

Janus: An Extensible Open-Source Software Package for Adaptive QM/MM Methods

Boyi Zhang,[†] Doaa Altarawy,^{‡,¶} Taylor Barnes,[‡] Justin M. Turney,[†] and Henry
F. Schaefer III^{*,†}

[†]*Center for Computational Quantum Chemistry, University of Georgia, Athens, GA, 30602*

[‡]*The Molecular Sciences Software Institute, Virginia Tech, Blacksburg, VA 24060*

[¶]*Department of Computer and Systems Engineering, Alexandria University, Alexandria,
Egypt*

E-mail: ccq@uga.edu

Abstract

Adaptive quantum mechanics/molecular mechanics (QM/MM) approaches are able to treat systems with dynamic or non-localized active centers by allowing for on-the-fly reassignment of the QM region. Although these approaches have been in active development, the inaccessibility of current software has caused slow adoption and limited applications. JANUS seeks to remedy the limitations of current software by providing a free and open-source Python library for adaptive methods that is modular and extensible. Our software has implementations of many existing adaptive methods and a user-friendly input structure that removes the hindrance of complicated set-up procedures. A Python API is made available to customize JANUS’s capabilities and implement novel adaptive approaches. JANUS currently interfaces with PSI4 and OPENMM, but its modular infrastructure enables easy extensibility to other molecular codes without major modifications to either code. The soft-

ware is freely available at <https://github.com/CCQC/janus>. Our goal is that JANUS will serve as a user-driven platform for adaptive QM/MM methods.

Introduction

Background

The use of combined quantum mechanical and molecular mechanical (QM/MM) methods to treat problems otherwise intractable by either QM or MM approaches alone has long been established.^{1,2} However, traditional QM/MM methods are only appropriate for systems with fixed QM regions. For systems with non-localized active centers, such as ion transport and solvent diffusion, an adaptive QM/MM approach that allows on-the-fly reassignment of the QM region during a molecular dynamics (MD) simulation should be used.^{3,4} These adaptive methods have been in active development, and utilized mostly in studies on the solvation properties of ions⁵⁻¹⁴ and select organic reactions.^{15,16} Recent achievements, such as studying the exchange of molecules in protein binding sites,¹⁷ the tracking of proton hopping in bulk water,^{18,19} the determination of explicit solvation effects on nucleophilic addition to carbonyl carbons,²⁰ and proton transfer in a protein channel,²¹ demonstrate the ability of adaptive QM/MM to address problems previously unreachable by traditional QM/MM.

The smoothing of energy and forces between steps in an MD simulation is a central problem in adaptive QM/MM. In order to have a dynamic QM region, an atom’s designation to be treated as a QM or MM particle can change between one step of a MD simulation and the next. This may cause a discontinuity in the energy and forces between the two steps that must be alleviated. In most adaptive QM/MM algorithms, this smoothing is achieved by defining a buffer zone between the QM and MM regions; the particles that fall within the buffer zone will be referred to as buffer groups. Various QM/MM configurations are determined from a method-dependent partitioning of each atom in the buffer group as either a QM or MM particle. A smoothing function is then applied to interpolate the various

QM/MM partitions. Buffer groups effectively have dual QM and MM character, enabling smoothing between time steps.

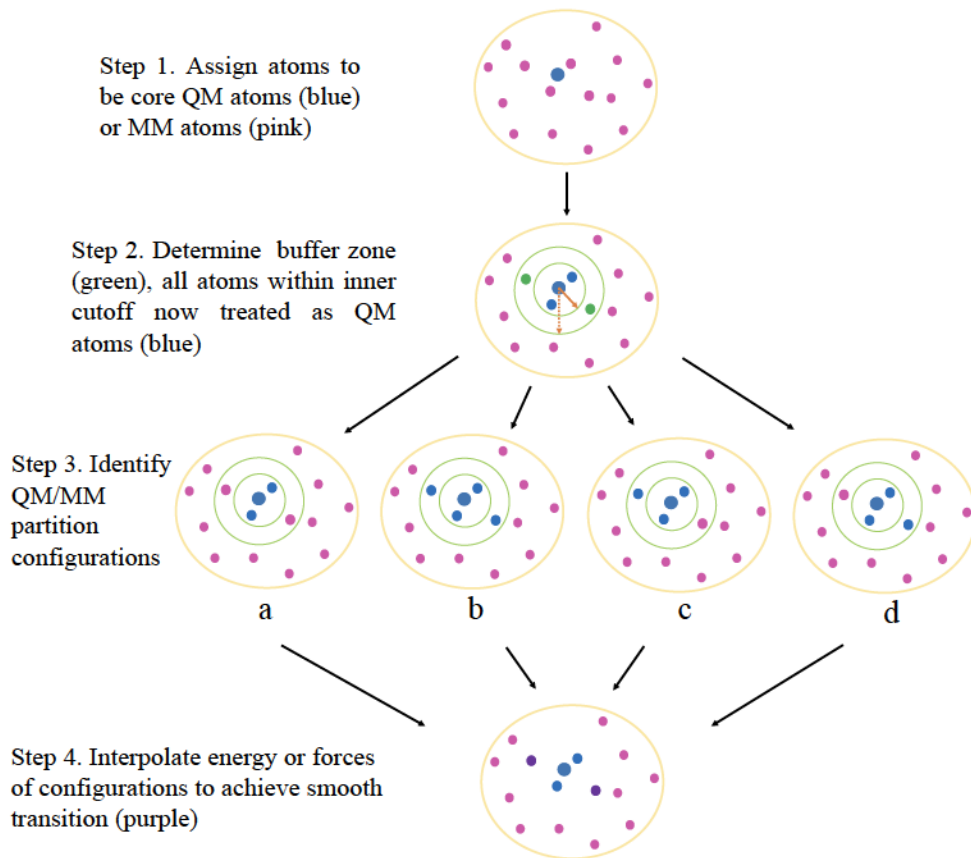


Figure 1: An example of a Permututed Adaptive Partitioning (PAP) scheme for a buffer zone with two groups.

Figure 1 shows an illustrative example of this process, and can be used to further understand the various parts of an adaptive QM/MM approach that need to be considered. Each step of Figure 1 is described below:

Step 1: Initial QM/MM designation. For a given system, atoms can be initially designated as a core QM atoms (blue) or a MM atom (pink). The core QM atoms will always be treated as QM throughout the course of the simulation, and are effectively what is being tracked or followed.

Step 2: **Determination of buffer groups.** Various approaches have been proposed for how to define the buffer zone; these include distance-,⁵ number-,^{22,23} and density-based²⁴ partitioning schemes. The distance-based scheme shown in Figure 1 is the most common, and involves the user defining a R_{min} (solid orange arrow) and R_{max} (dotted orange arrow) calculated from the core QM atoms to serve as the lower and upper radial boundary for the buffer zone. Everything within R_{min} now become designated as a QM atom (blue), while everything between R_{min} and R_{max} is a buffer group (green).

Step 3: **Identification of QM/MM configurations.** The groups within the buffer zone can be treated as either QM or MM, and QM/MM configurations may be identified based on this method-dependent decision. Figure 1 shows the scheme used by the Permuted Adaptive Partitioning (PAP) method, in which all combinations of how to assign buffer group identities are considered. For the two buffer groups in question, we can designate them to be: *a.* both MM atoms; *b.* both QM atoms; *c.* 1 QM atom and 1 MM atom; and *d.* 1 MM atom and 1 QM atom. A separate QM/MM computation is performed on the four resulting QM/MM configurations.

Step 4: **Interpolation scheme.** The separate QM/MM configurations can then be combined as a linear combination of either energies or forces with weights determined by the smoothing function. The smoothing function depends on the relative distances of the buffer groups to the QM center; if a buffer group is close to the QM region then the QM/MM configuration in which that buffer group is treated with QM is given more weight than the QM/MM configuration in which it is treated with MM.

Algorithms with energy-interpolation [e.g., ONIOM-XS,²⁵ DAS,²⁶ and the adaptive partitioning (AP) family of methods^{18,27,28}] conserve energy and momentum for the most part, but the presence of the smoothing function in the energy expression gives rise to extraneous forces due to the gradient of the smoothing function. Algorithms with force-interpolation

(e.g., Buffered Force,²⁹ Hot-Spot,⁷ and Size-Consistent Multipartitioning³⁰) often do not conserve energy or have a meaningful energy expression, but there are no extraneous forces.³ A number of reviews has been written on adaptive QM/MM methods, and we refer the interested reader to these for a more detailed discussion of these approaches.^{3,4,31,32}

Table 1: Time evolution of adaptive approaches and their availability in various software packages

Year ^a	Method	Software Package
1996	Hot-Spot ^{5,7,8,33}	FlexMD, ^{34,35} QMMM, ³⁶ Gaussian, ³⁷ Janus
2002	ONIOM-XS ²⁵	FlexMD, QMMM, Janus
2007	Permuted Adaptive Partitioning (PAP) ^{17,18,27,28,38}	QMMM, Janus
2007	Sorted Adaptive Partitioning (SAP) ^{17,18,27,28}	QMMM, Janus
2009	Difference-based Adaptive Solvation (DAS) ²⁶	FlexMD, Janus
2012	Buffered Force (BF) ^{29,39}	Flex-MD, CP2K, ⁴⁰ Amber, ⁴¹ Janus
2012	Number-Adaptive ²²	AG-IF ⁴²
2014	Density-Based Adaptive (DBA) ^{24,43}	Yoink ⁴⁴
2014	Size-Consistent Multipartitioning (SCMP), ^{30,45}	GROMACS, ⁴⁶ <i>b</i>
2015	Time-Adaptive ⁴⁷	CPMD ⁴⁸
2016	Hamiltonian Adaptive Many-Body Correction (HAMBC) ⁴⁹	FlexMD, <i>b</i>
2017	Scaled Interaction Single Partition Adaptive (SISPA) ⁵⁰	pDynamo ⁵¹

a. Methods are dated by the year in which they were first published, and subsequent modifications are cited.

b. Method will be implemented in Janus in the near future.

