

NAMD goes quantum: An integrative suite for hybrid simulations

Marcelo C. R. Melo^{1,2*}, Rafael C. Bernardi^{1*}, Till Rudack^{1,3},
Maximilian Scheurer^{4,5}, Christoph Riplinger⁶, James C. Phillips¹,
Julio D. C. Maia⁷, Gerd B. Rocha⁸, João V. Ribeiro¹, John E. Stone¹,
Frank Neese⁹, Klaus Schulten^{1,10+}, Zaida Luthey-Schulten^{1,2,9,11**}
* These authors contributed equally.
+ In memoriam.

¹ NIH Center for Macromolecular Modeling and Bioinformatics, Beckman Institute for Advanced Science and Technology, University of Illinois at Urbana-Champaign, 61801, USA

² Center for Biophysics and Computational Biology, University of Illinois at Urbana-Champaign, 61801, USA

³ Department of Biophysics, Ruhr-University Bochum, Germany

⁴ Biochemistry Center, Heidelberg University, Heidelberg, Germany

⁵ Interdisciplinary Center for Scientific Computing, Heidelberg, Germany

⁶ FAccTs GmbH, Köln, Germany

⁷ Center for Informatics, Federal University of Paraíba, João Pessoa, Brazil

⁸ Department of Chemistry, Federal University of Paraíba, João Pessoa, Brazil

⁹ Max Planck Institut für Kohlenforschung, Mülheim an der Ruhr, Germany

¹⁰ Department of Physics, University of Illinois at Urbana-Champaign, 61801, USA

¹¹ Department of Chemistry, University of Illinois at Urbana-Champaign, 61801, USA

** Corresponding author

Email: zan@illinois.edu

Applications of hybrid QM/MM methods range from reactions in active sites of small enzymes to multiple sites in large bioenergetic complexes. By combining the widely used molecular dynamics and visualization programs NAMD/VMD with the quantum chemistry packages ORCA, and MOPAC, we provide an integrated, comprehensive, customizable, and easy-to-use suite. Through the QwikMD interface, setup, execution, visualization, and analysis are streamlined for all levels of expertise.

Though molecular mechanics (MM) force-fields are based on quantum mechanical calculations and experimental observations, only quantum mechanics (QM) enables a complete and accurate understanding of many biochemical processes, particularly those involving chemical reactions or charge redistribution [1]. Nevertheless, even with the advanced hardware technology available today, the computational cost of studying nanosecond-long dynamics of entire systems relying solely on QM methodologies is usually prohibitive. A common route to circumvent this cost barrier is to confine the QM formalism to a subregion of a system and to include the effects of the surrounding system through MM simulations, leading to hybrid “QM/MM” simulations [2]. QM/MM calculations are now used broadly in enzymology [3], drug discovery [4], bioenergetic systems [5], as well as in combination with serial femtosecond crystallography [6] and other material sciences and structural biology techniques [7]. To deliver accurate results, these QM/MM studies require a carefully selected quantum region [5, 8]. Unfortunately, a majority of the available QM/MM implementations lack a comprehensive set of features that could make these calculations even more attractive to chemists, structural biologists, and material engineers.

NAMD [9] is one of the most widely used software packages for molecular dynamics (MD) simulations, particularly for large biomolecular systems on supercomputers. NAMD’s scalability and large array of enhanced sampling and free energy methods [10], as well as a seamless integration with VMD [11], which provides extended setup, visualization and analysis capabilities, makes it an invaluable tool for exploring complex biological systems. Here we present a comprehensive QM/MM suite, developed to provide easy setup, visualization and analysis of QM/MM simulations through the graphical user interface QwikMD [12] as well as a broad range of QM methods through NAMD’s new “QMForces” module. Our graphical interface makes our tool unique among widely employed MD codes. The QM/MM interface in NAMD supports the simulation of many independent QM regions, and smooth integration with a collection of enhanced sampling and alchemical methods, including the combination of QM/MM approaches with state-of-the-art free energy methods, such as extended adaptive biasing force (eABF) [13]. In addition, NAMD performs similarly to other QM/MM programs regarding energy conservation (**Supplementary Table 1**), and allows extreme scalability for free-energy calculations (**Supplementary Fig. 1**).

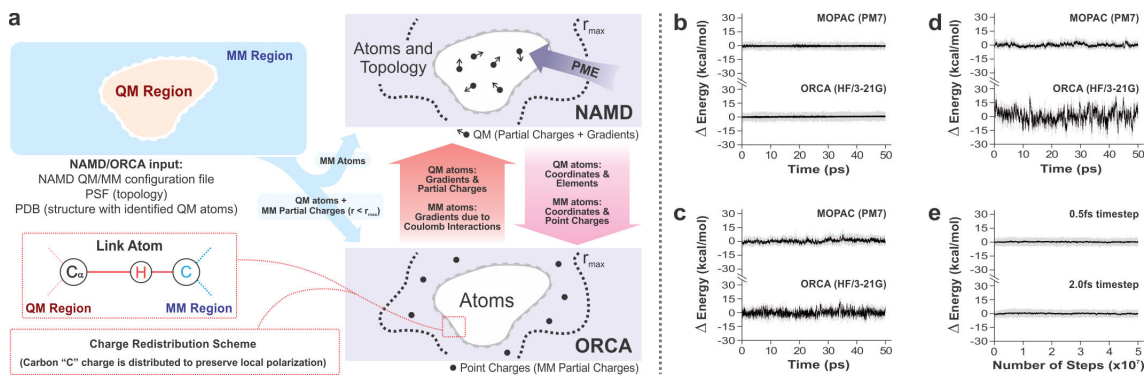


Figure 1: **Hybrid QM/MM NAMD.** **a)** Schematic of NAMD-ORCA interconnection. The contribution of MM charges beyond r_{max} are calculated by NAMD (via PME), while ORCA calculates direct electrostatics. **b)** Energy conservation test for a pure-QM alanine molecule. All energy conservation plots display the deviation from the mean system energy. The black line indicates a running average and the gray line indicates maximal fluctuation. **c)** Energy conservation test for a QM/MM tri-alanine molecule. **d)** Energy conservation test for a QM/MM tri-alanine molecule in water. The basis set employed in ORCA tests (3-21G) serves only as a technical test. **e)** Energy conservation test for a pure-QM NMA molecule using NAMD/MOPAC with PM7.

In hybrid QM/MM simulations, NAMD offloads part of its standard force and energy calculations to a QM program, either through native interfaces to MOPAC [14, 15] or ORCA [16], or through a flexible generic interface requiring a wrapper script (exemplary Python wrappers are provided for Gaussian, TeraChem, and Q-CHEM). Other MD software packages, such as Amber [17], additionally include their own code for QM calculations. This strategy was not adopted here as a major strength of our QM/MM interface is the flexibility to allow a user to easily fine tune the execution of any QM program, particularly using our scripted interface. Importantly, by performing file I/O in RAM, very little time is lost during communication between NAMD and the QM program when compared to the calculation of a QM step. In NAMD, multiple QM-MM coupling schemes have been implemented, allowing for both mechanically and electrostatically embedded QM regions to be used (see **Online Methods**). QM/MM simulations require the same input files used for classical MD, with additional options in the configuration file. Typically, QM and MM atoms that are covalently bound are treated by redistributing the MM atom’s charge over its nearest MM neighbors and by capping the QM atom with a hydrogen atom, known as the Link Atom method, as shown in **Fig. 1a** and **Supplementary Fig. 2** for a solvated tri-alanine QM/MM calculation using the NAMD/ORCA interface. For all Link Atom variations, see **Online Methods** and **Supplementary Fig. 3a-b, 4, 5a and 5b**.

To test the QM/MM interface for accuracy, stability, and performance, we carried out standard validation simulations with both MOPAC and ORCA. NAMD

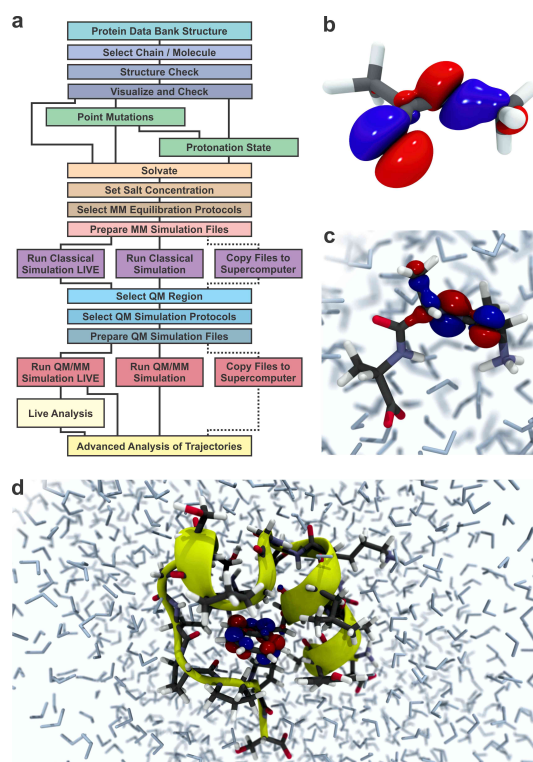


Figure 2: **Hybrid QM/MM VMD features.** **a)** QwikMD provides a graphical user interface (GUI) in VMD for performing QM/MM simulations. The image shows the workflow to prepare, run, analyze, and visualize a hybrid QM/MM molecular dynamics simulation. **b)** Highest occupied molecular orbital of an alanine molecule in vacuum. **c)** Alanine's highest occupied molecular orbital in a solvated QM/MM tri-alanine. **d)** Trp-Cage protein highest occupied molecular orbital in water solution.

achieved energy conservation for a pure QM alanine molecule (**Fig. 1b**), a hybrid QM/MM tri-alanine molecule in vacuum (**Fig. 1c**) and in water (**Fig. 1d**), and a pure QM N-methyl acetamide (NMA) molecule, the latter for up to 100 ns with both 0.5 fs and 2.0 fs time steps (**Fig. 1e**). A particle mesh Ewald (PME) treatment of long-range electrostatics was also observed to conserve energy (**Supplementary Table 2**). Compared, NAMD and Amber16 presented equally good QM/MM results (**Supplementary Fig. 6**). NAMD/MOPAC was shown to perform up to 10 ns/day of QM/MM simulation in a single desktop computer (**Supplementary Table 3**). In all tests NAMD was shown to perform efficiently while conserving energy.

