $2.0{\cdot}10^5$ a.u. (4.838 ps) and similar to the ammonia double well system, we will initialize the dynamics by simulating a short laser pulse. We will thus add a time-dependent perturbation to the system according to Eqs. 53 and 54. We will still only consider a perturbation directly in one displacement coordinate and we choose to use the symmetric bend $H_1 = q_0$. We use an amplitude of the pulse of $A = 2.00 \cdot 10^{-5}$ a.u., the peak position of the pulse was at $t_p = 2.0 \cdot 10^4$ a.u. and the phase was $\phi = 0$. The angular frequency of the pulse was chosen to match the excitation energy for the fundamental excitation of the symmetric bend obtained from a FVCI response calculation using the reference PES, $\omega = 7.2095 \cdot 10^{-3}$ a.u. Using the condition that the full width half maximum of the pulse equals ten times the

oscillation period, the standard deviation was determined to be $\sigma = 10\frac{\pi}{\omega\sqrt{2\ln 2}} = 3.6904 \cdot 10^3$ a.u.

We carry out three MCTDH simulations. A reference calculation using the reference potential, a TD-ADGA calculation, and a rerun calculation, where the final TD-ADGA PES is used to propagate the dynamics.

To initialize the TD-ADGA surface, the density of the reference VSCF state of the FVCI wave function was inputted to the TD-ADGA algoritmh as an initial density. The TD-ADGA initially requested 747 SPs to converge the three-mode PES using the initial density. At the end of the simulation the TD-ADGA has requested a total of 1585 SPs. Compared to the 2052 SPs requested by the TI-ADGA calculation used to obtain the reference PES, the TD-ADGA provides a reduction of 22.30%.

Fig. 6 shows the expectation value of the displacement coordinate for modes q_0 and q_1 (symmetric bend and symmetric stretch), as well as the perturbation parameter of the perturbing pulse during the three simulations. The expectation value of the q_2 mode is not shown since the changes observed are on the order of magnitude of 10^{-4} a.u. in the displacement coordinate. The lack of large oscillations in the q_2 mode shows that there is no strong coupling between the symmetric bend (q_0) and the asymmetric stretch modes, when the former is excited using the present perturbation. For all simulations, it is observed from Fig. 6 that as the perturbation increases in amplitude, both small and large period oscillations are induced in the q_0 mode. These oscillations persist throughout the rest of the simulation time. Shortly delayed after the initial oscillations in the q_0 mode are observed, oscillations are induced in the q_1 mode which also continues through the rest of the simulation. This redistribution of energy is to be expected since both q_0 and q_1 are totally symmetric.

By inspection of the first and second column of plots in Fig. 6, it can be seen that the overall structure of $\langle q_0 \rangle$ and $\langle q_1 \rangle$ are similar between the reference and TD-ADGA simulations. However, while $\langle q_0 \rangle$ is nearly identical for the two calculation throughout the simulation time, $\langle q_1 \rangle$ show a larger discrepancy. In the $\langle q_1 \rangle$ plots, the structure of the long period

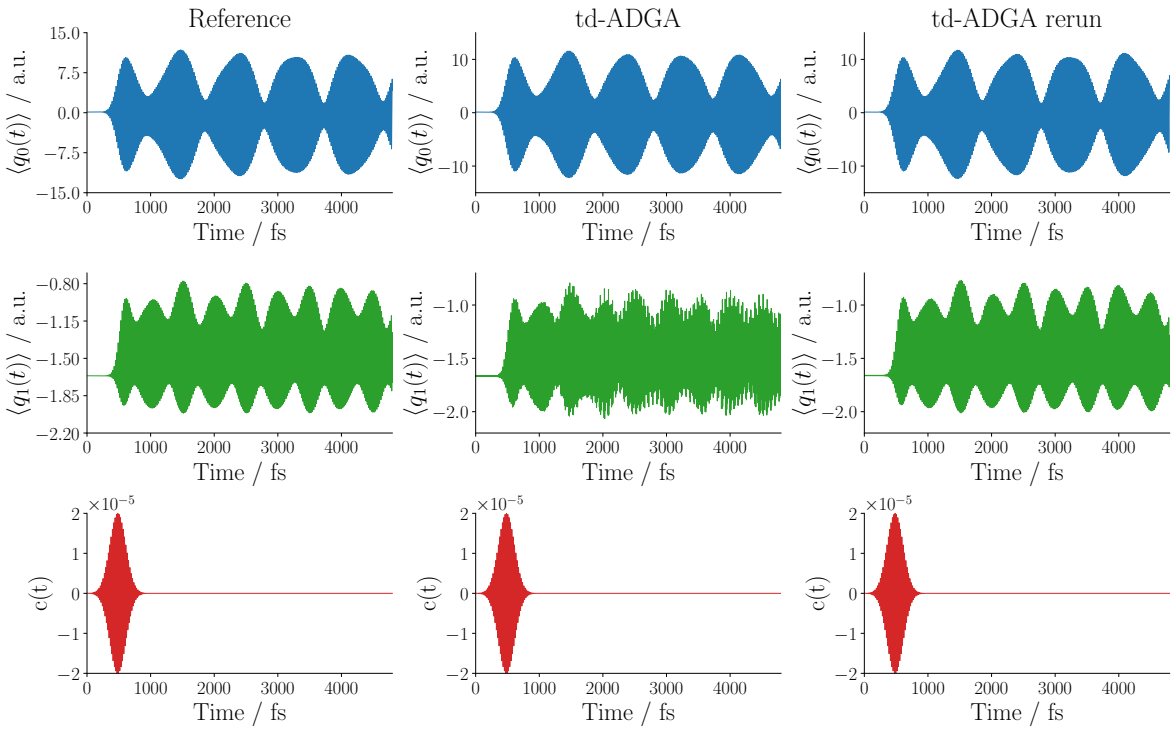


Figure 6: The expectation value of the displacement coordinates q_0 and q_1 and the value of the perturbing Gaussian pulse in the q_0 coordinate during the TD-ADGA and the reference simulation.

oscillations are similar in shape, but the reference results display "smooth" oscillations while the TD-ADGA results starts out smooth and the become more and more "jagged" as the simulation time proceed.

In the third column of Fig. 6, the TD-ADGA rerun results for the expectation values are shown. In the rerun calculation the evolution of the expectation values are very near to the reference results and in particular the $\langle q_1 \rangle$ results are here also in very good agreement with the reference. Similar to the results from the ammonia system, this large change shows that the restarts carried out during the TD-ADGA simulation is not sufficient to eliminate artifacts from fitting the PES on the fly. However, the obtained PES is in excellent agreement with the PES of the reference.

We finally note that in the beginning of the simulation (where the perturbation has not yet induced any oscillations in the q_0 mode), $\langle q_1 \rangle$ has a constant value in the reference simulation while small oscillations are observed in the same time-interval in the TD-ADGA and TD-ADGA rerun simulations. In the reference calculation, we expect the expectation value of q_1 to be constant in the beginning of the simulation because the initial wave function is a stationary state of the potential. This is not the case in the TD-ADGA and TD-ADGA rerun calculations, because the initial state is created from the reference potential and it is thus not exactly a stationary state for the TD-ADGA PES. Small oscillations in the $\langle q_0 \rangle$ values are similarly observed in the TD-ADGA results, but these are visible seen in Fig. 6 due to the larger scale on the y-axis.

We now consider the autocorrelation function and its normalized Fourier transform obtained from the three simulations. Figs. 7 and 8 shows these for the TD-ADGA and TD-ADGA rerun results, respectively, plotted on top of the reference results.

For all three simulations, the autocorrelation function remains one during the beginning of the simulation. Around the time where the time-dependent perturbation peaks the autocorrelation quickly drops to a value around 0.74, around which it stays for the rest of the simulation.