Current limitations to adaptive QM/MM

The development of new adaptive QM/MM methods is rapidly gaining traction, as more than half of all available methods were developed within the last six years.⁴ Table 1 shows current methods as well as the software packages in which they are available. Despite this progress, the number of published studies using adaptive QM/MM methods is far fewer than one would expect, considering the number of different algorithms available. There are many untapped areas of potential research that could benefit from the potential insights gained with an adaptive approach. The primary cause for the lack of usage stems from the limitations of existing software,^{4,32} which include:

1. **Restrictive licensing.** Most current software packages that have adaptive QM/MM capabilities offer restrictive licensing and/or is not free to use. Moreover, the QM or

MM codes with which QM/MM software packages interface may also have restrictive licensing and/or are not free to use. This causes a barrier to researchers.

2. **Lack of comprehensive method selection** Some codes used for traditional QM/MM, such as ChemShell,^{52,53} do not have adaptive QM/MM capabilities; others fail to offer a comprehensive selection of all available methods found in Table 1. Comparisons between selected methods require the use of different packages, which may all have their own approach to implementation. Reproducibility of results using various codes is thus difficult to achieve, and the advantages and disadvantages of energy or force-based approaches are still debated.
3. **Developer-focused software.** Most implementations are developer-focused, requiring additional programming skills (FLEX-MD,^{34,35} PDYNAMO⁵¹) or complicated system set-ups (QMMM³⁶). Such need of this prior knowledge is a significant hinderance to users. In addition, the actual implementation details are not always made clear by developers. As a result, existing methods have not been used widely enough to test their robustness and performance, and newer methods have seen little application outside of benchmarking on small systems.^{4,14,49} For example, the first application of the SCMP method was published just this year¹⁴ and the HAMBC method has not been used outside of a benchmarking context. Increased usage will allow for assessment of current methods and lead to the development of a standard adaptive QM/MM methodology.

The inaccessibility of adaptive methods has been the primary barrier to their adoption; as such, there have been repeated calls in the literature for user-friendly adaptive QM/MM software.^{4,31,32} The goals of such software would be as follows:

1. open-source code that interfaces with other open-source packages for easy access and community contribution;

2. user focused code with detailed documentation and tutorials for easy setup, with no prior programming knowledge required;
3. importable capabilities for flexible use by developers and advanced users;
4. the availability of most or all adaptive methods in one place for maximum choice and testing.

Several open-source programs exist for adaptive methods. The PDYNAMO program of Field⁵¹ is a Python-based library for general QM/MM which can be used to implement adaptive methods. However, knowledge of the Python programming language is needed to run this program and adaptive methods are not readily available. Q-REG⁵⁴ is a library written in C++ designed for running adaptive QM/MM specifically for electrochemistry. Standard adaptive methods are not implemented and would require extensive programming knowledge on the user’s part. Furthermore, neither PDYNAMO nor Q-REG currently interface with open-source codes for QM and MM computations.

Herein, we introduce JANUS, an open-source Python library that seeks to remedy the limitations of current programs by providing a unified platform for adaptive methods. (The name “Janus” is a reference to the Roman god of transition and duality, and has been used in the literature to describe QM/MM boundary atoms.⁵⁵) JANUS lowers the barrier to using adaptive QM/MM methods by eliminating prior programming knowledge and providing a simple set-up scheme. For more advanced users and developers, the full set of the software’s capabilities is available through a Python API to provide flexibility in usage. The modular design makes it possible for easy expansion of the code: only minimum modification to the code is required for implementing new methods and adding new interfaces. The key features of JANUS, as well as some illustrative benchmarking results are discussed in the following sections.

Software Features

Ease of use

```
system:
  system_info: water.pdb
  run_aqmmm: True
  run_md: True
  aqmmm_scheme: ONIOM-XS
  ll_program: OpenMM
  hl_program: Psi4

aqmmm:
  Rmin: 3.5
  Rmax: 4.0
  qm_center: [0,1,2]

md:
  start_qmmm: 30000
  end_qmmm: 70000
  md_steps: 70000
  md_ensemble: NVE
  step_size: 0.5

ll:
  rigid_water: False
  fric_coeff: 91
  temp: 298

hl:
  method: scf
  basis: cc-pvdz
```

Figure 2: An example of a minimal input file for running JANUS. The presented input tells the software to run adaptive QM/MM (“run_aqmmm”) with an ONIOM-XS scheme (“aqmmm_scheme”), using PSI4 for the QM computations (“hl_program”) and OPENMM for the MM computations (“ll_program”) and MD time-step integration (“run_md”). The QM center (“qm_center”) is defined as the atoms with indices 0, 1, and 2. 30000 MD steps with pure MM will run, after which the adaptive QM/MM computation will start (“start_qmmm”). 40000 steps will be taken with an adaptive QM/MM computation at every step (“end_qmmm”), resulting a total of 70000 MD steps (“md_steps”). The microcanonical (NVE) ensemble (“md_ensemble”) is specified. Additional MM (“ll”) and QM (“hl”) parameters are also given. No reinitialization is required.

JANUS does not require programming knowledge but instead supports input file submission through a simple command line interface (CLI). Figure 2 shows an example of a minimal input file that runs an adaptive QM/MM/MD simulation. The input has a YAML format with separate sections that specify the job instructions (“system”), any adaptive QM/MM

specific keywords (“aqmmm”), any MD simulation specific keywords (“md”), as well as any parameters specifically related to the QM (“h1”) or MM (“l1”) computations. Although there are many potential keywords for each section, there are sensible defaults for nearly all of them. Thus, the user only has to change the keywords they desire by including them in the appropriate section, resulting in a very simple input. Instructions on installation, how to set up an input file and run the program, along with the keywords of each section and their descriptions are provided in the manual (<https://ccqc.github.io/janus/>) to ensure that the learning barrier is as low as possible. For more advanced users, JANUS is available as an importable Python library.

JANUS offers a comprehensive selection of methods (see Table 1) for the user to choose from. Because the different types of adaptive QM/MM approaches can all be run in the same place, the methods can be compared on equal footing. This will allow for consistent benchmarking across different approaches, as well as easy comparisons between them to establish general adaptive procedures. Information on each method is provided in the manual, so the user can make an informed choice that is appropriate for their specific system.

Rapid testing using an API

In addition to input file submission, JANUS provides its own Python API as an alternative way to interact with the software, which is useful both to developers and general users. For developers, the API makes JANUS’s capabilities both accessible and customizable. Many of JANUS’s functionalities can be used independently as a starting point for new approaches. In addition, modifications to a currently existing method in JANUS can quickly be tested.

The following section of code is an example of how a general user might use the API:

```
from janus import qm_wrapper, mm_wrapper, qmmm

# instantiate a Psi4Wrapper object as the high level wrapper
hl_wrapper = qm_wrapper.Psi4Wrapper()

# instantiate an OpenMMWrapper object as the low level wrapper
```

```

ll_wrapper = mm_wrapper.OpenMMWrapper(sys_info="water.pdb")

# instantiate a Permuted Adaptive Partitioning (PAP) object,
# varying Rmin and Rmax (Angstroms)
p1 = qmmm.PAP(hl_wrapper, ll_wrapper, sys_info="water.pdb", Rmin=5.0, Rmax=5.5)
p2 = qmmm.PAP(hl_wrapper, ll_wrapper, sys_info="water.pdb", Rmin=5.0, Rmax=6.0)

# partition the QM and buffer zone atoms
p1.find_buffer_zone()
p2.find_buffer_zone()

# find QM/MM configurations that arise from buffer zone partitioning
p1.find_configurations()
p2.find_configurations()

#print Rmin, Rmax, number of qm groups, buffer groups, and QM/MM configurations
print(p1.Rmin, p1.Rmax, p1.n_qm_groups, p1.n_buffer_groups, p1.n_configs)
print(p2.Rmin, p2.Rmax, p2.n_qm_groups, p2.n_buffer_groups, p2.n_configs)

```

In this case, the API is used as a way to test how different values of `Rmin` and `Rmax` (using a distance partitioning scheme) affect the number of buffer groups that are designated and the number of QM/MM configurations. For a given system, it might not be immediately clear to the user what are reasonable `Rmin` and `Rmax` values to set. The value of `Rmin` and `Rmax` directly affect the number of atoms in the QM region and buffer zone, and thus also determine the number of QM/MM configurations that arise (see Figure 1). If `Rmin` is too large, the QM region might be too large and not feasible for some QM methods. If the distance between `Rmin` and `Rmax` is too large, there might be too many buffer groups and thus too many QM/MM configurations. Therefore, it is valuable to easily test different values of `Rmin` and `Rmax` to determine what is appropriate for a specific system before starting a longer simulation job. With just a little programming knowledge, a user can take advantage of the API to run jobs more efficiently.

