This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

SAND2019-13005J

KinBot: Automated stationary point search on potential energy surfaces

Ruben Van de Vijver[*] and Judit Zádor[†]

Combustion Research Facility, MS 9055, Sandia National Laboratories, Livermore, CA 94551-0969, USA

Declarations of interest: none

[*] Current address: Laboratory for Chemical Technology, Ghent University, Ghent, Belgium
[†] Corresponding author. E-mail: jzador@sandia.gov, 1 (925) 294-3603, Address: 7011 East Ave, Livermore, CA 94550, USA

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Abstract

KinBot is a Python code that automatically characterizes kinetically important stationary points on reactive potential energy surfaces and arranges the results into a form that lends itself easily to master equation calculations. This version of KinBot tackles C, H, O and S atom containing species and unimolecular (isomerization or dissociation) reactions. KinBot iteratively changes the geometry of the reactant to obtain initial guesses for reactive saddle points defined by KinBot's reaction types, which are then optimized by a third-party quantum chemistry package. KinBot verifies the connectivity of the saddle points with the reactant and identifies the products through intrinsic reaction coordinate calculations. New calculations can be automatically spawned from the products to obtain complete potential energy surfaces. The utilities of KinBot include conformer searches, projected frequency and hindered rotor calculations, and the automatic determination of the rotational symmetry numbers. Input files for popular RRKM master equation codes are automatically built, enabling an automated workflow all the way to the calculation of pressure and temperature dependent rate coefficients. Four examples are included. (*i*) [1,3]-sigmatropic H-migration reactions of unsaturated hydrocarbons and oxygenates are calculated to assess the relative importance of suprafacial and antrafacial reactions. (*ii*) Saddle points on three products of gamma-valerolactone thermal decomposition are studied and compared to literature potential energy surfaces. (*iii*) The previously published propene+OH reaction is reproduced to show the capability of building an entire potential energy surface. (*iv*) All species up to C4 in the Aramco Mech 2.0 are subjected to a KinBot search.

Keywords: Automation, Potential energy surface, Temperature and pressure dependent rate coefficients

PROGRAM SUMMARY	
Program title:	*KinBot*
Licensing provisions (please choose one): MIT license (MIT)/Apache License, 2.0 (Apache-2.0)/ Creative Commons by 4.0 (CC by 4.0)/ Creative Commons Zero (CC0)/BSD 3-Clause/BSD 2-Clause/ GNU General Public License 3 (GPL)/GNU General Public License 2/ LGPL/ CC by NC 3.0/ MPL-2.0	*BSD 3-Clause*
Programming language:	*Python*
Supplementary material (if any):	1. *A static version of the source code (KinBot.tar),* 2. *The manual for the static version (KinBot_Manual.pdf)* 3. *Geometries and energies of the stationary points on the potential energy surface of the sigmatropic reaction search (sigmatropic_H_shift.out)* 4. *Geometries and energies of the stationary points on the potential energy surface of the propene+ OH central and terminal addition reaction (propene+oh central addition.out, propene+oh terminal addition.out)* 5. *Geometries and energies of the stationary points on the potential energy surface of gamma valerolactone, 4-pentenoic acid and 3-pentenoic acid (GVL energies and geometries.out, 4PA energies and geometries.out, 3PA energies and geometries.out)* 6. *Example runs including all input and output files for a one-well search for propanol radical, full PES search for the n-pentyl radical, a search for all homolytic scission in propanol, and the reaction searches for GVL (output.zip)* 7. *Results of symmetry calculations for a literature benchmark dataset (Symmetry_correct.pdf, Symmetry_wrong.pdf)*
*Journal Reference of previous version:***	*None*
*Does the new version supersede the previous version?:***	*None*
*Reasons for the new version:***	*None*
*Summary of revisions:***	*None*
Nature of problem (approx. 50-250 words):	*Automatic discovery of unimolecular reaction pathways (isomerization and dissociation) for molecules and radicals relevant in gas-phase combustion and atmospheric chemistry, including oxidation and pyrolytic processes for structures including carbon, oxygen, sulfur and hydrogen atoms. The reactants, products, and transition states are characterized using a suite of tools coupled to electronic structure codes, and the results are provided in a format that lends itself easily to calculating rate coefficients based on statistical rate theories with other external codes.*

Solution method (approx. 50-250 words):	*Reaction pathways are identified using heuristic searches starting from a reactant by iteratively altering its geometry toward a good guess for a transition state for reactions with barriers. The transition state is identified as a first-order saddle point on the potential energy surface, which is located using local optimization methods of third-party quantum chemistry codes. We use intrinsic reaction coordinate calculations to verify the direct connectivity of the saddle point to the reactant and to identify the product species. Conformational searches, hindered rotor potentials, frequency calculations, and high-level optimizations yield the necessary data for RRKM master equation calculations.*
Additional comments including Restrictions and Unusual features (approx. 50-250 words):	*KinBot is designed to run on Unix clusters, and is written in Python, compatible with versions 2.7 and 3. it communicates with a PBS or SLURM workload manager to submit quantum chemistry calculations to third-party software. It makes use of a modified fork of ASE for the input writing, calling and output parsing of the quantum chemistry software which has been tested with Gaussian (G09RevD.01). OpenBabel (2.4.1) and RDKit (2018.09.01) are used to convert smiles to internal species representations and for species comparison and results visualization. The output of KinBot can be visualized with the PESViewer script, and graph structures are drawn using NetworkX. The master equation solvers MESS or MESMER are needed to calculate rate coefficients at the end of a given run. This version of KinBot can handle H, C, S, and O atom-containing molecules, and searches for isomerization and dissociation pathways.*
References:	

1. Introduction

The determination of accurate rate coefficients remains a key challenge for a wide variety of chemical research areas, such as heterogeneous catalysis, liquid-phase chemistry and gas-phase processes. While research in the past decades has focused intensively on how to accurately calculate rate coefficients [1-3], the developed methods often incorporate several manual actions and need expert user knowledge. This is a particularly severe hinderance for finding reaction pathways, which forms the basis of rate coefficient calculations. As a consequence, the large number of rate coefficients required to describe the relevant elementary reactions for complex chemical systems are not known from first principles calculations, and, therefore, the availability of detailed predictive models is limited especially for important gas-phase chemical environments. These are often governed by several thousand or even tens of thousands of reactions [4]. The characterization of such a large number of reactions using traditional methods is simply not feasible. Therefore, it is desirable to create tools that automatically search for reaction pathways and calculate rate coefficients, largely reducing the need for manual interventions, and speed up the rate at which scientific discovery can be made in these fields.

KinBot, our open-source code is one such tool, designed with gas-phase radical chain chemistry applications in mind. Radical chain chemistry is ubiquitous in many areas. For instance, understanding hydrocarbon and oxygenate combustion in detail is indispensable to optimize fuel blends, assess the combustion properties of new, sustainable fuel candidates, minimize pollutant production, and in combination with computational fluid dynamics, design and optimize combustion devices. Similarly, the reactions of organic molecules in the atmosphere determine their potential impact on human health and on the environment. These two complex systems, combustion and atmospheric chemistry, driven by radical chain reactions are the central applications for the present work. Our goal is to uncover reaction pathways computationally in a way that is convenient for automated reaction mechanism generator codes [5, 6] and related ab initio rate coefficient calculator codes [7-10] in these areas. The key challenges we address are the location and characterization of all chemically significant stationary points on a multidimensional and multiwell potential energy surface (PES).

Several computational tools with similar objectives have been developed in the past. We refer the reader to recent reviews for a detailed overview [11-13], here we only briefly discuss a

few methods most related to ours. Zimmermann [14] developed methods to automatically discover reaction steps using double-ended searches, i.e. with knowledge of the structure of the reactant(s) and product(s), and published several follow-up papers on similar methods [11, 15], including also single-ended searches [16]. The proposed reactions that the codes search for are generated from the reactant structure by applying a set of simple rules, which are system independent. Suleimanov and Green [17] followed a similar approach by combining double-ended and single-ended transition state searches applied to a large number of potential atom rearrangements in molecules. Both the methods of Zimmermann [11, 14-16] and of Suleimanov and Green [17] are essentially unbiased searches towards both expected and unexpected reactions and contain no heuristics.

The artificial force induced reaction (AFIR) method of Maeda et al. [18, 19] minimizes a function that consists of the PES and artificial forces. The artificial force depends on the type of the reactant(s) and the reaction and the method maps out the energy as a function of the AFIR path. The structure corresponding to the highest energy along the path is used as input for a true transition state search.

Bhoorasingh et al. [20, 21] developed AutoTST to locate the transition state of certain reaction classes, mostly abstraction reactions. Their rapid search algorithm is using tabulated information about the saddle points for the various reacting centers of the reactants to initiate the full saddle point search. This method is particularly interesting for large kinetic models or large kinetic databases containing many homologous reactions. Their method offers a systematic approach to improve rate coefficients using ab initio calculations, especially for simple abstraction reactions or for reactions in the high-pressure limit. Similarly, the program of Van de Vijver et al. [22] aims at improving the accuracy of the rate coefficients of expected reactions by building a database of known transition state geometries and applying these geometries to similar transition states. Again, this program is limited to high-pressure limit rate coefficients.

The approach by Kim et al. [23] consists of initially generating a large reaction network and selecting a subnetwork in the former that only contains kinetically significant steps. The subnetwork is subjected to transition state theory for a kinetic analysis of the reaction network.

A code called tsscds2018 [24] has been developed to find saddle point geometries based on graph theory using accelerated semi-empirical dynamics simulations and generate a reaction network from the stationary points.

EStokTP [25] is a code that integrates many aspects of potential energy surface exploration and kinetics calculation into one code. It features sophisticated calculations, such as multidimensional tunneling and multidimensional anharmonic treatments with many fallback options to make the code robust and capable of large-scale automated calculations in high performance computing environments. It targets bimolecular reactions and some classes of unimolecular reactions, one step at a time, and aims to yield pressure and temperature dependent rate coefficients with at most a factor of 2 error.

Our code, KinBot, can automatically locate wells and first-order saddle points on a PES and arrange the information about these stationary points in a way that is directly applicable in RRKM-based master equation codes, such as MESS [7, 8] or MESMER [9, 10]. The algorithms in KinBot include the search for conceivable pathways starting from a given reactant, the identification of the saddle points and the product species, conformational search, hindered rotor calculations, and symmetry perception. This version of KinBot is able to handle C, H, O and S atom containing structures, and elementary steps that are either isomerization or dissociation reactions, i.e., no bimolecular processes are searched for in this version.

KinBot has several distinguishing strengths. First, when using KinBot the user only needs to provide the structure of a reactant and KinBot's routines will guide the search towards a plethora of possible reaction pathways automatically. Second, KinBot can explore reactive saddle points on one PES in a single run providing an in-depth description of the reaction pathways. This enables KinBot to be directly coupled to master equation calculations to obtain accurate pressure- and temperature-dependent rate coefficients including well-skipping steps. Third, KinBot is relatively cheap computationally [27], the expense for a successful pathway search not being much more in most cases than it would be for an expert human kineticist. (Note that not all searches lead to a unique and valid saddle point, which results in overhead.) Finally, KinBot can run in a highly parallel mode suitable for high-performance computing applications, exploring not just one, but many reactive PESs at the same time.

The paper is organized as follows. In the Methods section we first present the workflow (section 2.1). In section 2.2 we describe the core functionality of finding reactive pathways. In 2.3 we describe how the species are represented (2.3), which includes discussion of chemically degenerate atoms and of resonance structures. In section 2.4 we describe how the conformational search is done. In sections 2.5 and 2.6 the anharmonic characterization of the species and the

automated calculation of the symmetry numbers are discussed. The part of the workflow that allows the automated calculation of pressure- and temperature-dependent rate coefficients is given in section 2.7. The visualization of the potential energy surfaces is described in section 2.8, and finally, code dependencies and installation instructions are detailed in sections 2.9 and 2.10.

In section 3 we include several examples to illustrate the capabilities of KinBot. First, KinBot is used to search for [1,3]-sigmatropic H-migration reactions to show how our code can be used to explore trends for a given reaction across many species (3.1). Second, the PES for the thermal decomposition of gamma-valerolactone is explored to demonstrate the use of KinBot for finding a plethora of pathways starting from a given well (3.2). Third, to also illustrate the capabilities of KinBot on open-shell species and its ability to explore a complete, fully connected PES, the propene + OH reaction is explored and compared to a comprehensive PES from the literature [28] (3.3). Finally, in section 3.4 the improvement of existing kinetic models with KinBot is shown for a propane combustion model taken from the Aramco Mech 2.0 [29].

Note that the previous version of KinBot published as a Sandia technical report [30] was also successfully used for many reactions, such as for the C_3H_4 + OH PES [31], the unimolecular reactions of 2,5-dimethylhex-1-yl [32], the low-temperature oxidation pathways of tetrahydrofurane [33], 2-butene + OH [34], 1-pentene + OH [35], and the decomposition of a small γ-ketohydroperoxide [27].

2. Methods

2.1. Workflow of calculations in KinBot

KinBot's aim is to go from a single structure all the way to pressure- and temperature-dependent rate coefficients without user intervention. The overall workflow is shown in Figure 1. The starting point for KinBot is a stable species, we refer to it as the reactant, specified in the input. The reactant geometry needs to be provided either directly in Cartesian coordinates (`structure` keyword in the input followed by the coordinates), or indirectly by the SMILES identifier of the reactant (`smiles` keyword followed by the smiles). The first step of the exploration process is that KinBot calls a third-party quantum chemistry software to optimize the starting geometry to a local minimum and confirms that this is a stable structure. For instance, some initial guess

structures might dissociate into fragments, and KinBot does not explore these PESs further. For all subsequently found wells we carry out the same test.

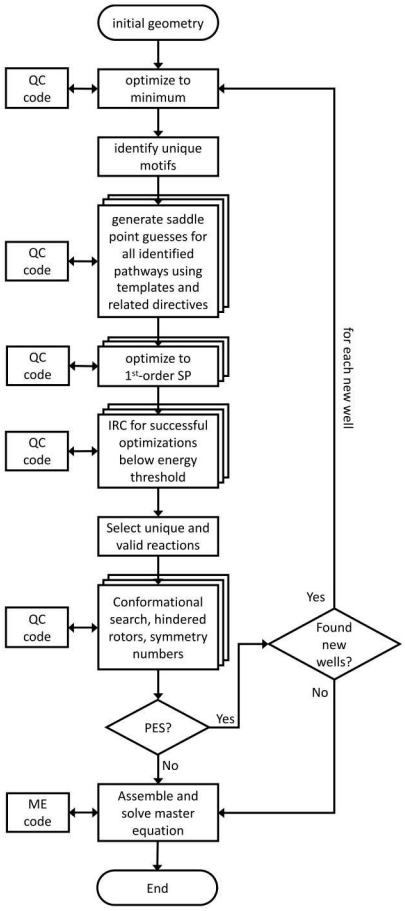


Figure 1: The workflow of KinBot. Tripled boxes indicate parallel operations. QC is the external quantum chemistry code, and ME is the external master equation solver. The PES yes/no option stands for whether the user requested a full search or only the search for channels starting from one well.

