# Supporting Information for Combining SchNet and SHARC: The SchNarc machine learning approach for excited-state dynamics

Julia Westermayr,[1] Michael Gastegger,[2] and Philipp Marquetand[1,3,4]

[1]University of Vienna, Institute of Theoretical Chemistry
Währinger Str. 17, 1090 Vienna, Austria
[2]Machine Learning Group, Technical University of Berlin, 10587 Berlin, Germany
[3]Vienna Research Platform on Accelerating Photoreaction Discovery, University of Vienna, Währinger Str. 17, 1090 Vienna, Austria.
[4]Data Science @ Uni Vienna, University of Vienna, Währinger Str. 29, 1090 Vienna, Austria.
E-mail: philipp.marquetand@univie.ac.at; michael.gastegger@tu-berlin.de

# Contents

## S1 SchNet for excited states

As a machine learning (ML) model, the deep continuous-filter convolutional neural network SchNet, that is described in detail in Ref.[1,2] is used and adapted for excited states to train excited-state energies, forces, spin-orbit couplings (SOCs), and nonadiabatic couplings (NACs).

The molecular descriptor is constructed by SchNet[1] that treats atoms in their chemical and structural environment. A cutoff is defined to specify the environment that is included for the description of an atom. Hence the molecular properties are obtained as atom-wise contributions. A continuous-filter convolutional layer and several additional interaction layers define and optimize the atom representations. These representations are mapped to different properties via fully connected layers with shifted softplus activation functions. These prediction blocks, which use a common descriptor network, are separately designed for energies, SOCs, and NACs, whereas the forces are derived with respect to atomic coordinates from outputs of the ML model for energies. The loss function is a combined loss function of all the properties. A trade-off is defined to weigh the properties according to their magnitude. The properties that should be learned have to be specified along with the corresponding trade-off in an additional input file.

### S1.1 Standard loss function, $L_2$

The overall $L_2$ loss function as implemented in SchNet for excited states, reads:

$$\begin{aligned} L_2 = t_E \mid\mid E^{QC} - E^{ML} \mid\mid^2 + t_F \mid\mid F^{QC} - F^{ML} \mid\mid^2 + \\ t_{\mathrm{SOC}} \mid\mid C^{QC}_{\mathrm{SOC}} - C^{ML}_{\mathrm{SOC}} \mid\mid^2 + t_{\mathrm{NAC}} \mid\mid C^{QC}_{\mathrm{NAC}} - C^{ML}_{\mathrm{NAC}} \mid\mid^2, \end{aligned} \tag{1}$$

where $t_E, t_F, t_{SOC}$, and $t_{NAC}$ define the trade-offs for the properties E (energies), F (forces), $C_{\mathrm{SOC}}$ (SOCs), and $C_{\mathrm{NAC}}$ (NACs), respectively. Corre-

sponding labels with an index "QC" refer to the reference value and with an index "ML" to the the SchNet predictions.

### S1.2 Phase-less loss function, $L_{ph}$

In order to train on inconsistent SOCs and NACs with respect to their sign, we have developed a phase-less loss-function. This is based on the $L_2$ loss, but here, the squared error of the predicted properties, $P$, is computed more often, i.e. $2^{N_S-1}$-times with $N_S$ being the total number states. The value, $L_P$, that enters the overall loss function, $L_{ph}$, is the minimum function of all possible squared errors, $\varepsilon_P^k$, for a given property, $P$:

$$L_P = \min\left(\{\varepsilon_P^k\}\right) \quad \text{with } 0 \leq k \leq 2^{N_S-1} \tag{2}$$

with

$$\varepsilon_P^k = \begin{cases} \sum_i^{N_S} \sum_{j\neq i}^{N_S} \frac{1}{N_A} \sum_m^{N_A} \mid\mid P_{ij,m}^{QC} - P_{ij,m}^{ML} \cdot p_i^k \cdot p_j^k \mid\mid^2 & \text{if } dim(\mathbf{P}) \geq 3 \\ \sum_i^{N_S} \sum_{j\neq i}^{N_S} \mid\mid P_{ij}^{QC} - P_{ij}^{ML} \cdot p_i^k \cdot p_j^k \mid\mid^2 & \text{if } dim(\mathbf{P}) \leq 2 \end{cases} \tag{3}$$

for vectorial and non-vectorial properties, respectively. The error $\varepsilon_P^k$ for a specific phase is computed as the mean squared error of a property $P$ from quantum chemistry (index $QC$) and machine learning (index $ML$). The property $P$ couples different states, indicated by $i$ and $j$. Since the wave function of each of the states can have an arbitrary phase, the property $P_{ij}$ that couples state $i$ and $j$ has to be multiplied with a product of the phases for these states, $p_i \cdot p_j$. The phases for all states together form a vector $p$ with entries of either +1 and -1. Which of the $2^{N_S-1}$ possible combinations for $p$ is chosen, is indicated by the index $k$, also defined in eq. (2). Since we are free to choose one of the phases, we set the phase of the first state always to +1. The relative signs within one vector remain and must be predicted

correctly for successful training.

The overall loss function used in this work is a combination of such phase-less errors and mean squared errors obtained for all properties with a trade-off factor to account for their relative magnitude (as already specified in equation 1):

$$L_{ph} = t_E \mid\mid E^{QC} - E^{ML} \mid\mid^2 + t_F \mid\mid F^{QC} - F^{ML} \mid\mid^2 + t_{SOC} \cdot L_{SOC} + t_{NAC} \cdot L_{NAC} \tag{4}$$

This cost function removes the influence of the arbitrary phase during the learning process of a ML model and further reduces the computational costs for the training set generation.