Minimum interface overhead with external software

Major disadvantages of current interface-based QM/MM packages such as QMMM,³⁶ PDYNAMO,⁵¹ and FLEX-MD,^{34,35} include the need to reinitialize the QM and MM codes at each time-step, as well as the number of file transfers needed for communication between different software. JANUS currently interfaces with OPENMM^{56,57} and PSI4,^{58,59} both of which have their own application program interface (API). By calling PSI4 and OPENMM’s functionalities through APIs, JANUS minimizes the reinitialization of external code and file-transfer mechanisms to achieve more efficient implementations of workflow. In sample computations performed, only a trivial amount of the time is spent as overhead ($< 2\%$). The software also uses MDTRAJ⁶⁰ to assist in intermediate trajectory storing and manipulation. Although not all molecular software packages have an API, there has been an increasing push to develop APIs for established software (e.g. AMBER,⁴¹ NWChem,⁶¹ GROMACS⁴⁶), which JANUS can take advantage of. Furthermore, initiatives from the Molecular Software Sciences Institute (MolSSI), such as the QCEngine package and the MolSSI Driver Interface Project, are encouraging API based interfaces for a variety of molecular software packages.^{62,63} These developments are expected to provide a way for JANUS to interface with more codes without the use of a file-based communication mechanism.

Easy software expansion through modular design

JANUS is designed with a modular approach to allow for easy method implementation and package expansion. Figure 3 shows the relationship between the four main modules in JANUS. The MM wrapper module (green) contains the interface to any molecular mechanics code, while the QM wrapper module (pink) contains the interface to any quantum mechanics code. As Figure 3 shows, each specific QM or MM software will be interfaced through its own subclass within the module. The QM/MM module (blue) includes the QMMM class as well as the AQMMM class, and contains all functionalities for traditional and adaptive QM/MM. Each adaptive method is a separate subclass that has its own set of method-

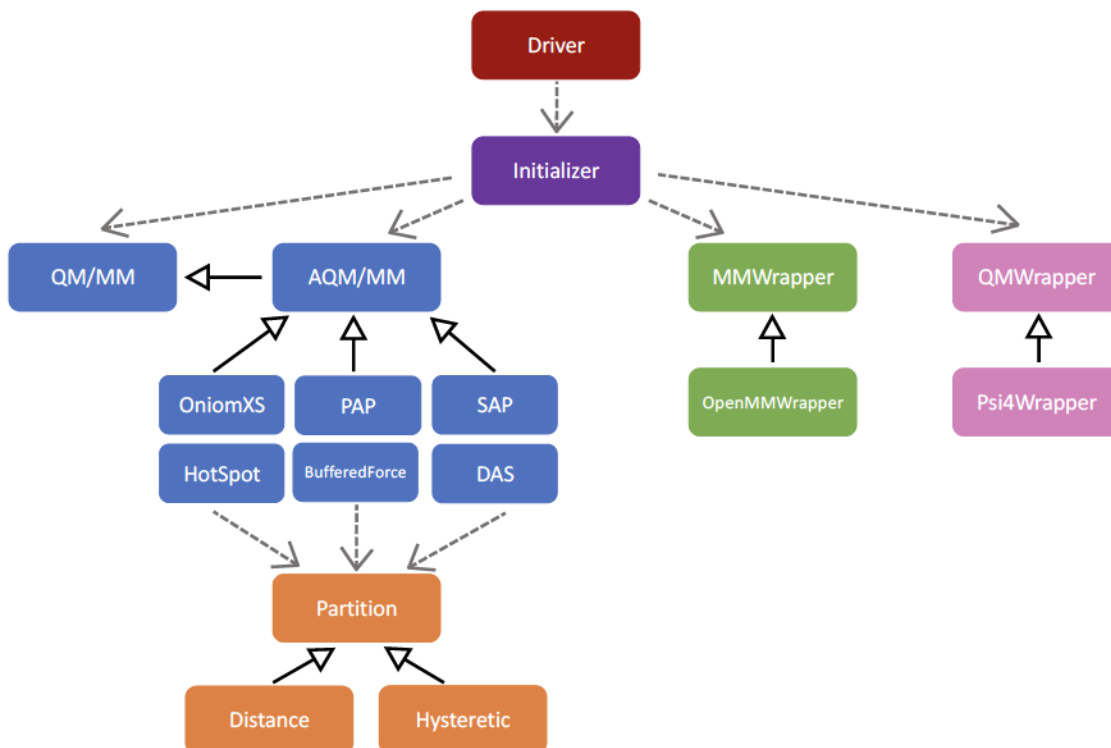


Figure 3: An overview of the class structure of JANUS. Dashed gray arrows from class A to B indicate that A calls functions from B . Solid black arrows from class A to B indicates that A is a subclass of B and inherits from B .

dependent functions in addition to the functions inherited from AQMMM. The Partition module contains schemes for defining the buffer zone, as described in Step 2 of Figure 1. Currently, two schemes are implemented: the distance-based scheme first pioneered by Rode and coworkers⁵ and the hysteretic scheme of Bernstein and coworkers.²⁹ Distance partitioning has been described previously in Figure 1. Hysteretic partitioning still involves defining a radial boundary for a QM and buffer region, but contains an additional set of boundaries (Figure 4). As the name suggests, the scheme uses information from the previous step to temper drastic changes in the number of QM and buffer atoms during a simulation.

The independence of separate modules allows for great flexibility both in application and implementation, and makes JANUS a great tool for testing new adaptive QM/MM approaches. In order to add a new adaptive method, add a new partitioning method, or

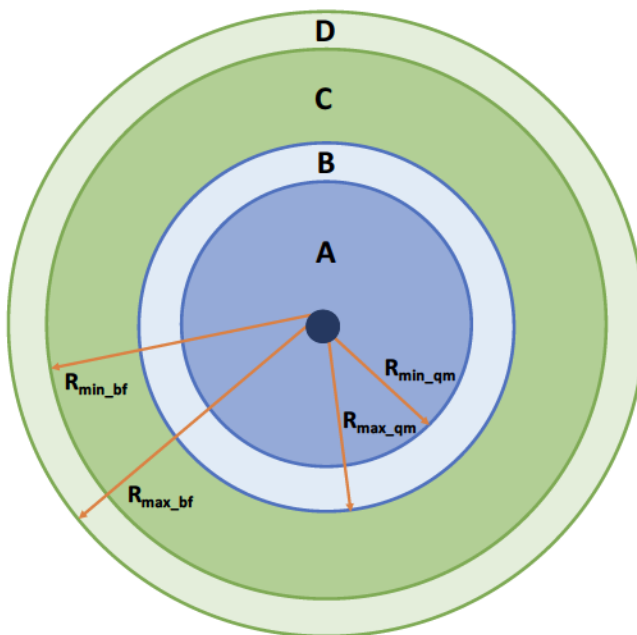


Figure 4: The hysteric partitioning scheme. All molecules in region A are designated as QM molecules. Molecules in region B are designated as QM molecules if they were QM molecules in the previous time-step, and buffer atoms otherwise. Molecules in region C are designated as buffer atoms. Molecules in region D are designated as buffer atoms if they were buffer atoms in the previous time-step, and MM molecules otherwise.

to interface to a new package, one simply has to create a subclass that takes advantage of the existing infrastructure without changing the rest of the code. This allows quick implementation of currently existing methods and relatively easy package expansion with other software. In addition, the separation of the Partition module from the QM/MM module allows users to test different combinations of partitioning schemes and adaptive methods to develop new approaches. For example, adaptive approaches such as ONIOM-XS²⁵ and the adaptive partitioning family (PAP, SAP)²⁷ are traditionally implemented with distance partitioning. However, with the modular implementation of different partitioning schemes one can easily test the use of hysteric partitioning in these methods. Thus, JANUS can be used both to develop new methods as well as test untried combinations of existing approaches.

Sample Applications

A primary application for adaptive QM/MM methods is to study the coordination number of a system in explicit solvent. A radial distribution function (RDF) is often used to quantify this property, and gives the probability of finding a molecule within a certain distance of another molecule. Peaks in the RDF correspond to the solvation shells around the center system.

To demonstrate this usage in JANUS, we performed test simulations on a cluster of 1099 water molecules. The flexible TIP3P⁶⁴ forcefield was used. No cutoff was used for nonbonded interactions. In all cases, the water box was equilibrated for 25 ps in the canonical (NVT) ensemble using a Langevin integrator at a temperature of 298 K, a friction coefficient of 1 ps^{-1} , and a step size of 0.5 fs. A production run in the canonical (NVT) ensemble was then performed. The pure MD simulation ran for 30 ps with a step size of 0.5 fs. For the QM/MM and adaptive QM/MM runs, an MD simulation using molecular mechanics forces only was run for 25 ps in the NVT ensemble before starting the QM/MM or adaptive QM/MM portion for 5 ps. For the QM/MM run, the QM region was set to be one water molecule. For the adaptive schemes, the same water molecule from QM/MM was set to be the QM center. Mechanical embedding was used. While electrostatic embedding is available for traditional QM/MM, electrostatic embedding for adaptive QM/MM is not yet implemented. The QM computations were performed using Hartree–Fock theory (using a RHF reference), along with either the 6-31+G*⁶⁵ or Dunning’s aug-cc-pVDZ⁶⁶ basis set. Other parameters used defaults set in JANUS, and can be found on the JANUS website (<https://ccqc.github.io/janus/>). Distance partitioning was used for adaptive test runs.