KinBot uses its hard-coded reaction template database to identify the various reaction pathways the structure can undergo (section 2.2). This requires the comprehension of the chemical structure (section 2.3), including the identification of chemically equivalent atoms, and resonance structures. The saddle point guesses are generated using geometry modification routines specific to the templates. These typically involve a systematic alteration of the dihedral angles followed by adjustments of bond lengths and bond angles. Once the structure is brought close to a proposed saddle point this way, we rely on the internal local optimizer of the quantum chemistry package to search for a 1st-order saddle point, and we keep the ones that are below a user-prescribed energy threshold. The validity of all saddle points is checked in IRC calculations. A valid saddle point is

one that connects the reactant to a new well or a pair of bimolecular products. We select the unique saddle points (i.e., duplicates are removed), and finally characterize all species: wells, bimolecular products, and saddle points that KinBot found. The characterization includes conformer search (section 2.4), hindered rotor scans (section 2.5) and symmetry number evaluation (section 2.6). If the user requested an all-over search (`pes` mode, see the PES rhomboid in Figure 1), then KinBot spawns new calculations from the newly found wells, until no more new wells are found within the prescribed energy range. Finally, the kinetic master equation is assembled and solved using external solvers to arrive at pressure- and temperature-dependent rate coefficients (section 2.7). KinBot's workflow is highly parallel, as indicated by the tripled process boxes in Figure 1.

2.2. Location of transition states

The central feature of KinBot is to find reaction pathways starting from a single structure. In order to find possible reaction pathways (`reaction_search` keyword), KinBot makes use of heuristics categorized in several reaction types (`families` keyword), given in Table 1, with many of the reaction types similar to the reaction families found in RMG [5], but are largely extended to include more possibilities.

This version of KinBot is tailored to gas-phase molecular and radical chemistry of hydrocarbons, oxygenates and sulfur-containing compounds, which means that the allowed atom types are carbon, oxygen, sulfur and hydrogen. The list of atoms will be expanded in future versions. To increase efficiency, KinBot uses four user-defined levels of single-reference theories when calculating the reaction pathways. The lower, cheaper levels are used for exploration, while the more expensive methods are used for refinement. Internal modifications of the structures are done at the AM1 level (L0). These are largely inconsequential to the search, and aim to prepare the geometries for a stationary point search. In our experience often a low level (L1, `method` and `basis` keywords) such as B3LYP/6-31G is sufficient to find the stationary points, which can then be directly refined at a higher level (L2, `high_level` keyword with `high_level_method` and `high_level_basis` options) such as M06-2X/6-311++G(d,p). Finally, as in typical dual-level calculations, accurate electronic energies can be obtained using e.g. coupled cluster theory (L3). The input for these L3 calculations are prepared automatically by KinBot in a Molpro [36] format. The calculations need to be run offline rather than via ASE in an active KinBot run due to

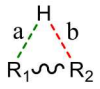
the large computational cost. (When these offline calculations are done, KinBot has to be restarted to use the results of these calculations.)

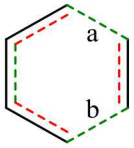
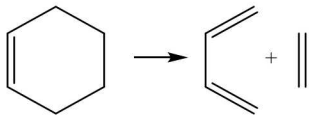
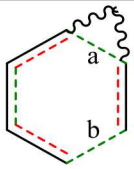
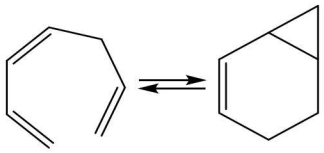
A species is eligible to undergo a transformation defined by a reaction template if a structural motif is found in the species which matches with the corresponding motif. Motifs are defined by a set of atoms connected in a particular way, possibly augmented with information about bond orders and the presence of cycles. For a given species all unique motifs are enumerated to find all possible pathways belonging to each reaction type. For example, since H migrations can proceed through different ring sizes, several motifs are defined to find H migration reactions. The first atom of this linear motif is a radical of any element, noted Xrad, and the final atom is a hydrogen noted H. To find a H migration transition state proceeding through a three-membered ring, i.e., a 1-2 H migration, the motif Xrad-X-H is searched for in the species, where X is an arbitrary atom. Similarly, the motifs Xrad-X-X-H, Xrad-X-X-X-H, and Xrad-X-X-X-X-H are used to find H migrations proceeding through four, five and six membered transition states, respectively. By searching for these motifs in the propanol radical $O\bullet CH_2CH_2CH_3$, several possible pathways are found. The shortest Xrad-X-H motif gives rise to two reaction possibilities, with both of the hydrogens in alpha position of the oxygen atom. These two reaction possibilities are equivalent since the two hydrogen atoms are equivalent, and only one of these motif matches is used for a reaction search. Similarly, the search for 1-3 and 1-4 H migration reactions leads to 2 and 3 reaction possibilities, respectively, of which only one of each needs to be constructed. Note that for the motif search of resonantly stabilized structures, each resonance isomer of the molecule is considered.

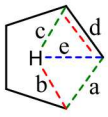
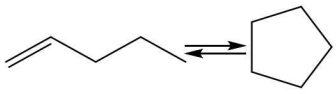
Once the key atoms that participate in a reaction are identified, a guess geometry for the transition state is generated starting from the reactant geometry based on the data tabulated in Table 1. The tabulated values are optimized for L1 = B3LYP/6-31G without S atoms, and for L1 = B3LYP/6-31+G(d) for S-atom-containing species. To arrive at the guess geometries, KinBot uses either one of the following three strategies, named as *direct*, *indirect*, and *scan*. Note that adding new reaction templates requires adding new routines to KinBot based on one of the `reac_[reaction_name].py`. For each template the motif that defines the reaction needs to be specified as well as the required geometrical modifications to generate good guess structures for the new pathway. The new pathway also has to be registered in the `reaction_finder.py` routine.

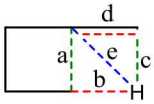
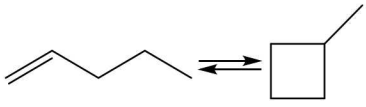
Table 1: Reaction templates in KinBot. The keyword(s) for a given template is (are) provided below the name in the Reaction column. Note that because of the internal structure of the code, multiple names may belong to one template. Only the unimolecular directions are searched for as shown by the arrows. The initial guess for the saddle point geometry are defined by the templates, with bond lengths in Angstroms. When no initial geometry is prescribed (Reverse 1,2 cycloaddition), the maximum energy structure is searched for by doing a constrained optimization along the reactive bonds. The dashed bonds represent breaking and forming bonds. Depending on the direction of the reaction, the red bonds are formed and the green ones are broken, or vice versa. The blue dashed lines represent ephemeral bonds present only in the vicinity of the saddle point. The template geometries were formulated using B3LYP/6-31G (B3LYP/6-31+G* for sulfur containing compounds) and tested at the same level of theory. We suggest using this level of theory as L1 for the reaction search.

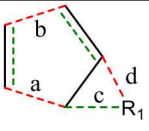
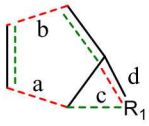
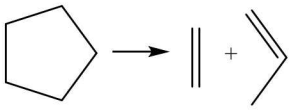
Reaction	Template geometries	Example

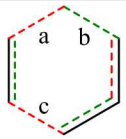
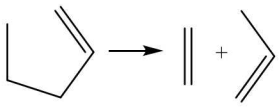
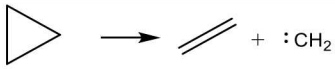
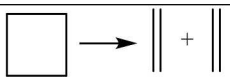
H migration in closed-shell molecules
`intra_H_migration_suprafacial`
`ketoenol`
`cpd_H_migration`
	
	R1=C: a = 1.35
	R2=C: b = 1.35
	R1=O: a = 1.20
	R2=O: b = 1.20
		

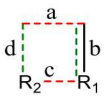
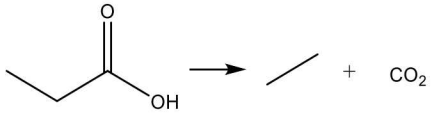
Reverse Diels-Alder `Diels_alder_addition`	 a = 2.2 b = 2.2	
Intra Diels-Alder `Intra_Diels_alder_R`	 a = 1.8 b = 2.2	

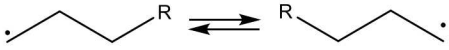
Endocyclic closed-shell cyclization
`Intra_RH_Add_Endocyclic_F`
`Intra_RH_Add_Endocyclic_R`
	
	a = 2.0
	b = 1.3
	c = 1.3
	d = 1.4
	e = 1.8
		

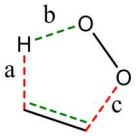
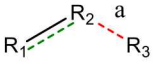
Exocyclic closed-shell cyclization
`Intra_RH_Add_Exocyclic_F`
`Intra_RH_Add_Exocyclic_R`
	
	a = 2.0
	b = 1.9
	c = 1.4
	d = 1.4
	e = 1.3
		

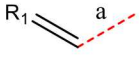
Generalized Korcek step 2 reaction `Korcek_step2`	 a = b = 2.0 R1 = H: c = d = 1.35 R1 = C,O: c = d = 1.8  a = b = 2.0 c = 1.8 d not specified	

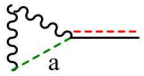
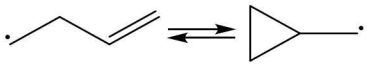
Retro-ene reaction `Retro_Ene`	 a = 1.35 b = 1.35 c = 2.0	
Reverse 1,2 cycloaddition `r12_cycloaddition`	 a = b = scan	
Reverse 2,2 cycloaddition `r22_cycloaddition`	 a = 2.2 b = 2.2	
Reverse 1,2 R insertion `r12_insertion_R`	 Rn = C a = 1.67 b = 2.2 c = 1.9 R1 = H a = 1.7 b = 1.09 c = 2.2 R1 = O a = 1.67 b = 1.8 c = 1.9	

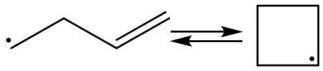
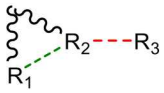
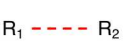
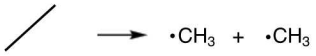
Reverse 1,3 insertion
`r13_insertion_CO2`
`r13_insertion_ROR`
`r13_insertion_RSR`
	 a = 2.0
	R1 = O: b = 1.3
	R1 = C: b = 1.45
	R2 = H: c = d = 1.3
	R2 = C, O: c = d = 2.0
		

R migration in radicals
`intra_R_migration`
`intra_OH_migration`
`intra_H_migration`
`12_shift_S_F`
`12_shift_S_R`
	 R1=C: a = 1.35
	R2=C: b = 1.35
	R1=O: a = 1.20
	R2=O: b = 1.20
		

HO_2 elimination `HO2_Elimination_from_PeroxyRadical`	 a = 1.3 b = 1.3 c = 2.0	
Radical β-scission `R_Addition_MultipleBond`	 R_1-R_2-R_3 C-C-C: a = 2.20 C-C-H: a = 1.79 C-C-O: a = 2.04 O-C-C: a = 2.12 O-C-H: a = 1.84 O-C-O: a = 2.04 C-O-C: a = 2.04 C-O-H: a = 1.42 C-O-O: a = 2.04 O-O-C: a = 2.04	

Radical α-scission
`R_Addition_COm3_R`
`R_Addition_CSm3_R`
	 a = 2.2	

Exocyclic intramolecular radical addition `Intra_R_Add_ExoTetCyclic_F`	 a = 2.2	

`Intra_R_Add_Exocyclic_F`		
Endocyclic intramolecular radical addition `Intra_R_Add_Endocyclic_F`	 a = 2.2	
Cyclization-elimination `Cyclic_Ether_Formation`	 Ri=C, Rj=C: Ri-Rj = 1.80 Ri=C, Rj=O: Ri-Rj = 1.68 Ri=C, Rj=H: Ri-Rj = 1.31 Ri=O, Rj=O: Ri-Rj = 1.78 Ri=O, Rj=H: Ri-Rj = 1.14	
Homolytic scission (note that this reaction type does not include a saddle point search, KinBot only lists the energies of the products. Invoked with the `homolytic_scissions` keyword)		

In the direct and indirect cases, the guess geometry shown in Table 1 is reached in *n* steps starting from the reactant. The guess cannot be constructed in a single step, because some large modifications result in atom clashes. Every step towards the guess consists of a geometry modification (GM) determined by KinBot and a constrained optimization (CO) carried out by the external quantum chemistry software at the L0 = AM1 level.

The GM consists of three steps. (1) Dihedral modifications are accomplished through rigid rotations of two groups relative to each other around the bond that connects them. If the bond around which the rotation is done is part of a cycle, the cycle is broken temporarily along the furthest bond in the cycle relative to the rotating bond. This leads to ring distortion, which is corrected in the third step of the GM. Each dihedral modification is followed by a CO, while fixing the dihedral(s). (2) The modification of the angles is also done by dividing the molecule into two rigid parts and changing their relative angles. Cycles are treated the same way as in the first step of the GM. Each angle modification is also followed by a CO, keeping the angle(s) and the previously modified dihedral(s) fixed. (3) The bond modifications adjust the bond lengths to the values listed in Table 1. There are two additional constraints: (i) the original connectivity of the atoms is kept unless stated otherwise in Table 1, and (ii) bonds and angles stay as close as possible to their original values. This requires the solution of a set of coupled equations. For simplicity, we define all of our coordinates as a distance between two atoms. Then a cost function A to be minimized can be defined as:

$$A(\boldsymbol{x}) = \sum_{i,j}^{N} \left(w_{ij} \left({d_{ij}}^2 - \left(x_i - x_j \right)^2 \right) \right)^2 \qquad \text{Eq. 1}$$

The sum includes all N atoms of the species, d_{ij} is the original distance between atoms i and j, x_i and x_j are the Cartesian coordinates of atom i and j, and w_{ij} is a weight factor. For atoms that are not connected to each other and do not have a common neighbor, an artificial repulsive potential is added to the cost function. This avoids two atoms that are not bonded to come too close to each other. If the distance between two atoms exceeds a threshold, the weight of this potential becomes 0.

