TAMkin: A Versatile Package for Vibrational Analysis and Chemical Kinetics

An Ghysels,* Toon Verstraelen, Karen Hemelsoet, Michel Waroquier, and Veronique Van Speybroeck*

Center for Molecular Modeling, QCMM Alliance Ghent-Brussels, Ghent University, Technologiepark 903, 9052 Zwijnaarde, Belgium

Received March 12, 2010

TAMkin is a program for the calculation and analysis of normal modes, thermochemical properties and chemical reaction rates. At present, the output from the frequently applied software programs ADF, CHARMM, CPMD, CP2K, Gaussian, Q-Chem, and VASP can be analyzed. The normal-mode analysis can be performed using a broad variety of advanced models, including the standard full Hessian, the Mobile Block Hessian, the Partial Hessian Vibrational approach, the Vibrational Subsystem Analysis with or without mass matrix correction, the Elastic Network Model, and other combinations. TAMkin is readily extensible because of its modular structure. Chemical kinetics of unimolecular and bimolecular reactions can be analyzed in a straightforward way using conventional transition state theory, including tunneling corrections and internal rotor refinements. A sensitivity analysis can also be performed, providing important insight into the theoretical error margins on the kinetic parameters. Two extensive examples demonstrate the capabilities of TAMkin: the conformational change of the biological system adenylate kinase is studied, as well as the reaction kinetics of the addition of ethene to the ethyl radical. The important feature of batch processing large amounts of data is highlighted by performing an extended level of theory study, which TAMkin can automate significantly.

I. INTRODUCTION

The calculation of thermodynamical properties based on normal-mode analysis (NMA) is a standard procedure in molecular simulations. The field of applications ranges from relatively small organic compounds, where the functional groups are responsible for characteristic peaks in the vibrational spectra,[1] to biomolecular systems, where conformational changes are assigned to the lowest frequency motions.[2] The molecular frequencies also serve as an input for the molecular partition function, which is the bridge to many macroscopic properties such as entropies. Predicted spectra, relative entropy and free energy differences, and reaction rates can then be compared with experimental results as a validation of the theoretical models.

In the last three decades, an increasing amount of simulation packages has become available, offering a variety of descriptions of the potential energy surface, that is, a pure quantum mechanical (QM), pure force field (MM), or hybrid QM/MM description. Examples include ADF,[3] CHARMM,[4] CPMD,[5] CP2K,[6] Gaussian03,[7] Q-Chem,[8] and VASP.[9] For geometry optimization purposes, these packages can calculate the first derivatives with respect to atomic displacements. Most packages also offer the possibility to calculate the matrix of second derivatives, or Hessian, either analytically or numerically by differentiation of the gradient. When the user requests a normal-mode analysis, the Hessian is mass-weighted and diagonalized, yielding the eigenfrequencies and the normal modes. Several packages add a thermochemical section to the output, specifying the translational, rotational and vibrational contribution to thermodynamical properties, such as the entropy and the free energy, at a specific temperature.

In addition to the basic functionalities available in the above-mentioned packages, researchers are in need of more advanced analysis tools. Theoretical investigations have made a large set of NMA methodologies available in addition to the plain diagonalization of the full Hessian.[2,10−16] A second aspect is that NMA differs slightly between a simulation in the gas phase and modes derived from a calculation with periodic boundary conditions, especially for the treatment of the zero frequencies. Another practical issue is that the thermodynamical properties often need to be recalculated for a series of different temperatures, for which repetition of the computationally expensive calculation and diagonalization of the Hessian should be avoided. Finally, for the estimation of reaction rate constants with Transition State Theory (TST),[17−20] the quantities resulting from different frequency calculation output files should be carefully combined. Although an experienced researcher can come up with self-programmed scripts to implement these specific analyses, we experienced the need for a coherent software program that performs the analysis in a flexible but organized way. Also lacking is a practical implementation of refinements because of the treatment of internal rotations when constructing partition functions. For instance, programs like pyVib[21] and PyVib2[22] provide a handy graphical interface for visualizing vibrational motion and vibrational spectra, but no link is made to chemical kinetics calculations. The POLYRATE program[23] developed at the University of Minnesota is specialized in the calculation of chemical reaction rates based on Variational Transition State Theory

* To whom correspondence should be addressed. E-mail: An.Ghysels@UGent.be (A.G.); Veronique.VanSpeybroeck@UGent.be (V.V.S.).

10.1021/ci100099g
Published on Web 08/26/2010

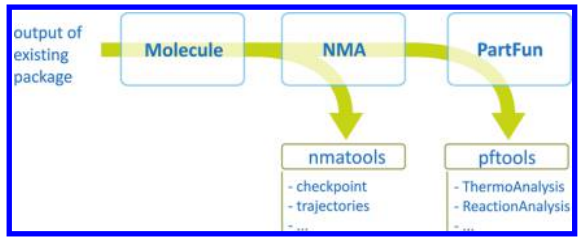

Figure 1. Modular philosophy of TAMkin. TAMkin's three central objects are the `Molecule`, the `NMA`, and the `PartFun` object.

(VTST). The version of 2008 can perform VTST calculations on reactions with both tight and loose transition states, in gas phase, in solid-phase, and at the gas−solid surface. The use of curvilinear coordinates along a minimum energy path or variationally optimized reaction path is implemented, as well as multidimensional tunneling corrections. TAMkin is limited to the Transition State Theory but has more flexibility in accessing the contributions to the partition function and has special approximate NMA models implemented. Our new program TAMkin facilitates the task of deriving normal modes and kinetic parameters from multiple molecular structures, calculated with diverse programs, with a broad choice in NMA models and refined partition function models.

TAMkin is furthermore a very useful tool to perform theoretical assessment studies, as will be illustrated in the present article. With the available plethora of contemporary techniques, these studies can become a tedious and time-consuming task. Because of the batch processing functionalities of TAMkin, this process can be automated significantly.

In the next section, we point out the workflow of the program, written in Python,[24] and its main characteristics. Once the second derivative information from the simulation is passed to the TAMkin program, the user can specify the desired NMA model. In the thermodynamical analysis, the user can choose which contributions are included in the partition function. Next, partition functions of various frequency calculations can be combined to derive kinetic parameters for the study of reaction kinetics. Several plotting functions are available, as well as functions to write mode trajectory files for the popular visualization programs VMD[25] and Molden.[26] The modular philosophy of TAMkin makes the program easily extensible for future NMA models. TAMkin's modules are based on three central objects: the molecule object (`molecule`), the NMA object (`NMA`), and the partition function object (`PartFun`), as illustrated in Figure 1. A checkpointing routine is available for the `NMA` object, allowing variation of the temperature and other parameters in the partition function without requiring recalculation of the frequencies. Other strong points of TAMkin are its consistent application programming interface (API), test routines essential for quality assurance and its potential to automate all computations with user-defined protocols.

In the Examples sections, two test cases illustrate the functionalities of the program. A first example is the adenylate kinase molecule, a protein of which the low frequency modes are responsible for the transition between the open and closed state.[27] TAMkin is used to study the configurational change between these two states by calculating the overlap with the normal modes and estimating the free energy and entropy differences. Several approximate NMA models are shown to produce qualitatively similar results. A second example highlights TAMkin's efficiency for performing an extended level-of-theory assessment. The chemical kinetics of the ethyl radical to ethene addition reaction is studied with Transition State Theory.[17−20] Model refinements due to internal hindered rotors and tunneling are included. A sensitivity tool is also discussed, assessing the effect of energetic and geometric variations on resulting rate coefficients.

In conclusion, TAMkin is an efficient versatile open-source program offering a wide variety of functionalities for NMA and facilitating the theoretical analysis of reaction kinetics.

Table 1. Overview of TAMkin Modules

module	functionality
molmod.periodic	database containing the periodic table of Mendeljev.
molmod.units	conversion factors between SI, popular units and the atomic units.
molmod.constants	some useful physicochemical constants in atomic units.
molmod.molecules	definition of a generic `Molecule` object.
unittest	tests for debugging.
tamkin.data	datatypes that store raw output from other simulation software; contains an extension of the `Molecule` object with Tamkin-specific features.
tamkin.geom	some convenient functions, for example, to determine whether a molecule is linear.
tamkin.timer	timer class to keep track of system and CPU times.
tamkin.io	reading functions for ADF, CHARMM, CP2K, CPMD, Gaussian03, Q-Chem, VASP; reading functions to specify the block choice for NMA methods; also dumping and loading to a standarized checkpoint file format.
tamkin.nma	performing the NMA, leading to an `NMA` object.
tamkin.nmatools	further analysis of the `NMA` object, for example, overlap of modes of different `NMA` objects.
tamkin.partf	definition of `PartFun` object, where it is possible to specify which partition function contributions are taken into account.
tamkin.pftools	further analysis of `PartFun` objects: analysis of equilibrium constant, reaction rate constant and kinetic parameters of a reaction.
tamkin.rotor	special treatment of internal rotors.
tamkin.tunneling	tunneling correction to the reaction kinetics.

II. IMPLEMENTATION AND FEATURES

TAMkin is developed as a modular program. The program makes use of three main Python classes: `Molecule`, `NMA`, and `PartFun`. The encapsulating programming style has been consciously followed, meaning that routines are grouped into classes and modules with their specific private or public settings. As a consequence, the user is protected from changing the program's essential work flow when adding a personal extension to the code. Figure 1 depicts the flow of data through the program, Table 1 lists the modules with their purpose, and Table 2 gives the main attributes of the three central classes.

The thermodynamical analysis is summarized in the following three basic commands:

- Load a molecule, for example, `mol = load_molecule_qchem(filename)` for reading from an output file generated with Q-Chem.

Table 2. Overview of the Main Attributes of the `Molecule`, `NMA`, and `PartFun` Class[a]

`Molecule` attributes	
Hessian	Cartesian Hessian
gradient	Cartesian gradient
energy	energy
mass	total mass
masses	atomic masses
numbers	atomic numbers
coordinates	Cartesian coordinates
inertia_tensor	inertia tensor
multiplicity	multiplicity 2S+1
symmetry_number	symmetry number
periodic	(logical) indicates whether periodic boundary conditions were used
`NMA` attributes	
freqs	frequencies
modes	mass-weighted normal modes
num_zeros	number of zero frequencies
zeros	indices of zero frequencies
`NMA` referenced attributes	
energy, mass, masses, numbers, coordinates, inertia_tensor, multiplicity, periodic (see `Molecule` object)	
`PartFun` attributes	
terms	list of contributions
vibrational	vibrational partition function
electronic	electronic partition function
`PartFun` referenced attributes	
energy (see `NMA` object)	
`PartFun` optional attributes	
translational	translational partition function
rotational	rotational partition function
rotors	list of partition functions of the 1D internal rotors
`PartFun` methods	
entropy(temp)	entropy
internal_energy(temp)	internal energy
enthalpy(temp)	enthalpy
gibbs_free_energy(temp)	Gibbs free energy
helmholtz_free_energy(temp)	Helmholtz free energy
heat_capacity(temp)	heat capacity
log(temp)	ln Q (logarithm of partition function)
dlog(temp)	$\partial \ln Q/\partial T$
ddlog(temp)	$\partial^2 \ln Q/\partial T^2$

[a] Referenced attributes do not store a copy of the content, but merely refer to the original.

- Perform a normal-mode analysis, for example, `nma = NMA(mol)`. By adding a keyword, an alternative NMA model can be chosen.
- Create the partition function, for example, `pf = PartFun(nma)`. Additional keywords can specify the desired contributions to the partition function.

Next, the user can either perform further analysis on the frequencies and modes, or use the partition function to analyze the reaction kinetics.