All computations were run using JANUS, calling PSI4 for the QM computations and OPENMM for the MM computations and time-step integration. Three separate simulations were run with the conditions above and averaged to obtain the RDF. RDFs were generated with PYTRAJ^{67,68} and used the distance between the central water molecule and the other water molecules. Figure 5 shows the Oc–O, Oc–H, and Hc–H RDF of the water box averaged

over the last 5 ps of the NVT simulation of three separate simulations for each test case. We demonstrate that JANUS not only runs adaptive QM/MM but can also perform traditional QM/MM and MD simulations to use for comparison purposes.

The results obtained are qualitatively in line with experiment and other computational RDFs previously published.⁶⁹ We note that for adaptive runs, a keyword change in the input is all that is necessary to redefine the buffer zone or QM approach.

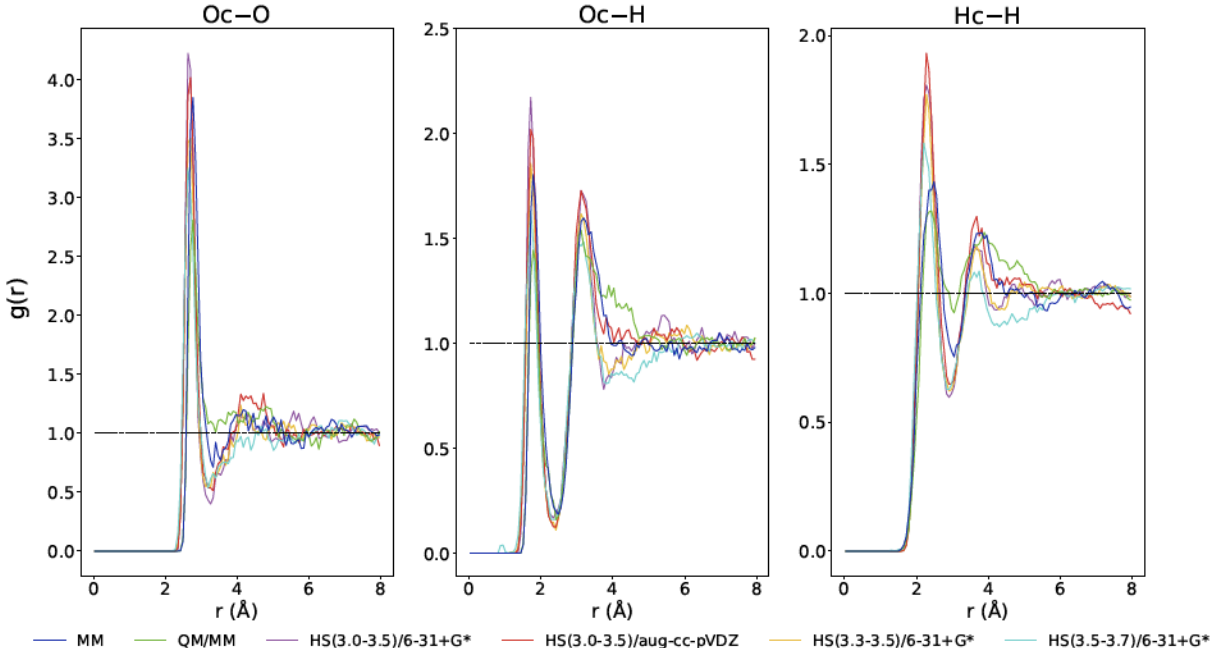


Figure 5: Radial distribution function for water. Oc-O was computed with the oxygen of the central water vs. all other water oxygens. Oc-H was computed with the oxygen of the central water vs. all other water hydrogens. Hc-H was computed with the hydrogens of the central water vs. all other water hydrogens. The blue curve corresponds to a pure MM simulation. The green curve corresponds to a non-adaptive QM/MM simulation. The purple curve corresponds to a Hot-Spot adaptive simulation with the buffer zone defined as $R_{min} = 3.0 \text{ \AA}$ and $R_{max} = 3.5 \text{ \AA}$, and using the 6-31+G* basis set for QM computations. The red curve corresponds to a Hot-Spot adaptive simulation with the buffer zone defined as $R_{min} = 3.0 \text{ \AA}$ and $R_{max} = 3.5 \text{ \AA}$, and using the aug-cc-pVDZ basis set. The yellow curve corresponds to a Hot-Spot adaptive simulation with the buffer zone defined as $R_{min} = 3.3 \text{ \AA}$ and $R_{max} = 3.5 \text{ \AA}$, and using the 6-31+G* basis set. The cyan curve corresponds to a Hot-Spot adaptive simulation with the buffer zone defined as $R_{min} = 3.5 \text{ \AA}$ and $R_{max} = 3.7 \text{ \AA}$, and using the 6-31+G* basis set.

Figure 6 shows how the QM region is changing for a 10 ps NVT simulation of the Hot-

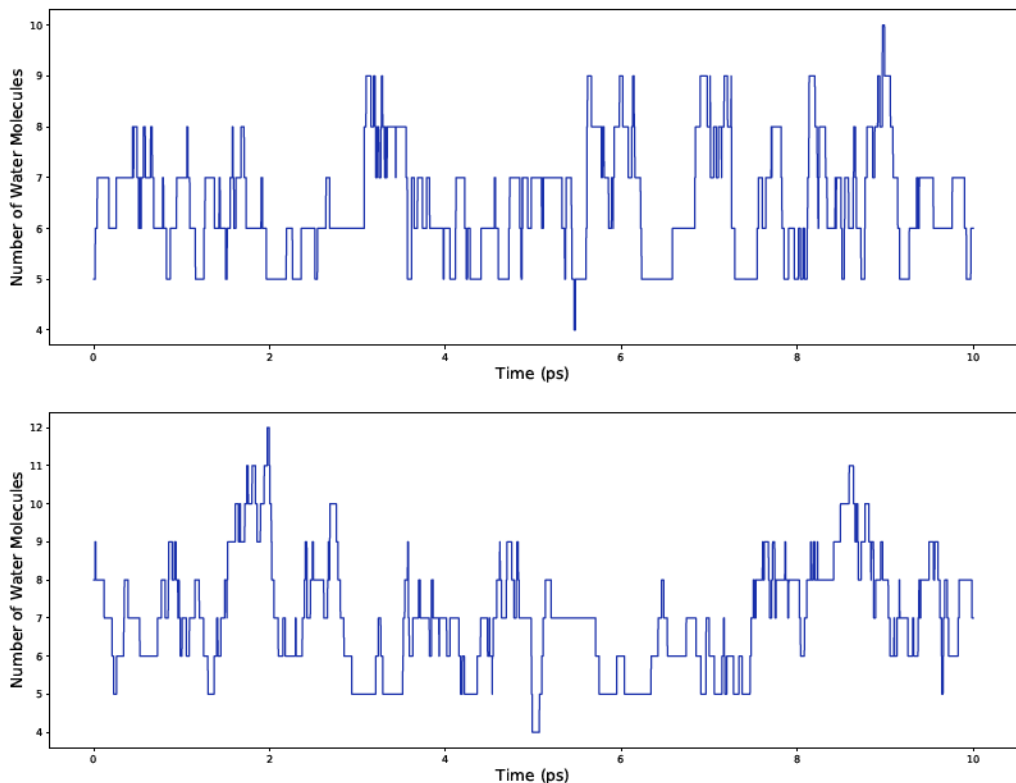


Figure 6: Number of water molecules in the QM region. The top corresponds to a Hot-Spot adaptive simulation with the buffer zone defined as $R_{min} = 3.3 \text{ \AA}$ and $R_{max} = 3.5 \text{ \AA}$, and using the 6-31+G* basis set for QM computations. The bottom corresponds to a Hot-Spot adaptive simulation with the buffer zone defined as $R_{min} = 3.5 \text{ \AA}$ and $R_{max} = 3.7 \text{ \AA}$, and using the 6-31+G* basis set for QM computations.

Spot adaptive approach with the SCF/6-31+G* QM method. The top graph shows the number of water molecules in a QM region (including buffer zone) with a radius of $3.5 \text{ \AA}$. The graph shows the number of water molecules in a QM region (including buffer zone) with a radius of $3.7 \text{ \AA}$. As the figure shows, throughout the simulation the number of waters in the smaller QM region varies from four to ten, while the number of molecules in the bigger QM region varies from four to twelve. Thus we see the adaptive approach correctly capture the movement of the water cluster and update the QM region as needed.

We also used JANUS to study the structure and solvation of N-methylacetamide (NMA) in

a water cluster of 576 water molecules. NMA is the simplest model for a peptide bond and as a result has been subject to numerous experimental and theoretical studies.^{70–82} The flexible TIP3P⁶⁴ forcefield was used for the water molecules and the Amber ff99SB^{83,84} protein forcefield was used for NMA. No cutoff was used for nonbonded interactions. The system was equilibrated for 25 ps in the canonical (NVT) ensemble using a Langevin integrator at a temperature of 300 K, a friction coefficient of 1 ps⁻¹, and a step size of 0.5 fs. A production run in the microcanonical (NVE) ensemble was then performed for 35 ps total. The use of QM/MM forces started after 15 ps. The QM/MM run treated NMA with QM and the rest with MM. Mechanical embedding was used. The QM portion was performed using both Hartree–Fock theory and density-fitted second-order Møller-Plesset (MP2) theory with an RHF reference, along with the STO-3G⁸⁵ and Dunning cc-pVDZ (DZ)⁶⁶ basis sets. For clarity purposes, QM/MM computations will be referenced using the notation [QM method] [QM basis set]/[MM] (e.g., MP2 STO-3G/MM).