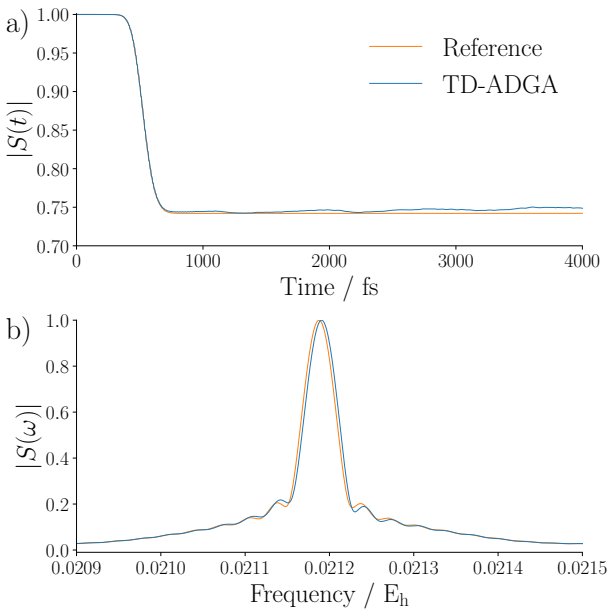


Figure 7: Absolute value of the autocorrelation function (a) and its Fourier transform (b) for the TD-ADGA calculation and the reference calculation. The autocorrelation function is only shown for the initial 4000 fs but the Fourier transform is obtained using the results from the entire simulation. The Fourier transforms have been normalized with respect to the maximum peak of the reference calculation.

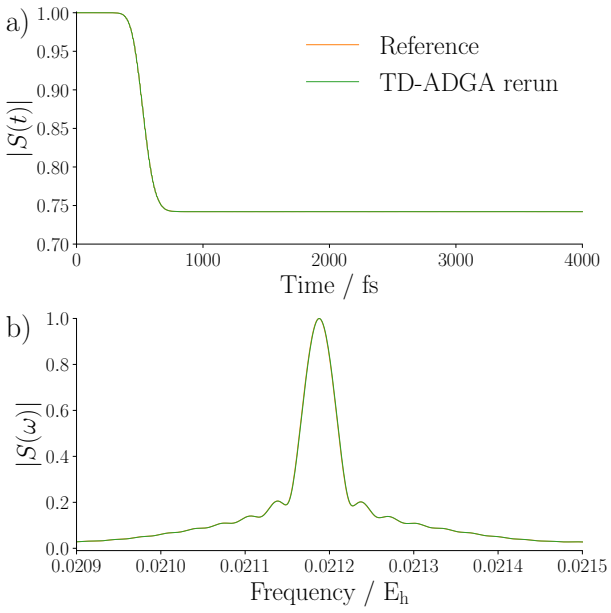


Figure 8: Absolute value of the autocorrelation function (a) and its Fourier transform (b) for the TD-ADGA rerun calculation and the reference calculation. The autocorrelation function is only shown for the initial 4000 fs but the Fourier transform is obtained using the results from the entire simulation. The Fourier transforms have been normalized with respect to the maximum peak of the reference calculation.

For the TD-ADGA results shown in Fig. 7a, a discrepancy from the reference simulation is again observed. On the scale showed in Fig. 7a, the reference autocorrelation function is at a constant value while the TD-ADGA results are non-constant and at a higher value. The Fourier transform shown in Fig. 7b are very similar for the TD-ADGA and reference simulations with the TD-ADGA spectrum being shifted slightly larger frequencies.

Fig. 8a shows the autocorelation function obtained in the TD-ADGA rerun simulation. These results are observed to be in excellent agreement with the reference and as the two results cannot be distinguished in Fig. 8a. The difference between the two autocorrelation functions are on the order of magnitude of 10^{-4} and from a plot with appropriate scaling on the y-axis, it can be observed that the TD-ADGA rerun autocorrelation function has a different oscillation pattern than the reference simulation. A plot displaying the difference between the TD-ADGA rerun and reference autocorrelation functions can be found in Fig. S6 of the SI. Considering the Fourier transform shown in Fig. 8b, the TD-ADGA rerun spectrum coincide almost perfectly with the reference spectrum.

Based on the above results we conclude for the water system that the TD-ADGA algorithm, combined with a final rerun, can provide quantitative dynamics and at the same time using only a reduced number of SPs.

4.5 Keto-enol tautomerization of salicylaldimine

As a final study, we will investigate the keto-enol tautomerization of salicylaldimine. This system allows us to test how the TD-ADGA framework fares when applied to a chemical problem with many modes and complex dynamics. In the tautomerization, a hydrogen is transferred between a nitrogen and an oxygen atom to form either the keto or enol form of salicylaldimine. We will in this subsection study the ability of the TD-ADGA to describe the reaction as salicylaldimine reacts from the enol to the keto form. The following description of the setup of the TD-ADGA calculation (excluding the construction of the full reference potential) outlines how the TD-ADGA can be applied to other chemical systems

and reactions.

To obtain a suitable set of coordinates to describe the intramolecular proton transfer, we initially optimized the salicylaldimine transition state (TS) for the transfer step. We then calculated the Hessian at the TS and used this to obtain a set of 42 normal mode coordinates. Following previous work on this system,[40,52] we use only a minimal number of these modes in order to describe the dynamics at the fully coupled MCTDH level of theory. We include the modes q_0 (TS mode), q_{15}, q_{17}, q_{19} (displacements of the ring structure), q_{37} (wagging of outer hydrogens, deformation of the ring structure, and internal displacement of the transfer hydrogen), and q_{41} (movement of the transfer hydrogen perpendicular to the TS mode). The six modes included in the reduced system are illustrated in Fig. 9 together with the associated harmonic wave numbers. The TS structure and the displacement coordinates can be found in the SI.

Since our goal is to benchmark the performance of the TD-ADGA method and not a high-accuracy investigation of the tautomerization, we will limit ourself to the Hartree−Fock method[53,54] with the 6-31G basis set[55] to describe the electronic structure of salicylaldimine. The structure optimization, Hessian calculation, and all SP calculations were all carried out using the Turbomole program.[56,57]

Computing the fully coupled six-mode PES for the reduced salicylaldimine system is an immense task, and we will accordingly represent the PES with restrictions in the included mode-couplings. We construct a MCR consisting of all one-mode (6) and two-mode combinations (15) and we further add all three-mode combinations including the TS mode (10). This is obtained through including MCs using Eq.(10) with $w_m = 1$ for all modes except the TS mode for which $w_m = 0$. This keeps the number of MCs at a manageable level (31) while giving a high coupling level for the TS mode. This MCR is used both in the TD-ADGA calculation and in the reference PES.