A. Molecule Object. The Molecular Modeling software library `molmod`, which was developed earlier as a part of the program Zeobuilder[28] and the molecular dynamics toolkit MD-Tracks,[29] contains the basic definition of the `Molecule` object: the molecular conformation, the mass distribution, the energy, and its derivatives. Some of the `molmod` databases are reused, such as the periodic table in `molmod.periodic`, a list of physical constants in `molmod.constants`, and unit conversion factors in `molmod.units` (see Table 1).

The input−output module `tamkin.io` reads the data from the output file of the simulation programs: the N atomic masses, coordinates, energy, gradient, and second derivatives (Hessian) are essential quantities. The format of the `Molecule` object is independent of the file from which the data is loaded. An additional attribute (`molecule.periodic`) keeps track of whether the simulation was obtained with periodic boundary conditions. The rotational symmetry number is derived from the molecular geometry, is filled in if available in the output file, or can be specified manually by the user. The atomic symbols (H, He, etc.) are either read from the output file or derived from the atomic masses, for eventual later visualization purposes. TAMkin can read in data from a series of simulation packages: ADF,[3] CHARMM,[4] CPMD,[5] CP2K,[6] Gaussian03,[7] Q-Chem,[8] and VASP.[9] The `tamkin.io` module can be easily extended with new reading functions to obtain data from any other package.

TAMkin works internally consistently with atomic units throughout the program. Whenever the user wants to generate output expressed in other units, the quantities can be simply converted with the `molmod.units` and `molmod.constants` modules. For instance, the print statement `print mol.gradient/(1e3*joule/angstrom/mol)` will print the gradient of the `Molecule` instance `mol` in kJ Å^{-1} mol^{-1}.

B. NMA Object. The `tamkin.nma` module uses the information of a `Molecule` object to construct a new `NMA` object that computes and stores the normal modes and frequencies. A variety of recent, advanced NMA approaches are implemented: the standard full Hessian NMA, the Mobile Block Hessian approach with or without adjoined blocks,[30,31] which includes the Rotation-Translation Block method,[32,33] the Partial Hessian Vibrational approach,[34−36] the Vibrational Subsystem Analysis with or without mass matrix correction,[37] the combined PHVA+MBH method, the Elastic Network Model,[38,39] and NMA with constraints, i.e., for projecting out the global translations, global rotations or a specific set of stretch motions. Some of those approaches are readily available in existing simulation packages, but TAMkin is the first program to bundle all. Moreover, the modular design of TAMkin allows the addition of new methods to `tamkin.nma` in the future without disturbing the flow of the program.

If an NMA method reduces the dimensionality of the eigenvalue problem, the reduced Hessian has a smaller $m \times m$ dimension ($m < 3N$, N is number of atoms). Solving the NMA equations results in m eigenvectors of length m, describing the displacement of m generalized coordinates. TAMkin applies the $3N \times m$ back-transform from general to Cartesian coordinates, such that the m returned eigenvectors have length $3N$ and are ready-to-use, for example, to create a mode trajectory file. This additional functionality of TAMKIN to perform the transformation from Cartesian to normal mode coordinates could be particularly useful for methods which are based on the use of the normal modes as the work set of coordinates. Examples are simulations packages that perform geometry optimizations in generalized coordinates or molecular dynamics packages.[40−43] Special attention was paid to the role of the zero eigenfrequencies of the Hessian, as they indicate the quality of the geometry optimization and the Hessian calculation. The eigenfrequencies corresponding to the global translational and rotational invariances of the system are automatically labeled as zero frequencies

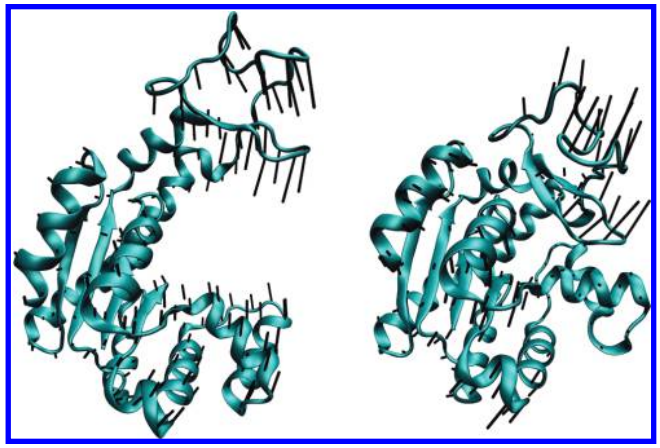

Figure 2. Adk conformations: open and closed. The lowest frequency mode of the open and closed conformations are drawn.

by comparing them to global translational and rotational displacement vectors.

TAMkin is capable of calculating frequencies for partially optimized molecular systems. An example is the NMA of a cluster where part of the system, the chemical environment, has been optimized at a lower level of theory, but consequently has been kept fixed during the high level-of-theory optimization. In this case, the chemically active part of the system is fully optimized, while residual forces still act on the environment atoms. For reactions occurring at a surface or in porous materials such as zeolites and MOFs, part of the system is often kept fixed at the lattice positions during the geometry optimization to simulate the solid state. A standard normal mode calculation based on the full Hessian is in this case unphysical.[44] With TAMkin, the user can apply the PHVA, the MBH or the combined PHVA/MBH approach to determine physical frequencies and derive reaction kinetics.[45]

With the functions available in `tamkin.nmatools`, the normal modes can be visualized by writing trajectory files, readable by VMD[25] and Molden.[26] Such trajectories have been used in VMD to create Figure 2 of the adenylate kinase example of section III. It is also possible to calculate the overlap between modes obtained from different NMA approaches, as well as the overlap with the conformational change of a two-state-system (see section III).

For large systems, the diagonalization of the Hessian becomes the most expensive step. Therefore TAMkin provides a checkpointing function, where the results of the expensive computation are stored in a file in text format. This encourages piece-wise analyses as a strategy for large systems.

C. PartFun Object. In the `tamkin.partf` module, partitions functions are constructed based on the frequencies, that is, within the harmonic oscillator approximation. The molecular partition function has contributions from the electronic, translational, rotational, and vibrational partition function

$$q_{\mathrm{mol}} = q_{\mathrm{elec}} q_{\mathrm{trans}} q_{\mathrm{rot}} q_{\mathrm{vib}} \tag{1}$$

with the vibrational partition function constructed from the vibrational frequencies ν_i

$$q_{\mathrm{vib}} = \prod_i q_{\nu_i} = \prod_i \frac{\exp\left(-\frac{h\nu_i}{2k_{\mathrm{B}}T}\right)}{1 - \exp\left(-\frac{h\nu_i}{k_{\mathrm{B}}T}\right)} \tag{2}$$

where k_{B} is the Boltzmann constant, h the Planck constant, and T the temperature. The workhorse of the module is the `PartFun` class. The `PartFun` object ensures the correct initialization of the molecular partition function with its translational, rotational, and vibrational contribution, using the frequencies stored in an `NMA` object. The user has the flexibility to decide upon which terms are active. One can choose for a classical or quantum mechanical treatment of the vibrations. A specific frequency can be replaced by a correction for the 1D Hindered Rotor model.[46−50] Frequencies corresponding to global translations and/or rotations are listed as zero frequencies and are consistently excluded from the vibrational partition function.

`PartFun` objects represent a partition function with an interface that does not depend on the different types of terms that are present in the partition function. This means that the same function call works for any of the derived partition function types. For instance, the evaluation of the derivative of the natural logarithm of the partition function with respect to the temperature `pf.dlog` of a `PartFun` instance `pf` ($\partial \ln q/\partial T$) will automatically call the corresponding `dlog` function of the translational, rotational and vibrational contribution. As a consequence, all applications that rely on partition functions work without knowledge of the individual contributions to the partition function.

The object oriented programming style prevents again useless recalculation of the same data. For instance, to obtain partition functions at different temperatures, the same `NMA` object can be reused. Furthermore, the user can request to write a typical thermochemical section to an output file.

One of the main objectives of the TAMkin program is the analysis of chemical kinetics. Multiple `PartFun` objects can be combined to study chemical equilibrium, reaction rate coefficients, kinetic parameters, and the various thermodynamic properties. Both unimolecular and bimolecular reactions are implemented. The program uses conventional transition state theory (TST).[17−20] The rate coefficient $k(T)$ of a bimolecular reaction $\mathrm{A} + \mathrm{B} \rightarrow \mathrm{C} + \mathrm{D}$ is hence computed as

$$k(T) = \kappa \frac{k_{\mathrm{B}}T}{h} \frac{(q_\dagger/V)}{(q_{\mathrm{A}}/V)(q_{\mathrm{B}}/V)} \exp\left(-\frac{\Delta E_0^\ddagger}{k_{\mathrm{B}}T}\right)$$

where κ is the tunneling coefficient, $q_\dagger$, q_{A}, and q_{B} relate to the partition functions without energetic and zero-point vibrational contribution, $\Delta E_0^\ddagger$ is the reaction barrier including zero-point corrections, and V is the reference volume used to evaluate the translational part of the partition function. As an important refinement to increase the accuracy of TST, tunneling corrections, both Wigner[51] and Eckart,[52] can be included during the TAMkin analysis. These methods only consider reaction stationary points and are hence compatible with TST. Besides the conventional TST based on the full Hessian frequencies, one can alter the vibrational partition function by using one of the extensions to NMA to compute the frequencies. The rate

coefficient can be fitted to the Arrhenius law to obtain a link with macroscopic properties:

$$k(T) \approx A \exp\left(-\frac{E_a}{k_B T}\right)$$

The barrier including zero-point corrections determines the magnitude of the activation energy E_a, while the remainder, which is mainly influenced by the molecular frequencies, determines the pre-exponential factor A. A sensitivity analysis is implemented using Monte Carlo sampling to test how the kinetic parameters in the Arrhenius law are affected by changes in molecular vibrations and reaction barrier. For instance, to see how an uncertainty of 1 cm^{-1} affects the kinetic parameters, the Monte Carlo routine adds a Gaussian noise with width 1 cm^{-1} to the original frequencies and recomputes the kinetic parameters 1000 times with those distorted vibrational frequencies. This latter feature provides valuable insight in the theoretical error margins.

D. Main Functionalities and Advantages of TAMkin.

- It is possible to checkpoint the results after using one of the modules, such that previous results can be reused. The main advantage of checkpointing after a certain step is that the next step can be repeatedly executed with different parameters, without the need to perform all previous steps again. This is especially useful for large systems when the computational load becomes challenging. For instance, in the adenylate kinase example in section III, the frequencies and modes were calculated once and stored in a checkpoint file, from which the data were read three times for the mode overlap, the free energy, and the entropy calculation.
- TAMkin is extensible. A large variety of NMA methods and contributions to the partition function are implemented, including advanced models, such as the PHVA model combined with the MBH model, or the 1D-HR model. The modular structure of the program allows the user to extend TAMkin to implement new NMA models or partition functions. The large set of test routines ensures high-quality code.
- The further analysis of the eigenfrequencies and normal modes is ready-to-use. All standard functions are implemented, such as writing the normal modes to a standard trajectory file, plotting the vibrational spectrum, calculating partition functions at any temperature, and estimating kinetic parameters.
- The basic commands are compact and self-explanatory.
- Interfacing the data is straightforward: masses, modes, etc., are readily available as attributes of the three main objects and can be easily accessed within the python script. This facilitates, for instance, a consistent comparative study of different geometries because the data are readily available for further processing and plotting. Moreover the user can be sure that the same formulas are used for the calculation when using distinctive simulation packages, without any risk for errors caused by (the lack of) unit conversions. This consistency makes TAMkin a reliable program.
- TAMkin can calculate the sensitivity of the frequencies to errors in the Hessian elements. Moreover, a Monte Carlo based sensitivity analysis tool is implemented for the kinetic parameters. This latter feature is until now nowhere available.
- Another strong point of TAMkin is the possibility of batch processing, that is, the execution of a series of jobs without manual intervention. Any computational protocol for a thermochemical property can be implemented in a Python script, which can be executed a large number of times. This guarantees consistent, reproducible and high-quality scientific output. This feature is illustrated in section IV with a level of theory study.