Figure 7 shows the labeled NMA molecule (used by Tables 2 and 3 and Figure 8) along with a snapshot of the MP2 DZ/MM simulation. The snapshot reveals two water molecules acting as hydrogen bond donors to the carbonyl oxygen and one water molecule acting as a hydrogen bond acceptor to the amide hydrogen. This is representative of the most common type of hydrogen bonded complex seen for all the simulations. The average numbers of water molecules hydrogen bonded with the carbonyl carbon during the 20 ps simulation are: 1.5 (MM), 1.7 (HF STO-3G/MM), 1.9 (HF DZ/MM), 1.7 (MP2 STO-3G/MM), and 1.7 (MP2 DZ/MM). For the amide hydrogen, the average numbers are: 1.0 (MM), 1.1 (HF STO-3G/MM), 1.0 (HF DZ/MM), 1.1 (MP2 STO-3G/MM), and 1.1 (MP2 DZ/MM). Average bonds were measure by tabulating the number of hydrogen bonds present at each time step and taking the mean. Hydrogen bonds were identified using the Baker–Hubbard⁸⁸ scheme as implemented in MDTRAJ⁶⁰ with the criteria $r_{H...Acceptor} < 2.5 \text{ \AA}$.

Table 2 shows the geometry parameters obtained by our simulations as compared to *ab initio* MD and experiment. We note that although most parameters are comparable with one

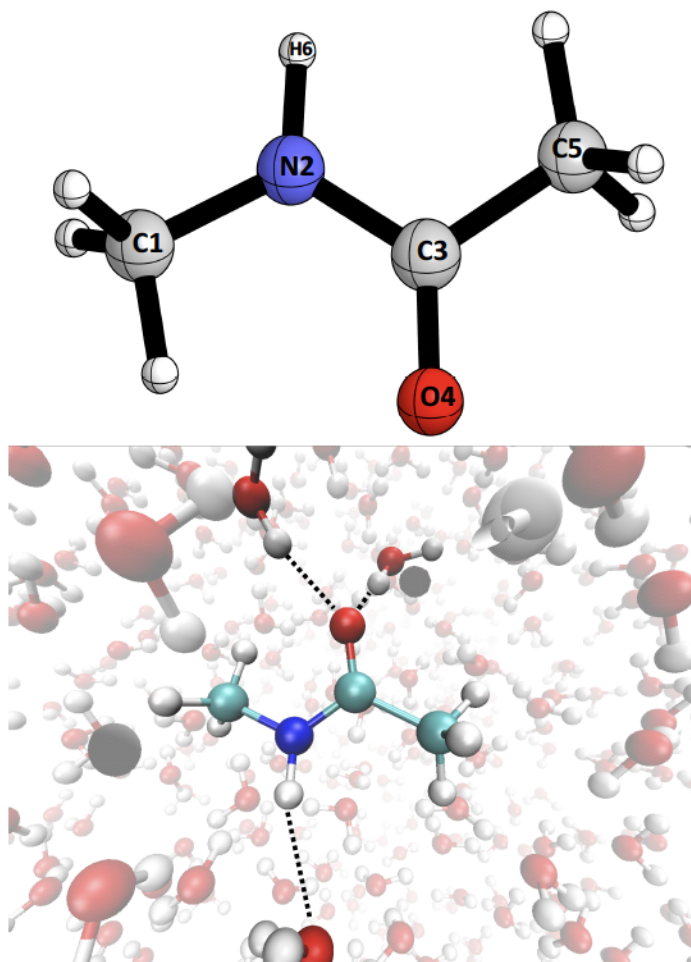


Figure 7: The NMA molecule and a snapshot of a hydrogen-bonded NMA–water complex during the MP2 cc-pVDZ/MM simulation generated with VMD.^{86,87}

another, the $\angle\text{H-N-C-O}$ torsion angle is highly dependent on the basis set. Table 3 shows the hydrogen bonding geometric parameters as compared with other QM/MM results in the literature. The average angles obtained by our results are lower than the ones in previous literature. The large variation in the standard deviation of the angles is partly due to the fact that we are averaging over all the waters that form hydrogen bonding interactions with the carbonyl carbon and amide hydrogen, as discussed above.

Figure 8 shows the RDF of various parts of the NMA molecule and the waters. Results obtained agree with those obtained by previous studies.^{70–74} Traditional^{70,74} and *ab initio* MD⁷¹ results qualitatively match our results for the amide hydrogen and water RDFs as

well as the carbonyl oxygen and water RDFs. Our QM/MM results also qualitatively match other QM/MM studies.^{72,73} The O4-H_{wat} RDF has a more pronounced peak than the one obtained with ABEEM/MM,⁷² but is in agreement with a HF 3-21G/MM study.⁷³

Table 2: Geometry parameters of NMA in a water cluster

	MM	HF STO-3G/MM	HF DZ/MM	MP2 STO-3G/MM	MP2 DZ/MM	Ref 70 ^a	Ref 76 ^b	Ref 73 ^c
Bond Lengths (Å)								
C1-N2	1.462 ± 0.034	1.478 ± 0.026	1.448 ± 0.024	1.5074 ± 0.029	1.4542 ± 0.030	1.477 ± 0.031	1.465(13)	1.458
N2-C3	1.333 ± 0.022	1.442 ± 0.029	1.356 ± 0.024	1.4793 ± 0.034	1.3744 ± 0.026	1.351 ± 0.028	1.290(13)	1.351
C3-C5	1.510 ± 0.031	1.543 ± 0.028	1.512 ± 0.029	1.5666 ± 0.031	1.5185 ± 0.031	1.520 ± 0.035	1.536(16)	1.515
C3-O4	1.230 ± 0.020	1.223 ± 0.020	1.205 ± 0.023	1.2658 ± 0.020	1.2351 ± 0.027	1.268 ± 0.022	1.236(12)	1.243
N2-H6	1.013 ± 0.027	1.036 ± 0.040	1.004 ± 0.028	1.0614 ± 0.030	1.0217 ± 0.029	1.031 ± 0.032		1.010
Bond Angles (°)								
∠C1-N2-C3	124.85 ± 3.23	113.84 ± 3.52	119.22 ± 3.16	110.59 ± 3.61	117.58 ± 3.63		120.5	122.4
∠N2-C3-O4	123.32 ± 2.80	120.38 ± 2.61	121.06 ± 2.44	120.45 ± 2.82	120.76 ± 2.84		123.0	121.7
∠N2-C3-C5	116.70 ± 2.93	115.58 ± 2.91	117.18 ± 2.78	114.57 ± 3.09	116.57 ± 3.11		116.5	116.9
∠O4-C3-C5	119.68 ± 3.02	123.64 ± 2.86	121.60 ± 2.66	124.31 ± 3.03	122.34 ± 3.10		120.5	
∠H6-N2-C3	116.98 ± 3.31	113.67 ± 4.65	119.73 ± 3.82	110.35 ± 5.05	119.68 ± 4.53		119.5	118.9
Torsion Angles (°)								
∠H-N-C-O	172.40 ± 5.74	147.44 ± 12.61	168.31 ± 8.40	141.12 ± 11.67	165.81 ± 9.68	173.98 ± 10.75		180.00
∠C-N-C-C	171.93 ± 6.06	166.28 ± 8.86	171.42 ± 6.16	166.11 ± 8.71	171.01 ± 6.72	178.46 ± 10.70		180.00

Number after ± is the standard deviation.

a. Parameters determined by *ab initio* MD.

b. Parameters determined by X-ray crystal diffraction.

c. Parameters determined by B3LYP/6-31+G(d,p) in the IEF continuum.

Table 3: Hydrogen bond parameters of NMA in a water cluster

	O4...H _{wat} (Å) ^a	O4-H _{wat} -O _{wat} (°)	C3-O4-H _{wat} (°)	H6...O _{wat} (Å)	N2-H _{wat} -O _{wat} (°)
MM	1.933 ± 0.220	156.03 ± 13.28	125.16 ± 19.49	2.141 ± 0.181	153.64 ± 13.36
HF STO-3G/MM	1.932 ± 0.223	156.02 ± 13.49	128.11 ± 18.78	2.134 ± 0.186	153.15 ± 12.98
HF DZ/MM	1.959 ± 0.212	155.29 ± 13.13	124.37 ± 19.61	2.198 ± 0.169	152.32 ± 13.79
MP2 STO-3G/MM	1.930 ± 0.218	157.66 ± 12.30	126.77 ± 19.42	2.173 ± 0.188	149.56 ± 14.01
MP2 DZ/MM	1.919 ± 0.220	157.34 ± 12.64	126.82 ± 18.34	2.195 ± 0.182	150.70 ± 13.86
OPLS ⁷³	1.78		141	1.94	175
AM1/MM ⁷³	1.70		144	1.77	176
HF 3-21G/MM ⁷³	1.99		143	2.10	168
^b B3LYP/6-31+G(d,p) ⁷⁴	1.83			1.90	

Number after ± is the standard deviation.

a. O4-H_{wat} related parameters are averaged over any hydrogen bond interactions with O4 at a given time.