To provide a quick and easy way of setting up QM/MM simulations, we incorporated the most widely employed features of a hybrid QM/MM molecular dynamics code to VMD’s QwikMD [12], setting up a comprehensive QM/MM work-flow (**Fig. 2a**). QwikMD automates the creation of input and configuration files while checking for common mistakes, and ensures reproducibility of the result by recording all simulation-related information. Moreover, QwikMD allows drugs, metabolites and other molecules lacking MM parameters to be easily added to a QM region. Once prepared, QM/MM simulations can be directly performed either in real-time using “live-view” mode or by calling NAMD using local computer resources or a supercomputing center (**Supplementary Fig. 7**).

Trajectories from classical or hybrid QM/MM simulations are easily read and analyzed by VMD, whose capabilities have been extended in this work to natively support MOPAC and ORCA outputs, which benefits not only the users of our QM/MM suite but any users of these two packages. New representation schemes were introduced to VMD allowing the selection and visualization of orbital trajectories throughout a simulation, and also to allow for the chemical bonds representation to be dynamically updated. The new capabilities in VMD make it a powerful tool to visualize and analyze output from QM/MM calculations as well as outputs directly from ORCA and MOPAC (see **Fig. 2b-d**, **Supplementary Tables 4, 5 and 6** and **Supplementary Video 1**).

Combining the new QM/MM capabilities of the NAMD/MOPAC interface with existing tools for enhanced sampling and free energy calculations, we investigated the aminoacylation reaction mechanism of *Thermus thermophilus* glutamyl-tRNA synthetase (GluRS) and its interactions with its cognate tRNA (tRNA^{Glu}), using PM7 together with the CHARMM36 force field. In order to establish the genetic code, GluRS reads the tRNA’s anticodon region and uses this information to rearrange its catalytic site, facilitating the transfer of an AMP-bound glutamyl to the 3’ end of the tRNA. A study of the allosteric information processing and transduction was previously conducted using classical MD [18]. However, because the anticodon binding domain and catalytic site of GluRS are over 50 angstroms apart, the treatment of

the entire pathway, with approximately 5,000 atoms, using QM accuracy would be unfeasible. Using QwikMD to prepare a system with two independent QM regions, we were able to investigate the allosteric signaling pathway of the GluRS:tRNA^{Glu}

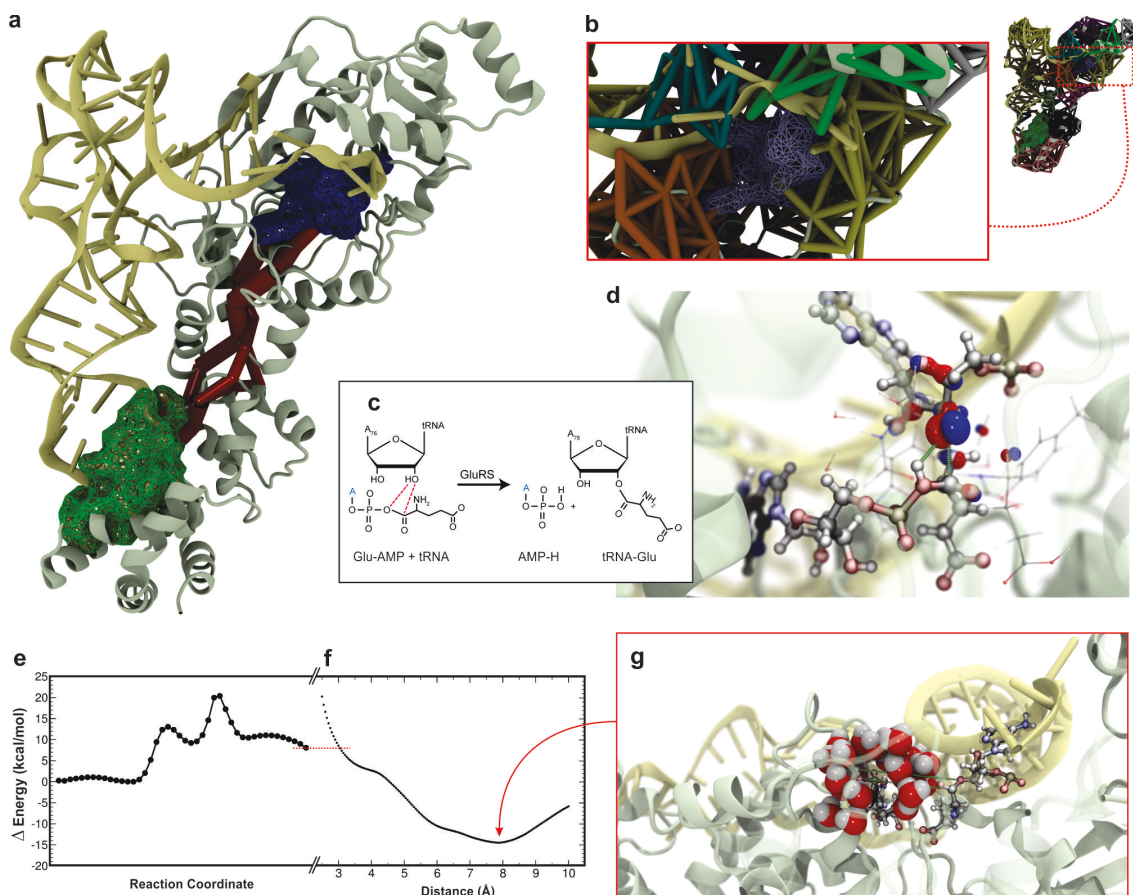


Figure 3: Mechanism of glutamyl-tRNA synthetase. **a)** Representation of the glutamyl-tRNA synthetase allosteric pathway (red). Two independent QM regions are highlighted, indicating the active site (blue) and the anticodon binding region (green). **b)** Community analysis showing multiple communities involved in the active site. **c)** Reaction mechanism of glutamyl-tRNA synthetase. **d)** Intermediate state of the glutamyl-tRNA synthetase reaction showing the highest occupied molecular orbital. **e)** Free energy profile of the glutamyl-tRNA synthetase reaction mechanism, calculated using eABF after a string method path optimization. Both eABF and the string method were performed using QM/MM MD simulations with NAMD/MOPAC and PM7. **f)** Free energy profile of the distancing and solvation of AMP calculated using ABF and classical MD simulations. **g)** Snapshot of the minimum energy state during the release of the AMP, showing the solvation of the phosphate group. The eABF free energy profile in **e** corresponds to the most favorable mechanism (**d**). The final state of the reaction revealed that the AMP-H phosphate was very close to the glutamyl-tRNA, keeping water from entering the active site. As the AMP-H moves away from the charged tRNA, water hydrates the AMP-H, as shown in **f** and **g**. Since in the final state of the eABF calculation the distance between Glu-AMP:Carb-C and Glu-AMP:P-O was of 3 Å, we used this distance as the connecting point (see red dashed line between **e** and **f**) between the free energy profiles calculated with eABF and ABF.

complex with an accurate QM treatment of the two most critical regions of the system (see **Fig. 3a**).

The communication pathways that lead to coordinated motion between functionally important QM regions were analyzed using cross-correlation based network analysis [18] (see **Online Methods** and **Supplementary Fig. 8**), and found to be significantly degenerate, as previously observed using classical MD [18] (**Fig. 3a**). The QM/MM treatment of critical regions lead to increased correlations between residues in the active site and anticodon binding site (**Supplementary Fig. 9**). Suboptimal communication pathways are based on the correlation of atom motion during the simulation and have been employed to calculate allosteric signaling and force propagation pathways. Tightly correlated groups of atoms are clustered into communities, indicating functional domains of biomolecules and important interfaces between multi-molecule complexes. In this work, the presence of multiple communities within a QM region and the occurrence of communities that cover both classical and quantum atoms, highlight the seamless integration of QM and MM regions (**Fig. 3b**).

The aminoacylation reaction mechanism was for the first time investigated in order to distinguish between four possible mechanisms (see **Supplementary Fig. 10**), all based on a previously proposed concerted exchange where the 3' or the 2'-OH oxygen attacks the carbonyl carbon of Glu-AMP while the proton shifts to the phosphate's oxygen [19]. In addition, the reaction was investigated while the amine group of the adenylyate was protonated or de-protonated.