The BFGS algorithm [37] has been implemented to minimize $A(\boldsymbol{x})$ to a local minimum close to the starting structure. The quasi-Newton steps are performed by a line search in the direction defined by the product of an approximate inverse Hessian matrix with the gradient. The approximate inverse Hessian (B^{-1}) is initialized to be the identity matrix and is updated in each step by

$$B_{k+1}^{-1} = B_k^{-1} + \frac{\left(s_k^T y_k + y_k^T B_k^{-1} y_k\right)\left(s_k s_k^T\right)}{\left(s_k^T y_k\right)^2} - \frac{B_k^{-1} y_k s_k^T + s_k y_k^T B_k^{-1}}{s_k^T y_k} \qquad \text{Eq. 2}$$

Here, k is the iteration step, s_k is the update of the Cartesian coordinates in step k, and y_k is the update of the gradient in step k.

Note that above optimization towards new coordinates is far from ideal, because neither angle nor dihedral contributions are incorporated in the cost function explicitly, and the weights are not parametrized to any true potential energy surface. However, this algorithm is sufficient for its current use in KinBot, because only small steps are taken in the coordinate modification, and also, a more accurate constrained optimization is followed after the modifications.

The direct strategy can be used when all atoms in the reactive center are connected to each other in the reactant, such as in the reverse Diels-Alder reaction. The Cartesian coordinates of the reactive atoms are changed in n steps by $\frac{d_{\mathrm{TS}} - d_{\mathrm{R}}}{n}$ per step, where d is the distance between two atoms in Angstroms and subscripts ‘TS’ and ‘R’ stand for transition state and reactant respectively. The

interatomic distances in the reactant are taken from the optimized geometry of the reactant, while the key interatomic distances at the transition state are tabulated, see Table 1.

The indirect strategy is used when two parts of a molecule react with one another in the absence of a direct chemical bond between these parts, such as in an R migration reaction where the cycle size is larger than three. In this case, the strategy has three stages. The first stage consists of changing the conformation of the molecule such that the reactive moieties get sufficiently close to each other. This is achieved by finding the shortest sequence of bonds that separate one reactive moiety from the other, and updating the m dihedral angles $\varphi_{\mathrm{R}i}$ along this connection in n steps, each by $-\frac{\varphi_{\mathrm{R}i}}{n}$. This leads to the formation of a planar structure with the reactive moieties close to each other. When $m > 5$, the structure is not made entirely planar to avoid non-bonded atoms to be too close, in which case the increments are $\frac{15-\varphi_{\mathrm{R}i}}{n}$. In the second stage of the indirect strategy, the bond angles of each bond in the planar ring are updated to a value of $\frac{180\cdot(m-2)}{m}$ in a single step, keeping the previously constrained dihedrals fixed. The final stage of the indirect strategy is the direct strategy: the bond distances are updated towards the tabulated initial guess.

The scan strategy is used for reaction families where a direct initial guess for the transition state often performs poorly, as is the case for the reverse 1,2-cycloaddition. The initial guess here is created by calculating the electronic energy along the active bond in steps of 0.1 Å until a maximum is found or the maximum of 30 steps is reached (`scan_step` keyword can be used to change this). The maximum energy structure, if found, is used as initial guess for the saddle point optimization.

The last step of all strategies is to optimize the guess structure to a first-order saddle point at L1 theory, and if a first-order saddle point is found, the structure is reoptimized at L2. The vibrational frequencies are assigned to the structure also at L2. The optimization is done with the internal saddle-point finding routine of the quantum chemistry code, such as the Berny algorithm in Gaussian. The idea is that KinBot has already brought the structure close enough to a proposed saddle point so that local optimizers can relatively easily find a nearby saddle point. In order to facilitate this, we request a full Hessian calculation at the beginning of the optimization.

The saddle points are verified by Intrinsic Reaction Coordinate (IRC) calculations in both directions starting from the optimized saddle point structure. A calculation is determined to be successful if one IRC calculation leads to the reactant, and the other one leads to a structure with a

different connectivity, which is in this case assigned as the product of the reaction. The product can either be unimolecular, or can consist of two or more fragments. In the latter case the structure is first optimized with all of the fragments in a single quantum chemistry calculation, and if they remain unbonded at the end of this calculation, they are split into separate species and reoptimized separately. The energies and frequencies are registered from these final, separated calculations.

For the saddle point and the products KinBot carries out a conformational search following the procedure described in 2.4. Importantly, however, the bonds at the saddle point need special treatment, because they cannot be obtained straightforwardly from the Cartesian coordinates as some forming or breaking bonds are considerably longer than in stable species. Instead, the bonding of the transition state is obtained by taking elementwise the maximum of the bond matrices of the reactant and product(s). For example, a bond transitioning from single to double in the course of a reaction is considered as a double bond by KinBot and is not treated as an internal rotor. Similarly, this approach treats the cycles in cyclic transition states as ordinary cycles from the point of view of conformational search.

An additional note has to be made concerning the quantum chemistry methods used to optimize to a transition state. DFT methods, especially the B3LYP functional, tend to underpredict barriers. At the same time, radical β-scission reactions often have very small reverse barriers, on the order of 1-2 kcal mol^{-1}, which can easily disappear when L1 = B3LYP/6-31G. To avoid this, β-scission transition state searches are automatically done at the MP2/6-31G level at the L1 stage. The energies are brought to the same zero by using the same L2 level further along the workflow.

2.3. Species representation

KinBot contains its own species and structure representation. However, the internally stored species can be easily translated into other representations to use the features of RDKit and OpenBabel. Species representation in KinBot is done by (1) the Cartesian coordinates of the atoms in the molecule, (2) a vector containing the element of each atom, and (3) a bond matrix constructed from the Cartesian coordinates, as shown in Figure 2 for formaldehyde. For two atoms to be bonded to each other we use the bond distance cutoff (BDC) values shown in Table 2. The multiplicity of the bond is determined in a localized π-electron model as described later in this subsection. Within KinBot, each species gets a unique ChemID, which is an integer number calculated from the bond matrix, and can be used to compare species and ensure species uniqueness during a PES search.

The ChemID is the sum of the atom ID's (aID's) of each atom, followed by the multiplicity of the species (1 for singlet, 2 for doublet, 3 for triplet, etc.). The calculation of aID is described below and is illustrated in Figure 3.

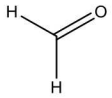
Table 2: Atoms *i* and *j* are considered bonded to each other in stable species if their interatomic distances are less than the corresponding bond distance cutoff ($BDC_{ij} = BDC_{ji}$). Units are Å.

j \ *i*	H	C	O	S
H	1.10			
C	1.31	1.80		
O	1.14	1.96	1.78	
S	1.608	2.18	2.18	2.48

The bonding environment of an atom is encoded in an integer called atom ID (aID). This is a layered integer number in which the n^{th} layer contains the mass number of the atom's n^{th} neighbors. Three digits are reserved per layer, but KinBot allows for the layers to cross each other. The number is preceded by the mass number of the atom itself as the zeroth layer so to speak. The number of layers is cut off at 7, which is sufficient for typical structures relevant for gas-phase radical chemistry. If there are less than 7 layers of neighbors for a given atom, the empty layers consist of three zeros. The aID values for 2-methylpropenol radical are shown in Figure 3, with the number of layers cut off at 4 for the sake of simplicity in this illustration.

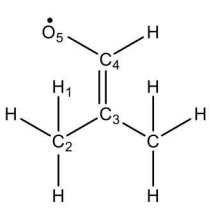
Starting from a reactant, there can be several chemically equivalent reaction pathways towards a product. In order to eliminate duplicate searches, it is necessary to know which atoms in a molecule are chemically equivalent. In KinBot, two atoms are equivalent (a) if they have the same aID (bonding environment) and (b) if the shortest path along bonds that connect the two atoms does not include atoms with double bonds or atom with bonds that are part of a ring.

```
0.98   0.08  -0.06
2.20   0.08  -0.06
2.79  -0.11  -0.97
2.79   0.27   0.86
```



$$\begin{bmatrix} O \\ C \\ H \\ H \end{bmatrix} \quad \begin{bmatrix} 0 & 2 & 0 & 0 \\ 2 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

Figure 2: Species representation in KinBot exemplified by formaldehyde. Top: Cartesian coordinates, Bottom: vector of chemical elements, and the correspondingly ordered bond matrix.



Layer	0	1	2	3	4	aID
H1	1	012	014	024	020	1012014024020
C2	12	015	024	020		12015024020000
C3	12	036	023			12036023000000
C4	12	029	024	006		12029024006000
O5	16	012	013	024	006	16012013024006
					ChemID:	531040980740262

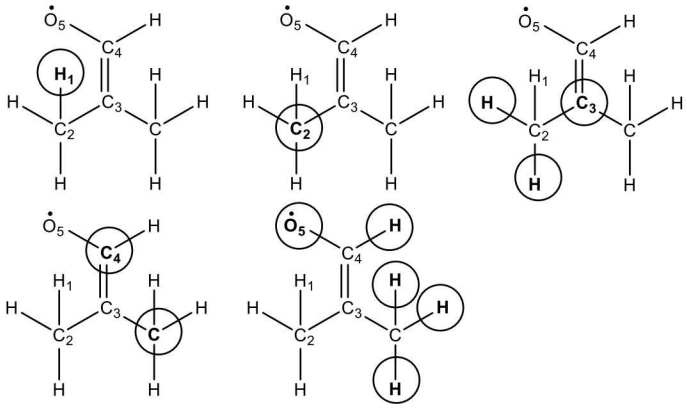


Figure 3: Illustration of the aID and ChemID calculation in the 2-methylpropenoyl radical. For the hydrogen atom H_1, the first number is its mass number, i.e. 1. Its only neighbor is a carbon atom with a mass number of 12, which is the number in the first layer. The next layer is 14 because C_1 has a carbon atom and two more hydrogen atoms bonded to it. The algorithm continues until the final id's are obtained. In case no more atoms are left to visit, the layer consists of zeros. In this example we only use 4 layers, but KinBot uses 7 layers. The ChemID is the sum of the atom ID's (aID's) of each atom, followed by the multiplicity of the species, which is 2 (doublet) in the example.

Two atoms with the same aID have the same bonding environment, however, the global structure of a molecule can still create differences between atoms with the same aID. By

considering the 2-methylpropenoyl radical (see Figure 3), the hydrogen atoms on each of the methyl groups have identical aIDs. The three hydrogen atoms bonded to C_2 can be abstracted internally by the radical site on the oxygen atom leading to three energetically identical reaction pathways within a hindered rotor representation. The hydrogens on the other methyl group are more difficult to abstract because of the presence of the double bond between C_3 and C_4. There is thus no six-fold degenerate pathway in a hindered rotor representation, rather there are two reaction pathways each with three-fold degeneracy. To perceive this difference, KinBot maps out the shortest pathways between every atom pair with identical aIDs and when at least one atom along that pathway is doubly bonded or is part of a cyclic structure, the two members of the atom pair are not deemed equivalent. Sometimes for highly symmetric molecules this strict definition causes KinBot to identify two equivalent atoms as different, which leads to the exploration of two equivalent pathways. This slightly decreases the computational efficiency, but does not lead to errors as equivalent pathways are filtered again later in the workflow of KinBot.

KinBot uses a localized π-electron model to represent structures: bonds are defined by an integer and no partial bond orders are allowed. Similarly, radical electrons are positioned on a single atom. However, to find all reactions of resonantly stabilized molecules, it is necessary in our framework to consider all of the possible resonance structures. The same is true for open shell triplets, where both radical sites can be reactive in the various reactions. For the 2-methylpropenoyl radical in Figure 3, for example, a hydrogen migration starting from the radical on the oxygen atom leads to the 2-methyl-1-propeno-3-yl radical. This reaction, however, is not considered when starting from the resonance isomer of the 2-methylpropenoyl radical, i.e., the 2-methylpropanon-2-yl radical. On the other hand, the latter leads to the 2-methylpropanon-3-yl radical, for which no pathway "exists" from the initial resonance isomer. Obviously, the resonance isomers correspond to the same species and both pathways exist from that species, but because how KinBot defines its molecules and reactions, it is necessary to uniquely identify each resonance isomer and to start the reaction search from each of them.

To generate resonance isomers of the stable species, KinBot starts with the connectivity as calculated based on Table 2, and initially assigns a bond order of 1 to all chemical bonds. Iteratively, for each atom in the molecule that has a lower valence compared to the expected valence (1 for H, 2 for O, 4 for C and 2, 4, or 6 for S), its neighbors are checked for their valences as well. Note that for S atoms KinBot tries all three valences, and can proceed with multiple different

feasible values if needed, just like in the case of resonantly stabilized structures. If a neighbor is also found with a valence lower than its expected value, the bond order between the atom and its neighbor is increased by one. If all neighbors have their expected valence, a radical electron is assigned to the atom. This procedure leads to one resonance isomer. To find all resonance isomers, the procedure needs to be repeated with a different atom order. In principle, all permutations of the unsaturated heavy atoms are necessary to find all resonance structures for sure, but doing this procedure is very time consuming for larger systems due to the $N!$ scaling. Therefore, we only search for $2N$ structures by enumerating all the atoms for which the initial valence is too low, and perform a search starting with each of these atoms, once in the forward direction (i.e. with increasing atom indices), and in the reverse direction (i.e. with decreasing atom indices). In our experience, this algorithm is sufficient. For highly fused rings with many unsaturated bonds, such as PAH's, KinBot is likely to miss some resonance isomers.

2.4. Conformational search

The optimization is followed by a conformational search of the reactant, if conformational search is requested by the user (`conformer_search` keyword). For acyclic structures we generate 3^n conformers by 120- and 240-degree rotations around each rotatable bonds with n being the number of rotors considered. KinBot considers all single bonds not in cycles and without symmetry, i.e., KinBot automatically ignores methyl, CH_2, *t*-butyl and other similar substructures from the conformational search. These substructures can be easily identified based on the aID's of the atoms in a rotating fragment.

For rings of size $N>3$, a measure for the flatness f of the ring is calculated by summing the absolute values of all the dihedrals along the ring and dividing that number by N. Since there are only $N–3$ independent dihedrals in a ring, only $N–3$ randomly chosen dihedral angles are sampled. We sample them on a $–f$, 0 and $+f$ grid building on the ideas of Kolossváry and Guida [38]. Thus we exhaustively enumerate all possible conformers of a cycle on this gird. If the structure contains rings and acyclic parts as well, first the generated ring conformers are relaxed in constrained optimization calculations by keeping the modified dihedrals fixed. Then every one of these partially relaxed ring conformers are combined with the 3^n conformers of the acyclic chains, and only then we relax the geometries fully.