b. Performed in IEF continuum

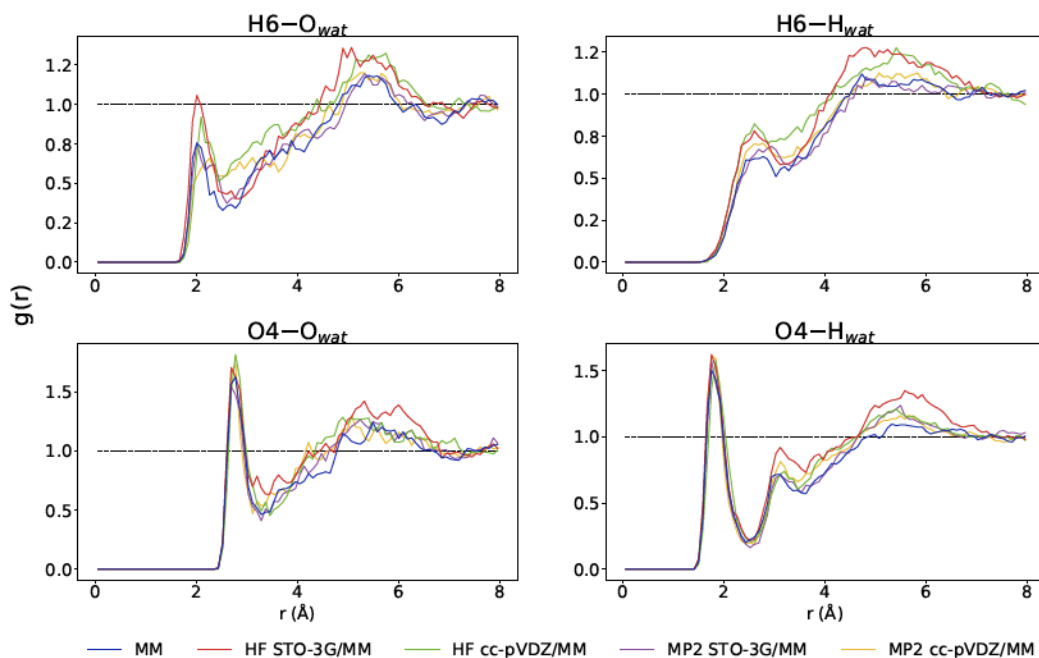


Figure 8: Radial distribution function of NMA and water. The $H6-O_{wat}$ RDF was computed with the amide hydrogen and water oxygen atoms. The $H6-H_{wat}$ RDF was computed with the amide hydrogen and water hydrogen atoms. The $O4-O_{wat}$ RDF was computed with the carbonyl oxygen and water oxygen atoms. The $O4-H_{wat}$ RDF was computed with the carbonyl oxygen and water hydrogen atoms. The blue curve corresponds to a pure MM simulation. The red, green, purple, and yellow curves correspond to QM/MM simulations at various levels of QM theory.

Summary

Several disadvantages exist in currently available software for adaptive QM/MM, including licensing restrictions, limited selection of methods, and developer-focused usage. The growing field of adaptive QM/MM can greatly benefit from a user-friendly platform with efficiently implemented methods, both to explore untested applications and to promote novel, problem-driven approaches. In the present paper, we describe JANUS, an adaptive QM/MM Python library that seeks emphasizes usability and promotes method development. JANUS is freely available to download and install at <https://github.com/CCQC/janus> and is made open-source. Contributions from the community are highly welcome and encouraged, facilitated by rigid automated testing and continuous integration. Simple input file structures and detailed documentation make it accessible to non-developers while advanced users can take advantage of the full capabilities of the Python library. An API is offered to give users great flexibility with how to use the code and allows rapid testing of various adaptive approaches. The modular design of JANUS makes it easy to either implement an existing method or test new methods. In addition, interfaces to other software packages can be added on with little modification to existing code. It is our hope that JANUS will promote the use of adaptive QM/MM methods and that the growth of the software will be sustained and guided by user needs.

Acknowledgements

B.Z. is supported by a fellowship from The Molecular Sciences Software Institute^{62,63} under NSF grant ACI-1547580. J.M.T and H.F.S acknowledge support by the U.S. National Science Foundation under grant CHE-1661604. B.Z. thanks Adam Abbott for careful reading of the manuscript.

References

- (1) Warshel, A.; Karplus, M. Calculation of ground and excited state potential surfaces of conjugated molecules. I. Formulation and parametrization. *J. Am. Chem. Soc.* **1972**, *94*, 5612–5625.
- (2) Warshel, A.; Levitt, M. Theoretical studies of enzymic reactions: Dielectric, electrostatic and steric stabilization of the carbonium ion in the reaction of lysozyme. *J. Mol. Biol.* **1976**, *103*, 227–249.
- (3) Pezeshki, S.; Lin, H. Recent developments in QM/MM methods towards open-boundary multi-scale simulations. *Mol. Sim.* **2015**, *41*, 168–189.
- (4) Duster, A. W.; Wang, C. H.; Garza, C. M.; Miller, D. E.; Lin, H. Adaptive quantum/-molecular mechanics: what have we learned, where are we, and where do we go from here? *Wiley Interdiscip. Rev.: Comput. Mol. Sci.* **2017**, *7*, 1–21.
- (5) Kerdcharoen, T.; Liedl, K. R.; Rode, B. M. A QM/MM simulation method applied to the solution of Li⁺ in liquid ammonia. *Chem. Phys.* **1996**, *211*, 313–323.
- (6) Schwenk, C. F.; Loeffler, H. H.; Rode, B. M. Structure and dynamics of metal ions in solution: QM/MM molecular dynamics simulations of Mn²⁺ and V²⁺. *J. Am. Chem. Soc.* **2003**, *125*, 1618–1624.
- (7) Hofer, T. S.; Pribil, A. B.; Randolph, B. R.; Rode, B. M. Structure and dynamics of solvated Sn(II) in aqueous solution: An ab initio QM/MM MD approach. *J. Am. Chem. Soc.* **2005**, *127*, 14231–14238.
- (8) Rode, B. M.; Hofer, T. S.; Randolph, B. R.; Schwenk, C. F.; Xenides, D.; Vchirawongkwin, V. Ab initio quantum mechanical charge field (QMCF) molecular dynamics: a QM/MM–MD procedure for accurate simulations of ions and complexes. *Theor. Chem. Acc.* **2006**, *115*, 77–85.

- (9) Azam, S. S.; Hofer, T. S.; Randolph, B. R.; Rode, B. M. Hydration of sodium (I) and potassium (I) revisited: a comparative QM/MM and QMCF MD simulation study of weakly hydrated ions. *J. Phys. Chem. A* **2009**, *113*, 1827–1834.
- (10) Rowley, C. N.; Roux, B. The solvation structure of Na^+ and K^+ in liquid water determined from high level ab initio molecular dynamics simulations. *J. Chem. Theory Comput.* **2012**, *8*, 3526–3535.
- (11) Lev, B.; Roux, B.; Noskov, S. Y. Relative free energies for hydration of monovalent ions from QM and QM/MM simulations. *J. Chem. Theory Comput.* **2013**, *9*, 4165–4175.
- (12) Kabbalee, P.; Sripa, P.; Tongraar, A.; Kerdcharoen, T. Solvation structure and dynamics of K^+ in aqueous ammonia solution: Insights from an ONIOM-XS MD simulation. *Chem. Phys. Lett.* **2015**, *633*, 152–157.
- (13) Kabbalee, P.; Tongraar, A.; Kerdcharoen, T. Preferential solvation and dynamics of Li^+ in aqueous ammonia solution: an ONIOM-XS MD simulation study. *Chem. Phys.* **2015**, *446*, 70–75.
- (14) Watanabe, H. C.; Kubillus, M.; Kuba, T.; Stach, R.; Mizaikoff, B.; Ishikita, H. Cation solvation with quantum chemical effects modeled by a size-consistent multi-partitioning quantum mechanics/molecular mechanics method. *Phys. Chem. Chem. Phys.* **2017**, *19*, 17985–17997.
- (15) Park, K.; Götz, A. W.; Walker, R. C.; Paesani, F. Application of adaptive QM/MM methods to molecular dynamics simulations of aqueous systems. *J. Chem. Theory Comput.* **2012**, *8*, 2868–2877.
- (16) Várnai, C.; Bernstein, N.; Mones, L.; Csányi, G. Tests of an adaptive QM/MM calculation on free energy profiles of chemical reactions in solution. *J. Phys. Chem. B* **2013**, *117*, 12202–12211.