QM/MM steered molecular dynamics (SMD) simulations were performed to induce the possible reaction mechanisms, and intermediate states were selected from the SMD trajectories to initiate a String Method optimization [20]. The method utilizes groups of biased MD simulations to find the path with the smallest barrier that describes the chemical reaction. Once an optimized string was achieved, the reaction path was used to perform a parallel extended-Adaptive Biasing Force [13] (eABF) calculation to determine the free energy transformation of each possible mechanism (see **Supplementary Fig. 11, 12**). The reaction charging the 3-hydroxyl group, with the amine group of the adenylyate being deprotonated, was observed to be the most favorable route with a barrier of ≈ 20 kcal/mol and final state at ≈ 8 kcal/mol (**Fig. 3c-e**).

Initially, all possible reaction mechanisms tested were observed to be endergonic (**Supplementary Fig. 13**). However, closer examination of the final states revealed that the AMP's phosphate was only 3 Å away from the charged tRNA. Dissociation of the AMP-H from the charged tRNA opens space for hydration of the active site, and consequently of its phosphate group (**Supplementary Fig. 14**). Since the

dissociation does not involve chemical reactions, we investigated this process using ABF and classical MD simulations. As the products move apart and water enters the active site, hydrating the phosphate group of AMP-H, we observed a ≈ 22 kcal/mol drop in free energy, making the entire reaction-solvation process exergonic, with a ≈ -15 kcal/mol free energy variation between reactants and products (**Fig. 3e-g**). It is worth noting that appending QM/MM eABF and MM ABF results leads to a small imprecision in the connection between free energy values (red dashed line in **Fig. 3e-f**).

Hybrid QM/MM protocols have been applied to a wide range of investigations, however, a comprehensive, customizable, easy-to-use, and freely available suite was not yet available to the broader computational biology and material sciences communities. Building upon the synergy between NAMD and VMD, we developed a robust and user-friendly QM/MM suite to prepare, perform, and analyze QM/MM simulations. Test applications presented here highlight the accuracy of this implementation, while the study of the GluRS:tRNA^{Glu}:Glu-AMP complex revealed sub-atomic details of its reaction mechanism. Uniting Network Analysis results for the full complex with the first combination of the string method, parallel eABF, and QM/MM simulations, we provide a unique view for essential steps in establishing the genetic code (**Supplementary Fig. 15**). In summary, the ease of access through QwikMD and modularity in NAMD will permit this suite to be used in a variety of applications and contexts, including as a teaching tool, a research interface, a platform for mixing-and-matching QM/MM with free-energy and enhanced sampling methods, and even as a sandbox for the development of new QM tools and QM/MM interactions schemes.

METHODS

Methods, including statements of data availability and any associated accession codes and references, are available in the online version of the paper.

ACKNOWLEDGMENTS

The authors thank M. F. Herbst, C. Chipot and G. Fiorin for helpful discussions. This work was supported by National Science Foundation (NSF) grants MCB-1616590 and MCB-1244570, and National Institutes of Health (NIH) grant 9P41GM104601. Z.L.S. acknowledges support from the NSF PHY1430124. M.C.R.M. and Z.L.S. acknowledge support from the Keck Foundation (grant no. 206231). T.R. acknowledges support as a Feodor Lynen von Humboldt Postdoctoral Fellow. J.D.C.M acknowledges a fellowship from the Brazilian Coordination for Improvement of Higher Educational Personnel (CAPES). G.B.R. acknowledges support from the Brazilian National Council for Scientific and Technological Development (CNPq 305271/2013-0),

and CAPES AUXPE1375/2014. F.N. and C.R. acknowledge support for the development of ORCA by the Max Planck society (MPG) and the Germans Science Foundation (DFG). This research made use of the Blue Waters sustained-petascale computing, which is supported by the state of Illinois and the NSF OCI-0725070 and ACI-1238993. This work is part of the Petascale Computational Resource (PRAC) grant, which is supported by the NSF ACI-1713784.

AUTHOR CONTRIBUTIONS

M.C.R.M, R.C.B., T.R., K.S. and Z.L.S. conceived the project. M.C.R.M implemented the QM/MM interface. R.C.B., T.R., M.C.R.M., G.B.R., and K.S. discussed QM/MM features. J.D.C.M., G.B.R., C.R. and F.N. provided guidance on the QM/MM interface development. J.C.P. assisted in adapting NAMD. M.C.R.M, and M.S. prepared Python scripts for the interfaces of selected QM software packages. R.C.B. and M.C.R.M performed all NAMD tests and simulations. J.D.C.M. and G.B.R. performed Amber calculations. R.C.B. and M.C.R.M performed all free energy calculations and analysis. M.S., and J.E.S. implemented the orbital visualization in VMD. J.V.R., and J.E.S. implemented QM/MM graphical interface in QwikMD. R.C.B., M.C.R.M., T.R., M.S., G.B.R., F.N., and Z.L.S. wrote and edited the manuscript. K.S., and Z.L.S. supervised the project.

COMPETING FINANCIAL INTERESTS

The authors declare no competing financial interests.

References

- [1] Senn, H. M. & Thiel, W. QM/MM methods for biomolecular systems. *Angew. Chem. Int. Ed. Engl.* **48**, 1198–229 (2009).
- [2] Field, M. J., Bash, P. A. & Karplus, M. A combined quantum mechanical and molecular mechanical potential for molecular dynamics simulations. *J. Comput. Chem.* **11**, 700–733 (1990).
- [3] van der Kamp, M. W. & Mulholland, A. J. Combined quantum mechanics/molecular mechanics (QM/MM) methods in computational enzymology. *Biochemistry* **52**, 2708–2728 (2013).
- [4] Bernardi, R. C. & Pascutti, P. G. Hybrid QM/MM Molecular Dynamics Study of Benzocaine in a Membrane Environment: How Does a Quantum Mechanical Treatment of Both Anesthetic and Lipids Affect Their Interaction. *J. Chem. Theor. Comp.* **8**, 2197–2203 (2012).

- [5] Retegan, M., Neese, F. & Pantazis, D. A. Convergence of QM/MM and Cluster Models for the Spectroscopic Properties of the Oxygen-Evolving Complex in Photosystem II. *J. Chem. Theory Comput.* **9**, 3832–3842 (2013).
- [6] Coquelle, N. *et al.* Chromophore twisting in the excited state of a photoswitchable fluorescent protein captured by time-resolved serial femtosecond crystallography. *Nature Chemistry* (2017).
- [7] Lin, H. & Truhlar, D. G. QM/MM: what have we learned, where are we, and where do we go from here? *Theor. Chim. Acta* **117**, 185–199 (2007).
- [8] Jindal, G. & Warshel, A. Exploring the dependence of QM/MM calculations of enzyme catalysis on the size of the QM region. *J. Phys. Chem. B* **120**, 9913–9921 (2016).
- [9] Phillips, J. C. *et al.* Scalable molecular dynamics with NAMD. *J. Comput. Chem.* **26**, 1781–1802 (2005).
- [10] Jiang, W. *et al.* Generalized scalable multiple copy algorithms for molecular dynamics simulations in NAMD. *Comput. Phys. Commun.* **185**, 908–916 (2014).
- [11] Humphrey, W., Dalke, A. & Schulten, K. VMD: visual molecular dynamics. *J. Mol. Graphics* **14**, 33–38 (1996).
- [12] Ribeiro, J. V. *et al.* QwikMD Integrative Molecular Dynamics Toolkit for Novices and Experts. *Sci. Rep.* **6**, 1–14 (2016).
- [13] Fu, H., Shao, X., Chipot, C. & Cai, W. Extended Adaptive Biasing Force Algorithm. An On-the-Fly Implementation for Accurate Free-Energy Calculations. *J. Chem. Theory Comput.* **12**, 3506–3513 (2016).
- [14] Stewart, J. J. MOPAC: a semiempirical molecular orbital program. *J. Comput.-Aided Mol. Des.* **4**, 1–103 (1990).
- [15] Maia, J. D. C. *et al.* GPU linear algebra libraries and GPGPU programming for accelerating MOPAC semiempirical quantum chemistry calculations. *J. Chem. Theor. Comp.* **8**, 3072–3081 (2012).
- [16] Neese, F. Software update: the ORCA program system, version 4.0. *Wiley Interdisciplinary Reviews: Computational Molecular Science* **8:e1327** (2018).
- [17] Götz, A. W., Clark, M. A. & Walker, R. C. An extensible interface for QM/MM molecular dynamics simulations with AMBER. *J. Comput. Chem.* **35**, 95–108 (2014).
- [18] Sethi, A., Eargle, J., Black, A. A. & Luthey-Schulten, Z. Dynamical networks in tRNA:protein complexes. *Proc. Natl. Acad. Sci.* **106**, 6620–6625 (2009).

- [19] Black Pyrkosz, A., Eargle, J., Sethi, A. & Luthey-Schulten, Z. Exit Strategies for Charged tRNA from GluRS. *J. Mol. Biol.* **397**, 1350–1371 (2010).
- [20] Pan, A. C., Sezer, D. & Roux, B. Finding Transition Pathways Using the String Method with Swarms of Trajectories. *J. Phys. Chem. B* **112**, 3432–3440 (2008).