We tested on several alternatives, such as a loss function that additionally includes the norm of a vector, variations of a minimum function and another type of phase-free loss function, that can be used if only one type of coupling, i.e. SOCs or NACs, or dipole moments are trained. This error is also implemented in SchNet for excited states and the error of a property that couples state i and j, $\varepsilon_P^k$ is computed as follows:

$$\varepsilon_P^{k,\pm} = \begin{cases} \mid\mid P_{ij}^{QC} \pm P_{ij}^{ML} \mid\mid^2 & \text{if } dim(\mathbf{P}) \leq 2 \\ \frac{1}{N_A} \sum_m^{N_A} \mid\mid P_{ij,m}^{QC} \pm P_{ij,m}^{ML} \mid\mid^2 & \text{if } dim(\mathbf{P}) \geq 3 \end{cases} \tag{5}$$

As can be seen, the error is computed twice – once assuming a correct phase of predicted properties and once a phase switch, that are both combined subsequently in case $dim(\mathbf{P}) \geq 3$:

$$e_P^k = \varepsilon_p^{k,-} \cdot C_{ij}^+ + \varepsilon_p^{k,+} \cdot C_{ij}^- \tag{6}$$

with

$$C_{ij}^{\pm} = \frac{\varepsilon_p^{k,\pm}}{\varepsilon_p^{k,-} + \varepsilon_p^{k,+}}. \tag{7}$$

The value, $L_P$, that enters the loss function is then either a combination of both possibilities for vectorial properties or a minimum function of the two possible errors, $\varepsilon_P^{k,\pm}$. This variation gives comparably accurate results and also leads to a phase-free training. Experiments have shown, that – e.g. in the case of the $SO_2$ molecule – more data points, but shorter training is necessary. This alternative error can be more favorable in cases, where only one coupling type is needed and many states are involved, since computation of all possible combinations can be omitted. All other tested variations turned out to be less successful in learning the shape of couplings.

### S1.3 Machine learning models

The model parameters for each molecular system are given in Table S1 including the number of data points and states trained. The errors for the remaining test set (i.e. the data points not used for training and validation) are listed as mean absolute values resulting from all states. For the $CH_2NH_2^+$ and $CSH_2$ models, the ML predictions reach chemical accuracy and in some cases the error is even below 0.043 eV (1 kcal/mol). If not stated otherwise, 256 features with 3 hidden layers are used for each model. The batch size ranges from 20 to 50 and is set in order to comply with the maximum allowed memory of a used GPU. The learning rate is set to 0.0001 and is reduced by a factor of 0.8 down to a value of 0.000001 with a patience of 15 steps. The maximum number of epochs is set to 5000. The trade-off for each property is defined, so that the mean squared errors in the first few epochs is equally large for all properties.

Table S1: Parameters for the trained SchNet models for excited states. For $SO_2$, $CSH_2$, and $CH_2NH_2^+$ 200 to 500, 200 and 100 data points are used for validation, respectively. Hence at least 200,000, 503, and 900 data points are used for testing. The MAEs and RMSEs are reported in eV, eV/Å, and a.u. for energies, gradients and all type of couplings, respectively. If not mentioned otherwise, the data sets are not phase corrected.

| Molecule | training points | S/T | properties (t) | cutoff [Å] | Loss | MAE (RMSE) |
|---|---|---|---|---|---|---|
| $SO_2$ | 5,000 | 3/0 | E (1.0) | 5.0 | L2 | 0.069 (0.20) |
| | | | F (0.25) | | | 0.20 (0.67) |
| $SO_2$ | 20,000 | 3/0 | E (1.0) | 8 .0 | $L_{ph}$ | 0.062 (0.184) |
| | | | F (0.1) | | | 0.24 (0.65) |
| | | | NAC (0.004) | | | 0.13 (1.15) |
| $SO_2$ | 5,000 | 3/3 | E (1.0) | 5.0 | $L_{ph}$ | 0.029 (0.068) |
| | | | F (0.25) | | | 0.12 (0.26) |
| | | | SOC (300) | | | $7.7 \cdot 10^{-6}$ ($3.1 \cdot 10^{-5}$) |
| $SO_2$ | 20,000 | 3/3 | E (1.0) | 8.0 | $L_{ph}$ | 0.027 (0.068) |
| | | | F (0.25) | | | 0.11 (0.26) |
| | | | NAC (0.0001) | | | 0.52 (23.8) |
| | | | SOC (300) | | | $1.2 \cdot 10^{-5}$ ($4.4 \cdot 10^{-5}$) |
| $CH_2NH_2^+$ | 3,000 | 3/0 | E (1.0) | 10.0 | L2 | 0.059 (0.13) |
| | | | F(1.0) | | | 0.15 (0.30) |
| | | | NAC (0.001) | | | 0.22 (0.89) |
| $CH_2NH_2^+$ | 3,000 | 3/0 | E (1.0) | 10.0 | $L_{ph}$ | 0.059 (0.14) |
| | | | F(1.0) | | | 0.14 (0.32) |
| | | | NAC (0.004) | | | 0.15 (0.55) |
| $CH_2NH_2^+$ | 3,000 | 3/0 | E (1.0) | 10.0 | L2 | 0.042 (0.087) |
| | *phase* | | F(1.0) | | | 0.096 (0.22) |
| | *corrected* | | NAC (0.0001) | | | 0.21 (0.83) |
| $CH_2NH_2^+$ | 3,000 | 3/0 | E (1.0) | 10.0 | $L_{ph}$ | 0.050 (0.16) |
| | *phase* | | F(1.0) | | | 0.13 (0.32) |
| | *corrected* | | NAC (0.004) | | | 0.15 (1.1) |
| $CH_2NH_2^+$ | 3,000 | 3/0 | E (1.0) | 10.0 | L2 | 0.048 (0.12) |
| | | | F(1.0) | | | 0.13 (0.30) |
| $CSH_2$ | 4,000 | 2/2 | E (1.0) | 10.0 | $L_{ph}$ | $4.1 \cdot 10^{-4}$ ($6.1 \cdot 10^{-4}$) |
| | | | F(1.0) | | | $6.2 \cdot 10^{-4}$ ($1.1 \cdot 10^{-3}$) |
| | | | SOC (500) | | | $6.1 \cdot 10^{-6}$ ($1.6 \cdot 10^{-5}$) |

## S2 Training sets and reference computations

### S2.1 Training set generation

SchNet models for excited states are trained on a linear vibronic coupling model (LVC) of $SO_2$ [3,4], the methylenimmonium cation, $CH_2NH_2^+$ and thioformaldehyde, $CSH_2$.