We generate the initial structure for each trial conformer via direct Z-matrix modifications in KinBot. This is accomplished by automatically creating Z-matrices. Let us assume that there is an A–B–C–D chain of atoms in the structure that defines a dihedral. In this case B is the first element of the Z-matrix, C is the second with the C–B distance, A is the third with the A–B distance and angle A–B–C, and D is the fourth element with D–C distance, D–C–B angle and D–C–B–A dihedral. If atom X≠A and is on the B side, it is defined in terms of the X–A distance, X–A–B angle and X–A–B–C dihedral; the corresponding logic is applied on the C side as well. In KinBot we do a series of transformations between Cartesian and Z-matrix representations to modify all dihedrals. In each step a given A–B–C–D dihedral is selected and the Z-matrix is built. Then, we modify the dihedrals as needed, which is a simple addition to the appropriate dihedral values. Because of the construction of the Z-matrix, this results in a perfect rigid rotation. The structure is converted then back to Cartesians. We repeat the procedure with the next A–B–C–D dihedral until all considered dihedrals are accounted for. This way the initial trial geometries only differ in the dihedrals; all bond lengths and most angles are the same as in the original structure. Each of the trial conformers are then fully optimized at the specified L1 theory to identify the lowest energy conformer, which is subsequently optimized at the L2 theory. Frequencies are also calculated at L2. The lowest energy conformer is selected based on the ZPE-corrected energies.

For species with many hindered rotors the exhaustive conformational search might become prohibitively expensive (`max_dihed` keyword). For instance, 7 nonsymmetric (e.g., not methyl) rotors invoke 2187 single point calculations just for one species on the PES. For this reason, we allow the user to maximize the number of samples during such a calculation (`max_dihed` keyword), and use random sampling (`random_conf` keyword) on the dihedral grid of 120 degrees instead of a systematic search. Should lower energy conformers arise during a hindered rotor scan, KinBot replaces that one for the one found during the random sampling (see section 2.8 and `rotation_restart` keyword).

Note that changes in dihedrals sometimes result in unwanted rearrangements due to, e.g., clashes of atoms. This problem is especially true for saddle points, which can relatively easily move away from the saddle so that the subsequent optimization converges to a different saddle point or diverges. For this reason, for each of the conformers we compare the interatomic distances of all bonded atoms with the interatomic distances of the original structure. If any of the distances differs

by more than 5%, the structure is deemed chemically different from the original one and is discarded.

2.5. Internal rotors and other anharmonic modes

KinBot can also automatically map out properties key to approximate torsional anharmonicities if requested by the user (`rotor_scan` keyword). This step is always necessary for calculating the rate coefficients in order for the symmetry numbers to match the vibrational treatment of the molecule, but if the user is only interested in the connectivity and the barrier heights of the species on a PES, the mapping of the torsional anharmonicities can be turned off in KinBot.

The potential energy is calculated along each relevant dihedral in a separable 1-D rotor framework: by default, twelve points (`nrotation` keyword) along each dihedral angle are evaluated, with increments of 30 degrees. The structures for these constrained optimizations are generated using a Z-matrix transformation, similarly to the strategy in the conformational search, and are run independently. After filtering out outliers along the scan, i.e. points that either did not converge, or for which the connectivity is different from the original connectivity, the remaining points are fitted, containing 6 sine and 6 cosine terms:

$$V = \sum_{i=1}^{6} V_{ci}(1 - \cos(i \cdot \varphi)) + \sum_{i=1}^{6} V_{si} \sin(i \cdot \varphi) \qquad \text{Eq. 3}$$

where V_{ci} and V_{si} are the fitting parameters. The value of the potential for dihedral angles that were filtered out are calculated based on the fitted potential. The Master Equation codes accept but do not require a fitted function to describe the hindering potentials, however, filtering is still necessary to eliminate outliers which could yield an unphysical potential and a likely inaccurate sum or density of states. Note that if a lower energy conformer relative to the starting geometry is found during the dihedral scan, the scans restart from this new minimum, for a maximum of 3 times (this number can be changed by the `rotation_restart` keyword). Lower energy conformers might be found for molecules with many hindered rotors, when the user does not request an exhaustive conformational search.

To remove the harmonic frequencies corresponding to the torsional modes, the modes corresponding to the *k* internal rotations need to be projected out from the mass-weighted Hessian

in addition to the three translational and three external rotational modes. The entire $3N \times 3N$ Hessian matrix H is read from the quantum chemistry output files and is converted to a mass-weighted Hessian matrix F:

$$F = M^{-1/2} H M^{-1/2} \qquad \text{Eq. 4}$$

where M is a $3N \times 3N$ diagonal matrix containing the masses of the atoms: the first three diagonal elements of M contain the mass of the first atom, the second three elements contain the mass of the second atom, etc.

From the optimized Cartesian coordinates, a translational and rotational basis is created. The translational vectors are:

$$t_{1,ij} = \begin{cases} {m_i}^{1/2} \text{ if } j = x \\ 0 \text{ otherwise} \end{cases} \qquad \text{Eq. 5}$$

$$t_{2,ij} = \begin{cases} {m_i}^{1/2} \text{ if } j = y \\ 0 \text{ otherwise} \end{cases} \qquad \text{Eq. 6}$$

$$t_{3,ij} = \begin{cases} {m_i}^{1/2} \text{ if } j = z \\ 0 \text{ otherwise} \end{cases} \qquad \text{Eq. 7}$$

with i being the index of the i^{th} atom and $j = x, y, z$. These vectors are concatenated and normalized to yield the three t vectors, each of length $3N$.

To calculate the overall rotational vectors, the inertia tensor is needed, which is decomposed to obtain its eigenvalues X. P is then calculated as the dot product of the atom positions (relative to the center of mass) and the corresponding row of X to finally calculate the rotational vectors as:

$$r_{1,ij} = \frac{P_{y,i} X_{j,3} - P_{z,i} X_{j,2}}{{m_i}^{1/2}} \qquad \text{Eq. 8}$$

$$r_{2,ij} = \frac{P_{z,i} X_{j,1} - P_{x,i} X_{j,3}}{{m_i}^{1/2}} \qquad \text{Eq. 9}$$

$$r_{3,ij} = \frac{P_{x,i} X_{j,2} - P_{y,i} X_{j,1}}{{m_i}^{1/2}} \qquad \text{Eq. 10}$$

which also need to be normalized.

The last set of vectors to be constructed are the vectors corresponding to the internal rotation along the rotatable bonds. The coordinates of the atoms in the molecule are first weighted by the atom masses, and the molecule is partitioned in its two parts, divided at the rotatable bond. The rotational axis is obtained by the two atoms on both sides of the rotatable bond, in mass weighted Cartesian coordinates. For each atom in the molecule, a vector perpendicular to this axis is constructed with length of the distance from the atom to the rotational axis. The rotational vector for an atom is obtained by the dot product of the latter vector with a normalized vector along the rotational axis, multiplied by its sign. The sign of all atoms within one partition is identical, and opposite to the signs of all atoms in the other partition. Again, concatenating and normalizing the rotational vectors is necessary. This procedure creates k vectors for the k rotatable bonds in the species.

A new basis v for the Hessian is created using a Gram-Schmidt orthonormalization of which each vector is orthogonal to the three t vectors, the three r vectors and the k internal rotation vectors. The projected Hessian F' is calculated as:

$$F' = v^T F v \qquad \text{Eq. 11}$$

Finally, F' is decomposed in its eigenvectors and eigenvalues, and the eigenvectors are projected back to the original basis through the dot product of the eigenvectors with the basis v, and they are then weighted with the mass of the atoms and normalized. The $3N - 6 - k$ eigenvalues of F' are converted from their atomic units to wavenumbers, yielding the list of harmonic frequencies not corresponding to internal rotors.

2.6. Symmetry numbers

Pollak and Pechukas [39] have shown that it is necessary to incorporate symmetry numbers for accurate state counts and thus to account for equivalent reaction pathways. This procedure requires the determination of the total (internal and external) rotational and optical symmetry number of the reactant and of the transition state. As shown by Gilson and Irikura [40], for flexible molecules it is sufficient to consider the symmetry of the graph of the molecule, and it is unnecessary to explore all of the conformers and assign symmetry numbers to them individually. However, determining these graph-based symmetry numbers automatically in an unsupervised

manner is not a straightforward task. Previous automated approaches include the work of Vandewiele et al. [41], Blurock et al. [42], Walters and Yalkowsky [43], and Muller et al. [44]. While these previous works present interesting and working algorithms, they are fairly expensive to evaluate, and are somewhat complicated to code. Here we present our own algorithm for the symmetry numbers along with a test to compare its performance to previous work and literature values.

In KinBot the external rotational symmetry σ_{ext}, the internal rotational symmetry σ_{int} and the number of optical isomers n_{opt} are calculated with a set of rules, including atom-centered, bond-centered and ring-centered contributions. We always assume that in the kinetics calculations the vibrational partition functions or densities of states include hindered rotor representations for all rotatable single bonds, pseudorotations and umbrella modes, and the symmetry numbers are determined accordingly.

For the atom-centered contributions, KinBot first identifies the equivalent atoms based on the aIDs, and the atom-centered contributions found in Table 3 are established for each atom by KinBot. These contributions are multiplied by one another to obtain the total atom-centered contribution. These contributions are only considered if the atom is not part of a cycle.

Table 3: Atom-centered symmetry calculation rules defining the contribution to the external symmetry number σ_{ext} and the number of optical isomers n_{opt}.

Number of neighbors	Equivalent neighbors	σ_{ext}	n_{opt}
1	0	1	1
2	2	2	1
2	0	1	1
3	3	6	1
3	2	1	1
3	0	1	1
4	4	12	1
4	3	1	1
4	2 + 2	2	1
4	2	1	1
4	0	1	2

The bond-centered symmetry is perceived by enumerating all of the bonds and linear substructures (e.g., >C=C=C<) and verifying if the two pivot atoms defining the bond or the linear substructure are equivalent to each other and are not part of the same cycle. If so, the external symmetry number is increased by a factor of 2. Furthermore, if the pivot atoms of the bond or linear substructure are not just equivalent, but have an equal number of equivalent neighbors, on both sides the atoms are equivalent to one another, and don't possess any atom-centered symmetry, a factor to the external symmetry number is added amounting the number of neighbors. An example of the latter is ethylene, where the first factor of 2 originates from the fact that both carbon atoms have identical aID's and the second factor of 2 comes from the equivalence of the two hydrogens on one carbon atom to the two hydrogens on the other carbon atom. This leads to a total symmetry number of 4 for ethylene.

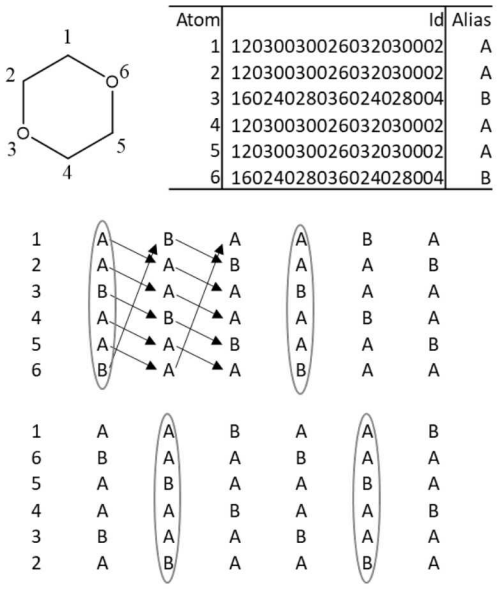


Figure 4: Illustration of the rotational symmetry perception for cyclic structures using 1,4-dioxane as example. There are two distinct aIDs in the ring, four equal ids for the carbon atoms and two equal ids for the oxygen atoms. They have been aliased 'A' and 'B' respectively for conciseness. First, the atoms are ordered corresponding to their bonding order and their aIDs are positioned in a vector. Next, the last element of the vector is placed at the start, moving all the elements down by one position ("rolling"). This is repeated five times for a six-membered ring with the current bond order, and another five times with a reverse bond order, giving rise to twelve vectors. These twelve vectors can be categorized in three sets of four equivalent vectors,

corresponding to a rotational symmetry number of four, which is in accord with the D_{2h} point group of the planar ring. This symmetry number is in accordance with the work of Gilson and Irikura [40].

To perceive ring-centered symmetry, each cycle of size *n* is considered individually within the chemical structure. We count how many configurations of the ring are identical if we rotate it around. E.g., for benzene, we want to perceive a symmetry of 12, while for 1,3,5-methyl-benzene we want to perceive a ring symmetry of 6. We start by constructing an *n*-long vector (id_1, id_2, ..., id_n) containing the aID's of the ring atoms in the order in which they are bonded to each other. This vector is "rolled" forward by taking the last element of the vector and placing it at the beginning of the vector. Then the procedure is repeated on the reverse of the original vector, (id_n, id_{n-1}, ..., id_1), yielding a total of $2n$-1 new vectors. The number of vectors identical to the initial vector plus 1 is the rotational symmetry of the ring, which is then factored into the external rotational symmetry number. Our procedure is illustrated for 1,4-dioxane in Figure 4. An additional rule is to include a factor of ½ if a ring atom has two equivalent neighbors on the ring, but two non-equivalent neighbors outside the ring. In the case an atom on the ring is part of an internal rotation, the symmetry of this ring is neglected, and in the case of fused rings, only the ring with the highest symmetry number is considered.

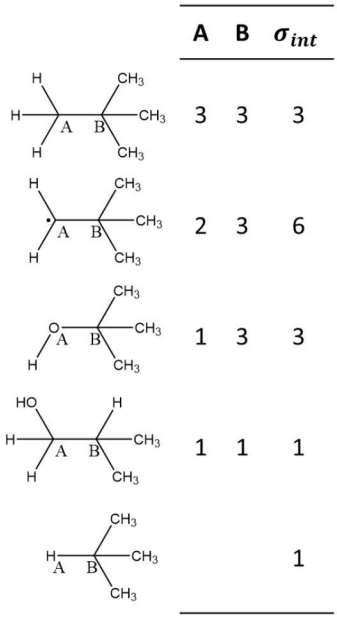


Figure 5: Illustration of the algorithm to calculate the internal rotational symmetry number along the A-B bond. On both sides of the bond, the individual symmetry is 1 if not all neighbors are equal (excluding A and B pivots

as neighbors) and is the number of (equivalent) neighbors otherwise. The internal symmetry number is 1 by definition for terminal A-B bonds as shown in the last row.

The internal rotational symmetry number is calculated for each non-terminal rotatable bond individually by taking the least common multiple of the rotational symmetry of the two bonded pivot atoms. We call a bond non-terminal if both pivot atoms have more than one neighbor. The rotational symmetry numbers on a given pivot atom is 1 if not all its non-pivot neighbors have the same aID, and is the number of (equivalent) neighbors otherwise. Additionally, if a pivot atom has external atom-centered symmetry, its rotational symmetry number is also set to 1. This is illustrated in Figure 5. The total internal rotational symmetry number for a structure is the product of the individual internal rotational symmetry numbers of the bonds.