ONLINE METHODS

In order to study the chemistry of complex biological systems, in particular using MD approaches, one needs to combine QM and MM methods to secure sub-atomic resolution within relevant time and length scales. Yet, recent advances in implementations of QM calculations and hardware improvements are continuously improving our capacity to expand the complexity and comprehensiveness of QM results [21]. Hybrid QM/MM simulations in NAMD divide the system into “MM” and “QM” regions, using a classical force field to treat the classical atoms (or “MM atoms”), and passing the information that describes the quantum atoms in the system (or “QM atoms”) to a quantum chemistry package, which is expected to calculate forces for all QM atoms, as well as the total energy of the QM region (and optionally partial charges). All bonded and non-bonded interactions among MM atoms are handled by NAMD’s CHARMM force field, whereas all interactions among QM atoms are handled by the quantum chemistry package in its chosen theory level.

The non-bonded interactions between QM and MM atoms can be modified and regulated by the user. Lennard-Jones interactions are always calculated by NAMD, and specific parameters can be provided for QM atoms. QM-specific modifications have been proposed in order to compensate for over-polarization that these atoms may exhibit in hybrid QM/MM simulations, though the importance of such modifications has been disputed [22]. In all simulations presented in this work, the standard CHARMM36 Lennard-Jones parameters were used for all MM and QM atoms.

Mechanical and electrostatic embedding It is long known that electrostatics, particularly polarization, plays a key role in many biochemical processes [23]. Electrostatic interactions between QM and MM atoms deserve a more detailed discussion due to the abundance and diversity of available alternatives. In the “mechanical embedding” scheme, no electrostatic influences from MM atoms are accounted for while the QM package computes forces, charge distribution and energy in the QM region. Only positions and elements of atoms in the QM region are passed on to the QM package, and QM and MM atoms interact only through NAMD-calculated Lennard-Jones and electrostatic potentials (see below for special treatment of QM-MM bonds).

In the “electrostatic embedding” scheme, on the other hand, the partial charges of MM atoms surrounding all QM atoms are used to approximate the electrostatic environment of the QM atoms. The selection of classical point charges can be done automatically by NAMD, in which case the “cutoff” value is used to effectively create a shell of point charges around the QM region. This type of embedding is the most frequently used in biomolecular simulations [1].

Classical point charges handed over to the QM program to be utilized in electrostatic embedding can be altered by NAMD in a variety of ways. First by applying a smoothing function that avoids abrupt changes in electrostatic forces due to the cutoff used in the selection of surrounding point charges. Second, partial charges can be further modified so that their sum is a whole number or so that it amounts to a complementary charge to that of the QM region, in which case the sum of charges from QM atoms and classical partial charges add to zero. In the latter, the user is also able to select, atom by atom, which classical partial charges should be used when building the electrostatic embedding, providing further flexibility to the application.

Irrespective of the chosen embedding method, the calculated charge distribution for QM atoms can be used by NAMD to update the partial charges of QM atoms for the calculation of short and long range electrostatic interactions. In particular, if particle-mesh Ewald (PME) [24] is being used, NAMD can apply the newly determined charges for QM atoms to the calculation of long range electrostatics in both QM and MM regions of the system. In this case, the necessary corrections are calculated as to subtract from the PME forces all interactions already calculated by the QM package between QM-QM atom pairs, and by NAMD’s QM module between QM-point charge pairs.

PME forces and energy calculations in NAMD are carried out using classical charges for all classical atoms and the user’s choice between classical charges or updated charges for QM atoms. The forces and energy derived from PME interactions between QM atoms, and between QM atoms and the surrounding point charges are re-calculated within the QM module and subtracted from the direct electrostatic calculations, to avoid double counting these contributions. In order to test the impact of such implementation on the simulations, an energy conservation analysis was carried out comparing the use of PME in NAMD when the QM calculations are done using PM3 [25, 26], HF-3C [27] and DFT methods (**Supplementary Table 2**).

Treatment of covalent bonds involving QM and MM atoms. Hybrid QM/MM simulations of biomolecular systems often present situations where only a part of a molecule should be treated quantum mechanically, usually to save computational resources since the cost of simulating QM regions rises rapidly with the number of atoms. In order to deal with chemical bonds that have one atom in the quantum mechanical (QM) region and another in the classical (MM) region (from here on called “QM-MM bonds”), NAMD offers several methods that can be combined to alter the molecular system in order to bridge differences in simulation type (QM vs MM), and minimize errors involved in the QM/MM division of the system. Irrespective of the selected methods to treat the QM-MM bond, the bonded term (between MM1 and QM1 atoms, see **Supplementary Fig. 3c**) will still be calculated by NAMD, along

with all proper and improper dihedral terms and angle terms that cross the QM/MM barrier but still involve at least one classical atom.

Link atoms. The most widely used method to cap the QM regions containing QM-MM bonds is the “link atom” approach [28], where an atom (usually a hydrogen atom) is placed along the bond between the QM atom and the MM atom (see **Supplementary Fig. 3c**), and does not exist in the classical simulation. The user can fine-tune this process by choosing the element and which method of link atom placement will be used: either fixed (the default) or variable. The former depends on a user-defined distance, which will be used throughout the simulation as d_{L-QM} (see **Supplementary Fig. 3c**). For the latter, a user-defined fraction is used to define C_L (see **Equation 2**), and this fixed value is used by NAMD to calculate d_{L-QM} at every step as a function of d_{MM-QM} , which in turn varies over the course of the simulation due to bond vibration.

In order to conserve force and energy, the total force acting on the link atom ($\vec{F}_L$), as calculated by the QM code, is redistributed by NAMD over the QM1 and MM1 atoms[29], and added to the total forces calculated on those atoms by the QM code and NAMD, respectively. **Equations 1** and **3** describe the calculation of the x component of the redistributed force applied on the QM1 and MM1 atoms, respectively. Analogous equations are used for the other two axis. Here, $\hat{i}_x$ is the unit vector on the x axis, $\hat{r}_{MM-QM}$ is the unit vector in the QM-MM bond direction, and x_{MM} and x_{QM} are the x components of the positions of the MM1 and QM1 atoms, respectively.

$$F'_{QM1x} = \vec{F}_L \cdot \left[(1 - C_L) \cdot \hat{i}_x + C_L \cdot \frac{x_{MM} - x_{QM}}{d_{MM-QM}} \cdot \hat{r}_{MM-QM} \right] \quad (1)$$

where

$$C_L = \frac{d_{L-QM}}{d_{MM-QM}} \quad (2)$$

Similarly,

$$F'_{MM1x} = \vec{F}_L \cdot \left[C_L \cdot \hat{i}_x - C_L \cdot \frac{x_{MM} - x_{QM}}{d_{MM-QM}} \cdot \hat{r}_{MM-QM} \right] \quad (3)$$

The link atom is not the only proposed method to handle QM-MM bonds, but it is the only one that relies entirely on the classical side of a QM/MM simulation, over which we have complete control. Therefore, irrespective of which QM package is chosen to carry out the QM calculations of the simulation, NAMD’s QM/MM inter-

face always guarantees the creation of proper conditions to simulate QM-MM bonds, which grants its great flexibility.

Point charge alteration and redistribution. In any system containing a QM-MM bond, the link atom will invariably be placed very near the MM1 atom, which would create very strong electrostatic repulsion forces (or attractions) in case the MM1 partial charge was sent to the QM package for its calculations. Under the “mechanical embedding” scheme, the QM package only receives the atoms in the QM region and the link atoms created to approximate QM-MM bonds, so no manipulation of partial charges is required. On the other hand, the more usual “electrostatic embedding” scheme requires special treatment of nearby classical partial charges.

Several methods have been proposed to handle the conditioning of classical partial charges surrounding a QM-MM bond, and the QM/MM interface developed here offers the most widely accepted ones (see **Supplementary Fig. 4**). In all methods implemented here, the classical atom participating in the QM-MM bond (MM1 atom) does not have its partial charge passed on to the QM package, since this would create excessive repulsion (or attraction) on the Link atom. This is, in fact, the entirety of the “Z1” method: ignoring the partial charge of the MM1 atom[30]. Analogously, Z2 ignores the MM1 partial charge and all partial charges of atoms bound to MM1, called “MM2 atoms”, and Z3 extends the approach by ignoring all partial charges of atoms bound to MM2 atoms, called “MM3 atoms” (see **Supplementary Fig. 4b-d**).

The Charge Shifting (CS) method[31] (see **Supplementary Fig. 4e**) is more elaborate, as it rearranges the partial charge of the MM1 atom (indicated here as q_{M1}) so that the total charge of the region is maintained, while approximating the dipole moments of the bonds between MM1 and MM2 atoms. This is done by creating “virtual” point charges that are passed to the QM package as if they represented partial charges of classical atoms. In this case, the MM1 partial charge is equally distributed across the MM2 atoms, as indicated in the figure with the placement of the charge q_1 at the position of MM2 atoms (where $q_1 = q_{M1}/2$), and the coefficient used in **Equations 4** and **5** to redistribute the force on this virtual charge is given by $C_1 = d_{M1-Q_1}/d_{M1-M2} = 1.0$. Moreover, two virtual point charges are placed along the direction of the MM1-MM2 bond, one before the MM2 atom ($q_+ = q_0$) and one after ($q_- = -1 \times q_0$). In our implementation, $C_+ = d_{M1-Q_+}/d_{M1-M2} = 0.94$, while $C_- = d_{M1-Q_-}/d_{M1-M2} = 1.06$ (see **Supplementary Fig. 4e**). This method will keep the total charge of the region constant while trying to preserve the local dipoles formed by all MM1-MM2 bonds.