- (17) Pezeshki, S.; Davis, C.; Heyden, A.; Lin, H. Adaptive-partitioning QM/MM dynamics simulations: 3. Solvent molecules entering and leaving protein binding sites. *J. Chem. Theory Comput.* **2014**, *10*, 4765–4776.
- (18) Pezeshki, S.; Lin, H. Adaptive-Partitioning QM/MM for Molecular Dynamics Simulations: 4. Proton Hopping in Bulk Water. *J. Chem. Theory Comput.* **2015**, *11*, 2398–2411.
- (19) Jiang, T.; Boereboom, J. M.; Michel, C.; Fleurat-Lessard, P.; Bulo, R. E. In *Quantum Modeling of Complex Molecular Systems*; Rivail, J.-L., Ruiz-Lopez, M., Assfeld, X., Eds.; Springer: Cham, 2015; Vol. 21; pp 51–91.
- (20) Boereboom, J. M.; Fleurat-Lessard, P.; Bulo, R. E. Explicit Solvation Matters: Performance of QM/MM Solvation Models in Nucleophilic Addition. *J. Chem. Theory Comput.* **2018**, *14*, 1841–1852.
- (21) Duster, A. W.; Garza, C. M.; Aydintug, B. O.; Negussie, M. B.; Lin, H. Adaptive Partitioning QM/MM for Molecular Dynamics Simulations: 6. Proton Transport through a Biological Channel. *J. Chem. Theory Comput.* **2019**, DOI: 10.1021/acs.jctc.8b01128.
- (22) Takenaka, N.; Kitamura, Y.; Koyano, Y.; Nagaoka, M. The number-adaptive multiscale QM/MM molecular dynamics simulation: Application to liquid water. *Chem. Phys. Lett.* **2012**, *524*, 56–61.
- (23) Takenaka, N.; Kitamura, Y.; Koyano, Y.; Nagaoka, M. An improvement in quantum mechanical description of solute-solvent interactions in condensed systems via the number-adaptive multiscale quantum mechanical/molecular mechanical-molecular dynamics method: Application to zwitterionic glycine in aqueous solution. *J. Chem. Phys.* **2012**, *137*, 024501.
- (24) Waller, M. P.; Kumbhar, S.; Yang, J. A density-based adaptive quantum mechanical/-molecular mechanical method. *ChemPhysChem* **2014**, *15*, 3218–3225.

- (25) Kerdcharoen, T.; Morokuma, K. ONIOM-XS: An extension of the ONIOM method for molecular simulation in condensed phase. *Chem. Phys. Lett.* **2002**, *355*, 257–262.
- (26) Buló, R. E.; Ensing, B.; Sikkema, J.; Visscher, L. Toward a Practical Method for Adaptive QM / MM Simulations. *J. Chem. Theory Comput.* **2009**, *5*, 2212–2221.
- (27) Heyden, A.; Lin, H.; Truhlar, D. G. Adaptive partitioning in combined quantum mechanical and molecular mechanical calculations of potential energy functions for multiscale simulations. *J. Phys. Chem. B* **2007**, *111*, 2231–2241.
- (28) Pezeshki, S.; Lin, H. Adaptive-partitioning redistributed charge and dipole schemes for QM/MM dynamics simulations: On-the-fly relocation of boundaries that pass through covalent bonds. *J. Chem. Theory Comput.* **2011**, *7*, 3625–3634.
- (29) Bernstein, N.; Várnai, C.; Solt, I.; Winfield, S. A.; Payne, M. C.; Simon, I.; Fuxreiter, M.; Csányi, G. QM/MM simulation of liquid water with an adaptive quantum region. *Phys. Chem. Chem. Phys.* **2012**, *14*, 646–656.
- (30) Watanabe, H. C.; Kubar, T.; Elstner, M. Size-consistent multipartitioning QM/MM: A stable and efficient adaptive QM/MM method. *J. Chem. Theory Comput.* **2014**, *10*, 4242–4252.
- (31) Buló, R. E.; Michel, C.; Fleurat-Lessard, P.; Sautet, P. Multiscale modeling of chemistry in water: Are we there yet? *J. Chem. Theory Comput.* **2013**, *9*, 5567–5577.
- (32) Zheng, M.; Waller, M. P. Adaptive quantum mechanics/molecular mechanics methods. *Wiley Interdiscip. Rev.: Comput. Mol. Sci.* **2016**, *6*, 369–385.
- (33) Csányi, G.; Albaret, T.; Payne, M. C.; De Vita, A. "Learn on the fly": A hybrid classical and quantum-mechanical molecular dynamics simulation. *Phys. Rev. Lett.* **2004**, *93*, 1–4.

- (34) Jacob, C. R.; Beyhan, S. M.; Bulo, R. E.; Gomes, A. S. P.; Götz, A. W.; Kiewisch, K.; Sikkema, J.; Visscher, L. PyADF - A scripting framework for multiscale quantum chemistry. *J. Comput. Chem.* **2011**, *32*, 2328–2338.
- (35) Te Velde, G.; Bickelhaupt, F. M.; Baerends, E. J.; Fonseca Guerra, C.; van Gisbergen, S. J.; Snijders, J. G.; Ziegler, T. Chemistry with ADF. *J. Comput. Chem.* **2001**, *22*, 931–967.
- (36) Lin, H.; Zhang, Y.; Pezeshki, S.; Wang, B.; Truhlar, D. G. *QMMM2015*; University of Minnesota: Minneapolis, 2015.
- (37) Gaussian16, Available at: <http://www.gaussian.com>.
- (38) Duster, A.; Wang, C.-H.; Lin, H. Adaptive QM/MM for Molecular Dynamics Simulations: 5. On the Energy-Conserved Permuted Adaptive-Partitioning Schemes. *Molecules* **2018**, *23*, 2170–2186.
- (39) Mones, L.; Jones, A.; Götz, A. W.; Laino, T.; Walker, R. C.; Leimkuhler, B.; Csányi, G.; Bernstein, N. The adaptive buffered force QM/MM method in the CP2K and amber software packages. *J. Comput. Chem.* **2015**, *36*, 633–648.
- (40) Hutter, J.; Iannuzzi, M.; Schiffmann, F.; Vandevondele, J. CP2K: Atomistic simulations of condensed matter systems. *Wiley Interdiscip. Rev.: Comput. Mol. Sci.* **2014**, *4*, 15–25.
- (41) Salomon-Ferrer, R.; Case, D. A.; Walker, R. C. An overview of the Amber biomolecular simulation package. *Wiley Interdiscip. Rev.: Comput. Mol. Sci.* **2013**, *3*, 198–210.
- (42) Okamoto, T.; Yamada, K.; Koyano, Y.; Asada, T.; Koga, N.; Nagaoka, M. A minimal implementation of the AMBER–GAUSSIAN interface for ab initio QM/MM-MD simulation. *J. Comput. Chem.* **2011**, *32*, 932–942.

- (43) Zheng, M.; Kuriappan, J. A.; Waller, M. P. Toward more efficient density-based adaptive QM/MM methods. *Int. J. Quant. Chem.* **2017**, *117*, e25336.
- (44) Zheng, M.; Waller, M. P. Yoink: An interaction-based partitioning API. *J. Comput. Chem.* **2018**, *39*, 799–806.
- (45) Watanabe, H. Improvement of performance, stability and continuity by modified size-consistent multipartitioning quantum mechanical/molecular mechanical method. *Molecules* **2018**, *23*, 1882–1897.
- (46) Abraham, M. J.; Murtola, T.; Schulz, R.; Páll, S.; Smith, J. C.; Hess, B.; Lindahl, E. GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. *SoftwareX* **2015**, *1-2*, 19–25.
- (47) Böckmann, M.; Doltsinis, N. L.; Marx, D. Adaptive switching of interaction potentials in the time domain: An extended Lagrangian approach tailored to transmute force field to QM/MM simulations and back. *J. Chem. Theory Comput.* **2015**, *11*, 2429–2439.
- (48) CPMD, Available at: <http://www.cpmc.org>.
- (49) Boereboom, J. M.; Potestio, R.; Donadio, D.; Bulo, R. E. Toward Hamiltonian Adaptive QM/MM: Accurate Solvent Structures Using Many-Body Potentials. *J. Chem. Theory Comput.* **2016**, *12*, 3441–3448.
- (50) Field, M. J. An Algorithm for Adaptive QC/MM Simulations. *J. Chem. Theory Comput.* **2017**, *13*, 2342–2351.
- (51) Field, M. J. The pDynamo Library for Molecular Simulations using Hybrid Quantum Mechanical and Molecular Mechanical Potentials,. *J. Chem. Theory Comput.* **2008**, *4*, 1151–1161.
- (52) Metz, S.; Kästner, J.; Sokol, A. A.; Keal, T. W.; Sherwood, P. ChemShell—a modular

- software package for QM/MM simulations. *Wiley Interdiscip. Rev.: Comput. Mol. Sci.* **2014**, *4*, 101–110.
- (53) Lu, Y.; Farrow, M. R.; Fayon, P.; Logsdail, A. J.; Sokol, A. A.; Catlow, C. R. A.; Sherwood, P.; Keal, T. W. Open-Source, Python-Based Redevelopment of the ChemShell Multiscale QM/MM Environment. *J. Chem. Theory Comput.* **2018**, DOI: 10.1021/acs.jctc.8b01036.
- (54) Dohm, S.; Spohr, E.; Korth, M. Developing adaptive QM/MM computer simulations for electrochemistry. *J. Comput. Chem.* **2017**, *38*, 51–58.
- (55) Senn, H. M.; Thiel, W. In *Atomistic approaches in modern biology*; Reiher, M., Ed.; Springer: Berlin, 2006; Vol. 268; pp 173–290.
- (56) Eastman, P.; Friedrichs, M. S.; Chodera, J. D.; Radmer, R. J.; Bruns, C. M.; Ku, J. P.; Beauchamp, K. A.; Lane, T. J.; Wang, L.-P.; Shuka, D.; Tye, T.; Houston, M.; Stich, T.; Klein, C.; Shirts, M. R.; Pande, V. S. OpenMM 4: a reusable, extensible, hardware independent library for high performance molecular simulation. *J. Chem. Theory Comput.* **2013**, *9*, 461–469.
- (57) Eastman, P.; Swails, J.; Chodera, J. D.; McGibbon, R. T.; Zhao, Y.; Beauchamp, K. A.; Wang, L. P.; Simmonett, A. C.; Harrigan, M. P.; Stern, C. D.; Wiewiora, R. P.; Brooks, B. R.; Pande, V. S. OpenMM 7: Rapid development of high performance algorithms for molecular dynamics. *PLOS Comput Biol* **2017**, *13*, 1–17.
- (58) Turney, J. M.; Simmonett, A. C.; Parrish, R. M.; Hohenstein, E. G.; Evangelista, F. A.; Fermann, J. T.; Mintz, B. J.; Burns, L. A.; Wilke, J. J.; Abrams, M. L.; Russ, N. J.; Leininger, M. L.; Janssen, C. L.; Seidl, E. T.; Allen, W. D.; Schaefer, H. F.; King, R. A.; Valeev, E. F.; Sherrill, C. D.; Crawford, T. D. Psi4: An open-source ab initio electronic structure program. *Wiley Interdiscip. Rev.: Comput. Mol. Sci.* **2012**, *2*, 556–565.