The molecular geometries and corresponding properties are saved in a database format provided by the atomic simulation environment [5]. No data points of the dynamics simulations to which we compare SchNarc models are included in the training sets. The phase corrected training set for the methylenimmonium cation, $CH_2NH_2^+$, is taken from Ref. [6] and consists of 4000 data points. The energies, gradients, and nonadiabatic couplings of geometries of this training set were recomputed with the same level of theory, MR-CISD(6,4)/aug-cc-pVDZ, but without applying any pre-processing, such as phase correction, in order to provide a non-phase corrected training set. The program suite COLUMBUS [7] was used for this purpose, resulting in 3998 converged single point calculations. For the dynamics simulations with SchNarc, 1000 trajectories (resulting from 20,000 initial conditions sampled from a Wigner distribution [8]) are propagated for 100 fs using a time step of 0.5 fs.

The training set for thioformaldehyde, $CSH_2$, is generated in the same way as it is done in Ref. [6] for $CH_2NH_2^+$. Initial configurations are sampled via scans of different reaction coordinates, such as normal modes. Additionally, adaptive sampling for excited states is carried out using two simple multi-layer feed-forward neural networks. At a number of 4855 data points, the networks seem to be converged and dynamics simulations can be reproduced. The training set consists of 4703 data points, where samples showing a smaller energy gap than 0.01 H between triplet-triplet states are sorted out

due to problematic data points in those regions. Without these points the NNs converge much better, in about half of the time. The RMSE of forces is slightly larger, the rest of the errors are comparable. The level of theory is CASSCF(6,5)/def2-SVP and 2 singlet and 2 triplet states are included. Quantum chemistry calculations are carried out using Molpro[9]. In addition to energies, forces, SOCs, and NACs, the permanent and transition dipole moments are included in the training set. For the dynamics simulations of $CSH_2$, 40,000 initial conditions are sampled from a Wigner distribution[8] and excited to the first excited singlet state (2.0-2.5 eV). 100 trajectories are propagated with the reference method for 3 ps with a time step of 0.5 fs. The resulting populations are compared to 959 trajectories obtained from SchNarc.

For the $SO_2$ molecule, that is based on a "one-shot" linear vibronic coupling model[3] and serves as a reference, we refer to dynamics simulations with the linear vibronic coupling model to generate the training set. After sampling of 10,000 initial conditions from a Wigner distribution[8] and excitation between 0 and 10 eV, surface hopping molecular dynamics simulations are carried out with SHARC. The first 200 trajectories are taken for the training set generation. This procedure is done twice - once only singlet states are considered and once singlet and triplet states are taken into account, resulting in 280,200 data points for each training set. Additional 1000 initially sampled geometries are excited and trajectories are simulated with SHARC to provide a comparison to SchNarc dynamics. All trajectories are propagated with NAC vectors for 700 fs with a time step of 0.5 fs. Due to symmetry, the $SO_2$ model contains NACs only between the $S_1$ state and the $S_2$ state as well as between the $T_1$ state and the $T_3$ state. We considered this restriction for the SchNarc computations by setting the other couplings to zero.

It is worth mentioning that SO2 needs more data points for training than $CSH_2$ and $CH_2NH_2^+$, since for the latter molecules, adaptive sampling was applied and for $SO_2$ we used data directly from dynamics simulations with the LVC model. The dynamics with the LVC model are extremely fast and hence for the training set generation the usual adaptive sampling approach[6,10,11] is far more costly and time-intensive. Since reference computations are even cheaper than ML predictions, it is not our goal to provide a perfect training set with a minimum number of data points for this model, but rather to provide an easy-to-use but yet challenging test system to validate our method.

## S2.2 Surface hopping molecular dynamics

In order to compute nonadiabatic molecular dynamics simulations, the SHARC[12–14] method, an extension of Tully's fewest switches algorithm[15], is applied. This mixed quantum-classical approach allows for on-the-fly computation of the PESs with electronic structure methods, on which the nuclei move according to Newton's second equation of motion. In order to account for nonadiabatic transitions between states of same spin multiplicity, instantaneous switches from one state to the others are allowed in regions of high hopping probability. After every simulation, the trajectories are analyzed using the SHARC diagnostic tools to check improper behaviour, such as energy fluctuations. A few reference trajectories are sorted out in each case – mainly due to improper convergence of quantum chemistry calculations in critical regions of the potential energy surfaces. Decoherence correction is applied[16] and the hopping probabilities are computed from SOCs and NACs from electronic structure calculations in case of reference dynamics or from ML models in case of SchNarc dynamics[14]. The velocities are corrected along the direction of the NAC vectors in each simulation and for dynamics with quantum

chemistry, the phase is tracked along an independent trajectory.

## S3 Nonadiabatic couplings

As mentioned in the main text, NACs are either approximated or derived from a virtual property built by SchNarc. In order to define the virtual property, which SchNarc builds internally, we start by the derivative of the electronic Hamiltonian, $H_{el}$, with respect to the atomic coordinates of a molecule, $\mathbf{R}$:

$$\begin{aligned} \frac{\partial H_{el_{ij}}(r,\mathbf{R})}{\partial \mathbf{R}} &= \frac{\partial}{\partial \mathbf{R}} \langle \Psi_i \mid H_{el}(r,\mathbf{R}) \mid \Psi_j \rangle \quad (8) \\ = \langle \frac{\partial}{\partial \mathbf{R}} \Psi_i \mid H_{el}(r,\mathbf{R}) \mid \Psi_j \rangle &+ \langle \Psi_i \mid \frac{\partial H_{el}(r,\mathbf{R})}{\partial \mathbf{R}} \mid \Psi_j \rangle + \langle \Psi_i \mid H_{el}(r,\mathbf{R}) \mid \frac{\partial}{\partial \mathbf{R}} \Psi_j \rangle \end{aligned}$$

Since the adiabatic wavefunctions are eigenfunctions of $H_{el}(r,\mathbf{R})$, we can reformulate equation (8):

$$\begin{aligned} \frac{\partial H_{el_{ij}}(r,\mathbf{R})}{\partial \mathbf{R}} &= E_j \langle \frac{\partial}{\partial \mathbf{R}} \Psi_i \mid \Psi_j \rangle + \langle \Psi_i \mid \frac{\partial H_{el}(r,\mathbf{R})}{\partial \mathbf{R}} \mid \Psi_j \rangle + E_i \langle \Psi_i \mid \frac{\partial}{\partial \mathbf{R}} \Psi_j \rangle \quad (9) \end{aligned}$$