In order to test the accuracy of our symmetry number assignment, we calculated the symmetry numbers for a subset of the 594 structures found in the supplementary information of the work of Vandewiele et al. [41] We selected the 363 C, H, O, and S containing species. We found that 356 of the calculated symmetry numbers are correct, which corresponds to a success rate of 98%. For 7 species an error was encountered. Five of these errors are the results of symmetric highly fused cyclic species. The sixth error is due to an incorrect perception of equivalent atom in large symmetric chains, which will be tackled by varying the aID depth based on the longest chain in the species. This depth is now set to 7 as default, leading to possible errors for highly symmetric species with a carbon chain length of 15 or longer. The last error results from resonance leading to a long linear feature in the molecule, not correctly perceived by KinBot. All calculated and expected symmetry numbers of the benchmark molecules are given in supplementary information. It is important to note that we adjusted some of the symmetry numbers from the benchmark of Vandewiele et al., because they were incorrect for the current considerations of symmetry. Those considerations include symmetry due to all real rotations, pseudo-rotations, and other anharmonic modes in a molecule, such as umbrella modes in pyramidal structures. The real rotations encompass all rotations along single non-cyclic bonds in the species, i.e. excluding any bond that has a partial double or triple character because of resonance. The pseudo-rotations are puckering modes in cyclic structures and breathing of cyclic structures such as cyclobutadiene. The symmetry perception in KinBot considers all of these modes non-harmonic.

Importantly, however, currently KinBot only characterizes the anharmonic potentials for the real rotations and not for the pseudo-rotations or the umbrella modes. To calculate accurate rate coefficients, the user has to remove the frequencies corresponding to all anharmonic modes from the harmonic oscillator approach and include the corresponding potentials for a correct treatment of all anharmonicities, or correct the symmetry number in case the anharmonic characters are neglected.

2.7. From PES to rate coefficients

After the L2 optimization and frequency calculations KinBot can be tasked to run Level 3 single point calculations to increase the accuracy of the electronic energy. The L3 calculations are typically very expensive, and are run in an offline mode. This is the final step before the results from all previously described steps done by KinBot are summarized in a format suitable for two master equation solvers, either the MESS [7] or the MESMER [9] code (selected by the `me_code` keyword). If L3 is not requested, the calculations proceed at the L2 level, which might be adequate for certain applications. These inputs contain the geometries of all stationary points, their frequencies, with the internal rotor motions projected out, the hindered rotor potentials, the zero-point corrected electronic energies at the prescribed level, the symmetry numbers, and for transition states the left and right barrier heights and the imaginary frequency for tunneling contributions via an Eckart barrier. The connectivity of the stationary points to each other is also encoded automatically. The connection to master equation calculations (`me` keyword to run them) is done through the appropriately formatted input files for the solvers, which only require minor manual interventions, for instance, the user needs to provide the energy transfer parameters and the temperature and pressure ranges for the calculations.

2.8. Visualization

KinBot produces a large amount of information during the exploration, e.g., the energies, geometries, frequencies, and interconnectivity of the wells and transition states. This information is saved into text files, which are organized for the machine to read, but these files are not easily digestible for the users because of the likely complexity of any given PES. To this end, we use the PESViewer software [45], which allows to automatically visualize the connectivity and the ZPE corrected electronic energies of the various stationary points along with the chemical structures of

the participating species in a conveniently organized way. KinBot automatically produces the input for PESViewer. All of the PES figures (see Figure 8 through Figure 13) were generated by PESViewer. KinBot also produces a graph representation of the interconnected wells and products using the python package called NetworkX.

2.9. Code dependencies

KinBot is an open-source code written in Python (compatible with Python 2.7 and Python 3) and licensed under the 3-clause BSD agreement. The KinBot code and its up-to-date manual are available through a GitHub repository (https://github.com/zadorlab/KinBot/wiki) and additional information and updates can be found on the https://kinbot.sandia.gov website. The manual describes all keywords and their default values. KinBot makes use of three open-source cheminformatics projects: the Atomic Simulation Environment (ASE) [46], which has been forked and slightly modified (https://github.com/zadorlab/ase), RDKit v. 2018.09.01 [47] (http://www.rdkit.org), and OpenBabel v. 2.4.1 [48] (http://openbabel.org). The ASE Python package is mainly used to bridge between KinBot and the electronic structure calculations: ASE contains input file writing and output file parsing features for many quantum chemistry packages. Gaussian 09 (G09RevD.01) [49] is currently the only thoroughly tested option used by KinBot, but ASE allows for KinBot to be agnostic about the quantum chemistry software, making it easy to include additional quantum chemistry software in the future (using the `qc` keyword). RDKit and OpenBabel are cheminformatics projects with robust graph representations of species, allowing species comparison, substructure searches, unambiguous species naming using line identifiers such as SMILES [50], 2D species visualization, etc.

Because of the large number of individual calculations required to explore a reactive PES, KinBot is programmed to communicate with job schedulers such as the Portable Batch System (PBS) and the SLURM Workload Manager, allowing parallel calculations across multiple nodes (toggle with the `queuing` keyword). In addition to this, the inherent parallelism of the quantum chemistry codes is used on a single node basis as well (set with the `ppn` keyword).

2.10. Installing and running KinBot

To install KinBot, the following software is necessary:

- Pybel v. 2.4.1, a Python distribution of OpenBabel found at https://openbabel.org/docs/dev/UseTheLibrary/PythonInstall.html)
- RDKit v. 2018.09.01 including Python bindings (https://www.rdkit.org/docs/Install.html
- A PBS or SLURM workload manager
- A quantum chemistry package. Currently only Gaussian 09 (G09RevD.01) is supported.
- The fork of ASE available on http://github.com/zadorlab/ase

Note that Pybel, RDKit and ASE need to be in the Python path. Optionally the following software is needed for full functionality:

- MESS (http://tcg.cse.anl.gov/papr/codes/mess.html) or MESMER (https://sourceforge.net/projects/mesmer/) to calculate rate coefficients
- PESViewer (https://github.com/rubenvdvijver/PESViewer) to visualize the PES

Once KinBot is downloaded from https://github.com/zadorlab/KinBot, the setup.py file needs to be run:

```
python setup.py build
python setup.py install
```

There are several test classes in the `tests/` directory, some of which are also supplied in the Supplementary Information. A `[name].json` input file with the keywords needs to be created in the directory where the calculations will be run. To run KinBot to explore reaction channels and other properties for one well, the command is

```
kinbot [name].json &
```

while to explore a full PES instead of only one well, run

```
pes [name].json &
```

For the `pes` script, separate directories are created for each well found, and the `kinbot` script is run for each of them separately. The `pes` script synthesizes the results in the end

Both modes, `kinbot` and `pes`, allow restarts. It is, however, important to keep certain parameters unchanged during restart, else there is no guarantee the code will run cleanly. Most importantly, the numbering of atoms should stay the same. Therefore, to restart a `pes` calculation from a different well, the Cartesian structure needs to be used instead of defining the SMILES in the input files, which scrambles the atom ordering. Only the following parameters can be changed at restart:

- `homolytic_scissions` (break all single non-cyclic bonds)
- `barrier_threshold` (reaction pathways with energy in kcal/mol higher than this threshold relative to the starting well are neglected)
- `simultaneous_kinbot` (maximum number of simultaneous kinbot runs during a full PES search)
- `high_level` (note that when using `pes`, the `barrier_threshold` that is propagated throughout the calculations will depend on whether high-level calculations are done or not)
- `conformer_search` (only switch this on if no `L2` calculations are done yet)
- `rotor_scan`

Additionally, the `pes` script can be run without calling `kinbot`.

```
pes [name].json no-kinbot &
```

This feature allows to quickly postprocess the already completed calculations, and create potential energy plots (see Section 2.8) that highlight for instance lowest energy pathways between species. The possible options are described in the manual.

The key output is the `summary_[chemid].out` file, which summarizes the reaction search (chemid is the id of the well the search started from). This file contains information about all successful and failed reaction searches, together with the barrier heights and the product identifiers. The `kinbot.log` files contains the logging information from the current and previous runs in a given directory. When `pes` is used, the `pes_summary.txt` file describe the status of the various `kinbot` jobs within the directory, while the `pes.log` file contains the logging information.

3. Results

3.1. Systematic reaction search for a range of species

KinBot enables the convenient systematic examination of a reaction type across several reactants to assess trends for instance. To demonstrate this, we have studied the simplest sigmatropic reaction, the [1,3]-sigmatropic H-migration reaction in closed-shell molecules, as illustrated in Figure 6.

The input for the simplest case, propene, is the following, when the antrafacial (more common) reaction is searched for:

```
{
    "title" : "propene",
    "smiles" : "CC=C",
    "mult" : 1,
    "reaction_search" : 1,
    "families" : ["intra_H_migration"]
    "high_level" : 1,
    "high_level_method" : "M062X",
    "high_level_basis" : "6-311++G**",
    "conformer_search" : 1,
    "rotor_scan" : 0,
    "barrier_threshold" : 200.0,
    "ppn" : 8
}
```

If the `families` keyword is set to `intra_H_migration_suprafacial`, a different kind of saddle point is searched for, as outlined here. This pericyclic reaction can proceed through two distinct transition state structures, one in which the hydrogen atom moves above and close to atom R_2, corresponding to a suprafacial reaction, and one in which the hydrogen atom moves close

to the plane of the three other atoms, which corresponds to an antrafacial reaction, hence we selected the corresponding two reaction families (`intra_H_migration` and `intra_H_migration_suprafacial`) to be investigated by KinBot. The transition state structures for these two cases are depicted in Figure 7 for propene.

According to the Woodward-Hoffmann rules [51], having 4 electrons in the electrocyclic reaction, the conrotatory reaction corresponding to the antrafacial transition state is allowed and the suprafacial reaction is forbidden. The Möbius-Hückel rules [52] lead to the same result since an even number of phase inversions is obtained for the suprafacial H-migration meaning it is thermally forbidden as opposed to the odd number of phase inversions in the antrafacial reaction.

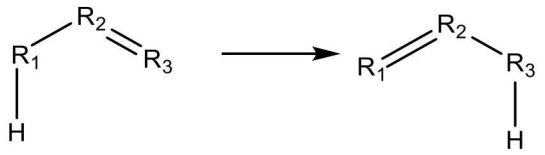


Figure 6: Illustration of a [1,3]-sigmatropic H-migration reaction in which a double bond moves one position.

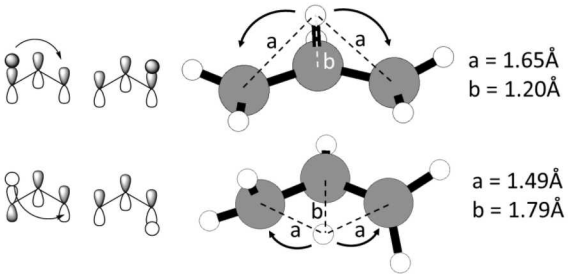


Figure 7: Transition state structures for suprafacial (top) and antrafacial (bottom) [1,3]-sigmatropic H-migration reactions in propene. The values for the C-H bond lengths of the hydrogen atom that changes in connectivity throughout the reactions are calculated at the M06-2X/6-311G++(d,p) level of theory.

Using KinBot, both suprafacial and antrafacial [1,3]-sigmatropic H-migration pathways have been located for 8 compounds shown in Figure 8. For three of the reactants, reactions to both the cis and the trans product are possible, and for 2-methyl-1-butene, the reaction can proceed with the original double bond in cis or in trans position of the terminal methyl group along the C4 chain in the transition state. Both lead to the same product but proceed through distinct transition states, hence the four reaction pathways.

All calculations have initially been done at the B3LYP/6-31G (L1) and have been refined at the M06-2X/6-311G++(d,p) level of theory (L2). For the reactions corresponding to keto-enol

tautomerization, the suprafacial reactions did not converge using the M06-2X functional and the B3LYP results are therefore given. For the identical reaction in formic acid, no suprafacial transition state was localized using either functionals. Additionally, the T1 diagnostics have been run at the CCSD(T)/cc-pVDZ level of theory resulting in values below 0.02 which confirmed the validity of single reference calculations of the transition states.

As expected, for vinyl alcohol, propene, 2-methyl-1-butene and propanal, the antrafacial allowed reaction has a lower reaction barrier height compared to the suprafacial, forbidden reaction pathways. The thermal rearrangement of these molecules thus preferentially proceeds through the allowed reactions as defined by both the Woodward-Hoffmann rules as well as the Möbius-Hückel rules. However, for 3-methyl butene, 4-pentenoic acid and 1-butene, the suprafacial reaction pathway to the cis-product is favored over the antrafacial one, and this pathway is also lower than the barriers to the thermodynamically more stable trans product.

The reason for the departure from the rules can be that the orbitals considered in the Möbius-Hückel rules only include the atoms that are part of the pericyclic ring structure; the ligands not part of the reacting center are not considered in the rules, nor does the cis/trans nature of the forming double bond is encoded in the Möbius-Hückel rules. Similarly, the Woodward-Hoffmann rules also cannot explain the differences observed here. The fact that in some cases the suprafacial reactions have a lower barrier must be due to electronic effects not captured by either of the rules.

This illustration shows that even within one reaction type it might be necessary to carry out a direct systematic search for transition states instead of relying on conclusions drawn from a small set of representatives of the reaction, because of potentially unexpected pathways even to expected products. KinBot is able to automatically and efficiently investigate such unexpected pathways and identify which one of them is kinetically more significant.