To obtain the reference PES, we carried out a TI-ADGA calculation using an average VSCF density obtained from the eight lowest energy modals in the TS mode and the four

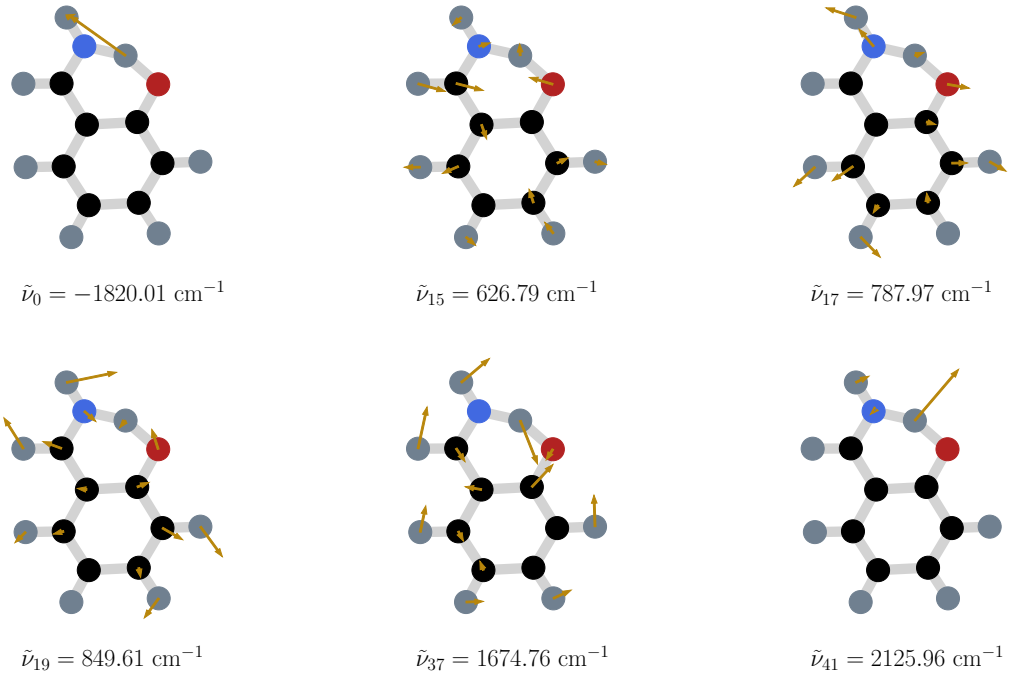


Figure 9: Illustration of the normal coordinates and the harmonic frequencies for each of the six modes included in the salicylaldimine simulations. Displacement vectors (arrows) with a length smaller than 0.05 a_0 have been excluded. The displacement vectors of mode 15, 17, 19, and 37 have been scaled with a factor of ten while the displacement vectors of mode 0 and 41 have been scaled with a factor of four. Atom labels: hydrogen (grey), carbon (black), nitrogen (blue), and oxygen (red).

lowest energy states in the remaining five modals of the system. Further, we did not use the default threshold for the relative convergence threshold ($\varepsilon_{\text{rel}} = 1.0 \cdot 10^{-2}$) but a tighter value of $1.0 \cdot 10^{-3}$ in order to get an accurate representation of the PES. To obtain this PES the TI-ADGA requested a total of 45787 SPs.

In the TD-ADGA simulation we also did not use the default values for the convergence threshold. Instead we used the values listed in Table 1, which will be repeated here for convenience. $\epsilon_{\text{rel}} = 1.0 \cdot 10^{-4}$, $\epsilon_{\text{abs}} = 1.0 \cdot 10^{-6}$, $\epsilon_{\rho} = 1.0 \cdot 10^{-4}$, and $\epsilon_{\text{lin}} = 1.0 \cdot 10^{-4}$. We note that these thresholds do not mirror the thresholds of the TI-ADGA calculation used to obtain the reference PES and we will discuss this further at the end of the current subsection. At the end of the TD-ADGA simulation a total of 64017 SPs had been requested which is an increase of 39.8 % compared to the TI-ADGA reference. This is not surprising considering that the TD-ADGA thresholds are considerably tighter than those used in the TI-ADGA calculation.

In the simulations, the dynamics of the six modes are described at the MCTDH level including all mode couplings (for the six-mode model) and the dynamics are propagated for a total of 5000 a.u. (120.9 fs). The initial state used in the dynamics is created from a ground state VSCF wave function obtained by using only the one-mode part of the reference potential and where the double well in the q_0 mode is replaced by a displaced harmonic potential. This harmonic potential is defined such that the harmonic frequency is 2493.55 cm^{-1} and the Gaussian wave function is centered around a displacement of 14.27 a.u. in the mass and frequency scaled coordinates used in the simulation. Thus, the initial state is a Hartree product of the VSCF modals obtained from the q_{15}, q_{17}, q_{19}, q_{37}, and q_{41} one-mode cuts and a Gaussian modal displaced away from the double well minima of the enol form towards the TS. The harmonic potential and the corresponding VSCF modal in the q_0 mode are shown in Fig. 10.

Firstly, we compare the flux over the transition state in the q_0 mode. The flux is obtained

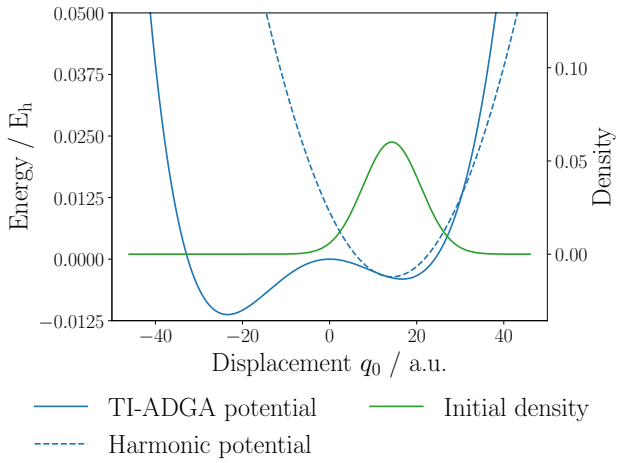


Figure 10: Initial state in the q_0 TS mode. The solid blue line is the one-mode potential determined by the reference TI-ADGA potential, the dashed blue line is the harmonic potential used to obtain the initial state, and the green solid line is the one-mode density of the state obtained from the harmonic potential.

during the two simulations from the expectation value of the flux operator[3]

$$F_{q_0=0} = i[H, \Theta(q_0)] = -\frac{i}{2}\left(\frac{\partial}{\partial q_0}\delta(q_0) + \delta(q_0)\frac{\partial}{\partial q_0}\right), \tag{55}$$

where the last equality is only true for rectilinear coordinates and where $\Theta(q_0)$ is the Heaviside step function and $\delta(q_0)$ is the Dirac-delta function.

The fluxes obtained in the two simulations are shown in Fig. 11. It is observed that the TD-ADGA and reference simulation results are not quantitatively identical, but they do display a similar qualitative behaviour. For both simulations, the flux initially grows negative to a minimum value after 3.35 fs after which the flux increases in values until it reaches a value of zero after approximately 11 fs. This corresponds to some part of the wave function (initially localized in the right well) passes through the TS to the left well, while the major part of the wave function is still localized in the right well. For the rest of the simulation the flux values fluctuates around zero. The fluctuations of the flux value arises from the wave function oscillating back and forth in both the left and the right well with enough energy to pass through the TS. At some times more of the wave function is moving

from the left well to the right well, resulting in a net positive flux, while at at other times the more of the wave function is moving from the right to the left well, resulting in a net negative flux. The fluctuations in the flux values decreases with the simulation time, as the vibrational energy is transferred to other vibrational modes in the system making it harder to pass through the TS.

During the first 50 fs the two simulations are in good agreement, but after 50 fs they differ from each other. The flux obtained in the TD-ADGA simulation decreases after the 50 fs, while the flux from the reference is roughly constant for another 10 fs before it starts to decrease. Thus, after 50 fs the fluxes obtained from the two simulations diverge and are no longer agreeing. However, the qualitative behaviour of a flux value fluctuating around zero is the same in the two simulations after this point.

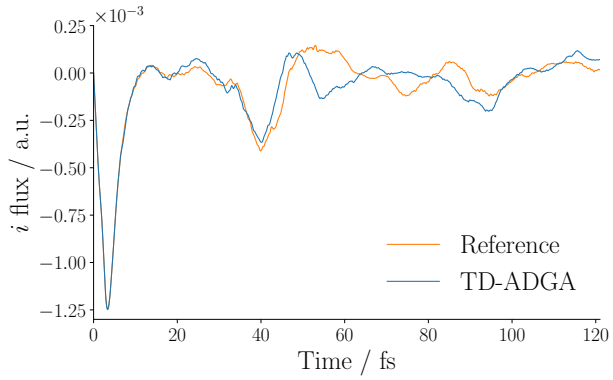


Figure 11: Flux over the transition state during the TD-ADGA and the reference simulation.