At present, the use of TAMkin requires the knowledge of basic Linux commands and the Python language. This should thus not cause limitations to its use because most computational scientists are familiar with the Unix environment and Python is known to be a straightforward programming language to use. In addition, the following examples provide a good starting point for new users. All data and scripts can be found in the `examples/` directory of the distribution of TAMkin.

III. EXAMPLE: CONFORMATIONAL CHANGE OF ADENYLATE KINASE (ADK)

Adenylate kinase is a phosphotransferase enzyme that catalyzes the conversion of adenosine-triphosphate (ATP). It is the model system of a dynamic molecular machine that cycles between well-defined states.[27,53,54] Its crucial role in signal pathways is made possible by the allosteric conformational change between open and closed structures, as shown in Figure 2 for *Escherichia coli* Adk. In general, conformational changes are largely determined by the low frequency modes of the protein, and thus the calculation of normal modes can clarify biological function. It is our aim to use TAMkin to determine the relevance of the lowest frequency modes to this "pacman motion" and to estimate the free energy and entropy difference between both conformations.

The geometries of the closed (Protein Databank 1ake) and open (Protein Databank 4ake) form were extensively optimized with the PARAM27 force field of CHARMM[4] to bring the root-mean-square of the residual gradient below 10^{-5} kJ/Å. The two structures were aligned according to the root-mean-square difference between the heavy atoms positions, and the rotated coordinates were stored in `closed.cor` and `open.cor`. The corresponding Hessian was calculated with the VIBRAN module of CHARMM and stored in `closed.hess` and `open.hess`. Adk consists of 214 residues, totaling 3339 atoms, which results in 10017 frequencies.

To analyze the system in TAMkin, the data are read from the CHARMM output files:

```
from tamkin import *
molecule1 = load_
molecule_charmm("open.cor", "open.hess")
```

The first line imports the TAMkin library. With the `Molecule` object, an `NMA` object can be constructed, for example

```
nma1 = NMA(molecule1)
```

performs the standard full Hessian calculation. The frequencies coincide with those obtained by a direct calculation with the DIAG command of the VIBRAN module in

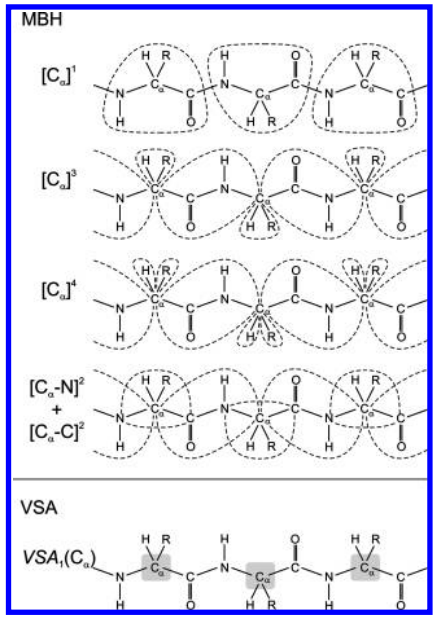

Figure 3. Adk block choices: MBH and VSA.

CHARMM. With similar commands, the closed form of Adk and its normal modes can be constructed (`molecule2`, `nma2`). The `NMA` object has the frequencies and modes as main attributes. The `tamkin.io` functions are used to store and reload the `NMA` object.

```
nma1.write_to_file("open.chk")
nma1 = load_chk("open.chk")
```

The checkpoint file can be reused multiple times, such that the expensive step of the computation, i.e., the diagonalization of the Hessian, need not be repeated.

TAMkin is now used to identify the normal modes that contribute most to the conformational change. In the spirit of a previous comparative paper,[15] the following 5 NMA schemes are selected: (1) the full Hessian; (2) the Rotation-Translation Blocks method RTB_n (where n is the number of residues in each block), originally proposed by Durand et al.[32] and later promoted for proteins by Tama et al.;[33] (3) the adjoined block schemes $[C_\alpha]^3$ and $[C_\alpha]^4$ proposed by Ghysels et al.,[31] where the first is the default MBH scheme for proteins, and where the second scheme is similar to the first, but does not fix the angle between the residue and the H atom attached to the C_α atom; (4) the adjoined block scheme $[C_\alpha{-}N]^2 + [C_\alpha{-}C]^2$ that selects the dihedral angles ψ and ϕ of the backbone as degrees of freedom, also proposed by Ghysels et al.;[31] (5) the Vibrational Subsystem Analysis VSA_n (C_α) (where n denotes that one C_α backbone carbon out of every n residues belongs to the subsystem), as proposed by Brooks et al.[37]

Figure 3 shows the division into blocks for MBH and the division into subsystem and environment for VSA. For a detailed description of these schemes, refer to ref 15. The predictable atom order in standard CHARMM coordinate files allows automated creation of the block divisions. Specifically, the following functions from the `tamkin.nmatools` library create the block choice or subsystem choice for RTB_1, $[C_\alpha]^3$, $[C_\alpha]^4$ $[C_\alpha{-}N]^2 + [C_\alpha{-}C]^2$, and VSA_1 (C_α), respectively:

```
blocks1 = create_blocks_peptide_charmm(
 "open.cor", "RTB", blocksize=1)
blocks2 = create_blocks_peptide_charmm(
 "open.cor","normal")
blocks3 = create_blocks_peptide_charmm(
 "open.cor","RHbending")
blocks4 = create_blocks_peptide_charmm(
 "open.cor", "dihedral")
subs = create_subs_peptide_charmm(
 "open.cor", frequency=1)
```

Next, normal modes are calculated with the following calls:

```
nma1 = NMA(molecule1) # standard
nma1_mbh1 = NMA(molecule1, MBH(blocks1))
nma1_mbh2 = NMA(molecule1, MBH(blocks2))
nma1_mbh3 = NMA(molecule1, MBH(blocks3))
nma1_mbh4 = NMA(molecule1, MBH(blocks4))
nma1_vsa = NMA(molecule1, VSA(subs))
```

The first (mandatory) argument of the `NMA` constructor is a molecule object. The second (optional) argument indicates which variant of the normal-mode analysis must be used, for example, MBH or VSA.

The normal modes are compared with the conformational change, which is characterized by the normalized, mass-weighted difference vector

$$\Delta r' = \frac{M^{1/2}\Delta r}{\Delta r^T M \Delta r} \tag{3}$$

where $\Delta r = r_{\text{open}} - r_{\text{closed}}$, the positions r_{open} and r_{closed} are $3N$-dimensional vectors, and M is the diagonal mass matrix. The square overlap $Q_j = |v_j^T \Delta r'|^2$ between the mass-weighted normal mode v_j and the difference vector is an adequate measure for the contribution of the jth mode to the conformational change. First the pacman vector is determined and next the dot product with the normal modes of the open or closed shape is calculated:

```
delta = get_delta_vector(
 nma1.coordinates, nma2.coordinates,
 normalize=True)
overlap1 = compute_overlap(nma1, delta)
```

and similarly for the overlap with modes of the other `NMA` instances (`nma2`, `nma1_mbh1`,...). Since the modes are normalized with respect to the mass tensor M, the square of the overlaps always lies in the range 0−100%.

Thermodynamic properties are obtained as the methods of a `PartFun` object, for instance at 300 K:

```
pf1 = PartFun(nma1, [ExtTrans(),
 ExtRot()])
G1 = pf1.gibbs_free_energy(300)
S1 = pf1.entropy(300)
```

With similar expressions for `molecule2`, one can calculate the theoretical free energy and entropy difference between the open and closed state for the several NMA models. In this way, TAMkin enabled a straightforward analysis of the performance of the models in reproducing the configurational pacman vector and the associated change in thermodynamic properties.

Interpretation of Results. Table 3 compares the value of the lowest nonzero frequency in cm^{-1} with the corre-

Table 3. Several Approximate NMA Methods (See Text) Applied on the Closed and Open Form of Adenylate Kinase, Compared to the Standard Full Hessian Calculation[a]

method	no. freq		ν_1 (Q_1) closed		ν_1 (Q_1) open		free energy ΔG	free energy error	entropy $T\Delta S$	entropy error
Full	10017	100%	3.07	(34%)	1.21	(63%)	−186.0	0.0	−22.51	v0.00
RTB_1	1284	13%	7.34	(33%)	2.07	(61%)	−184.8	1.2	−29.27	−6.76
RTB_2	642	6%	8.78	(33%)	2.66	(60%)	−187.5	−1.6	−27.23	−4.72
RTB_3	426	4%	10.22	(32%)	2.86	(63%)	−188.4	−2.5	−25.46	−2.95
RTB_4	318	3%	11.24	(26%)	3.44	(62%)	−192.2	−6.3	−22.18	0.33
RTB_5	252	3%	12.41	(24%)	4.22	(60%)	−184.3	1.6	−29.57	−7.07
VSA_1 (C_α)	636	6%	3.07	(34%)	1.21	(63%)	−193.9	−8.0	−19.53	2.98
VSA_2 (C_α)	325	3%	3.07	(34%)	1.21	(63%)	−194.5	−8.6	−19.95	2.56
VSA_3 (C_α)	210	2%	3.08	(34%)	1.21	(63%)	−200.0	−14.0	−14.56	7.95
VSA_4 (C_α)	156	2%	3.08	(34%)	1.21	(63%)	−192.9	−6.9	−21.56	0.95
VSA_5 (C_α)	123	1%	3.08	(34%)	1.21	(63%)	−185.4	0.6	−28.96	−6.45
$[C_\alpha]^4$	1685	17%	6.39	(34%)	1.73	(61%)	−187.5	−1.6	−29.10	−6.60
$[C_\alpha{-}N]^2 + [C_\alpha{-}C]^2$	426	4%	9.26	(35%)	2.55	(64%)	−180.6	5.4	−33.72	−11.22
$[C_\alpha]^3$	1278	13%	6.74	(34%)	1.82	(61%)	−182.2	3.7	−34.03	−11.52

[a] The lowest non-zero frequency of the closed and open state is given in cm^{-1} with the corresponding square overlap $Q_1 = |v_1^T \Delta r'|^2$ with the conformational change in % between parentheses. The free energy change $\Delta G = G_{closed} - G_{open}$ (kJ/mol) and entropy change $T\Delta S = TS_{closed} - TS_{open}$ (kJ/mol) at a temperature of 300 K are listed, as well as the error with respect to the full Hessian calculation.

sponding Q_1 overlap between parentheses for both conformations. The conformational change is indeed well reproduced by the lowest eigenvector, with an overlap of 63% for the lowest frequency mode of the open form. This number rapidly increases by including more low frequency modes. The closed form shows a lower overlap of 34% between the lowest frequency mode with the pacman vector, in agreement with the findings of Hinsen.[55] This is understood by the fact that the lowest modes are essentially determined by the shape of the protein. In the open form, the lobes are more distinct and the hinge bending motions of the two lobes with respect to each other is thus to be expected, whereas in the closed form the hinge motion is mixed up with other twisting motions. The results obtained with approximate NMA models are in accordance with the conclusions of ref 15: the block and subsystem choices as presented in Figure 3 are sufficient for an adequate description of the low frequency spectrum. Indeed, the estimates of the lowest nonzero frequency ν_1^{approx} and the square overlaps Q_1^{approx} lie close to the full Hessian values, despite the serious reduction in dimensionality to as low as 1% of the initial number of frequencies. The VSA, originally designed to describe the lower spectrum, produces the best frequency estimates, with a value of 3.07−3.08 cm^{-1} for the closed Adk and 1.21 cm^{-1} for the open Adk. There is a negligible difference with respect to the full Hessian values 3.07 cm^{-1} and 1.21 cm^{-1}.