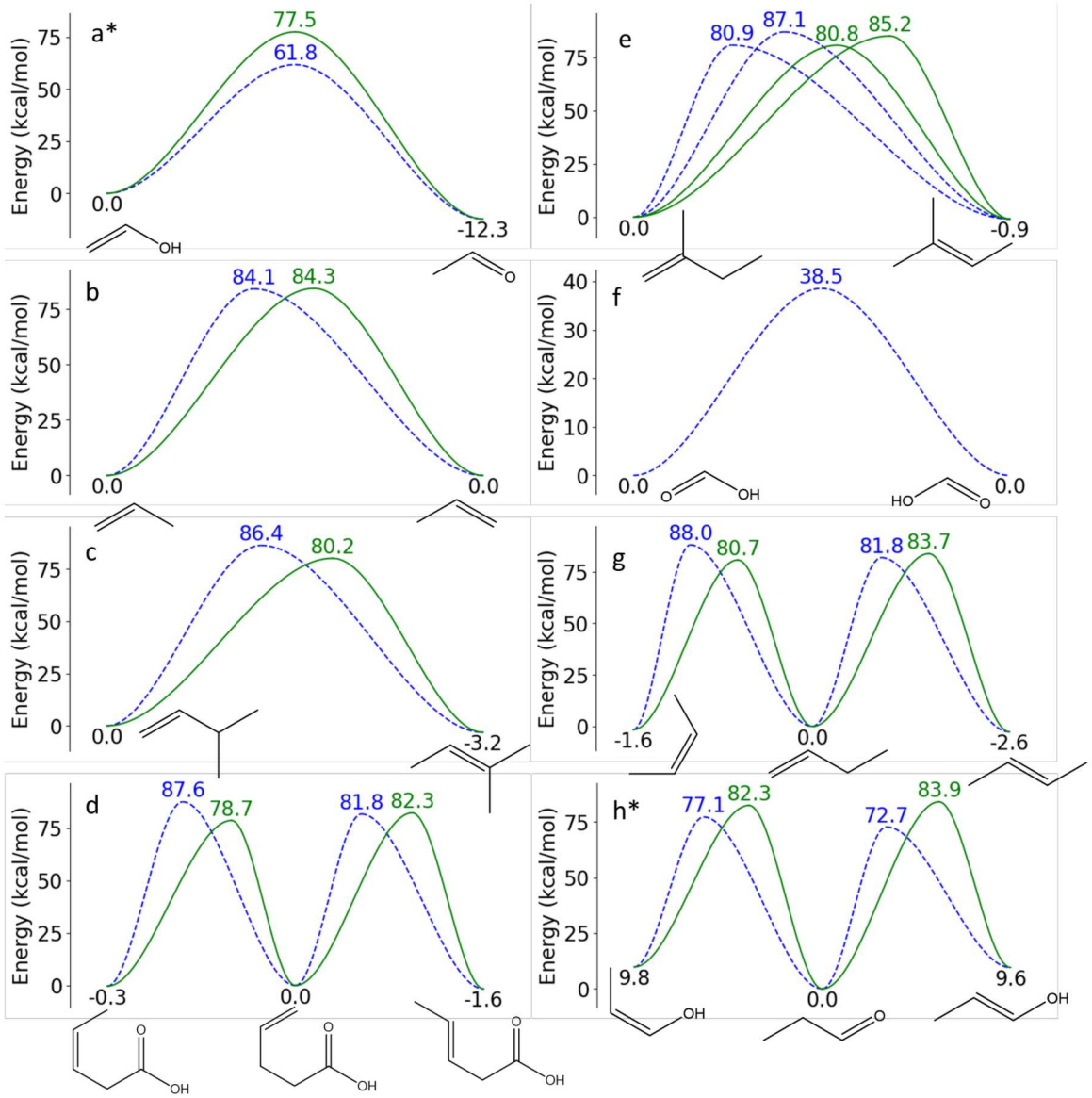


Figure 8: Electronic energies of the stationary points of [1,3]-sigmatropic H-migration reactions in (a) vinyl alcohol, (b) propene, (c) 3-methyl 1-butene, (d) 4-pentenoic acid, (e) 2-methyl 1-butene, (f) formic acid, (g) 1-butene, (h) propanal. The solid green lines correspond to the suprafacial reactions and the dashed blue lines correspond to the antrafacial reactions. For formic acid, no transition state was found for the suprafacial reaction. The energies are calculated at the M06-2X/6-311G++(d,p), except for vinyl alcohol and propanal, denoted by the asterisk, because no suprafacial reaction was found at this level of theory. Instead, the B3LYP/6-31G was used for these reactions.

3.2. Exploration of reaction channels starting from one well

The thermal decomposition of gamma-valerolactone (GVL), a potential biofuel, has been reported by two recent studies [53, 54]. Both studies were conducted without automation

procedures for transition state searches, and they both identified the same potential energy surface containing 9 saddle points and 2 barrierless steps with barrier heights ranging from 50.6 to 107.0 kcal mol^{-1} above GVL. GVL, 4-pentenoic acid and 3-pentenoic acid were used as input species in KinBot. The reaction searches and conformational scans were done at the L1 = B3LYP/6-31G level and the final optimizations and energy calculations were done at the L2 = B3LYP/6-311++G(d,p) level. KinBot also searched for the lowest energy conformers of the species. The input for the GVL is the following:

```
{
     "title" : "gamma_valerolactone",
     "smiles" : "O=C1CCC(C)O1",
     "mult" : 1,
     "method" : "b3lyp",
     "basis" : "6-31+g*",
     "qc" : "gauss",
     "conformer_search" : 0,
     "reaction_search" : 1,
     "rotor_scan" : 0,
     "high_level" : 0,
     "homolytic_scissions" : 0,
     "barrier_threshold" : 200.0,
     "ppn" : 8
}
```

The PES of GVL is shown in Figure 9; the well and barrier energies together with the Cartesian coordinates are given in the supplementary material. The lowest energy reaction pathway for GVL is a ring-opening reaction to 4PA, as was reported by De Bruycker et al. [53], with a barrier height of 62.5 kcal mol^{-1} calculated by KinBot. The next lowest pathway found by De Bruycker et al. leads to 2-oxiranone and propene for which KinBot calculated the barrier height at 73.5 kcal mol^{-1}. However, three pathways found by KinBot have lower barrier heights: 70.0 kcal mol^{-1} for the reaction to CO_2 and 1-butene, 70.2 kcal mol^{-1} for the reaction to CO_2 and methyl

cyclopropane, and 73.3 kcal mol^{-1} for the ring-opening reaction to 3PA. Additionally, for 4-pentenoic acid and 3-pentenoic acid, shown in Figure 10 and Figure 11 respectively, several new pathways were identified which have similar or lower barriers than the highest energy pathway reported in the literature. The geometries and energies of these reactions are also given in the supplementary material.

For 4PA and 3PA, several barrier heights are lower in energy than the corresponding products. This is an indication for the existence of van der Waals-wells along the reaction coordinate, which do influence the kinetics, especially at lower temperatures [28, 55, 56]. The barrier heights for these channels are high, making these reactions kinetically insignificant. We did not carry out further calculations to characterize the van der Waals-wells or the barrierless entrance into them. KinBot is currently not set up to study these complex regions of the PES.

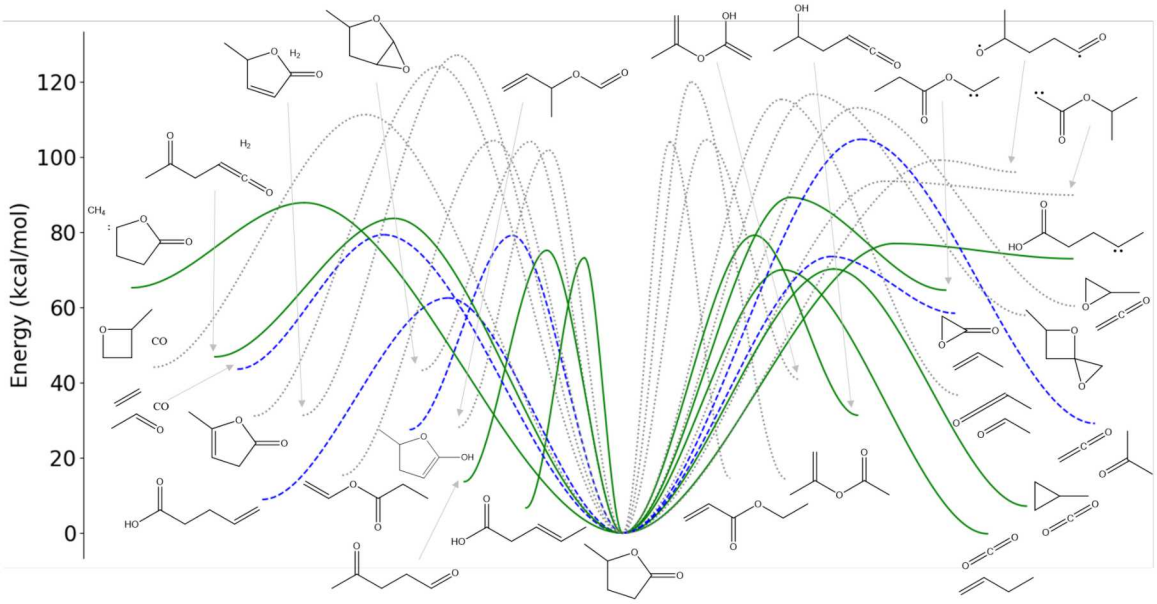


Figure 9: Potential energy surface of gamma-valerolactone as calculated by KinBot at the B3LYP/6-311++G(d,p) level of theory. Reactions marked with dashed blue lines are the ones reported in the literature[53, 54], reactions marked with solid green and the dotted gray lines are new reactions, not previously reported. The

green lines indicate a barrier height lower than the lowest hemolytic bond breaking pathway, which amounts to 88 kcal mol^{-1}. Exact energies and Cartesian coordinates are given in the supplementary material.

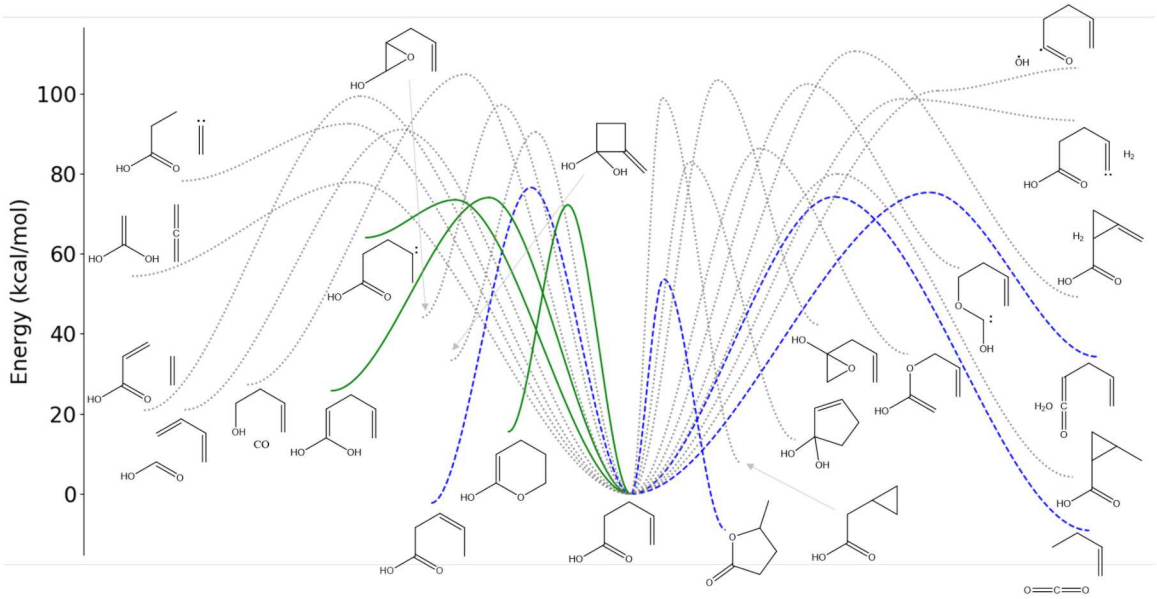


Figure 10: Potential energy surface of 4-pentenoic acid as calculated by KinBot at the B3LYP/6-311++G(d,p) level of theory. Reactions marked with dashed blue lines are the ones reported in the literature [53, 54], reactions marked with solid green and the dotted gray lines are new reactions, not previously reported. The green lines indicate a barrier height lower than the highest pathway previously reported, which amounts to 78 kcal mol^{-1}. Exact energies and Cartesian coordinates are given in the supplementary material.

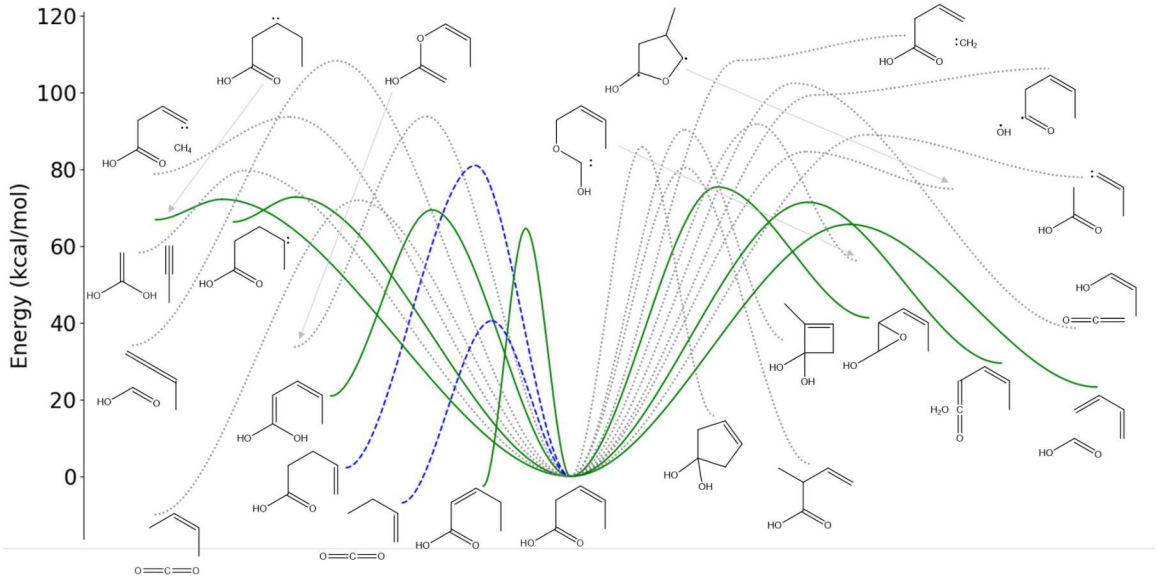


Figure 11: Potential energy surface of 3-pentenoic acid as calculated by KinBot at the B3LYP/6-311++G(d,p) level of theory. Reactions marked with dashed blue lines are the reactions reported in the literature [53, 54],

reactions marked with solid green and the dotted gray lines are new reactions, not previously reported. The green lines indicate a barrier height lower than the highest pathway previously reported, which amounts to 78 kcal mol^{-1}. Exact energies and Cartesian coordinates are given in the supplementary material.