- (59) Parrish, R. M.; Burns, L. A.; Smith, D. G.; Simmonett, A. C.; DePrince, A. E.; Hohenstein, E. G.; Bozkaya, U.; Sokolov, A. Y.; Di Remigio, R.; Richard, R. M.; Gonthier, J. F.; James, A. M.; McAlexander, H. R.; Kumar, A.; Saitow, M.; Wang, X.; Pritchard, B. P.; Verma, P.; Schaefer, H. F.; Patkowski, K.; King, R. A.; Valeev, E. F.; Evangelista, F. A.; Turney, J. M.; Crawford, T. D.; Sherrill, C. D. Psi4 1.1: An Open-Source Electronic Structure Program Emphasizing Automation, Advanced Libraries, and Interoperability. *J. Chem. Theory Comput.* **2017**, *13*, 3185–3197.
- (60) McGibbon, R. T.; Beauchamp, K. A.; Harrigan, M. P.; Klein, C.; Swails, J. M.; Hernández, C. X.; Schwantes, C. R.; Wang, L.-P.; Lane, T. J.; Pande, V. S. MD-Traj: A Modern Open Library for the Analysis of Molecular Dynamics Trajectories. *Biophys. J.* **2015**, *109*, 1528–1532.
- (61) Valiev, M.; Bylaska, E. J.; Govind, N.; Kowalski, K.; Straatsma, T. P.; Van Dam, H. J.; Wang, D.; Nieplocha, J.; Apra, E.; Windus, T. L.; De Jong, W. A. NWChem: A comprehensive and scalable open-source solution for large scale molecular simulations. *Comput. Phys. Commun.* **2010**, *181*, 1477–1489.
- (62) Wilkins-Diehr, N.; Crawford, T. D. NSF's Inaugural Software Institutes: The Science Gateways Community Institute and the Molecular Sciences Software Institute. *Comput. Sci. Eng.* **2018**, *20*, 26–38.
- (63) Krylov, A.; Windus, T. L.; Barnes, T.; Marin-Rimoldi, E.; Nash, J. A.; Pritchard, B.; Smith, D. G. A.; Altarawy, D.; Saxe, P.; Clementi, C.; Crawford, T. D.; Harrison, R. J.; Jha, S.; Pande, V. S.; Head-Gordon, T. Perspective: Computational chemistry software and its advancement as illustrated through three grand challenge cases for molecular science. *J. Chem. Phys.* **2018**, *149*, 180901–1–180901–11.
- (64) Jorgensen, W. L.; Chandrasekhar, J.; Madura, J. D.; Impey, R. W.; Klein, M. L.

- Comparison of simple potential functions for simulating liquid water. *J. Chem. Phys.* **1983**, *79*, 926–935.
- (65) Petersson, G.; Al-Laham, M. A. A complete basis set model chemistry. II. Open-shell systems and the total energies of the first-row atoms. *J. Chem. Phys.* **1991**, *94*, 6081–6090.
- (66) Dunning Jr, T. H. Gaussian basis sets for use in correlated molecular calculations. I. The atoms boron through neon and hydrogen. *J. Chem. Phys.* **1989**, *90*, 1007–1023.
- (67) Roe, D. R.; Cheatham III, T. E. PTRAJ and CPPTRAJ: software for processing and analysis of molecular dynamics trajectory data. *J. Chem. Theory Comput.* **2013**, *9*, 3084–3095.
- (68) Nguyen, H.; Roe, D.; Swails, J.; Case, D. PYTRAJ: Interactive data analysis for molecular dynamics simulations, Available at: <https://github.com/Amber-MD/pytraj>.
- (69) Head-Gordon, T.; Hura, G. Water structure from scattering experiments and simulation. *Chem. Rev.* **2002**, *102*, 2651–2670.
- (70) Yu, H. A.; Karplus, M.; Pettitt, B. M. Aqueous solvation of n-methylacetamide conformers: Comparison of simulations and integral equation theories. *J. Am. Chem. Soc.* **1991**, *113*, 2425–2434.
- (71) Gaigeot, M. P.; Vuilleumier, R.; Sprik, M.; Borgis, D. Infrared spectroscopy of N-methylacetamide revisited by ab initio molecular dynamics simulations. *J. Chem. Theory Comput.* **2005**, *1*, 772–789.
- (72) Qian, P.; Lu, L.-N.; Yang, Z.-Z. Molecular dynamics simulations of N-methylacetamide (NMA) in water by the ABEEM/MM model. *Can. J. Chem.* **2009**, *87*, 1738–1746.
- (73) Gao, J.; Freindorf, M. Hybrid ab initio QM/MM simulation of N-methylacetamide in aqueous solution. *J. Phys. Chem. A* **1997**, *101*, 3182–3188.

- (74) Mennucci, B.; Martínez, J. M. How to model solvation of peptides? Insights from a quantum-mechanical and molecular dynamics study of N-methylacetamide. 1. Geometries, infrared, and ultraviolet spectra in water. *J. Phys. Chem. B* **2005**, *109*, 9818–9829.
- (75) Guo, H.; Karplus, M. Ab initio studies of hydrogen bonding of N-methylacetamide: structure, cooperativity, and internal rotational barriers. *J. Phys. Chem.* **1992**, *96*, 7273–7287.
- (76) Han, W.-G.; Suhai, S. Density Functional Studies on N-Methylacetamide- Water Complexes. *J. Phys. Chem.* **1996**, *100*, 3942–3949.
- (77) Katz, J. L.; Post, B. The crystal structure and polymorphism of N-methylacetamide. *Acta Crystallogr.* **1960**, *13*, 624–628.
- (78) Kitano, M.; Fukuyama, T.; Kuchitsu, K. Molecular structure of N-methylacetamide as studied by gas electron diffraction. *Bull. Chem. Soc. Jpn.* **1973**, *46*, 384–387.
- (79) Minami, H.; Iwahashi, M. Molecular Self-Assembling of -Methylacetamide in Solvents. *Int. J. Spectrosc.* **2011**, *2011*.
- (80) Mirkin, N. G.; Krimm, S. Structure of trans-N-methylacetamide: planar or non-planar symmetry? *J. Mol. Struct.: THEOCHEM* **1995**, *334*, 1–6.
- (81) Buck, M.; Karplus, M. Hydrogen bond energetics: a simulation and statistical analysis of N-methyl acetamide (NMA), water, and human lysozyme. *J. Phys. Chem. B* **2001**, *105*, 11000–11015.
- (82) Xiao, X.; Tan, Y.; Zhu, L.; Guo, Y.; Wen, Z.; Li, M.; Pu, X.; Tian, A. Effects of the position and manner of hydration on the stability of solvated N-methylacetamides and the strength of binding between N-methylacetamide and water clusters: a computational study. *J. Mol. Model.* **2012**, *18*, 1389–1399.

- (83) Hornak, V.; Abel, R.; Okur, A.; Strockbine, B.; Roitberg, A.; Simmerling, C. Comparison of multiple Amber force fields and development of improved protein backbone parameters. *Proteins* **2006**, *65*, 712–725.
- (84) Hornak, V.; Okur, A.; Rizzo, R. C.; Simmerling, C. HIV-1 protease flaps spontaneously open and reclose in molecular dynamics simulations. *Proc. Natl. Acad. Sci.* **2006**, *103*, 915–920.
- (85) Hehre, W. J.; Stewart, R. F.; Pople, J. A. Self-consistent molecular-orbital methods. I. Use of gaussian expansions of Slater-type atomic orbitals. *J. Chem. Phys.* **1969**, *51*, 2657–2664.
- (86) Humphrey, W.; Dalke, A.; Schulten, K. VMD: visual molecular dynamics. *J. Mol. Graph.* **1996**, *14*, 33–38.
- (87) VMD, Available at: <http://www.ks.uiuc.edu/Research/vmd/>.
- (88) Baker, E.; Hubbard, R. Hydrogen bonding in globular proteins. *Prog. Biophys. Mol. Biol.* **1984**, *44*, 97–179.

For Table of Contents use only

Title: JANUS: An Extensible Open-Source Software Package for Adaptive QM/MM

Methods

Authors: Boyi Zhang, Doaa Altarawy, Taylor Barnes, Justin M. Turney, Henry F. Schaefer