Another interesting property is the difference in free energy $\Delta G = G_{closed} - G_{open}$ and entropy $\Delta S = S_{closed} - S_{open}$ between the closed and open conformations. Calculated thermodynamic quantities based on normal-mode analysis are rarely reported[56] and should be interpreted carefully. Because of the high dimensionality of the system, the potential energy surface has multiple minima (multistates[57−59]) and it is unrealistic that the thermodynamics are well described by the comparison of solely two minimum energy conformations. Such a description lacks the entropic contribution of hopping between multiple minima states with similar potential energies separated by low barriers. In addition, the harmonic approximation fails for low frequency vibrations. Nevertheless, the ΔG and ΔS differences calculated with NMA already give an indication of the realistic differences. The errors induced by the omission of the configurational hopping entropy and by the anharmonicity are expected to cancel out to a large extent by taking the difference between the two conformations.

Table 3 includes the ΔG and $T\Delta S$ values for the standard full Hessian at a temperature of 300 K. The difference in free energy, −186.0 kJ/mol, is governed by the internal energy, while the vibrational contribution ΔG_{vib} gives in comparison a rather small contribution of +27.4 kJ/mol (not in table). The closed form is the most favorable state. The entropy difference is given by $T\Delta S = -22.51$ kJ/mol and has a vibrational contribution $T\Delta S_{vib} = -5.13$ kJ/mol (not in table). The open structure is indeed less constrained than the closed structure, leading to some less constrained motions with a lower frequency, and thus a higher entropy.

Table 3 also lists ΔG and $T\Delta S$ obtained with the various approximate NMA models. The differences with respect to the full Hessian calculation are entirely due to the differences in the frequencies used to construct the partition function. Apparently, the approximate frequencies influence the values in a quite nonconsistent way, i.e. it is unpredictable whether ΔG and $T\Delta S$ increase or decrease by the use of MBH and VSA. Qualitatively, however, the same conclusions are drawn: the closed state is more favorable and has lower entropy. In fact, the role of the full Hessian as the benchmark is even questionable, because the high number of 10017 frequencies possibly leads to noise in the partition function. Indeed, small perturbations such as the reorientation of a side chain can influence a fraction of the frequencies. Given the large number of frequencies, the perturbations might imply a change in the thermodynamical properties, even when the small perturbations are irrelevant for the opening versus closing question. Therefore, it may be argued that a partition function based on the subset of the spectrum yields a better guess of the free energy or entropy difference beween two conformations, which is, intuitively, mostly dominated by the contributions of the large amplitude motions, rather

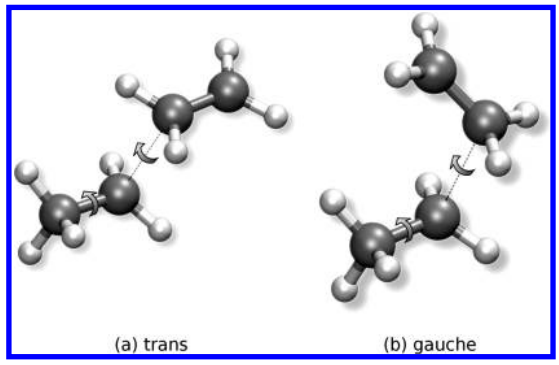

Figure 4. Trans and gauche transition states of the addition of the ethyl radical to ethene, with indication of the internal hindered rotors.

than the detailed local vibrations. The specific approximate NMA models considered in this paper perform best in the lower part of the spectrum, which may explain the (qualitatively) reasonable results.

The analysis performed here shows that TAMkin is ideally suited to generate very fast a number of postprocessing procedures on the Hessian. As a result, the impact of each scheme is visible on the conformational change, free energy, or entropy. Formerly, such analyses required a lot of manual intervention, whereas now they can be performed in an easy and transparent fashion.

IV. EXAMPLE: SYSTEMIC ANALYSIS OF CHEMICAL KINETICS OF THE ETHENE TO ETHYL RADICAL ADDITION

The addition reaction of the ethyl radical to ethene is used to demonstrate how TAMkin facilitates the systematic prediction of theoretical rate constants and kinetic properties. This reaction is the initial step of the free-radical polymerization of ethene and provides a well-studied reference both from an experimental[60,61] and theoretical[62−64] perspective. Figure 4 depicts the two possible transition states of the addition reaction, denoted trans and gauche. Special attention will be paid to the accurate treatment of internal rotations. This is a particular example of additional calculations and analyses that are beyond those available in common program packages. In the case of the ethyl reactant, the rotation of the methyl group is considered. Both transition states have two internal rotors, one corresponding to a methylene rotation, and one corresponding to the rotation of the approaching ethyl about the forming bond. In the following section, we compare the harmonic oscillator (HO) with the internal rotor (IR) implementation.

Moreover, a level of theory study illustrates the batch processing of a large series of kinetic calculations. Recent modifications on density functionals (DFT) methods focus on the inclusion of dispersion interactions, aiming at approaching the accuracy of high-level wave function theory techniques, at least for a variety of standard applications.[65−67] These developments, in addition to the availability of high-level composite procedures, have boosted assessment studies in recent years.[68−74] For the title reaction, already a lot of level of theory studies have been conducted. This type of study usually still requires a lot of manual intervention, which hampers performing the analysis systematically. TAMkin is the ideal tool to validate the newly developed functionals in a systematic way.

```
from tamkin import *

# 1) Load data from Gaussian03 files
# ----------------------------------------------
mol_ethyl = load_molecule_g03fchk("ethyl_freq.fchk")
mol_ethene = load_molecule_g03fchk("ethene_freq.fchk")
mol_ts = load_molecule_g03fchk("ts_freq.fchk")

# 2) Perform NMA
# ----------------------------------------------
nma_ethyl = NMA(mol_ethyl, ConstrainExt())
nma_ethene = NMA(mol_ethene, ConstrainExt())
nma_ts = NMA(mol_ts, ConstrainExt())

# 3) Construct the partition functions
# ----------------------------------------------
pf_ethyl = PartFun(nma_ethyl, [ExtTrans(), ExtRot()])
pf_ethene = PartFun(nma_ethene, [ExtTrans(), ExtRot()])
pf_ts = PartFun(nma_ts, [ExtTrans(), ExtRot()])

# 4) Analyze the partition functions (optional)
# ----------------------------------------------
ta_ethyl = ThermoAnalysis(pf_ethyl, [300, 400, 500, 600])
ta_ethyl.write_to_file("thermo_ethyl.csv")
ta_ethene = ThermoAnalysis(pf_ethene, [300, 400, 500, 600])
ta_ethene.write_to_file("thermo_ethene.csv")
ta_ts = ThermoAnalysis(pf_ts, [300, 400, 500, 600])
ta_ts.write_to_file("thermo_ts.csv")

# 5) Analyze the reaction
# ----------------------------------------------
ra = ReactionAnalysis([pf_ethyl, pf_ethene], pf_ts, 300, 600, 10)
ra.plot_arrhenius("arrhenius.png")
ra.write_to_file("reaction.txt")

# 6) Sensitivity Analysis
# ----------------------------------------------
ra.monte_carlo()
ra.plot_parameters("parameters.png")
```

Figure 5. Python script `reaction_ho.py` calculates reaction rates within the harmonic oscillator (HO) approximation.

A. Calculating and Analyzing Chemical Kinetics in TAMkin. The reactant and transition state geometries, energies and Hessians are computed with Gaussian03[7] using the B3LYP/6-31G(d) level of theory. Torsional energy profiles of the hindered rotors result from a relaxed potential energy surface scan at the same level of theory. The standard output is written to a log-file (e.g., `ethyl_scan_methyl.log`, `ts_scan_forming_bond.log`, or `ts_scan_methyl.log`) and the formatted checkpoint file is stored in a file with the extension fchk (e.g., `ethene_freq.fchk`, `ethyl_freq.fchk`, or `ts_freq.fchk`).

Figure 5 displays the `reaction_ho.py` script for the computation of the chemical kinetics of the trans attack within a pure harmonic-oscillator (HO) model, hence neglecting the presence of hindered internal rotors (IR). Comment lines starting with a # symbol are ignored when

the script is executed. We now discuss each section of the script in detail.

1. First, the data are loaded from the Gaussian03 formatted checkpoint files with the function `load_molecule_g03fchk`.
2. Next, an `NMA` object is constructed with the frequencies and modes as main atttributes. In principle, coupling between external rotations and the low-frequency modes should be absent when the geometry is properly optimized, but in practice, the limited accuracy of the second order derivatives in an ab initio calculation results in spurious coupling. Therefore we choose to perform the normal-mode analysis in constrained external degrees of freedom, which is specified as the second argument of the `NMA` constructor.
3. The `PartFun` constructor requires an `NMA` object as first argument. The second argument is a list (indicated by square brackets) of contributions to the partition function. The vibrational and electronic contribution are always implicitly included in the list, but can also be explicitly added in order to replace the default parameters by specific ones. For example, one can use scaled frequencies and scaled zero-point energies with the following line:

```
pf_ethyl = PartFun(nma_ethyl,
 [ExtTrans(), ExtRot(),
 Vibrations(freq_scaling=0.9614,
 zp_scaling=0.9806)])
```

For gas-phase molecules, one must explicitly include external translations and rotations to the list. The external rotation takes the rotational symmetry number as argument. When the rotational argument is omitted, it is derived from the molecular geometry.

All contributions to the partition function are stored as attributes of the `PartFun` object. Contributions themselves are also represented by objects that are very similar to a `PartFun` object and have the relevant parameters (inertia tensor, multiplicity etc.) stored in their attributes. For example, one prints the multiplicity of the electronic contribution as follows:

```
print pf_ethyl.electronic.multiplicity
```

The thermodynamic properties, however, are not implemented as attributes, but as methods of the `PartFun` object, because they depend on the temperature. The following two lines print the Gibbs free energy of the ethyl radical at 300 K and the vibrational entropy at 600 K in atomic units:

```
print pf_ethyl.gibbs_free_energy(300)
print pf_ethyl.vibrational.entropy(600)
```

The first line works for any `PartFun` object, no matter what contributions are included in its list. The same is true for other methods of the `PartFun` object, such as `entropy`, `internal_energy`, and so on. This reflects the fact that the interface to the partition function does not depend on the contributions present in the list. This level of abstraction allows us to implement *generic* routines to compute derived properties, such as kinetic parameters. Reasoning at this abstract level is natural, but writing compact and elegant programs at an abstract level requires a carefully designed library such as TAMkin and a modern programming language such as Python.

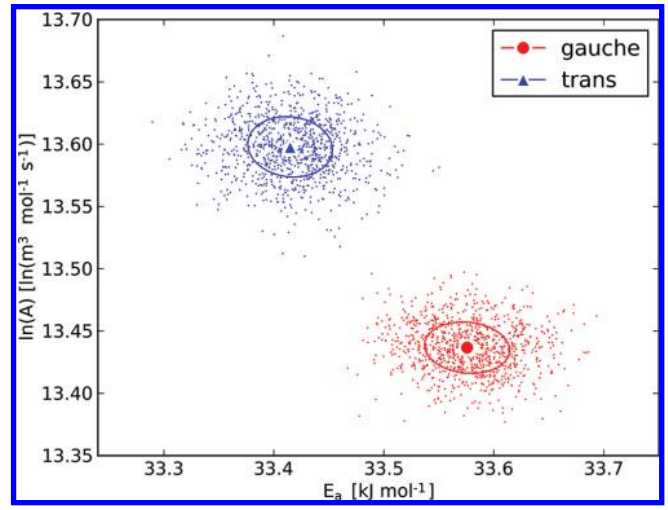

Figure 6. Ethyl radical addition to ethene: the estimated spread on the kinetic parameters because of uncertainty in the vibrational frequencies. Data points are obtained by a Monte Carlo run, assuming a Gaussian distributed error of 1 cm^{-1} on the frequencies.