3.3. Propene+OH

Similarly to the gamma-valerolactone reactions, the potential energy surface of the propene + OH reaction has been explored using KinBot and compared to the potential energy surface given in the literature [28, 57] to show the capabilities of KinBot for radical species. Since KinBot only considers unimolecular reactions, the initial adducts $CH_3CH•CH_2OH$ and $CH_3CH(OH)CH_2•$ were used as input species. The resulting PES's for the terminal and central additions are given in Figure 12 and Figure 13 respectively. The geometries and energies correspond to the B3LYP/6-311++G(d,p) level of theory. As mentioned above, KinBot does not consider van der Waals-wells as separate wells on the PES and, therefore, the entrance channels seem to have a barrier lower than the reactant energies. The entrance channels are complex and consist of a barrierless step into a van der Waals-well followed by an inner saddle point. Furthermore, the saddle point corresponding to the inner transition state was not found using the B3LYP functional and instead, second-order Møller-Plesset perturbation theory was used to locate them, which is the default method for radical β-scission reactions. A single point B3LYP calculation was used to obtain the energy values of -6.7 and -6.0 kcal mol^{-1} to approximate these barriers for demonstration purposes. The input for one of the radicals is the following, which contains now an almost complete set of parameters to enable going from a structure all the way to rate coefficients. The last 13 keywords are direct instructions to the master equation solver (MESS in this case) to set up the input. The `me` keyword automatically invokes the master equation solver, and the calculation ends when the rate coefficients have been produced.

```
{
     "title" : "propenoyl",
     "smiles" : "[CH2]CCO",
     "charge" : 0,
     "mult" : 2,
     "method" : "b3lyp",
```

```
    "basis" : "6-31+g*",
    "qc" : "gauss",
    "conformer_search" : 1,
    "reaction_search" : 1,
    "rotor_scan" : 1,
    "high_level" : 1,
    "high_level_method" : "B3LYP",
    "high_level_basis" : "6-311++G(d,p)",
    "homolytic_scissions" : 0,
    "barrier_threshold" : 200.0,
    "me" : 1,
    "me_code" : "mess",
    "ppn" : 8,
    "pes" : 1,
    "simultaneous_kinbot" : 5,


    "TemperatureList" : [500, 800, 1000, 1500, 2000],
    "PressureList" : [760],
    "EnergyStepOverTemperature" : 0.2,
    "ExcessEnergyOverTemperature" : 30,
    "ModelEnergyLimit" : 400,
    "CalculationMethod" : "direct",
    "ChemicalEigenvalueMax" : 0.2,
    "EnergyRelaxationFactor" : 200,
    "EnergyRelaxationPower" : 0.85,
    "EnergyRelaxationExponentCutoff" : 15,
    "Epsilons" : [7.08, 576.7],
    "Sigmas" : [2.576, 4.459],
    "Masses" : [4.0, 59.0],
}
```

All pathways reported previously [28] were found by KinBot, and KinBot also found pathways not previously reported. Most of them are kinetically insignificant high-energy pathways including cyclic formation reactions and water elimination reactions. KinBot also found one pathway which is low in energy (5.7 kcal mol^{-1} below the propene + OH reactants), which is an OH migration reaction from the terminal to the central carbon atom. This channel leads to more isomerization and dissociation channels, but we do not include further channels starting from this radical in this plot for the sake of clarity.

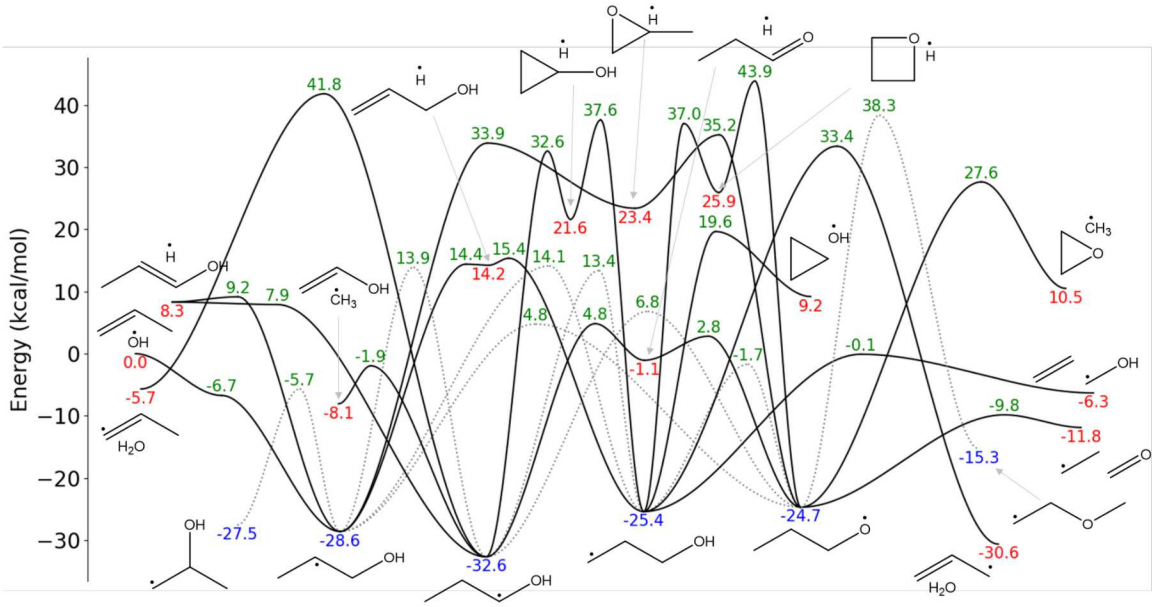


Figure 12: Potential energy surface of the OH addition on the terminal carbon atom in propene calculated at the B3LYP/6-311++G(d,p) level of theory. Dashed gray lines correspond to isomerization reactions, solid black

lines to bimolecular product formations. Blue values are the well energies, red values are the bimolecular product energies and green values are the transition state energies.

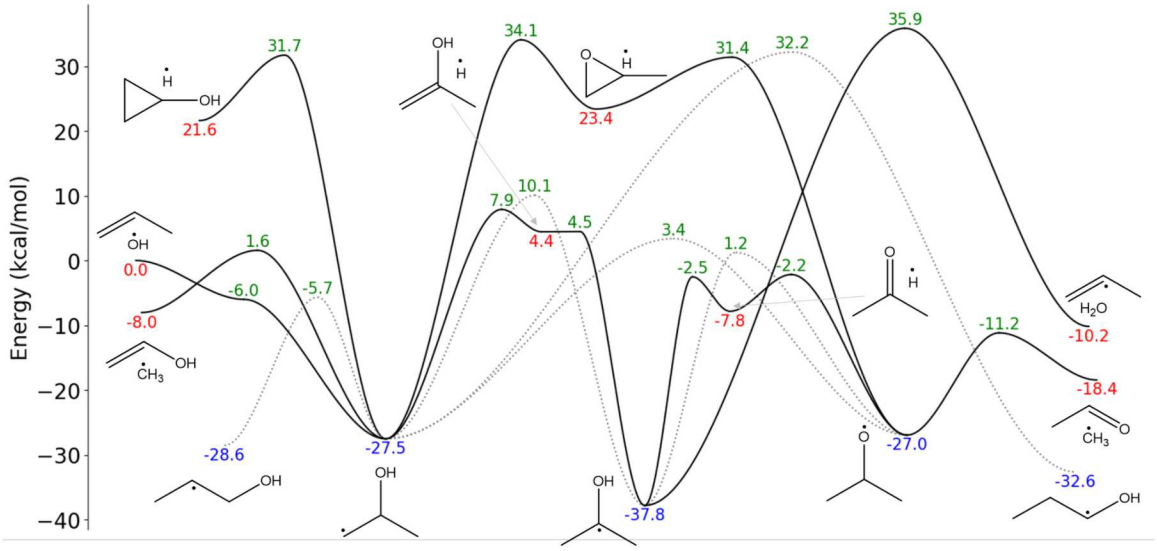


Figure 13: Potential energy surface of the OH addition on the central carbon atom in propene calculated at the B3LYP/6-311++G(d,p) level of theory. Dashed gray lines correspond to isomerization reactions, solid black lines to bimolecular product formations. Blue values are the well energies, red values are the bimolecular product energies and green values are the transition state energies.

3.4. Aramco Mech test up to C4

Our last demonstration case deals with the search for reactions in order to improve existing kinetic models. For this the Aramco Mech 2.0 [58] was used up to C4 species, which comes with the SMILES and InChI of the species. These were used to generate the initial structures for KinBot in a large batch calculation. For each species in the model, a KinBot reaction search was started to identify important pathways, with an input akin to the one shown for GVL (Section 3.2). The search is limited to unimolecular reactions that proceed through an energy barrier, i.e., no direct reactions, such as abstraction, and no barrierless reactions are included. The results of KinBot are compared to the reactions present in the Aramco Mech 2.0 model. For each species, the barrier heights of the reactions found (and calculated) by KinBot that are also in Aramco Mech (E_1) are compared to the barrier heights of the reactions found by KinBot that are not in the Aramco Mech (E_2). This comparison is plotted in Figure 14, where the species are ordered along the horizontal axis in increasing order of min(E_1). These min(E_1) values are connected with a continuous blue line, while all E_2 values are shown as dots. Points below the blue line can mean three things. First, it is possible

that reactions that are kinetically significant and have been found by KinBot are absent in Aramco Mech 2.0. Second, it is possible that the reactions on the blue line have a rate coefficient which is largely overestimated by Aramco Mech leading to the inclusion of the wrong reactions to the model. Third, although barrier heights of several reactions are lower compared to what is in Aramco Mech, it is possible that the final rate coefficient is kinetically insignificant due to either entropy effects along the reaction pathway or due to thermodynamically unstable products. In these cases, Aramco Mech is correct to leave the reactions out of the kinetic model.

Aramco Mech contains 359 C0-C4 species (only including H, C and O atoms), for 339 of those, KinBot found one or more reactions. The 20 species without reactions found are species which do not have any unimolecular reactions with barriers such as H, O_2, HO_2, CO, CH_4, C_2H_6, etc. For another 138 species, KinBot found reactions, but no unimolecular reactions are present in Aramco Mech, and finally, for 5 species, all reactions found by KinBot are also in Aramco Mech. This yields 196 species which are represented in Figure 14.

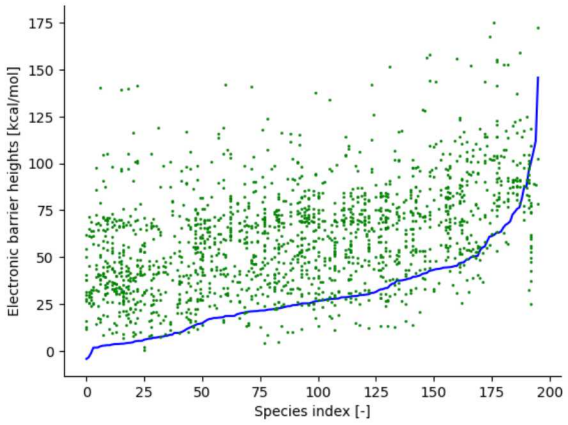


Figure 14: Comparison of the ZPE-exclusive B3LYP/6-31G barrier heights (vertical axis) for the C4- in the Aramco Mech 2.0 kinetic model (horizontal axis). The blue solid line is the lowest barrier height calculated by KinBot for all the reactions which are both found by KinBot and included in Aramco Mech (min(E_1)). The green dots are the barrier heights of the reactions that are only found by KinBot, E_2, and thus not present in Aramco Mech. Note that barrierless reactions and hydrogen abstractions are not included.

4. Conclusions

KinBot is a new code that automatically explores potential energy surfaces for molecules and radicals containing carbon, oxygen, sulfur, and hydrogen atoms relevant for combustion and

atmospheric chemistry. The reaction search is based on 18 reaction types, generalized from the reaction types in RMG. These families are defined by the necessary functional groups in reactants to start a reaction search and by the geometrical parameters for an initial guess of the transition state structure. This information is used to, for each potential reaction identified in the reactant, iteratively change the geometry of the reactant towards that initial guess, which is followed by a saddle point optimization on the potential energy surface. Intrinsic Reaction Coordinate calculations are used to verify the connectivity of the saddle point to the initial well and to identify the product. For all of the stationary points found on a potential energy surface, a systematic conformational search, hindered rotor calculations, high-level optimization, frequency calculations and symmetry perceptions are carried out. These results are written into input files for Master Equation calculations to obtain temperature and pressure dependent rate coefficients.

The characterization of a reactive PES for kinetics requires three general steps. First, reaction pathways need to be identified, second, the structures need to be brought close to the relevant saddle points, and third, the structures need to be fully optimized to first-order saddle points. In KinBot the first two steps are accomplished very efficiently through our templates aided with considerations for chemically identical atoms. KinBot does not directly cut down on the third local optimization step. The strength of KinBot is the ability to propose and process many, potentially thousands of pathways in parallel. Inefficiencies in KinBot are largely related to the proposal of reactions that do not exist in reality, to searches for saddle points that are too high in energy for a given chemical problem, or searches that eventually converge to the same saddle point. These inefficiencies, however, are to some extent inevitable in a comprehensive search. Notably, KinBot drastically decreases the time the user has to spend on exploring pathways.

We illustrated KinBot's capabilities via several examples by searching for a given reaction type in different reactants to compare the barrier heights, exploring all direct pathways starting from one well, exploring an entire multiwell potential energy surface, and searching for all of the reactions of the C0-C4 species in a literature model for combustion and compare the barrier heights of the newly found reactions to the ones in the model.

Features added to KinBot after the publication of this article will be posted to the kinbot.sandia.gov site and documented in the online manual at https://github.com/zadorlab/KinBot/wiki.

Supporting Information

1. A static version of the source code (KinBot.tar),
2. The manual (KinBot_Manual.pdf)
3. Geometries and energies of the stationary points on the potential energy surface of the sigmatropic reaction search (sigmatropic_H_shift.out)
4. Geometries and energies of the stationary points on the potential energy surface of the propene+OH central and terminal addition reaction (propene+oh central addition.out, propene+oh terminal addition.out)
5. Geometries and energies of the stationary points on the potential energy surface of gamma valerolactone, 4-pentenoic acid and 3-pentenoic acid (GVL energies and geometries.out, 4PA energies and geometries.out, 3PA energies and geometries.out)
6. Example runs including all input and output files for a one-well search for propanol radical, full PES search for the *n*-pentyl radical, a search for all homolytic scission in propanol, and the reaction searches for GVL (output.zip)
7. Results of symmetry calculations for a literature benchmark dataset (Symmetry_correct.pdf, Symmetry_wrong.pdf)

Acknowledgement

RVdV was supported by the Exascale Computing Project (ECP), Project Number 17-SC-20-SC, a collaborative effort of two U.S. Department of Energy (DOE) organizations, the Office of Science and the National Nuclear Security Administration (NNSA), responsible for the planning and preparation of a capable exascale ecosystem including software, applications, hardware, advanced system engineering, and early test bed platforms to support the nation's exascale computing imperative, and by the AITSTME project as part of the Predictive Theory and Modeling component of the Materials Genome Initiative. JZ was funded under DOE Basic Energy Sciences (BES), the Division of Chemical Sciences, Geosciences, and Biosciences. We thank Eric D. Hermes for the useful discussions. Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. DOE's NNSA under

contract DE-NA0003525. The views expressed in the article do not necessarily represent the views of the U.S. DOE or the United States Government.