The results for the autocorrelation function and its Fourier transform obtained from the two simulations are shown in Fig. 12. Fig. 12a shows the values of the autocorrelation function during the two simulations. Similar to the results for the flux, we observe a very good agreement for between the two simulations for the initial 50 fs and after this point the similarity decreases. The results are however, qualitatively the same for the entire simulation time. Looking at the Fourier transforms in Fig. 12b, a very good agreement between the TD-ADGA and Reference autocorrelation spectra is observed. The largest difference between the two autocorrelation spectra is found in the frequency interval between 0.015 and 0.020

E_h, where two peaks are present. In this region the TD-ADGA peaks are shifted to a higher frequency and the amplitude of the second peak is smaller.

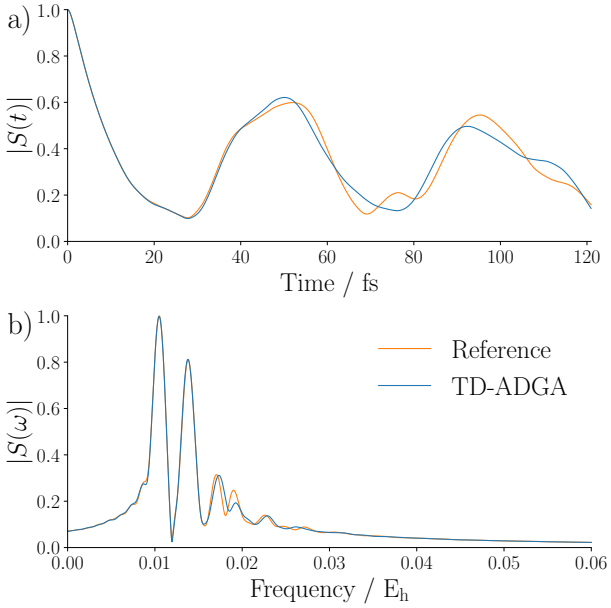


Figure 12: Absolute value of the autocorrelation function (a) and its Fourier transform (b) for the TD-ADGA calculation and the reference calculation. The Fourier transforms have been normalized to one with respect to the maximum peak of the reference calculation.

We have also carried out a TD-ADGA rerun simulation similar to the other molecular systems but we observe no changes in the results compared to the TD-ADGA results. Figures with the TD-ADGA rerun results can be found in the SI.

We have observed that the results of the TD-ADGA simulation varies to some extent when the convergence thresholds are varied. I.e. qualitatively similar results are obtained, but the quantitative agreement between different settings varies as the flux-changes appear at different times. The TD-ADGA thresholds presented above have been chosen as they give reasonable results while not being so tight that the number of required SPs becomes very large. However, it seems difficult to push accuracy further, with respect to both converging internally and with respect to the results obtained with the TI-ADGA PES. In the present fits we use global polynomial fit functions for each MCs and the details of number, position, and overall distribution of points influence the obtained fit, and lead to small global variations.

As expected, tightened thresholds leads to more SPs being required in all MCs. But more SPs are added in the right well regions, where the wave function is initially located, than in the left well. It is seemingly delicate to obtain hard convergence in TD-ADGA in such cases, perhaps due to the global polynomial fit basis causing slightly unbalanced fits and/or distributing tiny changes globally. We have also observed that varying the order of the polynomials used in the fitting leads to significant changes in the results even when using the same thresholds. Contrary, the simulations carried out with the TI-ADGA PES is converged when using the tighter value described above and varies only slightly when the thresholds are tightened further. However, changing the fit order in the TI-ADGA calculation also changes the obtained results. Some dependency of details in the placement of points and choice of fit-bases in principle applies to any fitting approach. However, the level of instability found in TD-ADGA is larger than in the TI-ADGA PES. In the latter, the full region of the PES is determined from the beginning. Thus after the one-mode cuts are converged, SPs in the higher order MCs are only placed by subdividing boxes and this leads to a much more even distribution of the SPs in the TI-ADGA. This opens the question whether the TD-ADGA would be more stable, and even more efficient, by i) introducing protocols for keeping SPs somewhat more evenly distributed, or ii) replacing the global polynomial fits in each MC with some kind of local fit functions, such that only nearby SPs dominate the fitted PES in each region. The latter can support better possibilities for improving the PES locally without giving global effects that may be partially artificial and potentially cause the PES to be hard to converge. The implementation and tests of such approaches and the further uncovering of these finer details are major undertakings and will be included in future research.

5 Summary and Conclusion

We have introduced a quasi-direct quantum dynamics method, where the PES is constructed as needed by the dynamics by on the fly acquisition of new SPs together with refittings and

continuous updates of the PES. The suggested TD-ADGA is a generalization of the previous ADGA (TI-ADGA) where the evolving nuclear density is put central in the decisions of when and where to update the PES with new SPs. We have verified the approach for both single- and double-minimum PESs, one- and multi-dimensional PESs, and with and without time-dependent terms in the Hamiltonian. In all cases, the results obtained with TD-ADGA are near to the results obtained in reference computations where extensive reference PESs have been used. In particular, an approach where first the whole dynamics is studied with a partial restart followed by a final full restart dynamics calculation has shown a high level of agreement with the results of accurate reference computations. We believe the method opens a whole new framework for computing quantum dynamics accurately, also for larger systems. As long as the electronic structure code can run fairly automatic, the whole procedure is essentially black box, and is a much needed mid-position between using a pre-computed PES and doing full direct (quantum) dynamics. Still, the method needs to be tested in many new contexts.

It is important to emphasize that the method is not limited to normal coordinates. If the density can be provided in any other set of coordinates, this coordinate set can equally well be used, and this will be exploited in forthcoming work. In the present work, we have employed the developed methods in conjunction with the TDH and MCTDH methods. Any other wave function propagation method capable of providing time-dependent reduced one-body densities will also be applicable in this context. We will in future work use the methodology in conjunction with the time-dependent vibrational coupled cluster with time-dependent modals methods,[34–36] offering new attractive compromises between accuracy and efficiency.

In the present work we considered bound dynamics on a single PES. Consideration of for example photodissociation dynamics should also be fully possible, but requires dynamics method capable of handling simultaneously the different wave function components without generating artificial density distributions. The extension to simultaneous dynamics on

multiple PESs and including non-adiabatic couplings is another interesting but complicated topic of future research, where clearly extension to multi-state ADGA selection criteria are needed.

Further tuning of the algorithms for the distribution of grid points and the optimal fit bases for representing the PES can likely lead to improved options for obtaining good compromises between accuracy and efficiency. Finally, we have in this work emphasised the learning of the PES based on physical ideas and refitting to known/given analytical forms. Similar to how TI-ADGA has been combined with probabilistic ML methods like GPR with encouraging results,[18] it is a very promising perspective to use GPR to accelerate the accurate construction of the PES relative to the number of SPs. This does, however, require some further extension on the decision criteria as well as significant numerical exploration.

Supporting Information Available

The following files are available free of charge.