4. The thermochemical analysis is another example of a generic implementation that works for any `PartFun` object. A `ThermoAnalysis` object is constructed with two mandatory arguments: a `PartFun` object and a list of temperatures. If requested, the method `write_to_file` writes all thermodynamic quantities to a csv-file that can be loaded into conventional spreadsheet software.
5. The next three lines compute the rate constants at different temperatures, fit the kinetic parameters to these rate constants, and write all the output to disk. The `ReactionAnalysis` constructor takes five arguments: (i) a list of reactant partition functions, (ii) the partition function of the transition state, (iii) a begin temperature, (iv) an end temperature, and (v) a temperature step. The number of reactant partition functions is in principle unlimited. The last three arguments are used to construct a grid on the temperature axis, in order to fit the Arrhenius law. The method `plot_arrhenius` plots the rate constants and the fitted Arrhenius law in a typical $\ln(k)$ versus $1/T$ plot. A comprehensive summary of the kinetics computation and the partition functions can be written to a text-file with the method `write_to_file`. We furthermore note that the `ReactionAnalysis` constructor can take two optional arguments: a flag whether the reaction occurs under constant volume or constant pressure and a tunneling correction object. Wigner and Eckart tunneling corrections can be computed.
6. The last section in the example code performs a sensitivity analysis of the kinetic parameters for fluctuations in the vibrational frequencies. We observed that different initial geometries for the geometry optimization of the reactants and the transition state result in slightly different optimized geometries and that, in turn, these lead to an uncertainty of the computed vibrational frequencies of about 1 cm^{-1}. The Monte Carlo routine recomputes the kinetic parameters 1000 times with distorted vibrational frequencies, that is, Gaussian noise with width 1 cm^{-1} is added to the original frequencies. Figure 6 illustrates the result for both the trans and the gauche reaction.

B. Refinement: The One-Dimensional Hindered Rotor Model. Figure 7 displays the `reaction_ir.py` script for the computation of the trans reaction rate with

```
from tamkin import *

# 1) Load data from Gaussian03 files
# ----------------------------------------------
mol_ethyl = load_molecule_g03fchk("ethyl_freq.fchk")
mol_ethene = load_molecule_g03fchk("ethene_freq.fchk")
mol_ts = load_molecule_g03fchk("ts_freq.fchk")
scan_ethyl = load_scan_g03log("ethyl_scan.fchk")
scan_ts_methyl = load_scan_g03log("ts_scan_methyl.fchk")
scan_ts_forming_bond = load_scan_g03log("ts_scan_forming_bond.fchk")

# 2) Perform NMA
# ----------------------------------------------
nma_ethyl = NMA(mol_ethyl, ConstrainExt())
nma_ethene = NMA(mol_ethene, ConstrainExt())
nma_ts = NMA(mol_ts, ConstrainExt())

# 3) Construct the partition functions
# ----------------------------------------------
rotor_ethyl = Rotor(scan_ethyl, mol_ethyl, 6, True)
rotor_ts_methyl = Rotor(scan_ts_methyl, mol_ts, 3, False)
rotor_ts_forming_bond = Rotor(scan_ts_forming_bond, mol_ts, 1, True)
pf_ethyl = PartFun(nma_ethyl, [ExtTrans(), ExtRot(), rotor_ethyl])
pf_ethene = PartFun(nma_ethene, [ExtTrans(), ExtRot()])
pf_ts = PartFun(nma_ts, [ExtTrans(), ExtRot(), rotor_ts_methyl,
    rotor_ts_forming_bond])

# 4) Analyze the partition functions (optional)
# ----------------------------------------------
ta_ethyl = ThermoAnalysis(pf_ethyl, [300, 400, 500, 600])
ta_ethyl.write_to_file("thermo_ethyl.csv")
ta_ethene = ThermoAnalysis(pf_ethene, [300, 400, 500, 600])
ta_ethene.write_to_file("thermo_ethene.csv")
ta_ts = ThermoAnalysis(pf_ts, [300, 400, 500, 600])
ta_ts.write_to_file("thermo_ts.csv")

# 5) Analyze the reaction
# ----------------------------------------------
ra = ReactionAnalysis([pf_ethyl, pf_ethene], pf_ts, 300, 600, 10)
ra.plot_arrhenius("arrhenius.png")
ra.write_to_file("reaction.txt")

# 6) Plot the energy levels of the rotors
# ----------------------------------------------
rotor_ethyl.plot_levels("rotor_ethyl_energy_levels.png")
rotor_ts_methyl.plot_levels("rotor_ts_methyl_energy_levels.png")
rotor_ts_forming_bond.plot_levels("rotor_ts_forming_bond_levels.png")
```

Figure 7. The Python script `reaction_ir.py` calculates reaction rates with a 1D internal rotors (IR) refinement for the internal rotations.

corrections for the internal rotations (IR). In the one-dimensional rotor model, the internal rotation is assumed to be decoupled from all other modes in the molecule. Further refinements to higher dimensionality do not result in significant changes in the thermodynamic properties.[47,49,50,75] We discuss the changes with respect to the previous script.

1. The rotational scan data are loaded from the Gaussian03 log-files with the function `load_scan_g03log` into a `Scan` object. For some molecular systems, it is nontrivial to perform relaxed potential energy surface scans. It is then possible to join output files from multiple conventional constrained optimizations with the Unix `cat` command, and load the resulting file with the function `load_scan_g03`.
2. The 1D-rotor contribution to the partition function is represented by the `Rotor` object, which is initialized with four arguments: (i) a `Scan` object, (ii) the `Molecule` object to which the rotor applies, (iii) the rotational symmetry number of the rotor, and (iv) whether the torsional energy profile should be an even function of the torsional angle. The torsional energy profile and the wave function are expanded in a Fourier basis, taking into account the rotational symmetry and parity, to determine the energy levels quantum mechanically. The Fourier expansion of the potential is obtained with a least-squares fit to the torsional scan data. Singular value decomposition is used to obtain a well-behaved fit.[76] The implementation is such that the data points from the scan do not have to be equidistant, as opposed to the Fast Fourier Transform (FFT) algorithm.

   The `Rotor` object not only adds the contribution $q_{\mathrm{rot,1D}}$ of the 1D rotor to the partition function, but also cancels the vibrational mode: q_{tot} is replaced by $q_{\mathrm{tot}}\, q_{\mathrm{rot,1D}}/q_{\nu,\,\mathrm{cancel}}$. It is not recommended to simply pick ν_{cancel} as the frequency whose eigenmode corresponds approximately with the internal rotation.[77] Instead, TAMkin derives ν_{cancel} from the rotational scan, by evaluating the curvature of the Fourier fit at the reference point and mass-weighting by the appropriate inertia moment. Alternatively, TAMkin can use the linked blocks version of MBH[31,78] to estimate ν_{cancel} with a smart block choice. The first block contains the rotor (e.g., the methyl group) and the two atoms defining the rotation axis. The second block contains the two atoms defining the rotation axis and the remainder of the molecule. The resulting system has thus the internal rotation as the only degree of freedom. Using the `write_to_file` method, the details of each rotor treatment are included in the file `reaction.txt`.
3. The last part of the script `reaction_ir.py` plots the energy levels, the fitted torsional potential and the original input data of the torsional scan.

With this analysis, we easily found the following results for the addition reaction of the ethyl radical to ethene, through the trans transition state. The reaction rate was found to be 2.4×10^1 m^3 mol^{-1} s^{-1} at 500 K with a fully harmonic TST calculation. The 1D-rotor model shifts this value to 1.7×10^2 m^3 mol^{-1} s^{-1}.

C. Batch Processing. The addition of the ethyl radical to ethene (Figure 4) is furthermore studied using a wide range of theoretical methods to demonstrate TAMkin's possibilities of batch processing. In total, 41 different (post-)Hartree−Fock (HF) and Density Functional (DFT) methods were assessed, using the Gaussian03 software package.[7] Geometries, frequencies, and energies, for both the trans and gauche attack, are computed using the electronic structure methods in conjunction with two different basis sets (6-31G(d)[79,80] and 6-311+G(3df,2p)[81−83]). The use of this methodology, that is, all computations at the same level of theory, are referred to as the "consistent" approach. The influence of energy refinements is singled out using a "2-component" method, with optimized B3LYP/6-31G(d) geometries, frequencies and rotational scans, but with energy barriers at a different level of theory and the larger 6-311+G(3df,2p) basis set. For all

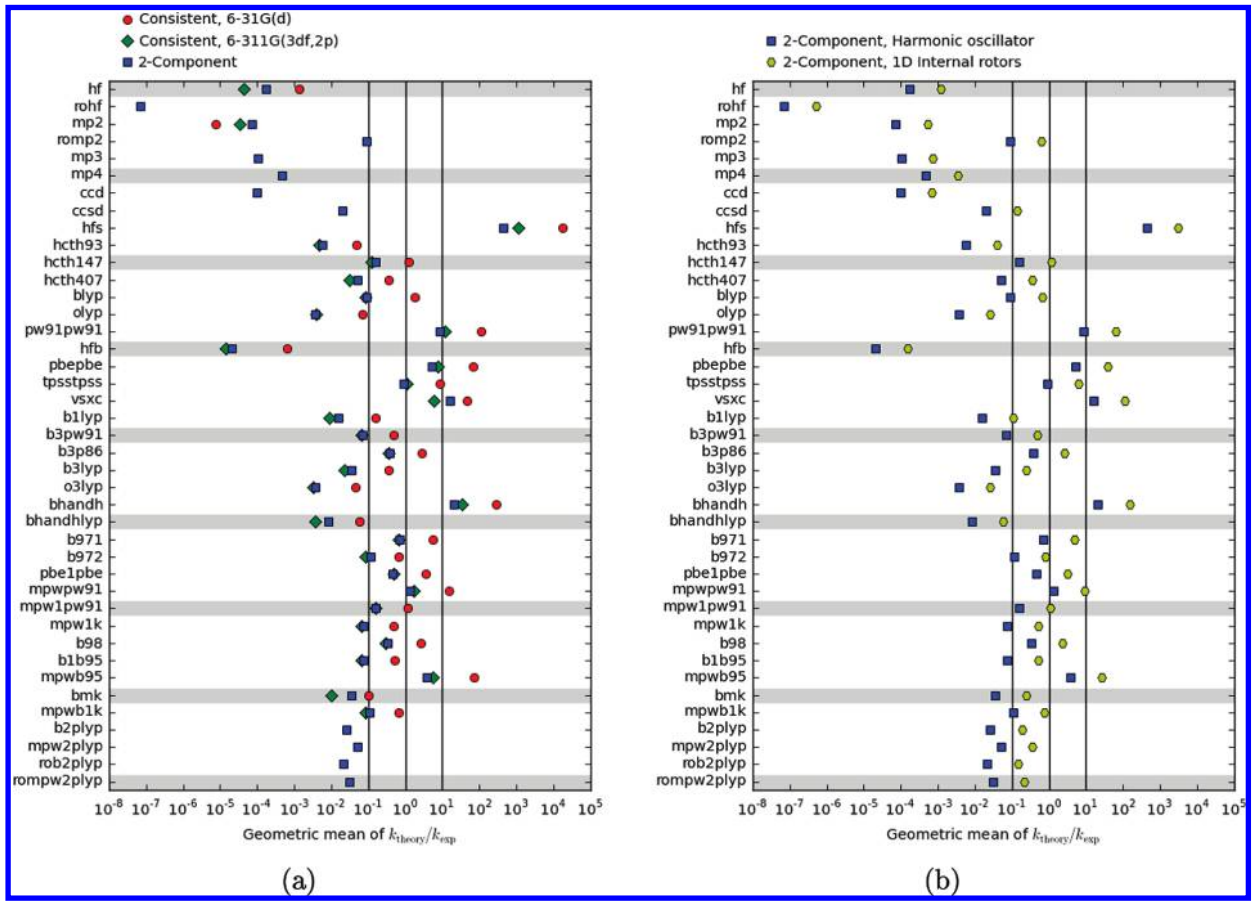

Figure 8. Computed reaction rates k_{theory} of the ethyl radical addition to ethene are compared with the experimental rate k_{exp}. The average ratio $f = k_{theory}/k_{exp}$ taken over a temperature range 300−600 K and the trans/gauche reaction (geometric mean, see text) are plotted on a logarithmic scale for each level of theory. (a) Influence of methodology. (b) Influence of rotor treatment (data from 2-component methodology): HO versus IR.

methodologies the effect of the internal rotor approach, as well as the inclusion of counter-poise corrections[84] for the reaction barrier is tested. In total, more than 1350 Gian03 computations were carried out and 800 reaction rates were computed.