By using the relation, $\langle \Psi_i \mid \frac{\partial}{\partial \mathbf{R}} \Psi_j \rangle = -\langle \frac{\partial}{\partial \mathbf{R}} \Psi_i \mid \Psi_j \rangle$, we can write:

$$\frac{\partial H_{el_{ij}}(r,\mathbf{R})}{\partial \mathbf{R}} = (E_i - E_j) \cdot \langle \Psi_i \mid \frac{\partial}{\partial \mathbf{R}} \Psi_j \rangle + \langle \Psi_i \mid \frac{\partial H_{el}}{\partial \mathbf{R}} \mid \Psi_j \rangle. \quad (10)$$

Applying the Hellmann-Feynman theorem[17], we obtain the diagonal el-

ements as the gradients,

$$\frac{\partial}{\partial \mathbf{R}} H_{el_{ij}} = \nabla E_{ij} \quad for\ i = j, \tag{11}$$

and obtain the NAC terms as properties that are inversely proportional to the corresponding energy gap of two adiabatic electronic states:

$$C_{ij}^{\mathrm{NAC}} \approx \langle \Psi_i \mid \frac{\partial}{\partial \mathbf{R}} \Psi_j \rangle = \frac{1}{E_i - E_j} \langle \Psi_i \mid \frac{\partial H_{el}}{\partial \mathbf{R}} \mid \Psi_j \rangle \quad \text{for } i \neq j. \tag{12}$$

The virtual property that SchNarc is generating is then the multi-dimensional anti-derivative of the latter expression in equation 12, $\langle \Psi_i \mid \frac{\partial H_{el}}{\mathbf{R}} \mid \Psi_j \rangle$. eably, due to the Berry phase[18–20] the NAC vector field is not conservative[21] and a line integral remains path dependent. Hence this approach does not include the effects of the Berry phase, which is also neglected in approaches such as the Zhu-Nakamura approximation[22,23] that does not contain a phase at all, or the phase correction algorithm[6,24]. The mixed ML-classical dynamics are thus assumed to be mostly unaffected[6,24], which might not be the case in quantum dynamics simulations.

### S3.1 NAC approximation and timing

The approximation of NAC vectors, as explained in the main text and adapted from Refs.[21,25,26], relies on the approximation of the Hessian from energy potentials between two states. It is especially powerful for ML models trained on quantum chemistry methods, where implementations of NAC vectors are largely missing, such as linear-response methods and here especially between the first excited state and the ground state, with the ADC(2) method being a prominent example[27]. Such an ML approach could further be used to pave the way towards efficient Hessian computations for all the states treated in quantum dynamics simulations using the variational

multi-configurational Gaussian method, where the direct Hessian computation often remains the time limiting step[28,29].

The used NAC approximation is valid in the vicinity of a conical intersection and hence relies on a threshold to define the energy gap, for which the approximation is applied. In order to avoid additional computations, the Hessians are thus only computed if one of the energy differences between all possible singlet-singlet or triplet-triplet potentials is within the given threshold (as default we set 0.5 eV and 1.0 eV for singlet-singlet and triplet-triplet gaps). Since the gaps of the PESs are overestimated in case of $CH_2NH_2^+$[6], this threshold is increased by 30%. This means, that based on this predefined threshold, SchNarc decides whether NAC vectors are computed or not. In all other cases, the NAC vectors are set to zero. It is advisable to check the used thresholds for certain cases and adapt them, where necessary. It is worth mentioning, that this approach is limited to same-symmetry electronic states, which are, nevertheless, the most probable avoided state crossings in case of real, polyatomic systems[26].

## Timing

For the thioformaldehyde molecule, the evaluation of the 4 Hessians takes approximately 2 seconds, for the methylenimmonium cation, the computation of 3 Hessians takes 3-4 seconds, both on a CPU. A test computation of a 24 atom molecule was further carried out with SchNarc, which showed that the evaluation of 1 Hessian took around 45 sec on a CPU, which could be reduced to 14 sec on a GPU. In future work, we thus seek to adapt the code in order to compute only the relevant Hessians, i.e., those of close-lying states with respect to the active state during a dynamics simulation. To this aim, we seek to give the information of the active state to the SchNarc model, which is not yet implemented for our pySHARC[3,6] wrapper (python

Table S2: Comparison of the timings of 100 steps of a dynamics simulations using SchNarc with learned and approximated NACs as well as SHARC with quantum chemistry or the LVC model for $SO_2$. We used 2x Intel Xeon E5-2650 v3 CPUs, and GeForce GTX 1080 Ti GPUs.

| | # States S/T | 100 time steps [s/CPU] *(Training[h/GPU]/data points)* **SchNarc** NAC learned | **SchNarc** NAC excluded | **SHARC** |
|---|---|---|---|---|
| $SO_2$ | 3/0 | 5 (*50.8/20,000*) | 6 (*3.7/5,000*) | 1-2 |
| $SO_2$ | 3/3 | 13 (*308/200,000*) | 17(*18.3/5,000*) | 1-2 |
| $CSH_2$ | 2/2 | 7 (*13.0/4,000*) | 8 (*12.9/4,000*) | 52 |
| $CH_2NH_2^+$ | 3/0 | 12 (*19.9/3,000*) | 126 (*11.1/3,000*) | 37,112 |

wrapper for the SHARC code, which avoids heavy file I/O). Hence the Hessians would only be computed for the states that are close enough to the active state and the dynamics simulations are then still very efficient compared to pure quantum chemistry dynamics simulations. A comparison of the timings of 100 time steps for a dynamics simulation with the current SchNarc implementations and SHARC is given in Table S2.

As can be seen, the LVC dynamics using the pySHARC wrapper are very cheap and only serve as a test system. It is however clearly visible that the training of NACs in this case needs way more data points and hence it takes longer to train the models. The dynamics of the $CH_2NH_2^+$ molecule using the MR-CISD method are expensive and ML can substantially decrease the simulation time. It can be seen that the Hessian computation with ML becomes more expensive, the larger the molecule becomes or the more states are involved in a simulation.