- Additional computational details and results (PDF). Table S1: Nuclear wave function basis set details. Table S2: Fitting coefficient for each of the bromine PES fits. Figure S1: Fitted PES and required single points for the A state PES obtained by the TI-ADGA and TD-ADGA methods. Figure S2: Difference plots for the autocorrelation function and the Fourier transform of the autocorrelation function for the bromine dimer. Figure S3: Expectation value of the displacement coordinate for the bromine dimer. Figure S4: Animation of selected TD-ADGA iterations in the ammonia simulation. Figure S5: Expectation value of the displacement coordinate in the ammonia inversion mode obtained using the TD-ADGA with the full restart scheme. Figure S6: Absolute value of the autocorrelation function for for water in the TD-ADGA rerun calculation and the reference calculation. Figure S7: Flux over the transition state for salicylaldimine in the TD-ADGA rerun calculation and the reference calculation.

Figure S8: Absolute value of the autocorrelation function and the Fourier transform of the autocorrelation function for salicylaldimine in the TD-ADGA rerun calculation and the reference calculation.

- salicylaldimine_6d.mmol (TXT): Transition state structure and corresponding normal coordinates in Midas molecule format

Acknowledgement

The authors acknowledge support from the Independent Research Fund Denmark through Grant No. 1026-00122B. Computations were performed at the Centre for Scientific Computing Aarhus (CSCAA).

This work was supported by the Danish National Research Foundation through the Center of Excellence for Chemistry of Clouds (Grant Agreement No: DNRF172).

A Obtaining VSCF solutions localized in space by using a split B-spline basis

In this appendix we will show how local VSCF solutions can be obtained by dividing a B-spline basis set into smaller non-overlapping basis sets and then finally re-obtaining the completeness of the basis.

We consider a basis set of B-spline functions in one-dimension $\{\mathcal{B}^m\}$, where the basis functions are B-spline functions of the displacement coordinate of the vibrational mode that we consider $B_i^m(q_m)$. The B-spline have a limited domain where they are non-vanishing and are thus localized in space. We want to divide $\{\mathcal{B}^m\}$ into a number of subsets with no overlap so we can find solutions to the time-independent nuclear Schrödinger equation that are localized in these basis functions and thus localised in space. Due to the localized nature of B-spline functions we can at any given value q_m^{split} divide $\{\mathcal{B}^m\}$ into three subsets, where

two of the sets, S_1 and S_2, are completely orthogonal to one-another and the a third set, which we denote the boundary set b_{12}, is non-orthogonal to both S_1 and S_2. The three sets are uniquely defined by defining b_{12} as containing only B-spline functions with a non-vanishing value at q_m^{split}.

$$B_i^m(q_m) \in b_{12} \quad | \quad B_i^m(q_m^{\text{split}}) \neq 0. \tag{56}$$

The S_1 and S_2 sets are then subsequently defined as containing all the remaining B-spline functions which are localized at q_m values that are smaller than and larger than q_m^{split}, respectively.

$$B_i^m(q_m) \in S_1 \quad | \quad B_i^m(q_m) = 0 \,\forall\, q_m \geq q_m^{\text{split}}, \tag{57}$$

$$B_i^m(q_m) \in S_2 \quad | \quad B_i^m(q_m) = 0 \,\forall\, q_m \leq q_m^{\text{split}}. \tag{58}$$

It is possible to come up with other partitioning schemes but for our purposes it is convenient to make the boundary basis as small as possible as we are interested in solutions in the S_1 and S_2 sets. The boundary basis thus takes the role of an auxiliary basis that we need to ensure completeness in the basis.

The three bases now obtained can each be further sub-divided in order to create a further division of the basis. A primitive B-spline basis of n subdivisions can generally be written as a set of the basis sets

$$\{\mathcal{B}^m\} = \Big\{\{B_i^m(q_m) \in S_k\},\, \{B_i^m(q_m) \in b_{kl}\}\Big\}, \quad 1 \leq k \leq n+1,\, l = k+1 \leq n+1. \tag{59}$$

We will in the following keep to the simple example of a single division of the basis as this is relevant for localizing a solution in a double well potential, but the results can easily be generalized to further subdivisions.

As mentioned above we have at this point ensured that the two sub-basis sets S_1 and S_2

are completely orthogonal to each other as they span different domains in space

$$\left\langle B_i^{m,S_k} \middle| B_j^{m,S_l} \right\rangle = 0, \text{ if } k \neq l. \tag{60}$$

We now consider a VSCF calculation in the basis $\{\mathcal{B}^m\}$. In the VSCF algorithm we need, for each mode in the system, to solve the eigenvalue equation

$$\mathbf{F}^m \mathbf{C}^m = \mathbf{S}^m \mathbf{C}^m \mathbf{E}^m, \tag{61}$$

where $\mathbf{F}^m$ is the Fock matrix containing all Fock matrix elements between the primitive B-splines, $\mathbf{S}^m$ is the overlap matrix containing the overlap of all the primitive B-splines, $\mathbf{C}^m$ is the coefficient matrix that contain the eigenfunctions as column vectors, and finally $\mathbf{E}^m$ is a diagonal matrix containing the eigenvalues for each eigenstate.

Employing the division of $\{\mathcal{B}^m\}$ in Eq. 59, we want to solve the VSCF eigenvalue equation for each of the sub-basis sets independently. This can be done if the $\mathbf{F}^m$ and $\mathbf{S}^m$ matrices are block diagonal in the split basis. In fact it holds that

$$F_{ij} = S_{ij} = 0 \text{ for } \left(|B_i^m\rangle \in S_k \wedge \left|B_j^m\right\rangle \in S_l \right) \wedge k \neq l. \tag{62}$$

However, there are formally some non-zero matrix elements between the boundary region and the neighboring B-spline functions. We thus enforce additional block diagonality at this point by requiring that also all other coupling elements between different basis blocks are set to zero in the Fock and overlap matrices,

$$F_{ij} = S_{ij} \to 0 \text{ if } \left(|B_i^m\rangle \vee \left|B_j^m\right\rangle \in b_{k,l} \right) \wedge \left(|B_i^m\rangle \vee \left|B_j^m\right\rangle \in S_k \right). \tag{63}$$

This yields a block-diagonal eigenvalue equation and the VSCF modals that are obtained are given as linear combinations of the primitive B-spline basis functions where only basis

functions from one set contributes

$$\left|\tilde{\phi}_k^{m,U}\right\rangle = \sum_i C_{ik}^m \left|B_i^m\right\rangle, \quad C_{ik}^m = 0 \text{ if } \left|B_i^m\right\rangle \notin U. \tag{64}$$

The coefficient matrix can thus be divided into three matrices $\mathbf{C}^{m,S_1}$, $\mathbf{C}^{m,b_{12}}$, and $\mathbf{C}^{m,S_2}$ which have dimensions $N \times N_U$ where N is the number of B-spline function in $\{\mathcal{B}\}$ and N_U is the number of eigenstates in one of the set U.

VSCF modals from the S_1 and S_2 sets are all orthonormal

$$\left\langle \tilde{\phi}_i^{m,S_k} \middle| \tilde{\phi}_j^{m,S_l} \right\rangle = \delta_{ij}\delta_{kl}, \tag{65}$$

but the VSCF modals from the b_{12} set are not orthogonal to the VSCF modals from the S_1 and S_2 sets since Eq. 63 is enforced and not exact. Often we prefer to use an orthonormal basis so if we want to use a localized VSCF basis in subsequent calculations it is desirable to orthonormalize it.