The methodological assessment includes four typical tasks, which are all automated using Python scripts: (i) the generation of input files, (ii) the inspection of the output files for error and warning messages, (iii) the computation of the rate constants and kinetic parameters with TAMkin, and (iv) the compilation of tables and figures based on the output generated by TAMkin. The main advantage of TAMkin is that it automates the entire process of performing an extended level-of-theory assessment, which is a time-consuming task. The present illustration makes use of Gaussian03 only, but results from other software packages can easily be included.

Given the amount of data generated and processed, it is crucial to pay extra attention to quality control of the Gaussian03 log-files which serve as input to TAMkin. A first check is the use of scripts to check each output for common signs of malfunction. In case of correct termination of the calculation, typical problems to anticipate are spin contamination in the HF and post-HF methods and the convergence toward the wrong transition state. These issues are controlled by generating tables of $\langle S^2 \rangle$ values and the most relevant internal coordinates obtained with each method. Other sanity checks include the correct number of imaginary frequencies, and a reasonable value for the barrier height (to detect issues in the SCF cycle). One can also compare results with literature data or manually computed outcomes for further validation. When internal rotations are present, corrupt data points of the rotational scans can easily be detected with an outlier test and by checking the goodness of fit of the Fourier expansion of the torsional potential. In case of sufficient other data points, these corrupt points can be omitted; or the computation can be investigated in detail and redone manually whenever necessary. The correct periodicity of the rotational potential can also be checked. The TAMkin library itself is provided with extensive test routines according to the "unit testing principle" and should not be of a concern.

The construction of the Figures 8 and 9 is facilitated through TAMkin. A straightforward analysis of the influence of various aspects such as the applied methodology for geometry optimization/energy calculation, the HO versus IR approach, the inclusion of basis set superposition error (BSSE) corrections and the performance of various levels of theory is now possible.

Interpretation of Results. Although the main purpose of this exposition is to highlight the functionalities of TAMkin, a comparison of the theoretical results with experimental data is indispensable. Rate constants of the addition of ethene to the ethyl radical have been measured in a heated tubular flow reactor coupled to a photoionization mass spectrometer and

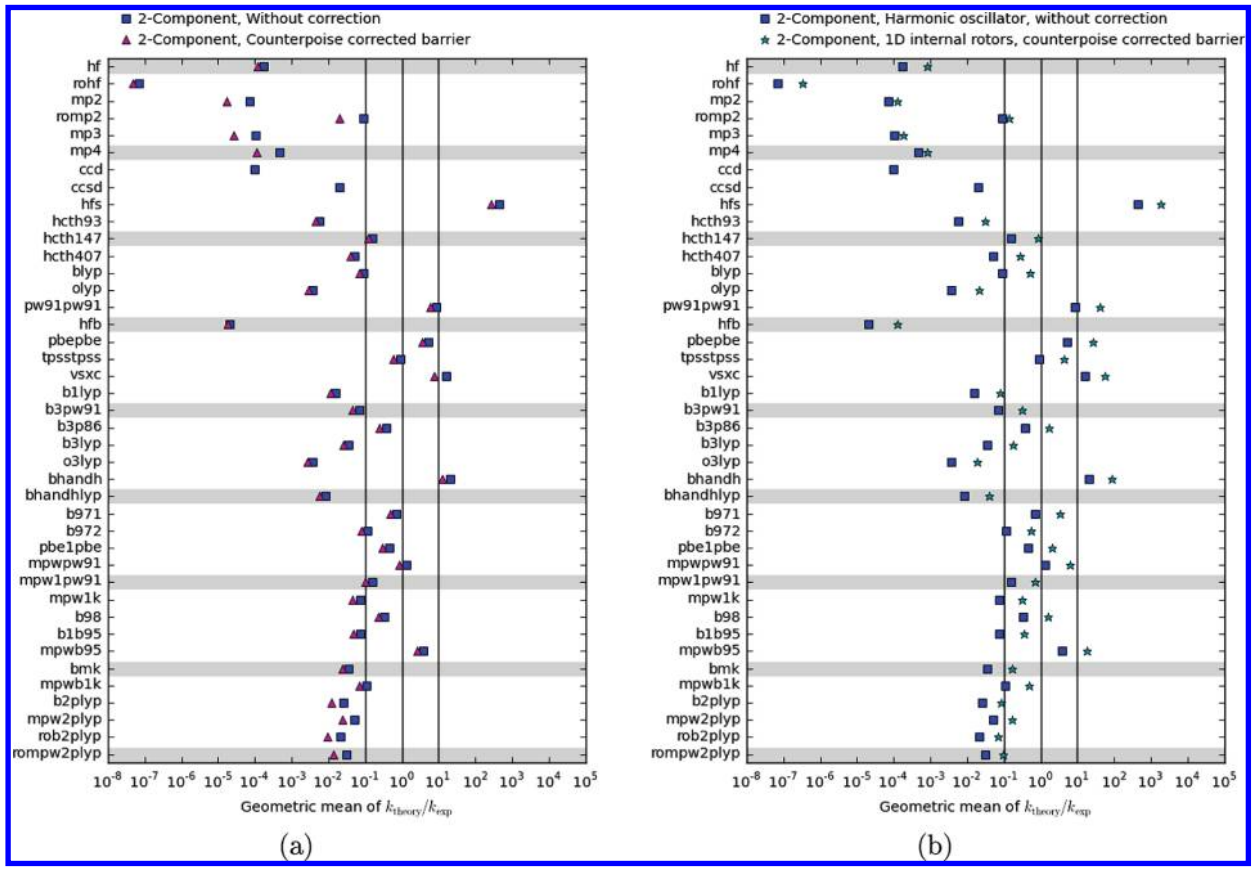

Figure 9. Computed reaction rates k_{theory} of the ethyl radical addition to ethene are compared with the experimental rate k_{exp}. The average ratio $f = k_{theory}/k_{exp}$ taken over a temperature range 300−600 K and the trans/gauche reaction (geometric mean, see text) is plotted on a logarithmic scale for each level of theory. (a) Influence of the counterpoise correction. (b) Combined effect of rotor treatment and counterpoise correction.

serve as a reference.[60] The experimentally observed reaction rate is

$$k_{exp}(T) = 4.33 \times 10^3 [\mathrm{m^3\ mol^{-1}\ s^{-1}}] \left(\frac{T}{298\ \mathrm{K}}\right)^{2.44} \times \exp\left(-\frac{22.45[\mathrm{kJ\ mol^{-1}}]}{RT}\right)$$

where R is the universal gas constant. When the conventional Arrhenius law is fitted within the temperature interval between 300 and 600 K the pre-exponential factor is 1.21×10^5 m^3 mol^{-1} s^{-1} and the activation energy becomes 31.0 kJ mol^{-1}. The kinetic parameters can vary to a large extent depending on the range in which they have been fitted. Instead, we will focus on the reproduction of the rate constant curve itself in the relevant temperature interval. The introduction of a ratio $f = k_{theory}/k_{experiment}$ allows a direct comparison between theoretical and experimental rate coefficients. A deviation up to a factor of 10 is commonly interpreted as acceptable. Figures 8 and 9 present the geometric mean of the ratio f, taken over the temperatures 300, 400, 500, and 600 K, and the trans/gauche rate. Note that the usual arithmetic mean is not suitable since f varies over several orders of magnitude. The computed values of the rate constants, kinetic parameters and deviations between theory and experiment are summarized in the Supporting Information (Tables S1−S4).

In all calculations, the trans reaction is slightly faster compared to the gauche attack; however, both reactions will occur competitively. The findings for the reaction under investigation are a consistent extension of the results in ref 63.

Methodology. Figure 8a shows that the cost-effective two-component methods lead to very similar results as the fully consistent approach using the large 6-311+G(3df,2p) basis.

HO/IR. Using the two-component methods, Figure 8b shows that the inclusion of IR corrections (in this case derived from B3LYP frequencies) increases the theoretical rate coefficients. In the HO approximation, there is a cancellation of errors for the neglected methylene internal rotor, because it is present in both the reactant and the transition state. This error cancellation is not possible for the rotation about the forming bond and a more accurate IR description of this transitional mode is therefore more important.[85,86] When the IR model is used, more density functional methods (21 versus 12) give a good reproduction of the experimental rate constant.

BSSE Corrections. The effect of the counterpoise method to correct for basis set superposition errors is significant in case of small basis sets. The correction is on average 9 kJ/mol when using the 6-31G(d) basis, except for the MP2 method, which is clearly more sensitive with a correction of 17 kJ/mol. For the present reaction a small basis set can thus be used for geometry optimization when this is combined with BSSE corrections for the reaction barrier. Individual BSSE corrections are included in the Supporting Information. The correction is almost negligible when a larger basis set is used (average of 2 kJ/mol), with MP2 being more sensitive (5 kJ/mol). Figure 9a shows the HO and IR results using the two-component methods. Also, here, inclusion of the BSSE corrections leads to a small decrease of the theoretical rate coefficient;

however, this refinement is less important than taking into account a more detailed IR model.

Levels of Theory. Using our most accurate results, including IR and BSSE corrections, Figure 9b shows that the several hybrid, meta hybrid, and double hybrid functionals are suitable choices for the computation of accurate energetics for the studied reaction.

The above application shows the case of systematically calculating reaction parameters for a level of theory study and other varying parameters (e.g., the influence of internal rotations) with automatic generation of the requested results and analysis plots.

V. PROGRAM AVAILABILITY

The TAMkin toolkit is distributed as open source software under the conditions of the GNU General Public License, version 3. Publications and communications based in parts on this program are required to cite the present paper. The software can be downloaded from the code Web site of the Center for Molecular Modeling: `http://molmod.ugent.be/code/`. Installation instructions and technical support are also available on this Web site. In addition, there is a web-interface to the revision control system that logs all changes in the source code. The auxiliary package `MolMod` can be found on the same Web site.