The Redistributed Charge and Dipole (RCD) method [30] (see **Supplementary Fig. 4f**) follows a similar arrangement. A virtual point charge is created in the middle

of all MM1-MM2 bonds ($C_2 = d_{M1-Q_2}/d_{M1-M2} = 0.5$) with a charge $q_2 = 2 \times q_{M1}/2$, and a charge $q_1 = -1 \times q_{M1}/2$ is placed in the positions of all MM2 atoms ($C_1 = 1$). This arrangement still maintains the total charge of the region constant, while attempting to mimic local dipoles.

In all cases where a virtual charge is created, a force is calculated on it as to balance its electrostatic interactions with QM atoms. Similarly to the Link atom, the total force acting on, e.g., the virtual point charge “ q_+ ” ($\vec{F}_{q_+}$) is redistributed over the MM1 and MM2 atoms that were used to define it. The fraction of the total force that is applied on the MM1 and MM2 atoms is defined, respectively, by

$$\vec{F}_{MM_1} = (1 - C_+) \cdot \vec{F}_{q_+} \quad (4)$$

and

$$\vec{F}_{MM_2} = C_+ \cdot \vec{F}_{q_+} \quad (5)$$

Implementation and interaction with quantum chemistry packages. The implementation of the QM/MM module in NAMD was done entirely in C/C++, making use of Charm++ message-driven load balancing tools so that simulations can be run in parallel over one or several computer nodes. All quantum mechanical calculations are offloaded to a QM package, either through hard-coded interfaces to ORCA [16] or MOPAC [14, 15], or through a standardized interface that utilizes Python scripts (or any other tool) to wrap and convert input and output formats between NAMD and any other arbitrary QM package. A standardized interface was created for data I/O so that external wrapper scripts can be called, providing a translation layer for input and output files from any other QM package. We provide Python scripts that wrap Gaussian [32], TeraChem [33], and Q-CHEM [34], which also serve as templates for the development of new wrapper scripts.

In general, positions and elements of QM atoms are passed on the QM package along with positions and magnitudes of partial charges representing the local MM environment. In return, NAMD expects forces, total energy, and partial charges for QM atoms, and possibly forces acting on MM partial charges due to electrostatic interactions. The exchange of information between NAMD and quantum chemistry packages is preferably done through files written to RAM, which take an insignificant amount of time (milliseconds) when compared to the time scale of QM calculations (seconds or more). By not embedding the QM code directly into NAMD, we allow users to choose their preferred quantum chemistry package and level of theory.

Additionally, NAMD takes advantage of the current advanced state of quantum chemistry packages, which are prepared to be sequentially called by an external soft-

ware, such as in a QM/MM context. After an SCF calculation on a given set of atoms, a “state-files” containing the result of the SCF procedure is saved. A much faster convergence is achieved in the following SCF calculation when the QM package is re-initialized using the saved state-files, as it is already initialized with a very close “guess” of what the SCF solution should be. This occurs because atom positions vary only slightly between consecutive calculations.

Multiple QM regions can be simulated through simultaneous and independent executions of the chosen QM package, one per independent QM region defined in the biomolecular system.

MD simulations of test systems and ttGluRS Structures for the test systems and for ttGluRS were prepared for classical and QM/MM MD simulations employing VMD’s [11] QwikMD [12] graphical interface. While QwikMD assists users in selecting QM regions residue-by-residue through a point-and-click interface, other methods have been developed to provide automated QM region selection [35]. The structure of the pre-transfer complex ttGluRS:tRNA^{Glu}:Glu-AMP had been previously solved by means of X-ray crystallography at 2.1 Å resolution and is available at the protein data bank (PDB:1N78) [36], with the replacement of the inert analog for the active Glu-AMP [19]. The simulations in the present study were performed employing the NAMD molecular dynamics package [9], MOPAC 2016 [14, 15], using PM7 [37], and ORCA 4.0 [16]. The CHARMM36 force field [38] along with the TIP3P water model, for solvated tests, was used to describe all systems. All calculations involving the ttGluRS:tRNA^{Glu}:Glu-AMP system, were done in the NpT ensemble. In energy conservation tests, the simulations used either the PM3 [25, 26] method (using NAMD’s ORCA interface) or RM1 [39] (using NAMD’s MOPAC interface). Results from Amber used its SQM implementation for both PM3 and RM1.

For solvated systems, the simulations were carried out under periodic boundary conditions. Simulations were performed in many steps to ensure a reasonable starting conformation for QM-based simulations: (1) First, for systems explicitly solvated, we employed classical MD in the NpT ensemble with temperature maintained at 300 K using Langevin dynamics for both pressure, kept at 1 bar, and temperature coupling. (2) Using the same parameters from classical NpT simulations, QM-based NpT simulations (either pure-QM or hybrid QM/MM) were carried out. (3) For energy conservation tests, QM-based simulations were performed in the NVE ensemble.

A distance cut-off of 12.0 Å or 17.0 Å (eABF calculations) was applied to short-range, non-bonded interactions, whereas long-range electrostatic interactions were treated using the particle-mesh Ewald (PME) [24] method. For equilibration and classical MM simulations, the equations of motion were integrated using the r-RESPA

multiple time step scheme [9]. For all production QM/MM simulations, the van der Waals interactions as well as electrostatic interactions were updated every time step. The time step of integration was chosen to be either 0.5 fs or 2.0 fs (see details bellow).

The MonoAlanine system is composed of a single alanine residue in vacuum (gas phase). In **Fig. 1b**, the simulations for verification of energy conservation were carried out as described above with 0.5 fs time step.

The PolyAla system is composed of 3 Alanine residues. The same system was used in different ways, either in vacuum or in solution (TIP3 water), with either all 3 residues treated quantum mechanically, or just the central residue as a QM region, leading to the use of 2 Link atoms (**Supplementary Fig. 3a and b**). In **Fig. 1c**, the system is used in vacuum, with the middle residue being treated quantum mechanically. In **Fig. 1e**, the system is used in solution, with the all three residues treated quantum mechanically, and the water molecules treated classically, as also shown in **Fig. 1a** for the NAMD/ORCA interface. Simulations were performed with 0.5 fs time step and the Charge Shift treatment was used for Link Atoms. To reproduce previously reported energy conservation results in Amber, the SPC/FW water model was used in all Amber QM/MM simulations.

Analogously, the N-methyl acetamide (NMA) system was simulated either in vacuum or in solution (TIP3 water), but always entirely treated with QM formalism. In **Fig. 1d**, the system is used in vacuum (gas phase), with the entire molecule being treated quantum mechanically. Simulations were performed with both 0.5 fs and 2.0 fs time step, as shown in **Fig. 1d**.

Network Analysis Suboptimal communication pathways and communities were calculated based on Pearson cross-correlation of atom motion, using a similar protocol to the one described in [18]. Our calculations were based on ten QM/MM trajectories that used 2 fs time steps for a total of 100 ps, totaling 1 ns of QM/MM simulations. For this system two independent QM regions were employed, one in the anticodon domain with approximately 500 QM atoms, one in the catalytic region with approximately 200 QM atoms. Briefly, the network analysis protocol uses central atoms (alpha-carbons in amino acids, the N1/N9 in nucleic bases, and the sugar phosphate P) called “nodes” as proxy for the motion of residues, and their positions throughout MD simulations are used to calculate the Pearson cross-correlation of motion. Only nodes from residues whose non-hydrogen atoms were less than 4.5 Å apart for more than 75% of the total simulated time are considered connected [40]. Nodes that do not fit this cutoff have their cross-correlation multiplied by zero through a mask applied on the cross-correlation matrix. The resulting set of correlations between connected nodes forms a weighted matrix that is used by the Floyd–Warshall algorithm to find

shortest communication paths between nodes (called “suboptimal paths”), and by the Girvan–Newman algorithm to find optimal communities between highly interacting nodes. The suboptimal paths represent the shortest set of connected nodes that could transmit information from two distant sites, such as an allosteric regulator site and an enzyme’s active site.

QM/MM Simulations To study the reaction mechanism of the ttGluRS:tRNA^{Glu}:Glu-AMP system, first a 200 ps QM/MM simulation was performed with 2 fs time step, followed by a 50 ps QM/MM simulation with 0.5 fs time step. These two simulations were used to equilibrate the system in a conformation that was favorable for the reaction mechanism to be investigated. Four mechanisms were tested by inducing the reaction steps to occur (**Supplementary Fig. 10**). The four biased simulations were performed with 0.5 fs time step for 20 to 100 ps. All simulations were carried out using the Charge Shift method to treat Link Atoms, and a “Shift” function was applied to surrounding classical partial charges in the electrostatic embedding scheme.

String Method and eABF In order to study a transformation that occurs in a biomolecular system, such as a chemical reaction or a conformational change, one defines collective variables such as distances between atoms, between centers of mass of groups of atoms, or even angles between subdomains of a molecular structure. The collective variables (colvars for short) are used to track the changes in the system as it undergoes the transformation being studied, and can be used to define a reaction coordinate.