References

[1] A. Fernandez-Ramos, J.A. Miller, S.J. Klippenstein, D.G. Truhlar, Modeling the kinetics of bimolecular reactions, Chem. Rev., 106 (2006) 4518-4584.

[2] S.J. Klippenstein, V.S. Pande, D.G. Truhlar, Chemical Kinetics and Mechanisms of Complex Systems: A Perspective on Recent Theoretical Advances, J. Am. Chem. Soc., 136 (2014) 528-546.

[3] S.J. Klippenstein, From Theoretical Reaction Dynamics to Chemical Modeling of Combustion, Proc. Combust. Inst., 36 (2016) 77-111.

[4] R. Van de Vijver, N.M. Vandewiele, P.L. Bhoorasingh, B.L. Slakman, F.S. Khanshan, H.-H. Carstensen, M.-F. Reyniers, G.B. Marin, R.H. West, K.M. Van Geem, Automatic Mechanism and Kinetic Model Generation for Gas- and Solution-Phase Processes: A Perspective on Best Practices, Recent Advances, and Future Challenges, Int. J. Chem. Kinet., 47 (2015) 199-231.

[5] C.W. Gao, J.W. Allen, W.H. Green, R.H. West, Reaction Mechanism Generator: Automatic construction of chemical kinetic mechanisms, Computer Physics Communications, 203 (2016) 212-225.

[6] N.M. Vandewiele, K.M. Van Geem, M.-F. Reyniers, G.B. Marin, Genesys: Kinetic model construction using chemo-informatics, Chemical Engineering Journal, 207-208 (2012) 526-538.

[7] Y. Georgievskii, S.J. Klippenstein, MESS.2016.3.23, 2016.

[8] Y. Georgievskii, J.A. Miller, M.P. Burke, S.J. Klippenstein, Reformulation and solution of the master equation for multiple-well chemical reactions, J. Phys. Chem. A, 117 (2013) 12146-12154.

[9] D.R. Glowacki, C.-H. Liang, C. Morley, M.J. Pilling, S.H. Robertson, MESMER: An Open-Source Master Equation Solver for Multi-Energy Well Reactions, J. Phys. Chem. A, 116 (2012) 9545-9560.

[10] S.H. Robertson, D.R. Glowacki, C.-H. Liang, C. Morley, M.J. Pilling, MESMER (Master Equation Solver for Multi-Energy Well Reactions), an object oriented C++ program for carrying out ME calculations and eigenvalue-eigenvector analysis on arbitrary multiple well systems, University of Leeds, 2008.

[11] A.L. Dewyer, A.J. Argüelles, P.M. Zimmerman, Methods for exploring reaction space in molecular systems, Wiley Interdiscip. Rev. Comput. Mol. Sci., 8 (2018) e1354.

[12] A.L. Dewyer, P.M. Zimmerman, Finding reaction mechanisms, intuitive or otherwise, Org Biomol Chem, 15 (2017) 501-504.
[13] G.N. Simm, A.C. Vaucher, M. Reiher, Exploration of Reaction Pathways and Chemical Transformation Networks, J. Phys. Chem. A, (2018).
[14] P.M. Zimmerman, Automated discovery of chemically reasonable elementary reaction steps, J. Comput. Chem., 34 (2013) 1385-1392.
[15] M. Jafari, P.M. Zimmerman, Uncovering reaction sequences on surfaces through graphical methods, Phys. Chem. Chem. Phys., 20 (2018) 7721-7729.
[16] P.M. Zimmerman, Single-Ended Transition State Finding with the Growing String Method, J. Comput. Chem., 36 (2015) 601-611.
[17] Y.V. Suleimanov, W.H. Green, Automated Discovery of Elementary Chemical Reaction Steps Using Freezing String and Berny Optimization Methods, J. Chem. Theory Comp., 11 (2015) 4248-4259.
[18] S. Maeda, T. Taketsugu, K. Morokuma, Exploring transition state structures for intramolecular pathways by the artificial force induced reaction method, J. Comput. Chem., 35 (2014) 166-173.
[19] S. Maeda, Y. Harabuchi, M. Takagi, T. Taketsugu, K. Morokuma, Artificial Force Induced Reaction (AFIR) Method for Exploring Quantum Chemical Potential Energy Surfaces, Chem. Rec., 16 (2016) 2232-2248.
[20] P.L. Bhoorasingh, R.H. West, Transition state geometry prediction using molecular group contributions, Phys. Chem. Chem. Phys., 17 (2015) 32173-32182.
[21] P.L. Bhoorasingh, B.L. Slakman, F. Seyedzadeh Khanshan, J.Y. Cain, R.H. West, Automated Transition State Theory Calculations for High-Throughput Kinetics, J. Phys. Chem. A, 121 (2017) 6896-6904.
[22] R. Van de Vijver, K.M. Van Geem, G.B. Marin, On-the-fly ab initio calculations toward accurate rate coefficients, Proc. Combust. Inst., (2018).
[23] Y. Kim, J.W. Kim, Z. Kim, W.Y. Kim, Efficient prediction of reaction paths through molecular graph and reaction network analysis, Chem. Sci., 9 (2018) 825-835.
[24] A. Rodríguez, R. Rodríguez-Fernández, S. A. Vázquez, G. L. Barnes, J. J. P. Stewart, E. Martínez-Núñez, tsscds2018: A code for automated discovery of chemical reaction mechanisms and solving the kinetics, J. Comput. Chem., 39 (2018) 1922-1930.

[25] C. Cavalloti, M. Pelucchi, Y. Georgievskii, S.J. Klippenstein, From electronic structure calculations to temperature and pressure dependent rate constants: a new computational environment, 25th International Conference on Chemical Reaction EngineeringFlorence, Italy, 2018.
[26] M. Yang, L. Yang, G. Wang, Y. Zhou, D. Xie, S. Li, Combined Molecular Dynamics and Coordinate Driving Method for Automatic Reaction Pathway Search of Reactions in Solution, J. Chem. Theory Comput., 14 (2018) 5787-5796.
[27] C.A. Grambow, A. Jamal, Y.-P. Li, W.H. Green, J. Zádor, Y.V. Suleimanov, Unimolecular Reaction Pathways of a γ-Ketohydroperoxide from Combined Application of Automated Reaction Discovery Methods, Journal of the American Chemical Society, 140 (2018) 1035-1048.
[28] J. Zádor, A.W. Jasper, J.A. Miller, The reaction between propene and hydroxyl, Phys. Chem. Chem. Phys., 11 (2009) 11040-11053.
[29] Y. Li, C.-W. Zhou, K.P. Somers, K. Zhang, H.J. Curran, The oxidation of 2-butene: A high pressure ignition delay, kinetic modeling study and reactivity comparison with isobutene and 1-butene, Proc. Combust. Inst., 36 (2017) 403-411.
[30] J. Zádor, H.N. Najm, Automated exploration of the mechanism of elementary reactions, Sandia Report, Sandia National Laboratories, 2012.
[31] J. Zádor, J.A. Miller, Adventures on the C_3H_5O potential energy surface: OH + propyne, OH + allene and related reactions, Proc. Combust. Inst., 35 (2015) 181-188.
[32] B. Rotavera, J. Zádor, O. Welz, L. Sheps, A.M. Scheer, J.D. Savee, M.A. Ali, T.S. Lee, B.A. Simmons, D.L. Osborn, A. Violi, C.A. Taatjes, Photoionization Mass Spectrometric Measurements of Initial Reaction Pathways in Low-Temperature Oxidation of 2,5-dimethylhexane J. Phys. Chem. A, 118 (2014) 10188-10200.
[33] I.O. Antonov, J. Zádor, B. Rotavera, E. Papajak, D.L. Osborn, C.A. Taatjes, L. Sheps, Pressure-Dependent Competition among Reaction Pathways from First- and Second-O_2 Additions in the Low-Temperature Oxidation of Tetrahydrofuran, J. Phys. Chem. A, 119 (2016) 7742–7752.
[34] I.O. Antonov, J. Kwok, J. Zador, L. Sheps, A Combined Experimental and Theoretical Study of the Reaction OH+2-Butene in the 400-800 K Temperature Range, J. Phys. Chem. A, 119 (2015) 7742-7752.
[35] R. Van de Vijver, K.M. Van Geem, G.B. Marin, J. Zádor, Decomposition and isomerization of 1-pentanol radicals and the pyrolysis of 1-pentanol, Combust. Flame, 196 (2017) 500-514.

[36] H.-J. Werner, P.J. Knowles, G. Knizia, F.R. Manby, M. Schütz, Molpro: a general-purpose quantum chemistry program package, Wiley Interdiscip. Rev. Comput. Mol. Sci., 2 (2012) 242-253.
[37] R. Fletcher, Practical methods of optimization, 2 ed., John Wiley & Sons, New York, 1987.
[38] I. Kolossváry, W.C. Guida, Torsional flexing: Conformational searching of cyclic molecules in biased internal coordinate space, J. Comput. Chem., 14 (1993) 691-698.
[39] E. Pollak, P. Pechukas, Symmetry Numbers, Not Statistical Factors, Should Be Used in Absolute Rate Theory and in Brønsted Relations, J. Am. Chem. Soc., 100 (1978) 2984-2991.
[40] M.K. Gilson, K.K. Irikura, Symmetry Numbers for Rigid, Flexible, and Fluxional Molecules: Theory and Applications, The Journal of Physical Chemistry B, 114 (2010) 16304-16317.
[41] N.M. Vandewiele, R. Van de Vijver, K.M. Van Geem, M.-F. Reyniers, G.B. Marin, Symmetry calculation for molecules and transition states, J. Comput. Chem., 36 (2015) 181-192.
[42] E.S. Blurock, V. Warth, X. Grandmougin, R. Bounaceur, P.A. Glaude, F. Battin-Leclerc, JTHERGAS: Thermodynamic estimation from 2D graphical representations of molecules, Energy, 43 (2012) 161-171.
[43] W.P. Walters, S.H. Yalkowsky, ESCHERA Computer Program for the Determination of External Rotational Symmetry Numbers from Molecular Topology, J. Chem. Inform. Comput. Sci., 36 (1996) 1015-1017.
[44] C. Muller, G. Scacchi, G.M. Côme, A topological method for determining the external symmetry number of molecules, Comput. Chem., 15 (1991) 17-27.
[45] R. Van de Vijver, PESViewer, https://github.com/rubenvdvijver/PESViewer, 2018.
[46] A. Hjorth Larsen, J. Jorgen Mortensen, J. Blomqvist, I.E. Castelli, R. Christensen, M. Dulak, J. Friis, M.N. Groves, B. Hammer, C. Hargus, E.D. Hermes, P.C. Jennings, P. Bjerre Jensen, J. Kermode, J.R. Kitchin, E. Leonhard Kolsbjerg, J. Kubal, K. Kaasbjerg, S. Lysgaard, J. Bergmann Maronsson, T. Maxson, T. Olsen, L. Pastewka, A. Peterson, C. Rostgaard, J. Schiotz, O. Schutt, M. Strange, K.S. Thygesen, T. Vegge, L. Vilhelmsen, M. Walter, Z. Zeng, K.W. Jacobsen, The atomic simulation environment-a Python library for working with atoms, J. Phys.: Condens. Matter, 29 (2017) 273002.
[47] RDKit: Open-source cheminformatics, http://www.rdkit.org, 2018.
[48] N.M. O'Boyle, M. Banck, C.A. James, C. Morley, T. Vandermeersch, G.R. Hutchison, Open Babel: An open chemical toolbox, J. Cheminformatics, 3 (2011).

[49] M.J. Frisch, G.W. Trucks, H.B. Schlegel, G.E. Scuseria, M.A. Robb, J.R. Cheeseman, G. Scalmani, V. Barone, B. Mennucci, G.A. Petersson, H. Nakatsuji, M. Caricato, X. Li, H.P. Hratchian, A.F. Izmaylov, J. Bloino, G. Zheng, J.L. Sonnenberg, M. Hada, M. Ehara, K. Toyota, R. Fukuda, J. Hasegawa, M. Ishida, T. Nakajima, Y. Honda, O. Kitao, H. Nakai, T. Vreven, J. Montgomery, J. A., J.E. Peralta, F. Ogliaro, M. Bearpark, J.J. Heyd, E. Brothers, K.N. Kudin, V.N. Staroverov, R. Kobayashi, J. Normand, K. Raghavachari, A. Rendell, J.C. Burant, S.S. Iyengar, J. Tomasi, M. Cossi, N. Rega, N.J. Millam, M. Klene, J.E. Knox, J.B. Cross, V. Bakken, C. Adamo, J. Jaramillo, R. Gomperts, R.E. Stratmann, O. Yazyev, A.J. Austin, R. Cammi, C. Pomelli, J.W. Ochterski, R.L. Martin, K. Morokuma, V.G. Zakrzewski, G.A. Voth, P. Salvador, J.J. Dannenberg, S. Dapprich, A.D. Daniels, Ö. Farkas, J.B. Foresman, J.V. Ortiz, J. Cioslowski, D.J. Fox, Gaussian 09, Revision D.01, Gaussian, Inc., Wallingford CT, 2009.

[50] D. Weininger, SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules, J. Chem. Inform. Comput. Sci., 28 (1988) 31-36.

[51] R.B. Woodward, R. Hoffmann, The Conservation of Orbital Symmetry, Angew. Chem. Int. Ed., 8 (1969) 781-853.

[52] H.E. Zimmerman, Möbius-Hückel concept in organic chemistry. Application of organic molecules and reactions, Acc. Chem. Res., 4 (1971) 272-280.

[53] R. De Bruycker, H.-H. Carstensen, J.M. Simmie, K.M. Van Geem, G.B. Marin, Experimental and computational study of the initial decomposition of gamma-valerolactone, Proc. Combust. Inst., 35 (2015) 515-523.

[54] L. Ye, W. Li, F. Qi, Pressure-dependent branching in initial decomposition of gamma-valerolactone: a quantum chemical/RRKM study, RSC Adv., 8 (2018) 12975-12983.

[55] E.E. Greenwald, S.W. North, Y. Georgievskii, S.J. Klippenstein, A two transition state model for radical-molecule reactions: A case study of the addition of OH to C_2H_4, J. Phys. Chem. A, 109 (2005) 6031-6044.

[56] W.H. Miller, Unified statistical model for "complex" and "direct" reaction mechanisms, J. Chem. Phys., 65 (1976) 2216-2223.

[57] C. Kappler, J. Zádor, O. Welz, X. Fernandez Ravi, M. Olzmann, A. Taatjes Craig, Competing Channels in the Propene + OH Reaction: Experiment and Validated Modeling over a Broad Temperature and Pressure Range, Z. Phys. Chem., 225 (2011) 1271-1291.

[58] AramcoMech 2.0, 2018.