VI. CONCLUSION

In this paper, we have presented TAMkin, a freely available program for the calculation and analysis of normal modes and chemical reaction rates. TAMkin is the first program supporting a broad variety of advanced models to perform a normal-mode analysis (NMA). The implemented methods include the standard full Hessian NMA, the Mobile Block Hessian (MBH) approach with or without adjoined blocks, which includes the Rotation-Translation Block method, the Partial Hessian Vibrational approach (PHVA), the Vibrational Subsystem Analysis with or without mass matrix correction, the combined PHVA+MBH method, the Elastic Network Model, and NMA with constraints. The user has at its disposal research tools such as a sensitivity analysis of the frequencies, the calculation of mode overlaps, or the visualization of selected modes through the generation of trajectory files. Chemical kinetics of unimolecular and bimolecular reactions can be analyzed in a straightforward way using conventional transition state theory, including tunneling corrections and the 1-D hindered rotor corrections whenever necessary. The chemical kinetics calculation also includes a sensitivity analysis, which provides important insight into the theoretical error margins on the kinetic parameters. TAMkin supports output from all frequently used software programs, such as ADF, CHARMM, CPMD, CP2K, Gaussian, Q-Chem, and VASP. The program is moreover readily extensible because of its modular structure. The present paper demonstrates all aforementioned capabilities of TAMkin through two examples. The important feature of batch processing large amounts of data is also demonstrated by performing an extended level of theory study. TAMkin can automate the analysis to a large extent and errors because of manual processing are as such avoided.

ACKNOWLEDGMENT

T.V. and K.H. are postdoctoral Fellows of the Fund for Scientific Research, Flanders (FWO). This work is supported by the Fund for Scientific Research, Flanders (FWO), the Research Board of Ghent University (BOF), and BELSPO in the frame of IAP/6/27. Funding was also received from the European Research Council under the European Community's Seventh Framework Programme (FP7(2007-2013) ERC grant agreement number 240483). Computational resources and services used in this work were provided by the Lobos cluster of the National Institutes of Health and by Ghent University.

Supporting Information Available: Tables S1a−d contain the theoretical rate constants of the trans and gauche reaction at several temperatures calculated with the pure harmonic oscillator (HO) approximation or the internal rotor (IR) corrections and with or without the BSSE corrections. Rate constants were calculated in three ways, (1) with the consistent approach using the 6-31G(d) basis set, (2) with the consistent approach using the larger 6-311+G(3df,2p) basis, and (3) with the 2-compound approach where geometries, frequencies and torsional potentials are obtained at the B3LYP/6-31G(d) level, and the barriers are computed with other levels using the larger 6-311+G(3df,2p) basis. To facilitate the comparison with the experimental rate constants, Tables S2a−d express the deviation between theory and experiment on a logarithmic scale (decibels). Tables S3a−d are similar to Tables S1a−d, but contain the pre-exponential factor and the activation energy fitted to the reaction rates in the temperature interval 300−600 K. This information is available free of charge via the Internet at http://pubs.acs.org/.

REFERENCES AND NOTES

(1) Fessenden, R. J.; Fessenden, J. S. *Organic Chemistry*, 4th ed.; Brooks/Cole Publishing Company: Belmont, CA, 1990. 323−339.
(2) Cui, Q.; Bahar, I. *Normal Mode Analysis: Theory and Applications to Biological and Chemical Systems*; Mathematical and Computational Biology Series; Chapman & Hall/CRC, Taylor & Francis Group: Boca Raton, FL, 2006.
(3) ADF: Density Functional Theory (DFT) software for chemists. http://www.scm.com (Accessed May 5, 2010).
(4) Brooks, B. R.; Brooks, C. L.; Mackerell, A. D.; Nilsson, L.; Petrella, R. J.; Roux, B.; Won, Y.; Archontis, G.; Bartels, C.; Boresch, S.; Caflisch, A.; Caves, L.; Cui, Q.; Dinner, A. R.; Feig, M.; Fischer, S.; Gao, J.; Hodoscek, M.; Im, W.; Kuczera, K.; Lazaridis, T.; Ma, J.; Ovchinnikov, V.; Paci, E.; Pastor, R. W.; Post, C. B.; Pu, J. Z.; Schaefer, M.; Tidor, B.; Venable, R. M.; Woodcock, H. L.; Wu, X.; Yang, W.; York, D. M.; Karplus, M. CHARMM: The biomolecular simulation program. *J. Comput. Chem.* **2009**, *30* (10), 1545–1614.
(5) CPMD consortium page. http://www.cpmd.org (Accessed May 5, 2010).
(6) CP2K project page. http://cp2k.berlios.de (Accessed May 5, 2010).
(7) Frisch, M. J.; Trucks, G. W.; Schlegel, H. B.; Scuseria, G. E.; Robb, M. A.; Cheeseman, J. R.; Montgomery, J. A., Jr.; Vreven, T.; Kudin, K. N.; Burant, J. C.; Millam, J. M.; Iyengar, S. S.; Tomasi, J.; Barone, V.; Mennucci, B.; Cossi, M.; Scalmani, G.; Rega, N.; Petersson, G. A.; Nakatsuji, H.; Hada, M.; Ehara, M.; Toyota, K.; Fukuda, R.; Hasegawa, J.; Ishida, M.; Nakajima, T.; Honda, Y.; Kitao, O.; Nakai, H.; Klene, M.; Li, X.; Knox, J. E.; Hratchian, H. P.; Cross, J. B.; Bakken, V.; Adamo, C.; Jaramillo, J.; Gomperts, R.; Stratmann, R. E.; Yazyev, O.; Austin, A. J.; Cammi, R.; Pomelli, C.; Ochterski, J. W.; Ayala, P. Y.; Morokuma, K.; Voth, G. A.; Salvador, P.; Dannenberg, J. J.; Zakrzewski, V. G.; Dapprich, S.; Daniels, A. D.; Strain, M. C.; Farkas, O.; Malick, D. K.; Rabuck, A. D.; Raghavachari, K.; Foresman, J. B.; Ortiz, J. V.; Cui, Q.; Baboul, A. G.; Clifford, S.; Cioslowski, J.; Stefanov, B. B.; Liu, G.; Liashenko, A.; Piskorz, P.; Komaromi, I.; Martin, R. L.; Fox, D. J.; Keith, T.; Al-Laham, M. A.; Peng, C. Y.; Nanayakkara, A.; Challacombe, M.; Gill, P. M. W.; Johnson, B.; Chen,

W.; Wong, M. W.; Gonzalez, C.; Pople, J. A. *Gaussian 03, Revision C.02*, Gaussian, Inc., Wallingford, CT, 2004.

(8) Shao, Y.; et al. Advances in methods and algorithms in a modern quantum chemistry program package. *Phys. Chem. Chem. Phys.* **2006**, *8*, 3172–3191.

(9) Kresse, G.; Furthmller, J. Efficient iterative schemes for ab initio total-energy calculations using a plane-wave basis set. *Phys. Rev. B* **1996**, *54*, 11169–11186.

(10) Brooks, B. R.; Janezic, D.; Karplus, M. Harmonic analysis of large systems. 1. Methodology. *J. Comput. Chem.* **1995**, *16*, 1522–1542.

(11) Janezic, D.; Brooks, B. R. Harmonic analysis of large systems. 2. Comparison of different protein models. *J. Comput. Chem.* **1995**, *16*, 1543–1553.

(12) Janezic, D.; Venable, R. M.; Brooks, B. R. Harmonic analysis of large systems. 3. Comparison with molecular dynamics. *J. Comput. Chem.* **1995**, *16*, 1554–1566.

(13) Neugebauer, J.; Hess, B. A. Fundamental vibrational frequencies of small polyatomic molecules from density-functional calculations and vibrational perturbation theory. *J. Chem. Phys.* **2003**, *118*, 7215–7225.

(14) Herrmann, C.; Neugebauer, J.; Reiher, M. Finding a needle in a haystack: Direct determination of vibrational signatures in complex systems. *New J. Chem.* **2007**, *31*, 818–831.

(15) Ghysels, A.; Van Speybroeck, V.; Pauwels, E.; Catak, S.; Brooks, B. R.; Van Neck, D.; Waroquier, M. Comparative study of various normal mode analysis techniques based on partial Hessians. *J. Comput. Chem.* **2010**, *31*, 994–1007.

(16) Jacob, C. R.; Reiher, M. Localizing normal modes in large molecules. *J. Chem. Phys.* **2009**, *130*, 15.

(17) Eyring, H. The activated complex in chemical reactions. *J. Chem. Phys.* **1935**, *3*, 107.

(18) Evans, M. G.; Polanyi, M. Some applications of the transition state method to the calculation of reaction velocities, especially in solution. *Trans. Faraday Soc.* **1935**, *31*, 875.

(19) Wynne-Jones, W. F. K.; Eyring, H. J. The absolute rate of reactions in condensed phases. *J. Chem. Phys.* **1935**, *3*, 492.

(20) Laidler, K. J. *Chemical Kinetics*, 3rd ed.; Harper Collins Publishers, Inc.: New York, NY, 1987.

(21) Zerara, M. pyVib Molecular Graphics Program. http://pyvib.sourceforge.net (Accessed May 5, 2010).

(22) Fedorovsky, M. PyVib2, A program for analyzing vibrational motion and vibrational spectra. http://pyvib2.sourceforge.net (Accessed May 5, 2010).

(23) Zheng, J. et al. *POLYRATE*, version 2008; University of Minnesota: Minneapolis, MN, 2008.

(24) Python Programming Language—Official website. http://www.python.org (Accessed May 5, 2010).

(25) Humphrey, W.; Dalke, A.; Schulten, K. VMD—Visual molecular dynamics. *J. Mol. Graphics* **1996**, *14*, 33–38.

(26) Schaftenaar, G.; Noordik, J. H. Molden: A pre- and post-processing program for molecular and electronic structures. *J. Comput.-Aided Mol. Des.* **2000**, *14*, 123–134.

(27) Maragakis, P.; Karplus, M. Large amplitude conformational change in proteins explored with a plastic network model: Adenylate Kinase. *J. Mol. Biol.* **2005**, *352*, 807–822.

(28) Verstraelen, T.; Van Speybroeck, V.; Waroquier, M. ZEOBUILDER: A GUI toolkit for the construction of complex molecular structures on the nanoscale with building blocks. *J. Chem. Inf. Model* **2008**, *48*, 1530–1541.

(29) Verstraelen, T.; Van Houteghem, M.; Van Speybroeck, V.; Waroquier, M. MD-TRACKS: A productive solution for the advanced analysis of molecular dynamics and Monte Carlo simulations. *J. Chem. Inf. Model* **2008**, *48*, 2414–2424.

(30) Ghysels, A.; Van Neck, D.; Van Speybroeck, V.; Verstraelen, T.; Waroquier, M. Vibrational modes in partially optimized molecular systems. *J. Chem. Phys.* **2007**, *126*, 224102.

(31) Ghysels, A.; Van Speybroeck, V.; Pauwels, E.; Van Neck, D.; Brooks, B. R.; Waroquier, M. Mobile Block Hessian approach with adjoined blocks: an efficient approach for the calculation of frequencies in macromolecules. *J. Chem. Theory Comput.* **2009**, *5*, 1203–1215.

(32) Durand, P.; Trinquier, G.; Sanejouand, Y. H. New approach for determining low-frequency normal-modes in macromolecules. *Biopolymers* **1994**, *34*, 759–771.

(33) Tama, F.; Gadea, F. X.; Marques, O.; Sanejouand, Y. H. Building-block approach for determining low-frequency normal modes of macromolecules Proteins. *Struct. Funct. Genet.* **2000**, *41*, 1–7.

(34) Jin, S. Q.; Head, J. D. Theoretical investigation of molecular water-adsorption on the Al(111). *Surf. Sci.* **1994**, *318*, 204–216.

(35) Calvin, M. D.; Head, J. D.; Jin, S. Q. Theoretically modelling the water bilayer on the Al(111) surface using cluster calculations. *Surf. Sci.* **1996**, *345*, 161–172.

(36) Head, J. D. Computation of vibrational frequencies for adsorbates on surfaces. *Int. J. Quantum Chem.* **1997**, *65*, 827–838.