The string method is an iterative process that optimizes a reaction coordinate in order to find the path of least resistance from the initial to the final state of the system. The method uses a discretized representation of the reaction coordinate composed of “images”, where each image is a copy of the entire simulated system at different stages of the transformation. At each iteration, multiple independent MD simulations are initiated from each image, allowing the systems to explore their energy surface and drift toward local minima. Then, average values for the colvars are determined, and biases are applied to keep consecutive images approximately equidistant in colvars space, and to smooth the reaction coordinate. The iterations proceed until consecutive calculations do not produce significant changes in the mean colvar values, indicating convergence of the method.

Once the string has been optimized, the images are used to define a continuous path that defines the transformation, and eABF (extended-Adaptive Biasing Force) is used to calculate free energy changes. Using the images that were optimized through the string method, new path collective variables S and Z are created, which con-

strain the dynamics of the system so it follows the chosen reaction coordinate. The S variable indicates progression along the path, while the Z variable indicates a perpendicular distance to the path. Taking advantage of NAMD’s outstanding scalability and NCSA’s Blue Waters super-computer, we used a parallel strategy that initiates multiple walkers from the different images, we could conduct extensive sampling over the defined path.

Code availability. The reported QM/MM features are publicly available in NAMD 2.12 or later, and VMD 1.9.4 or later. See www.ks.uiuc.edu/Research/qmmm for more details. A Tutorial is available in the same website.

ORCA is available at <https://orcaforum.cec.mpg.de> and <https://www.faccts.de>.

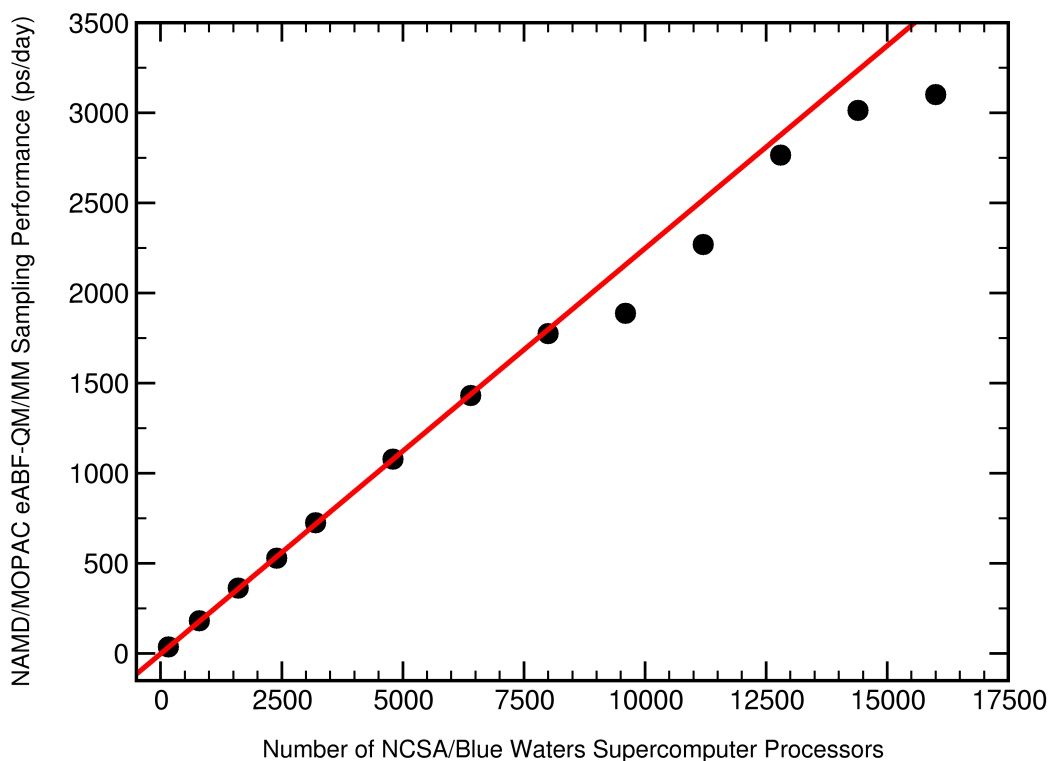
Data availability. The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

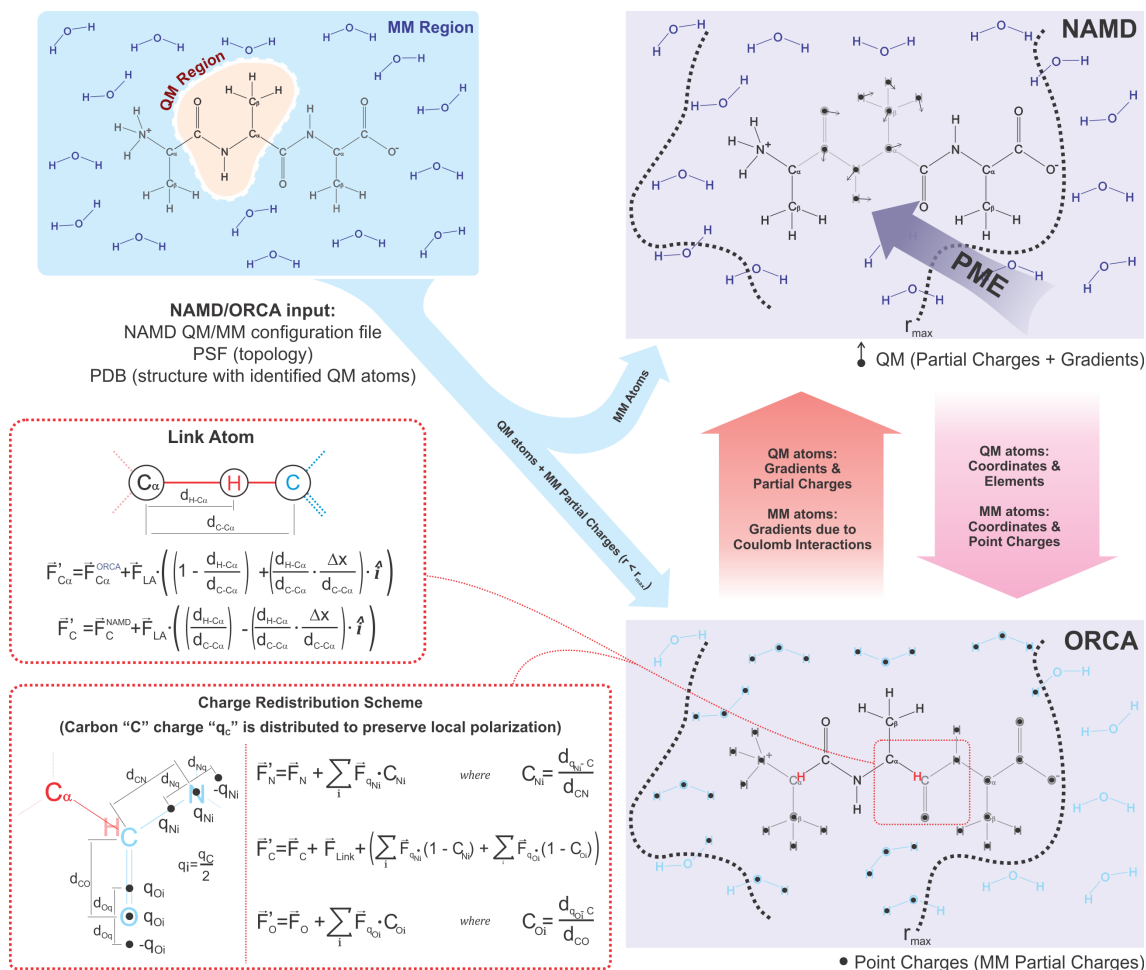
- [21] Riplinger, C., Pinski, P., Becker, U., Valeev, E. F. & Neese, F. Sparse mapsA systematic infrastructure for reduced-scaling electronic structure methods. II. Linear scaling domain based pair natural orbital coupled cluster theory. *J. Chem. Phys.* **144**, 024109 (2016).
- [22] Riccardi, D., Li, G. & Cui, Q. Importance of van der Waals Interactions in QM/MM Simulations. *J. Phys. Chem. B* **108**, 6467–6478 (2004).
- [23] 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.* **103**, 227–249 (1976).
- [24] Darden, T., York, D. & Pedersen, L. Particle mesh Ewald: An Nlog(N) method for Ewald sums in large systems. *J. Chem. Phys.* **98**, 10089–10092 (1993).
- [25] Stewart, J. J. P. Optimization of parameters for semiempirical methods I. Method. *J. Comput. Chem.* **10**, 209–220 (1989). URL <http://doi.wiley.com/10.1002/jcc.540100208>.
- [26] Stewart, J. J. P. Optimization of parameters for semiempirical methods II. Applications. *J. Comput. Chem.* **10**, 221–264 (1989). URL <http://doi.wiley.com/10.1002/jcc.540100209>.