# S4 Quality of ML models

## S4.1 Excited-states of sulfur dioxide, $SO_2$

To assess the quality of approximated NACs and learned SOCs, a linear vibronic coupling (LVC) model[3,4] of sulfur dioxide, $SO_2$, is used. The LVC model of sulfur dioxide, $SO_2$,[3] contains 3 singlet states and 3 triplet states, with symmetry allowed NACs between the first and second excited singlet states as well as the first and third triplet states, as well as SOCs between singlet and triplet states. It is used to train two ML models: One, where the NACs are learned and another, where the NACs are approximated according to equation 7 in the main text. The ML models for dynamics simulations with approximated NACs, i.e. ML models that are trained only on energies and gradients for singlet states only and energies, gradients, and SOCs for singlet and triplet states, used 5,000 data points. To train also on NACs 20,000 data points are required. Those training points are randomly selected from data sets consisting of 280,200 points.

Fig. S1 shows the quality of ML energies (A), gradients (B), SOCs between singlet and triplets (C), and NACs between singlet states (D, E). The NACs in panel (D) are obtained from an ML model trained on energies, gradients, and NACs, whereas the NACs in panel (E) are approximated from an ML model trained on energies and gradients. Since the test set is very large (i.e. it contains more than 250,000 data points), the first 1,000 data points of the randomly shuffled test set are plotted.

The accuracy of the ML Hessians is assessed by a computation of the frequencies and normal mode coordinates of the equilibrium conformation. The results are included as a supplementary file in molden format and the frequencies are compared to reference values in Table S3. The first normal mode is underestimated by about 10% and the other two normal modes agree

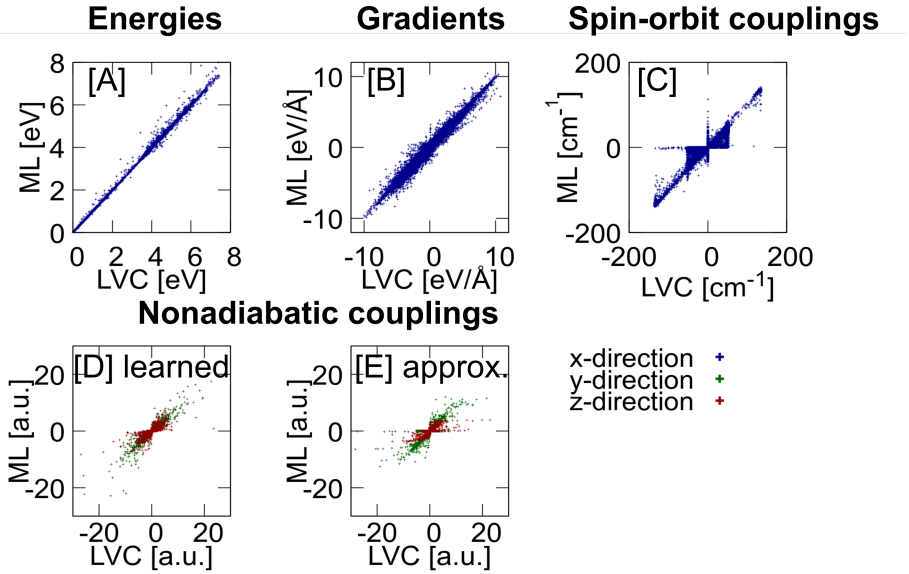


Figure S1: Scatter plots for $SO_2$ showing the distribution of (A) energies, (B) gradients, (C) spin-orbit couplings, and (D) NACs from ML models that are trained on NACs and (E) from ML models that are solely trained on energies and gradients. The NACs are approximated from energies, gradients and Hessians of the ML model in the latter case.

to the reference very well within about 98% accuracy.

Table S3: Comparison of frequencies of $SO_2$ obtained from LVC and SchNarc Hessians.

| Normal mode | LVC [$cm^{-1}$] | SchNarc [$cm^{-1}$] |
|---|---|---|
| 1 | 519 | 451 |
| 2 | 1165 | 1189 |
| 3 | 1405 | 1409 |

In Fig. S2, the potential energy curves from both LVC and ML along the asymmetric stretching mode of the singlet states (left plot) and the triplet states (right plot) are shown along with the norm of the respective NAC vectors. As can be seen, the shape as well as the height of the peak of the norm of trained NACs (dashed lines) and approximated NACs (dotted

lines) are comparable to those of the LVC model (continuous lines). The approximated NACs approach zero faster than the trained NACs, which is due to the applied threshold that is set for the energy gap to compute NACs. Noticeably, the learned NACs between the triplet states show a decrease, when the corresponding triplet energies are close to each other. This might be an effect due to the Berry phase that can not be captured with our approach leading to artifacts in some regions.

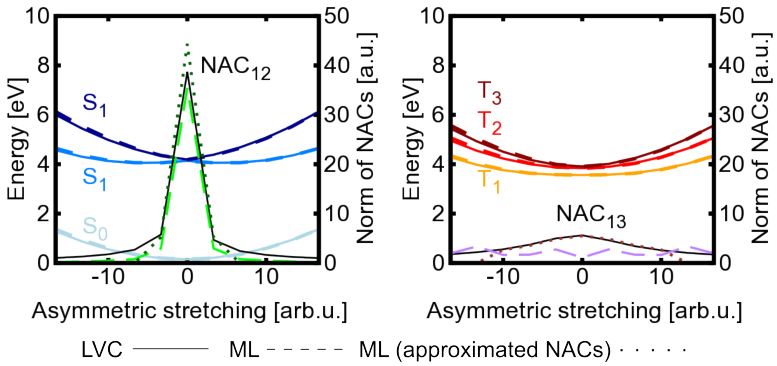


Figure S2: Potential energy curves and the norm of the NAC vectors between singlet states (A) and triplet states (B) along the asymmetric stretching mode of $SO_2$. Continuous line represent LVC(MR-CISD) and dotted (dashed) lines show results obtained from SchNarc models trained on only energies and gradients (as well as NACs for comparison) of 3 singlets and 3 triplet states.

Importantly, the NACs can be predicted accurately with ML in most of the regions around the conical intersections. We first consider a singlet-only model in order to support this assumption with dynamics simulations, see Fig. S3. LVC populations show minor population transfer between the second excited singlet state and the first excited singlet state, which can be reproduced with both SchNarc models (upper panels (A)-(C)).