To obtain a complete orthonormal basis we must thus ensure that the eigenstates in the b_{12} set are orthogonal to the VSCF eigenstates from the S_1 and S_2 sets. We carry out this reorthogonalization of the b_{12} VSCF modals by projecting out the overlap with the VSCF modals in S_1 and S_2 sets

$$\left|\check{\phi}_k^{m,b_{12}}\right\rangle = O^{S_2} O^{S_1} \left|\tilde{\phi}_k^{m,b_{12}}\right\rangle, \tag{66}$$

where O^{S_1} and O^{S_2} are the orthogonal compliment to the S_1 and S_2 sets respectively

$$O^{S_k} = 1 - P^{S_k}, \tag{67}$$

$$P^{S_k} = \sum_i \left|\tilde{\phi}_i^{m,S_k}\right\rangle \left\langle \tilde{\phi}_i^{m,S_k} \right|. \tag{68}$$

After this projection has been carried out, the new functions spanning b_{12} are no longer

eigenstates of the VSCF equation we solved and thus no longer orthonormal among themselves. This can be cured by for example a Gram-Schmidt orhogonalization. However, we follow an approach that preserves as much as possible of the VSCF nature of the basis while maintaining that as many as possible of the basis functions are purely local. First a normalization is carried out

$$\left|\hat{\phi}_k^{m,b_{12}}\right\rangle = \left|\check{\phi}_k^{m,b_{12}}\right\rangle \left(\left\langle \check{\phi}_k^{m,b_{12}} \middle| \check{\phi}_k^{m,b_{12}} \right\rangle \right)^{-1/2}. \tag{69}$$

The purified coefficient matrix is a matrix containing the coefficients for expressing $\left|\hat{\phi}_k^{m,b_{12}}\right\rangle$ in the primitive B-spline basis

$$\left|\hat{\phi}_k^{m,b_{12}}\right\rangle = \sum_i \hat{C}_{ik}^m \left|B_i^m\right\rangle. \tag{70}$$

We note that these functions can have contributions from all B-spline basis functions due the projection carried out and the $\hat{\mathbf{C}}^m$ matrix is thus a $N \times N_{b_{12}}$ matrix. Next, we chose to orthonormalize the functions by solving the VSCF eigenvalue equation in this purified b_{12} basis, which is now orthogonal to all VSCF modals in the S_1 and S_2 sets. We thus solve the equations

$$\bar{\mathbf{F}}^m \bar{\mathbf{C}}^m = \bar{\mathbf{S}}^m \bar{\mathbf{C}}^m \bar{\mathbf{E}}^m, \tag{71}$$

where the purified basis Fock and overlap matrices are obtained by transforming the matrices in the primitive basis with the coefficient matrices for the purified basis functions in the b_{12} set

$$\bar{\mathbf{F}}^m = (\hat{\mathbf{C}}^m)^T \mathbf{F}^m \hat{\mathbf{C}}^m, \tag{72}$$

$$\bar{\mathbf{S}}^m = (\hat{\mathbf{C}}^m)^T \mathbf{S}^m \hat{\mathbf{C}}^m. \tag{73}$$

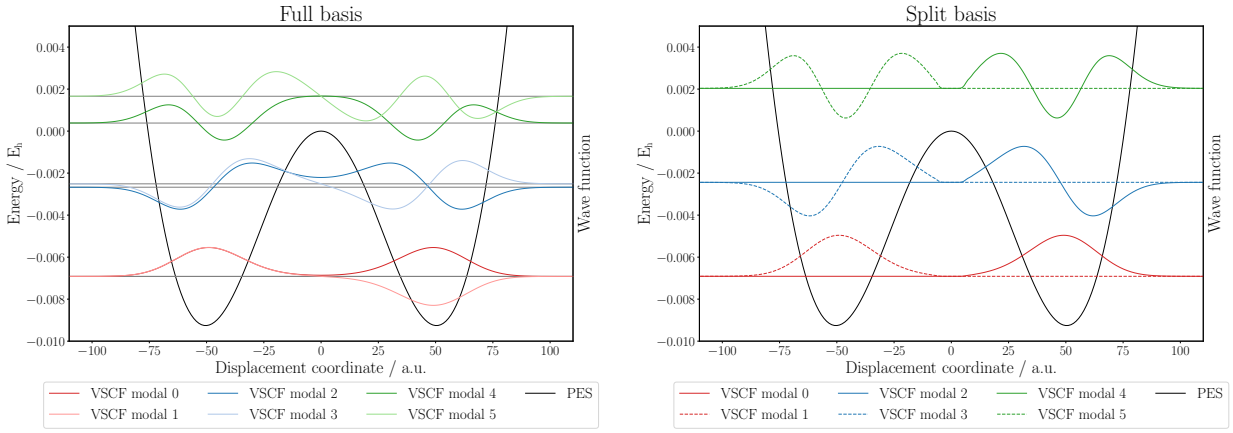
After these VSCF-like eigenvalue equations have been solved we have obtained a set of orthonormal modals that can be expressed in the primitive B-spline basis through the coefficients

$$\mathbf{C}^{m,b_{12}} = \hat{\mathbf{C}}^{m}\bar{\mathbf{C}}^{m}. \tag{74}$$

This extra VSCF calculation is inexpensive since the boundary basis by construction is very small.

The above procedure thereby determines a set of localized orthonormal VSCF modals for the mode m, where we have VSCF modals strictly localized in the two sub-basis sets S_1 and S_2 and in addition boundary modals in basis b_{12} that are orthonormal and have the major amplitude in the boundary region. The localized VSCF modals are not the optimal VSCF modals found if they had been optimized freely in the entire basis but for the use in dynamics calculation the basis is complete and the localization properties can be used to define physically meaningful basis functions and initial wave functions. This allows for example the wave packet to be initialized to a VSCF state localized to only one of the wells in a double well potential. Fig. 13 shows the VSCF solutions for the one-dimensional double well potential corresponding to the NH_3 inversion mode. In Fig. 13a the VSCF solutions when using the full global basis is shown. Here we observe the well known character of wave functions delocalized over both wells with alternating parity. In Fig. 13b the localized VSCF states obtained by using the split basis is shown. Here it is observed that the VSCF modals are only non-zero in one of the wells a the time. Below the barrier in Fig. 13a, the VSCF solutions are approximately pairwise degenerate, but this is no longer the case for energies higher than the barrier. For the localized solutions in Fig. 13b, the solutions localized in each well are pairwise degenerate for all energies in this system. We note that the low energy localized VSCF modals have no contribution in the b_{12} region. This is because the modals which are mainly localized to the b_{12} region have a much higher in energy than the modals

in the S_1 and S_2 regions.. The two lowest-energy VSCF modals in the b_{12} regions are shown in Fig. 14, together with the six split basis VSCF modals also found in Fig. 13b. From Fig. 14 it can be seen that the b_{12} modals are primarily localized at a displacement value around zero where the modals from the S_1 and S_2 sets are zero.



(a) (b)

Figure 13: Lowest-energy VSCF modals for a full and split basis. All modals are displaced along the energy axis corresponding to their energy. (a) The six lowest-energy VSCF modals obtained using the full basis. The modals are pairwise colored for states of similar energy but different parity and the horizontal grey lines indicate the energy of each modal. (b) The six lowest-energy VSCF modals obtained in the split basis. The modals are pairwise degenerate and localized in one of the wells. Degenerate modals have the same color and dashed lines refer to modals localized in the left well while full lines refer to modals localized in the right well.

References

(1) Jäckle, A.; Meyer, H.-D. Product representation of potential energy surfaces. II. *J. Chem. Phys.* **1998**, *109*, 3772–3779, Publisher: American Institute of Physics.

(2) Christiansen, O. Vibrational structure theory: new vibrational wave function methods for calculation of anharmonic vibrational energies and vibrational contributions to molecular properties. *Phys. Chem. Chem. Phys.* **2007**, *9*, 2942–2953.