- [27] Sure, R. & Grimme, S. Corrected small basis set Hartree-Fock method for large systems. *J. Comput. Chem.* **34**, 1672–1685 (2013). URL <http://doi.wiley.com/10.1002/jcc.23317>.
- [28] Singh, U. C. & Kollman, P. A. A combined ab initio quantum mechanical and molecular mechanical method for carrying out simulations on complex molecular systems: Applications to the CH₃Cl + Cl exchange reaction and gas phase protonation of polyethers. *J. Comput. Chem.* **7**, 718–730 (1986).
- [29] Walker, R. C., Crowley, M. F. & Case, D. A. The implementation of a fast and accurate QM/MM potential method in Amber. *J. Comput. Chem.* **29**, 1019–1031 (2008).
- [30] Lin, H. & Truhlar, D. G. Redistributed charge and dipole schemes for combined quantum mechanical and molecular mechanical calculations. *J. Phys. Chem. A* **109**, 3991–4004 (2005).
- [31] Sherwood, P. *et al.* Computer simulation of zeolite structure and reactivity using embedded cluster methods. *Faraday Discuss.* **106**, 79–92 (1997).
- [32] Frisch, M. J. *et al.* Gaussian 09, Revision A.02. *Gaussian, Inc. Wallingford CT* (2016).
- [33] Titov, A. V., Ufimtsev, I. S., Luehr, N. & Martinez, T. J. Generating Efficient Quantum Chemistry Codes for Novel Architectures. *J. Chem. Theory Comput.* **9**, 213–221 (2013).
- [34] Shao, Y. *et al.* Advances in molecular quantum chemistry contained in the Q-Chem 4 program package. *Mol. Phys.* **113**, 184–215 (2015).
- [35] Kulik, H. J., Zhang, J., Klinman, J. P. & Martínez, T. J. How Large Should the QM Region Be in QM/MM Calculations? The Case of Catechol O -Methyltransferase. *J. Phys. Chem. B* **120**, 11381–11394 (2016). URL <http://pubs.acs.org/doi/10.1021/acs.jpcc.6b07814>.
- [36] Sekine, S.-i. *et al.* ATP binding by glutamyl-tRNA synthetase is switched to the productive mode by tRNA binding. *EMBO J.* **22**, 676–688 (2003).
- [37] Stewart, J. J. P. Optimization of parameters for semiempirical methods VI: more modifications to the NDDO approximations and re-optimization of parameters. *J. Mol. Model.* **19**, 1–32 (2013). URL <http://link.springer.com/10.1007/s00894-012-1667-x>.
- [38] Best, R. B. *et al.* Optimization of the Additive CHARMM All-Atom Protein Force Field Targeting Improved Sampling of the Backbone ϕ , ψ and Side-Chain χ 1 and χ 2 Dihedral Angles. *J. Chem. Theor. Comp.* **8**, 3257–3273 (2012).

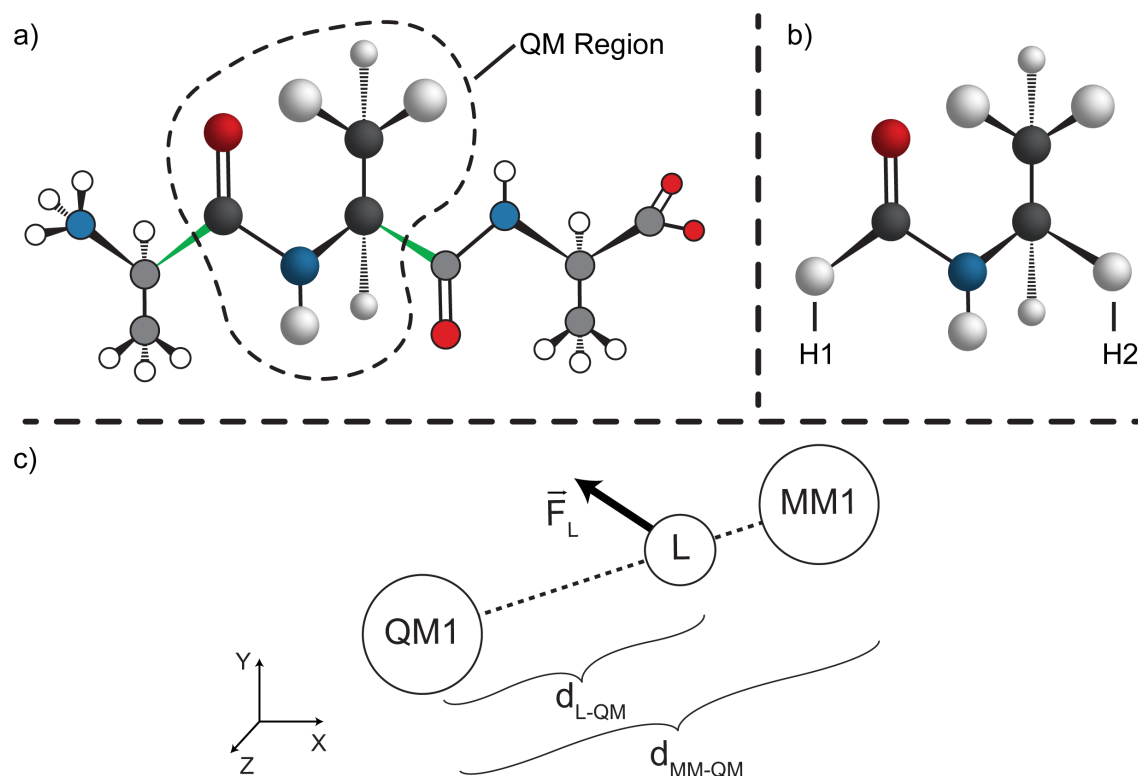
- [39] Dutra, J. D. L., Filho, M. A. M., Rocha, G. B., Simas, A. M. & Freire, R. O. RM1 Semiempirical Quantum Chemistry: Parameters for Trivalent Lanthanum, Cerium and Praseodymium. *PLoS One* **10**, e0124372 (2015). URL <http://www.ncbi.nlm.nih.gov/pubmed/26132289><http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC4489505><http://dx.plos.org/10.1371/journal.pone.0124372>.
- [40] Schoeler, C. *et al.* Mapping mechanical force propagation through biomolecular complexes. *Nano Lett.* **15**, 7370–7376 (2015).



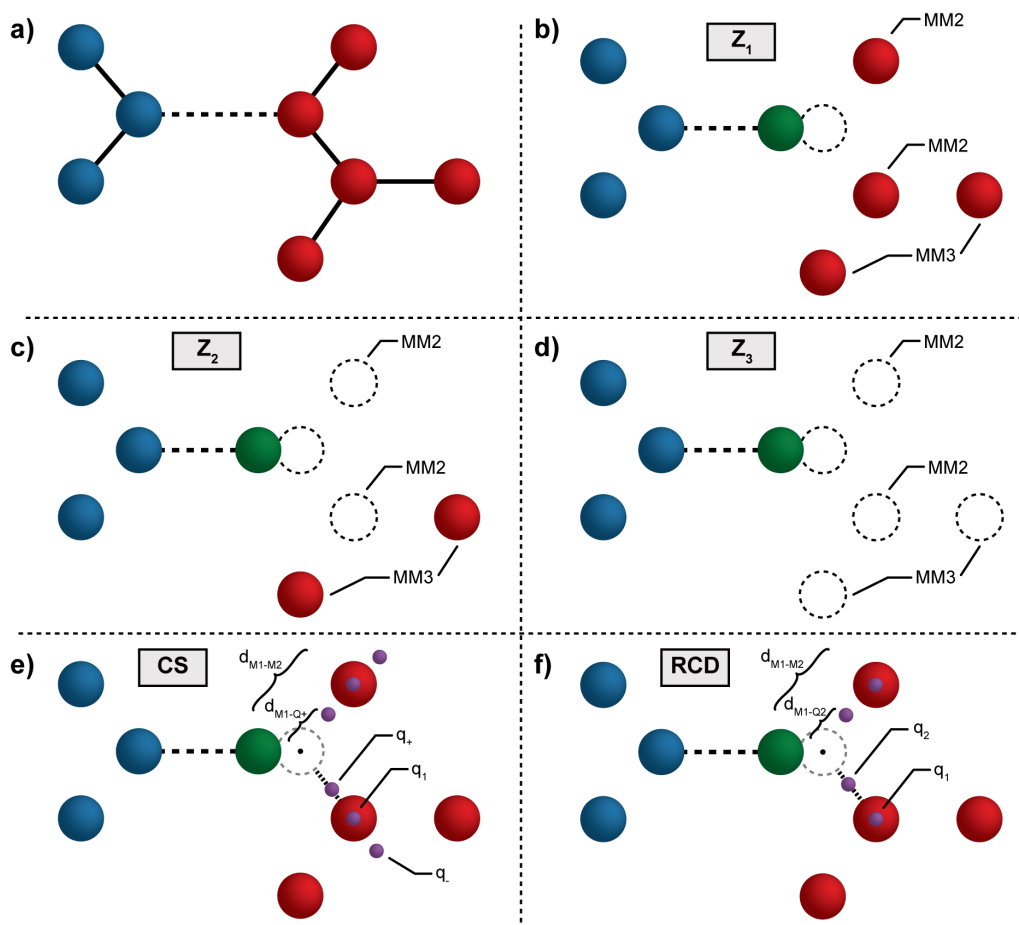
Supplementary Figure 1: **QM/MM eABF Performance Using NAMD/MOPAC.** We performed parallel scaling benchmark calculations of NAMD on NCSA/Blue Waters Supercomputer. The plot shows that NAMD/MOPAC scales very well over thousands of Blue Waters processor cores, obtaining near perfect scaling (red line). The GluRS:tRNA^{Glu}:Glu-AMP complex was employed, in total 263,577 atoms were simulated, with 226 atoms treated using the semiempirical PM7 method. The simulation is the same used to investigate the reaction mechanisms for charging of the tRNA. The extended adaptive biasing force (eABF) method allows for great scalability due to the small communication between simulation replicas. Here each simulation replica uses 10 cores. For each Blue Waters node (32 processor cores), 3 simulation replicas were launched, with 2 cores not assigned for the simulations and assisting with I/O and communication. We tested from 15 (one replica per image) to 1500 replicas (one hundred replicas per image).