Also when including triplet states, the SchNarc models can reproduce the dynamics (lower panels (D)-(F)). Here, population is mostly transferred from

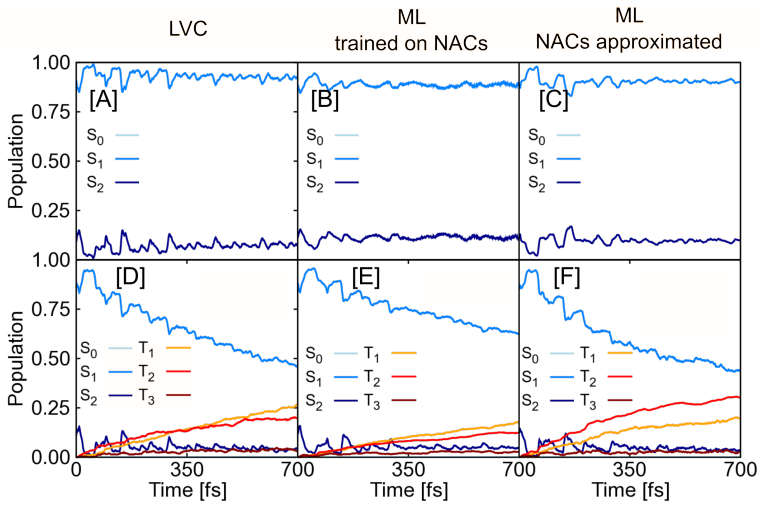


Figure S3: Quantum populations using LVC(MR-CISD) of $SO_2$ (left panels), SchNarc trained on NACs (middle panels) and SchNarc using approximated NACs from energies and gradients (right panels). Dynamics are shown of 1000 initially excited configurations considering only singlet states (upper line) and additionally triplet states (lower line).

the first excited singlet state to the triplet states. Note that populations in surface hopping are often only accurate to within 10%, such that we judge the deviations of the ML populations from the LVC reference as small (see Ref. 30 for examples, where dynamics is not reproduced by ML although potentials seemingly are).

### S4.2 Excited states of the methylenimmonium cation, $CH_2NH_2^+$

The quality of the ML excited-state energies (A), gradients (B), and NAC values (C, D) of the methylenimmonium cation, $CH_2NH_2^+$, are assessed via scatter plots of the test set in Fig. S4. A scan along a reaction coordinate in Fig. S5, which includes two conical intersections, shows the ability of ML to reproduce energies and couplings also in critical regions.

The reference energies and couplings obtained with quantum chemistry

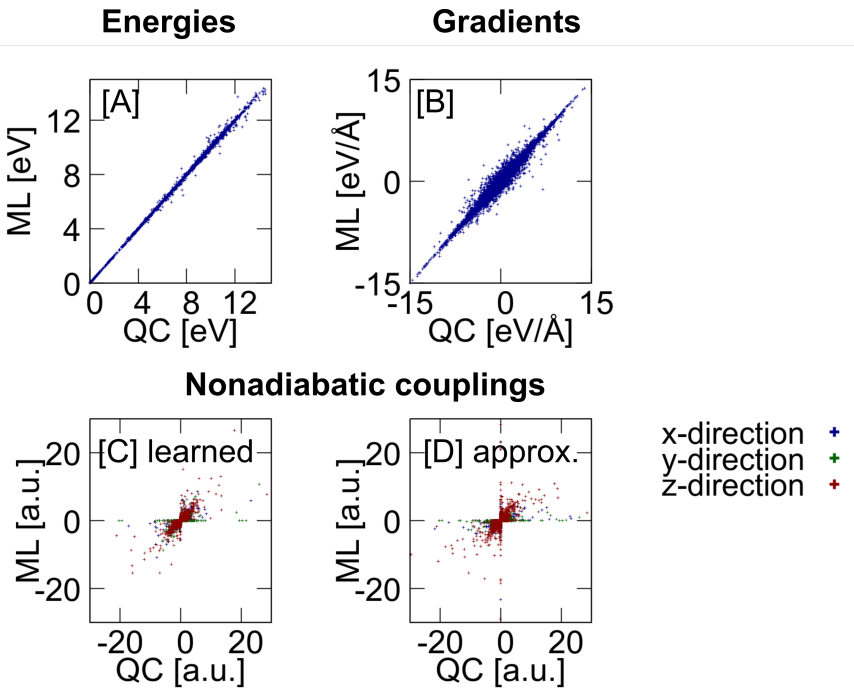


Figure S4: Scatter plots showing the distribution of (A) energies, (B) gradients, and (C) NACs of $CH_2NH_2^+$ obtained from ML models that are trained on NACs and (D) from ML models that are solely trained on energies and gradients. The NACs are approximated from energies, gradients and Hessians in the latter case.

clearly show an artifact in the $S_2$ state for geometries around the avoided crossing between the $S_0$ and the $S_1$ state. This artifact is not reproduced with ML, leading to artificially larger errors in such cases. Hence scatter plots have to be considered with care and large errors might not result from inaccurate ML potentials, but from inconsistencies within quantum chemical calculations. Such artifacts can be a result of an intruder state, i.e. an electronic state, that is very high in energy at the reference equilibrium geometry, but is low in energy for another geometry, or of a change in the active space along certain reaction coordinates. With regards to nonadiabatic

molecular dynamics simulations, it is important to predict the magnitude of the couplings approximately correctly to compute an accurate hopping probability. The exact value of NACs is less important, which we have discussed in detail in Ref.[30]. It is more relevant that the couplings are large when two states are close to each other and that they are small elsewhere. This behaviour can be reproduced with both versions of ML NACs – learned NACs and approximated ones.

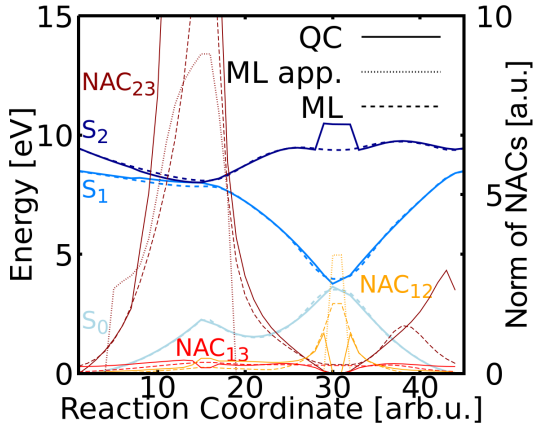


Figure S5: Potential energy curves and the norm of the NAC vectors between singlet states along a reaction coordinate of $CH_2NH_2^+$ with two critical points. Continuous lines represent the quantum chemical reference method MR-CISD(6,4)/aug-cc-pVDZ and dashed lines the ML predictions. Dotted lines show the norm of NACs obtained from approximated NACs using ML predicted energies, gradients, and Hessians.