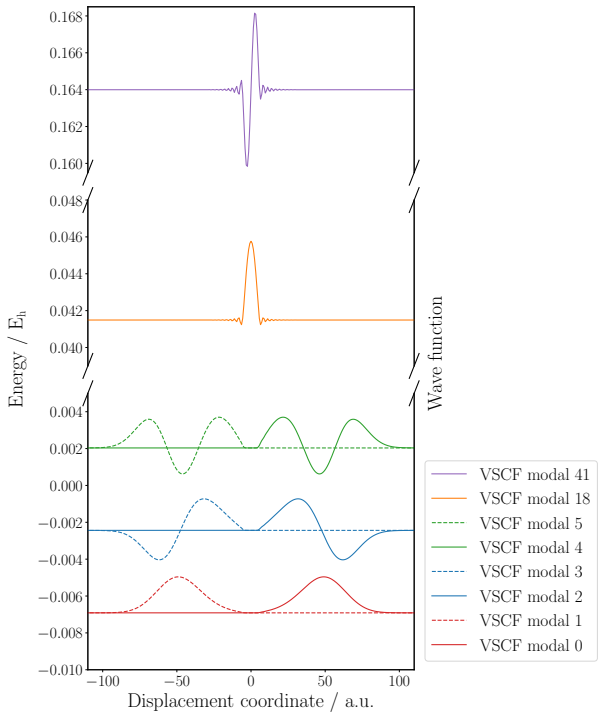


Figure 14: The six lowest-energy VSCF modals in the split basis set and the two lowest-energy modals localized in the boundary basis. All modals are displaced along the energy axis corresponding to their energy. Degenerate modals have the same color and dashed lines refer to modals localized in the left well while full lines refer to modals localized in the right well.

(3) Beck, M.; Jäckle, A.; Worth, G.; Meyer, H.-D. The multiconfiguration time-dependent Hartree (MCTDH) method: a highly efficient algorithm for propagating wavepackets. *Phys. Rep.* **2000**, *324*, 1–105.

(4) Ben-Nun, M.; Quenneville, J.; Martinez, T. Ab initio multiple spawning: Photochemistry from first principles quantum molecular dynamics. *J. Phys. Chem. A* **2000**, *104*, 5161–5175.

(5) Lasorne, B.; Bearpark, M.; Robb, M.; Worth, G. Direct quantum dynamics using variational multi-configuration Gaussian wavepackets. *Chem. Phys. Lett.* **2006**, *432*, 604–609.

(6) Lasorne, B.; Bearpark, M.; Robb, M.; Worth, G. Controlling S/S 0 Decay and the Balance between Photochemistry and Photostability in Benzene: A Direct Quantum Dynamics Study. *J. Phys. Chem. A* **2008**, *112*, 13017–13027.

(7) Worth, G.; Robb, M.; Lasorne, B. Solving the time-dependent Schrödinger equation for nuclear motion in one step: direct dynamics of non-adiabatic systems. *Mol. Phys.* **2008**, *106*, 2077–2091.

(8) Frankcombe, T. J.; Collins, M. A.; Worth, G. A. Converged quantum dynamics with modified Shepard interpolation and Gaussian wave packets. *Chem. Phys. Lett.* **2010**, *489*, 242–247.

(9) Richings, G.; Polyak, I.; Spinlove, K.; Worth, G.; Burghardt, I.; Lasorne, B. Quantum dynamics simulations using Gaussian wavepackets: the vMCG method. *Int. Rev. Phys. Chem.* **2015**, *34*, 269–308.

(10) Christopoulou, G.; Freibert, A.; Worth, G. A. Improved algorithm for the direct dynamics variational multi-configurational Gaussian method. *J. Chem. Phys.* **2021**, *154*, 124127, Publisher: American Institute of Physics.

(11) Crespo-Otero, R.; Barbatti, M. Recent Advances and Perspectives on Nonadiabatic Mixed Quantum–Classical Dynamics. *Chem. Rev.* **2018**, *118*, 7026–7068, Publisher: American Chemical Society.

(12) Worth, G. A.; Lasorne, B. *Quantum Chemistry and Dynamics of Excited States*; John Wiley & Sons, Ltd, 2020; pp 413–433.

(13) Richings, G. W.; Habershon, S. MCTDH on-the-fly: Efficient grid-based quantum dynamics without pre-computed potential energy surfaces. *J. Chem. Phys.* **2018**, *148*, 134116.

(14) Richings, G. W.; Robertson, C.; Habershon, S. Improved on-the-Fly MCTDH Simulations with Many-Body-Potential Tensor Decomposition and Projection Diabatization. *J. Chem. Theory Comput.* **2019**, *15*, 857–870.

(15) Sparta, M.; Toffoli, D.; Christiansen, O. An adaptive density-guided approach for the generation of potential energy surfaces of polyatomic molecules. *Theor. Chem. Acc.* **2009**, *123*.

(16) Klinting, E. L.; Thomsen, B.; Godtliebsen, I. H.; Christiansen, O. Employing general fit-bases for construction of potential energy surfaces with an adaptive density-guided approach. *J. Chem. Phys.* **2018**, *148*, 064113.

(17) Artiukhin, D. G.; Klinting, E. L.; König, C.; Christiansen, O. Adaptive density-guided approach to double incremental potential energy surface construction. *J. Chem. Phys.* **2020**, *152*, 194105.

(18) Schmitz, G.; Klinting, E. L.; Christiansen, O. A Gaussian process regression adaptive density guided approach for potential energy surface construction. *J. Chem. Phys.* **2020**, *153*, 064105.

(19) Carter, S.; Culik, S. J.; Bowman, J. M. Vibrational Self-Consistent Field Method for Many-Mode Systems: A New Approach and Application to the Vibrations of CO Adsorbed on Cu(100). *J. Chem. Phys.* **1997**, *107*, 10458.

(20) Carter, S.; Bowman, J. M.; Harding, L. B. Ab Initio Calculations of Force Fields for H_2CN and C_1HCN and Vibrational Energies of H_2CN. *Spectrochim. Acta Mol. Biomol. Spectrosc.* **1997**, *53*, 1179–1188.

(21) Rauhut, G. Efficient Calculation of Potential Energy Surfaces for the Generation of Vibrational Wave Functions. *J. Chem. Phys.* **2004**, *121*, 9313.

(22) Kongsted, J.; Christiansen, O. Automatic Generation of Force Fields and Property Surfaces for Use in Variational Vibrational Calculations of Anharmonic Vibrational Energies and Zero-Point Vibrational Averaged Properties. *J. Chem. Phys.* **2006**, *125*, 124108–124123.

(23) Bowman, J. M.; Carter, S.; Huang, X. MULTIMODE: A Code to Calculate Rovibrational Energies of Polyatomic Molecules. *Int. Rev. Phys. Chem.* **2003**, *2*, 533–549.

(24) Rabitz, H.; Aliş, Ö. F. General Foundations of High-Dimensional Model Representations. *J. Math. Chem.* **1999**, *25*, 197–233.

(25) Schatz, G. C. The Analytical Representation of Electronic Potential-Energy Surfaces. *Rev. Mod. Phys.* **1989**, *61*, 669.

(26) Jung, J. O.; Gerber, R. B. Vibrational Wave Functions and Spectroscopy of $(H_2O)_n$, $n = 2, 3, 4, 5$: Vibrational Self-Consistent Field with Correlation Corrections. *J. Chem. Phys.* **1996**, *105*, 10332.

(27) Bowman, J. M. Self-consistent field energies and wavefunctions for coupled oscillators. *J. Chem. Phys.* **1978**, *68*, 608–610.

(28) Gerber, R.; Ratner, M. A semiclassical self-consistent field (SC SCF) approximation for eigenvalues of coupled-vibration systems. *Chem. Phys. Lett.* **1979**, *68*, 195–198.

(29) Christiansen, O. A second quantization formulation of multimode dynamics. *J. Chem. Phys.* **2004**, *120*, 2140–2148.