(37) Woodcock, H. L.; Zheng, W. J.; Ghysels, A.; Shao, Y. H.; Kong, J.; Brooks, B. R. Vibrational subsystem analysis: A method for probing free energies and correlations in the harmonic limit. *J. Chem. Phys.* **2008**, *129*, 214109.

(38) Tirion, M. M. Large amplitude elastic motions in proteins from a single-parameter, atomic analysis. *Phys. Rev. Lett.* **1996**, *77*, 1905–1908.

(39) Bahar, I.; Atilgan, A. R.; Erman, B. Direct evaluation of thermal fluctuations in proteins using a single-parameter harmonic potential. *Fold. Des.* **1997**, *2*, 173–181.

(40) Janezic, D.; Praprotnik, M.; Merzel, F. Molecular dynamics integration and molecular vibrational theory. I. New symplectic integrators. *J. Chem. Phys.* **2005**, *122*, 174101.

(41) Praprotnik, M.; Janezic, D. Molecular dynamics integration and molecular vibrational theory. II. Simulation of nonlinear molecules. *J. Chem. Phys.* **2005**, *122*, 174102.

(42) Praprotnik, M.; Janezic, D. Molecular dynamics integration and molecular vibrational theory. III. The infrared spectrum of water. *J. Chem. Phys.* **2005**, *122*, 174103.

(43) Praprotnik, M.; Janezic, D. Molecular dynamics integration meets standard theory of molecular vibrations. *J. Chem. Inf. Model.* **2005**, *45*, 1571–1579.

(44) Ghysels, A.; Van Neck, D.; Waroquier, M. Cartesian formulation of the Mobile Block Hessian approach to vibrational analysis in partially optimized systems. *J. Chem. Phys.* **2007**, *127*, 164108.

(45) Ghysels, A.; Van Speybroeck, V.; Verstraelen, T.; Van Neck, D.; Waroquier, M. Calculating reaction rates with partial Hessians: Validation of the mobile block Hessian approach. *J. Chem. Theory Comput.* **2008**, *4*, 614–625.

(46) Van Speybroeck, V.; Van Neck, D.; Waroquier, M. Ab initio study of radical addition reactions: addition of a primary ethylbenzene radical to ethene (I). *J. Phys. Chem. A* **2000**, *104*, 10939–10950.

(47) Van Speybroeck, V.; Van Neck, D.; Waroquier, M. Ab initio study of radical reactions: role of coupled internal rotations on the reaction kinetics (III). *J. Phys. Chem. A* **2002**, *106*, 8945–8950.

(48) Vansteenkiste, P.; Van Speybroeck, V.; Marin, G. B.; Waroquier, M. Ab initio calculation of entropy and heat capacity of gas-phase *n*-alkanes using internal rotations. *J. Phys. Chem. A* **2003**, *107*, 3139–3145.

(49) Van Speybroeck, V.; Vansteenkiste, P.; Van Neck, D.; Waroquier, M. Why does the uncoupled hindered rotor model work well for the thermodynamics of *n*-alkanes? *Chem. Phys. Lett.* **2005**, *402*, 479–484.

(50) Vansteenkiste, P.; Verstraelen, T.; Van Speybroeck, V.; Waroquier, M. Ab initio calculation of entropy and heat capacity of gas-phase *n*-alkanes with hetero-elements O and S: Ethers/alcohols and sulfides/thiols. *Chem. Phys.* **2006**, *328*, 251–258.

(51) Wigner, E. P. *Z. Phys. Chem.* **1932**, *B19*, 203.

(52) Eckart, C. The penetration of a potential barrier by electrons. *Phys. Rev.* **1930**, *35*, 1303–1309.

(53) Miyashita, O.; Onuchic, J. N.; Wolynes, P. G. Nonlinear elasticity, proteinquakes, and the energy landscapes of functional transitions in proteins. *Proc. Natl. Acad. Sci. U. S. A.* **2003**, *100*, 12570–12575.

(54) Beckstein, O.; Denning, E. J.; Perilla, J. R.; Woolf, T. B. Zipping and unzipping of adenylate kinase: atomistic insights into the ensemble of open−closed transitions. *J. Mol. Biol.* **2009**, *394*, 160–176.

(55) Hinsen, K. Analysis of domain motions by approximate normal mode calculations. *Prot. Struct. Funct. Genet.* **1998**, *33*, 417–429.

(56) Carrington, B. J.; Mancera, R. L. Comparative estimation of vibrational entropy changes in proteins through normal mode analysis. *J. Mol. Graphics Modell.* **2004**, *23*, 167–174.

(57) Elber, R.; Karplus, M. Multiple conformational states of proteins—A molecular-dynamics analysis of myoglobin. *Science* **1987**, *235*, 318–321.

(58) Frauenfelder, H.; Parak, F.; Young, R. D. Conformational substates in proteins. *Annu. Rev. Biophys. Biophys. Chem.* **1988**, *17*, 451–479.

(59) Kitao, A.; Hayward, S.; Go, N. Energy landscape of a native protein: Jumping-among-minima model. *Prot. Struct. Funct. Genet.* **1998**, *33*, 496–517.

(60) Knyazev, V. D.; Slagle, I. R. Unimolecular decomposition of *n*-C_4H_9 and *iso*-C_4H_9 radicals. *J. Phys. Chem.* **1996**, *100*, 5318–5328.

(61) Beuermann, S. Rate coefficients of free-radical polymerization deduced from pulsed laser experiments. *Prog. Polym. Sci.* **2002**, *27*, 191–254.

(62) Saeys, M.; Reyniers, M.-F.; Marin, G. B.; Van Speybroeck, V.; Waroquier, M. Ab initio group contribution method for activation energies for radical additions. *AIChE J.* **2004**, *50*, 426–444.

(63) Van Speybroeck, V.; Van Cauter, K.; Coussens, B.; Waroquier, M. Ab initio study of free-radical polymerizations: Cost-effective

methods to determine the reaction rates. *ChemPhysChem* **2005**, *6*, 180–189.
(64) Van Cauter, K.; Van Speybroeck, V.; Vansteenkiste, P.; Reyniers, M. F.; Waroquier, M. Ab initio study of free-radical polymerization: Polyethylene propagation kinetics. *ChemPhysChem* **2006**, *7*, 131–140.
(65) Grimme, S. Accurate description of van der Waals complexes by density functional theory including empirical corrections. *J. Comput. Chem.* **2004**, *25*, 1463–1473.
(66) Schwabe, T.; Grimme, S. Double-hybrid density functionals with long-range dispersion corrections: Higher accuracy and extended applicability. *Phys. Chem. Chem. Phys.* **2007**, *9*, 3397–3406.
(67) Zhao, Y.; Truhlar, D. G. The M06 suite of density functionals for main group thermochemistry, thermochemical kinetics, noncovalent interactions, excited states, and transition elements: two new functionals and systematic testing of four M06-class functionals and 12 other functionals. *Theor. Chem. Acc.* **2008**, *120*, 215–241.
(68) Izgorodina, E. I.; Brittain, D. R. B.; Hodgson, J. L.; Krenske, E. H.; Lin, C. Y.; Namazian, M.; Coote, M. L. Should contemporary density functional theory methods be used to study the thermodynamics of radical reactions? *J. Phys. Chem. A* **2007**, *111*, 10754–10768.
(69) Hemelsoet, K.; Moran, D.; Van Speybroeck, V.; Waroquier, M.; Radom, L. An assessment of theoretical procedures for predicting the thermochemistry and kinetics of hydrogen abstraction by methyl radical from benzene. *J. Phys. Chem. A* **2006**, *110*, 8942–8951.
(70) Hemelsoet, K.; Van Speybroeck, V.; Waroquier, M. Bond dissociation enthalpies of large aromatic carbon-centered radicals. *J. Phys. Chem. A* **2008**, *112*, 13566–13573.
(71) Menon, A. S.; Wood, G. P. F.; Moran, D.; Radom, L. Bond dissociation energies and radical stabilization energies: An assessment of contemporary theoretical procedures. *J. Phys. Chem. A* **2007**, *111*, 13638–13644.
(72) Wodrich, M. D.; Corminboeuf, C.; Schreiner, P. R.; Fokin, A. A.; Schleyer, P. V. How accurate are DFT treatments of organic energies. *Org. Lett.* **2007**, *9*, 1851–1854.
(73) Karton, A.; Tarnopolsky, A.; Lamere, J. F.; Schatz, G. C.; Martin, J. M. L. Highly accurate first-principles benchmark data sets for the parametrization and validation of density functional and other approximate methods. Derivation of a robust, generally applicable, double-hybrid functional for thermochemistry and thermochemical kinetics. *J. Phys. Chem. A* **2008**, *112*, 12868–12886.
(74) Korth, M.; Grimme, S. "Mindless" DFT benchmarking. *J. Chem. Theory Comput.* **2009**, *5*, 993–1003.
(75) Vansteenkiste, P.; Van Neck, D.; Van Speybroeck, V.; Waroquier, M. An extended hindered-rotor model with incorporation of Coriolis and vibrational-rotational coupling for calculation partition functions and derived quantities. *J. Chem. Phys.* **2006**, *124*, 044314.
(76) Press, W. H.; Teukolsky, S. A.; Vetterling, W. T.; Flannery, B. P. *Numerical Recipes in C—The Art of Scientific Computing*, 2nd ed.; Press Syndicate of the University of Cambridge: Cambridge, U. K., 1992; Chapter 15.
(77) Pfaendtner, J.; Yu, X.; Broadbelt, L. J. The 1-D hindered rotor approximation. *Theor. Chem. Acc.* **2007**, *118*, 881–898.
(78) Ghysels, A.; Van Neck, D.; Van Speybroeck, V.; Brooks, B. R.; Waroquier, M. Normal modes for large molecules with arbitrary link constraints in the Mobile Block Hessian approach. *J. Chem. Phys.* **2009**, *130*, 084107.
(79) Hariharan, P. C.; Pople, J. A. The influence of polarization functions on molecular orbital hydrogenation energies. *Theor. Chim. Acta* **1973**, *28*, 213–222.
(80) Francl, M. M.; Pietro, W. J.; Hehre, W. J.; Binkley, J. S.; Gordon, M. S.; DeFrees, D. J.; Pople, J. A. Self-consistent molecular orbital methods. XXIII. A polarization-type basis set for second-row elements. *J. Chem. Phys.* **1982**, *77*, 3654–3665.
(81) Krishnan, R.; Binkley, J. S.; Seeger, R.; Pople, J. A. Self-consistent molecular orbital methods. XX. A basis set for correlated wave functions. *J. Chem. Phys.* **1980**, *72*, 650–654.
(82) Frisch, M. J.; Pople, J. A.; Binkley, J. S. Self-consistent molecular orbital methods 25. Supplementary functions for Gaussian basis sets. *J. Chem. Phys.* **1984**, *80*, 3265–3269.
(83) Clark, T.; Chandrasekhar, J.; Spitznagel, G. W., III. The 3-21+G basis set for first-row elements, Li—F. *J. Comput. Chem.* **1983**, *4*, 294–301.
(84) Boys, S. F.; Bernardi, F. The calculation of small molecular interactions by the differences of separate total energies. Some procedures with reduced errors. *Mol. Phys.* **1970**, *19*, 553–566.
(85) Heuts, J. P. A.; Gilbert, R. G.; Radom, L. A priori prediction of propagation rate coefficients in free-radical polymerizations: Propagation of ethylene. *Macromolecules* **1995**, *28*, 8771–8781.
(86) Sabbe, M. K.; Vandeputte, A. G.; Reyniers, M.-F.; Van Speybroeck, V.; Waroquier, M.; Marin, G. B. Ab initio thermochemistry and kinetics for carbon-centered radical addition and β-scission reactions. *J. Phys. Chem. A* **2007**, *111*, 8416–8428.

CI100099G