The Hessians are further used to compute the frequencies of the equilibrium geometry of $CH_2NH_2^+$. The results are included as a supplementary file in molden format and are compared to reference quantum chemistry values in Table S4. In addition, we compare to frequencies that are computed with previously[6] trained multi-layer feed-forward neural networks on energies and gradients of $CH_2NH_2^+$. The Hessian is is obtained via numerical differentiation in the latter case. Both ML models agree well with the reference method.

Table S4: Comparison of frequencies of $CH_2NH_2^+$ obtained from the quantum chemistry (QC) reference method MR-CISD(6,4)/aug-cc-pVDZ, from SchNarc Hessians and from previously trained multi-layer feed-forward neural networks (MLFF)[6]. In the latter case, Hessians are obtained via numerical differentiation.

| Normal mode | QC [$cm^{-1}$] | SchNarc [$cm^{-1}$] | MLFF [$cm^{-1}$] |
|---|---|---|---|
| 1 | 940 | 926 | 936 |
| 2 | 969 | 1090 | 1006 |
| 3 | 1074 | 1096 | 1165 |
| 4 | 1174 | 1142 | 1211 |
| 5 | 1369 | 1451 | 1389 |
| 6 | 1473 | 1520 | 1494 |
| 7 | 1612 | 1632 | 1582 |
| 8 | 1790 | 1831 | 1771 |
| 9 | 3223 | 3189 | 2924 |
| 10 | 3357 | 3295 | 3385 |
| 11 | 3550 | 3547 | 3432 |
| 12 | 3663 | 3631 | 3501 |

### S4.3 Excited states of thioformaldehyde, $CSH_2$

The scatter plots of the ML excited-state energies (A), gradients (B), and SOCs (C) of thioformaldehyde, $CSH_2$, are given in Fig. S6 for the test set. A scan along the C-S bond-stretching, where the $S_1$ and $T_1$ states approach each other, is shown in Fig. S5 and compares the potential energy curves and SOCs of ML models and quantum chemistry. The reference energies and SOCs obtained with quantum chemistry and ML agree very well.

The Hessians are further used to compute the frequencies of the equilibrium geometry of $CSH_2$. The results are included as a supplementary file in molden format and are compared to reference quantum chemistry values in Table S5. They agree well to reference values, but are slightly underestimated.

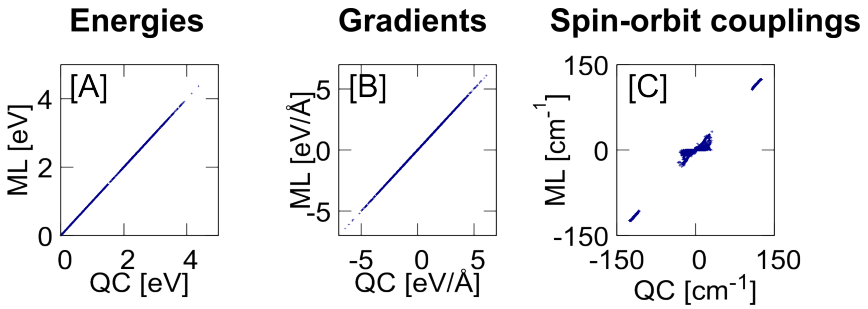


Figure S6: Scatter plots showing the distribution of energies (A), gradients (B), and spin-orbit couplings (C) from ML models used to compute Fig. 4 in the main text.

Table S5: Comparison of frequencies of $CSH_2$ obtained from the quantum chemistry (QC) reference method CASSCF(6,5)/def2-SVP and from SchNarc Hessians.

| Normal mode | QC [$cm^{-1}$] | SchNarc [$cm^{-1}$] |
|---|---|---|
| 1 | 1014 | 956 |
| 2 | 1026 | 990 |
| 3 | 1038 | 1008 |
| 4 | 1577 | 1531 |
| 5 | 3267 | 3247 |
| 6 | 3380 | 3366 |

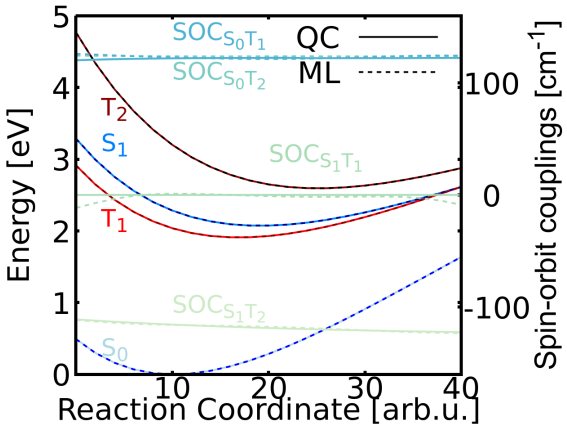


Figure S7: Potential energy curves and the spin-orbit couplings (summed real and imaginary values) between singlets and triplets along the C-S stretching coordinate of $CSH_2$. Continuous lines represent the quantum chemical reference method CASSCF(6,5)/def2-SVP and dashed lines the ML predictions.

## References


[1] Schütt, K. T.; Sauceda, H. E.; Kindermans, P.-J.; Tkatchenko, A.; Müller, K.-R. SchNet – A Deep Learning Architecture for Molecules and Materials. *J. Chem. Phys.* **2018**, *148*, 241722.

[2] Schütt, K. T.; Kessel, P.; Gastegger, M.; Nicoli, K. A.; Tkatchenko, A.; Müller, K.-R. SchNetPack: A Deep Learning Toolbox For Atomistic Systems. *J. Chem. Theory Comput.* **2019**, *15*, 448–455.