(30) Hansen, M. B.; Sparta, M.; Seidler, P.; Toffoli, D.; Christiansen, O. New Formulation and Implementation of Vibrational Self-Consistent Field Theory. *J. Chem. Theory Comput.* **2010**, *6*, 235–248.

(31) Madsen, N. K.; Hansen, M. B.; Zoccante, A.; Monrad, K.; Hansen, M. B.; Christiansen, O. Exponential parameterization of wave functions for quantum dynamics: Time-dependent Hartree in second quantization. *J. Chem. Phys.* **2018**, *149*, 134110.

(32) Meyer, H.-D.; Manthe, U.; Cederbaum, L. The multi-configurational time-dependent Hartree approach. *Chem. Phys. Lett.* **1990**, *165*, 73–78.

(33) Meyer, H.-D.; Gatti, F.; Worth, G. A. *Multidimensional Quantum Dynamics: MCTDH Theory and Applications*; Wiley-VCH: Weinheim, 2009.

(34) Hansen, M. B.; Madsen, N. K.; Zoccante, A.; Christiansen, O. Time-dependent vibrational coupled cluster theory: Theory and implementation at the two-mode coupling level. *J. Chem. Phys.* **2019**, *151*, 154116.

(35) Madsen, N. K.; Jensen, A. B.; Hansen, M. B.; Christiansen, O. A general implementation of time-dependent vibrational coupled-cluster theory. *J. Chem. Phys.* **2020**, *153*, 234109.

(36) Madsen, N. K.; Hansen, M. B.; Christiansen, O.; Zoccante, A. Time-dependent vibrational coupled cluster with variationally optimized time-dependent basis sets. *J. Chem. Phys.* **2020**, *153*, 174108.

(37) Christiansen, O.; Artiukhin, D. G.; Bader, F.; Godtliebsen, I. H.; Gras, E. M.; Győrffy, W.; Hansen, M. B.; Hansen, M. B.; Højlund, M. G.; Høyer, N. M.; Jensen, R. B.; Jensen, A. B.; Klinting, E. L.; Kongsted, J.; König, C.; Madsen, D.; Madsen, N. K.; Monrad, K.; Schmitz, G.; Seidler, P.; Sneskov, K.; Sparta, M.; Thomsen, B.; Toffoli, D.; Zoccante, A. MidasCpp. 2023.04.0; `https://source.coderefinery.org/midascpp/midascpp`.

(38) Kongsted, J.; Christiansen, O. Automatic generation of force fields and property surfaces for use in variational vibrational calculations of anharmonic vibrational energies and zero-point vibrational averaged properties. *J. Chem. Phys.* **2006**, *125*, 124108.

(39) Toffoli, D.; Kongsted, J.; Christiansen, O. Automatic generation of potential energy and property surfaces of polyatomic molecules in normal coordinates. *J. Chem. Phys.* **2007**, *127*, 204106.

(40) Madsen, N. K.; Hansen, M. B.; Worth, G. A.; Christiansen, O. MR-MCTDH[n]: Flexible Configuration Spaces and Nonadiabatic Dynamics within the MCTDH[n] Framework. *J. Chem. Theory Comput.* **2020**, *16*, 4087–4097.

(41) Battocchio, G.; Madsen, N. K.; Christiansen, O. Density matrices and iterative natural modals in vibrational structure theory. *Mol. Phys.* **2017**, *115*, 228–240.

(42) Helgaker, T.; Jørgensen, P.; Olsen, J. *Molecular electronic-structure theory*; John Wiley & Sons, 2014.

(43) Christiansen, O. A second quantization formulation of multimode dynamics. *J. Chem. Phys.* **2004**, *120*, 2140.

(44) Madsen, N. K.; Hansen, M. B.; Worth, G. A.; Christiansn, O. Systematic and variational truncation of the configuration space in the multiconfiguration time-dependent Hartree method: The MCTDH[n] hierarchy. *J. Chem. Phys.* **2020**, *152*, 084101.

(45) Hairer, E.; Nørsett, S. P.; Wanner, G. *Solving Ordinary Differential Equations I: Non-stiff Problems*, 2nd ed.; Springer Series in Computational Mathematics 8; Springer: Heidelberg ; London, 2009.

(46) Toffoli, D.; Sparta, M.; Christiansen, O. Accurate multimode vibrational calculations using a B-spline basis: theory, tests and application to dioxirane and diazirinone. *Mol. Phys.* **2011**, *109*, 673–685.

(47) Gomes, J. d. S.; Gargano, R.; Martins, J. B. L.; M. de Macedo, L. G. Relativistic Four-Component Potential Energy Curves for the Lowest 23 Covalent States of Molecular Bromine (Br_2). *J. Phys. Chem. A* **2014**, *118*, 5818–5822.

(48) Wilson, A. K.; Woon, D. E.; Peterson, K. A.; Dunning, J., Thom H. Gaussian basis sets for use in correlated molecular calculations. IX. The atoms gallium through krypton. *J. Chem. Phys.* **1999**, *110*, 7667–7676.

(49) Murrell, J. N.; Carter, S.; Farantos, S. C.; Huxley, P.; Varandas, A. J. C. Molecular Potential Energy Functions. *Berichte der Bunsengesellschaft für physikalische Chemie* **1985**, *89*, 1122–1122.

(50) Coon, J.; Naugle, N.; McKenzie, R. The investigation of double-minimum potentials in molecules. *J. Mol. Spectrosc.* **1966**, *20*, 107–129.

(51) Partridge, H.; Schwenke, D. W. The determination of an accurate isotope dependent potential energy surface for water from extensive ab initio calculations and experimental data. *J. Chem. Phys.* **1997**, *106*, 4618–4639.

(52) Polyak, I.; Allan, C. S. M.; Worth, G. A. A complete description of tunnelling using direct quantum dynamics simulation: Salicylaldimine proton transfer. *J. Chem. Phys.* **2015**, *143*, 084121.

(53) Hartree, D. R. The Wave Mechanics of an Atom with a Non-Coulomb Central Field. Part I. Theory and Methods. *Mathematical Proceedings of the Cambridge Philosophical Society* **1928**, *24*, 89–110.

(54) Fock, V. Näherungsmethode zur Lösung des quantenmechanischen Mehrkörperproblems. *Zeitschrift für Physik* **1930**, *61*, 126–148.

(55) Hehre, W. J.; Ditchfield, R.; Pople, J. A. Self—Consistent Molecular Orbital Methods. XII. Further Extensions of Gaussian—Type Basis Sets for Use in Molecular Orbital Studies of Organic Molecules. *J. Chem. Phys.* **1972**, *56*, 2257–2261.

(56) Balasubramani, S. G.; Chen, G. P.; Coriani, S.; Diedenhofen, M.; Frank, M. S.; Franzke, Y. J.; Furche, F.; Grotjahn, R.; Harding, M. E.; Hättig, C.; Hellweg, A.; Helmich-Paris, B.; Holzer, C.; Huniar, U.; Kaupp, M.; Marefat Khah, A.; Karbalaei Khani, S.; Müller, T.; Mack, F.; Nguyen, B. D.; Parker, S. M.; Perlt, E.; Rappoport, D.; Reiter, K.; Roy, S.; Rückert, M.; Schmitz, G.; Sierka, M.; Tapavicza, E.; Tew, D. P.; van Wüllen, C.; Voora, V. K.; Weigend, F.; Wodyński, A.; Yu, J. M. TURBOMOLE: Modular program suite for *ab initio* quantum-chemical and condensed-matter simulations. *J. Chem. Phys.* **2020**, *152*, 184107.

(57) Ahlrichs, R.; Bär, M.; Häser, M.; Horn, H.; Kölmel, C. Electronic structure calculations on workstation computers: The program system turbomole. *Chem. Phys. Lett.* **1989**, *162*, 165–169.