[3] Plasser, F.; GÃşmez, S.; Menger, M. F. S. J.; Mai, S.; González, L. Highly Efficient Surface Hopping Dynamics Using a Linear Vibronic Coupling Model. *Phys. Chem. Chem. Phys.* **2019**, *21*, 57–69.

[4] Köppel, H.; Domcke, W.; Cederbaum, L. S. Multimode molecular dynamics beyond the Born-Oppenheimer approximation. *Adv. Chem. Phys.* **1984**, *57*, 59–246.

[5] A. H. Larsen *et al.*, The Atomic Simulation Environment—a Python Library for Working with Atoms. *J. Phys.: Condens. Matter* **2017**, *29*, 273002.

[6] Westermayr, J.; Gastegger, M.; Menger, M. F. S. J.; Mai, S.; González, L.; Marquetand, P. Machine Learning Enables Long Time Scale Molecular Photodynamics Simulations. *Chem. Sci.* **2019**, *10*, 8100–8107.

[7] H. Lischka *et al.*, High-Level Multireference Methods in the Quantum-Chemistry Program System COLUMBUS: Analytic MR-CISD and MR-AQCC Gradients and MR-AQCC-LRT for Excited States, GUGA Spin-Orbit CI and Parallel CI Density. *Phys. Chem. Chem. Phys.* **2001**, *3*, 664–673.

[8] Wigner, E. On The Quantum Correction For Thermodynamic Equilibrium. *Phys. Rev.* **1932**, *40*, 749–750.

[9] H.-J. Werner *et al.*, MOLPRO, version 2012.1, A Package of Ab Initio Programs. 2012; see http://www.molpro.net.

[10] Behler, J. Constructing High-Dimensional Neural Network Potentials: A Tutorial Review. *Int. J. Quantum Chem.* **2015**, *115*, 1032–1050.

[11] Gastegger, M.; Behler, J.; Marquetand, P. Machine Learning Molecular Dynamics for the Simulation of Infrared Spectra. *Chem. Sci.* **2017**, *8*, 6924–6935.

[12] Mai, S.; Marquetand, P.; González, L. Nonadiabatic Dynamics: The SHARC Approach. *WIREs Comput. Mol. Sci.* **2018**, *8*, e1370.

[13] Richter, M.; Marquetand, P.; González-Vázquez, J.; Sola, I.; González, L. SHARC: Ab Initio Molecular Dynamics with Surface Hopping in the Adiabatic Representation Including Arbitrary Couplings. *J. Chem. Theory Comput.* **2011**, *7*, 1253–1258.

[14] Mai, S.; Richter, M.; Ruckenbauer, M.; Oppel, M.; Marquetand, P.; González, L. SHARC2.0: Surface Hopping Including Arbitrary Couplings – Program Package for Non-Adiabatic Dynamics. sharc-md.org, 2018.

[15] Tully, J. C. Molecular Dynamics with Electronic Transitions. *J. Chem. Phys.* **1990**, *93*, 1061–1071.

[16] Granucci, G.; Persico, M.; Zoccante, A. Including Quantum Decoherence in Surface Hopping. *J. Chem. Phys.* **2010**, *133*, 134111.

[17] Feynman, R. P. Forces in Molecules. *Phys. Rev.* **1939**, *56*, 340–343.

[18] Herzberg, G.; Longuet-Higgins, H. C. Intersection of Potential Energy Surfaces in Polyatomic Molecules. *Discuss. Faraday Soc.* **1963**, *35*, 77–82.

[19] Worth, G. A.; Cederbaum, L. S. Beyond Born-Oppenheimer: Molecular Dynamics Through a Conical Intersection. *Ann. Rev. Phys. Chem.* **2004**, *55*, 127–158.

[20] Ryabinkin, I. G.; Joubert-Doriol, L.; Izmaylov, A. F. Geometric Phase Effects in Nonadiabatic Dynamics near Conical Intersections. *Acc. Chem. Res.* **2017**, *50*, 1785–1793.

[21] Köppel, H.; Gronki, J.; Mahapatra, S. Construction Scheme for Regularized Diabatic States. *J. Chem. Phys.* **2001**, *115*, 2377–2388.

[22] Oloyede, P.; MilâĂŹnikov, G.; Nakamura, H. Generalized Trajectory Surface Hopping Method Based on the Zhu-Nakamura Theory. *J. Chem. Phys.* **2006**, *124*, 144110.

[23] Ishida, T.; Nanbu, S.; Nakamura, H. Clarification of Nonadiabatic Chemical Dynamics by the Zhu-Nakamura Theory of Nonadiabatic Transition: From Tri-Atomic Systems to Reactions in Solutions. *Int. Rev. Phys. Chem.* **2017**, *36*, 229–285.

[24] Akimov, A. V. A Simple Phase Correction Makes a Big Difference in Nonadiabatic Molecular Dynamics. *J. Phys. Chem. Lett.* **2018**, *9*, 6096–6102.

[25] An, H.; Baeck, K. K. Practical and Reliable Approximation of Nonadiabatic Coupling Terms between Triplet Electronic States Using Only Adiabatic Potential Energies. *Chem. Phys. Lett.* **2018**, *696*, 100 – 105.

[26] Baeck, K. K.; An, H. Practical Approximation of the Non-Adiabatic Coupling Terms for Same-Symmetry Interstate Crossings by Using Adiabatic Potential Energies Only. *J. Chem. Phys.* **2017**, *146*, 064107.

[27] Dreuw, A.; Wormit, M. The Algebraic Diagrammatic Construction Scheme for the Polarization Propagator for the Calculation of Excited States. *WIREs Comput. Mol. Sci.* **2015**, *5*, 82–95.

[28] 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.

[29] Frankcombe, T. J. Using Hessian Update Formulae to Construct Modified Shepard Interpolated Potential Energy Surfaces: Application to Vibrating Surface Atoms. *J. Chem. Phys.* **2014**, *140*, 114108.

[30] Westermayr, J.; Faber, F.; Christensen, A. S.; von Lilienfeld, A.; Marquetand, P. Neural Networks and Kernel Ridge Regression for Excited States Dynamics of $CH_2NH_2^+$: From Single-State to Multi-State Representations and Multi-Property Machine Learning Models. *Mach. Learn.: Sci. Technol.* **2020**, *in press*, DOI:10.1088/2632–2153/ab88d0.