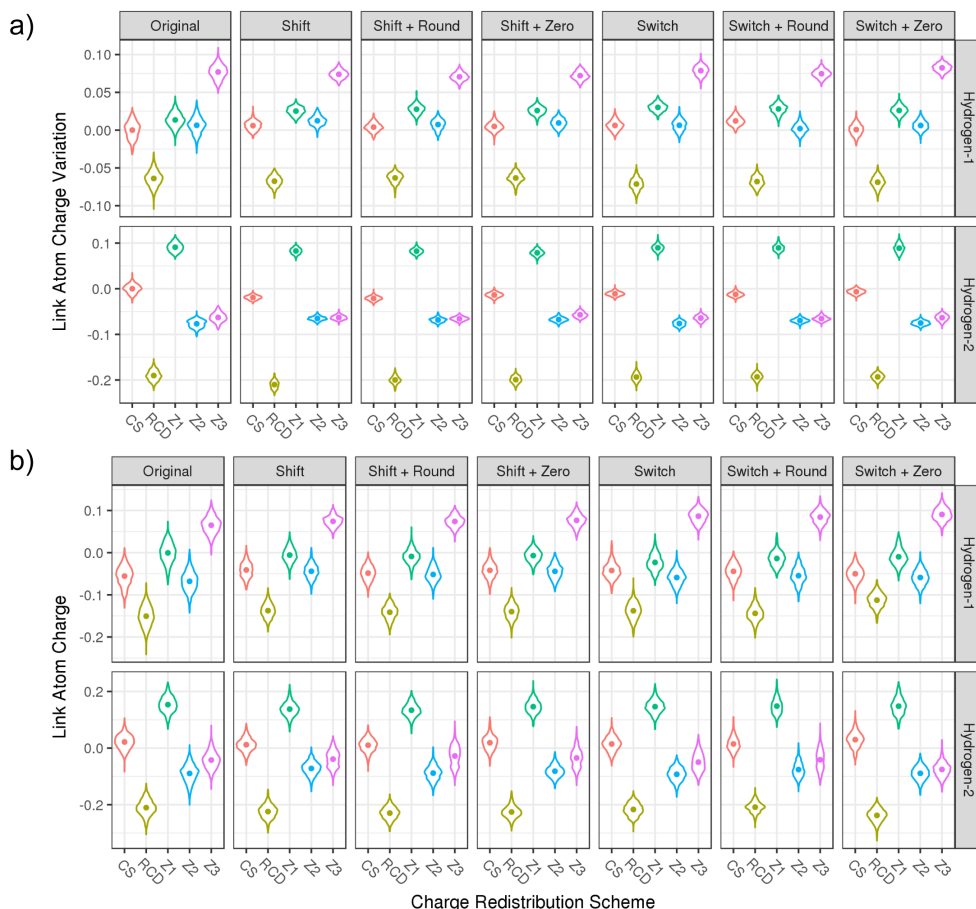
Supplementary Figure 2: **Hybrid QM/MM NAMD - Extended.** Extended schematic of the interconnection between NAMD and ORCA. The contribution of MM charges beyond r_{max} are calculated by NAMD (via PME), while ORCA calculates direct electrostatics. In the representation, we assume the "charge shift" redistribution scheme, where the partial charge of the linking MM atom is shifted to its nearest MM neighbors.



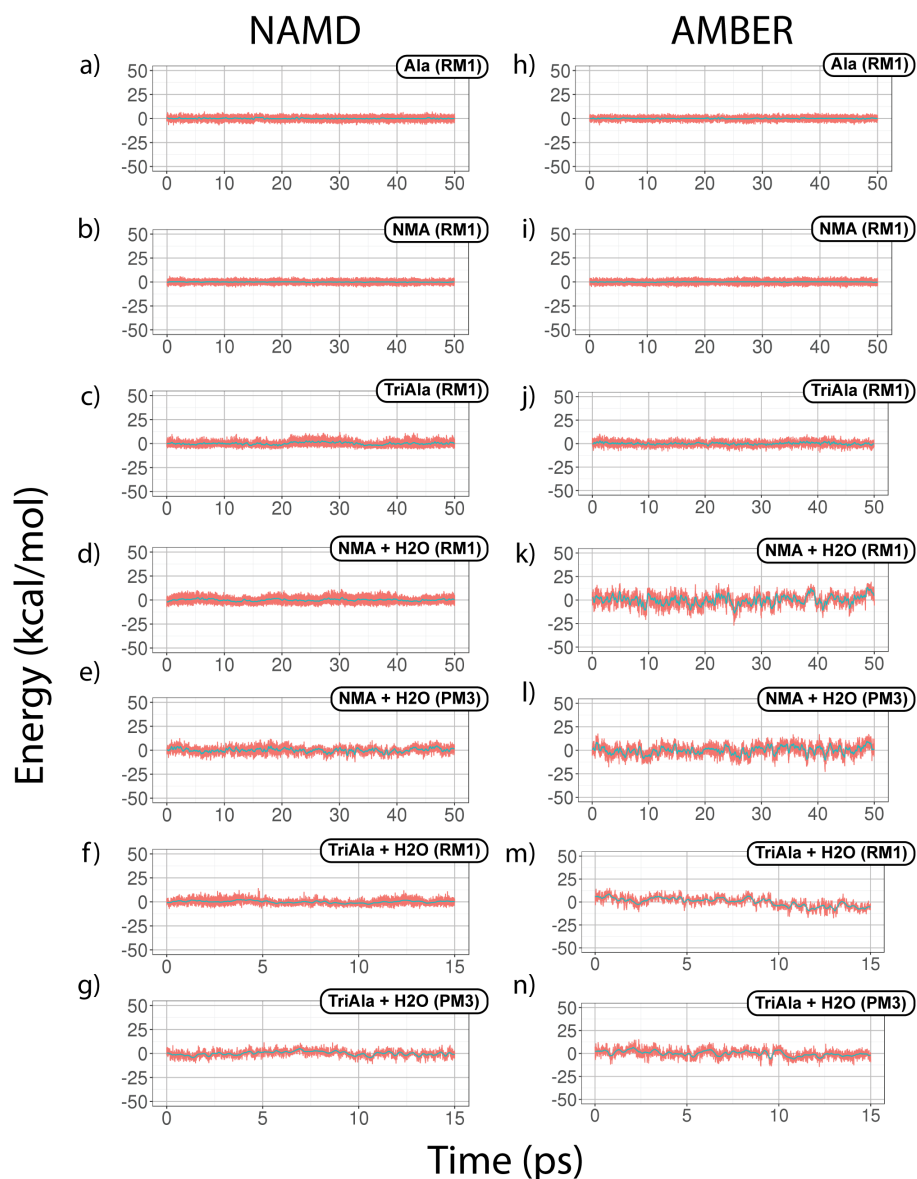
Supplementary Figure 3: **System Used to Compare QM-MM Bond Treatments.** **a)** The QM region in the tri-alanine (TriAla) system is highlighted, indicating which carbon-carbon bonds are being treated with the link atom method. Carbon atoms are in gray, nitrogen atoms in blue, oxygen atoms in red and hydrogen atoms in white. We avoid “breaking” the peptide bond with QM-MM bonds, and instead break along the C-alpha carboxylic-carbon bonds (in green), as to maintain CHARMM’s charge groups intact. **b)** Depiction of the QM region with hydrogen atoms capping the QM-MM bonds. Hydrogen 1 and 2 are indicated, as they are used in Link Atom charge analysis (**Supplementary Fig. 5a and b**). **c)** Link atom placement with respect to QM1 and MM1 atoms. Distances between Link atom and QM1 atom (d_{L-QM}), and between QM1 and MM1 atoms (d_{MM-QM}) are indicated in the figure.



Supplementary Figure 4: **QM-MM Bonds and Charge Redistribution Schemes.** **a)** Illustration of all atoms in the vicinity of the QM-MM bond (dashed line). In blue, the region treated quantum mechanically. In red, the region treated classically. **b)** Atoms marked as “MM2” are directly bonded to the “MM1” atom, and atoms marked “MM3” are directly bonded to “MM2” atoms. The green sphere represents the Link atom, while the dashed circle indicates the position of MM1. This figure depicts the “Z1” scheme, where we simply ignore the partial charge of the MM1 atom. **c)** “Z2” method, where partial charges of MM2 atom are also ignored. **d)** “Z3” method, where partial charges up to MM3 atoms are ignored. **e)** CS method. Virtual point charges, are represented in purple spheres. Distances between the virtual charge and MM1 (d_{M1-Q}) and between MM1 and MM2 (d_{M1-M2}) are indicated in the figure. Charges placed along the MM1-MM2 bond are also indicated (see text). **f)** RCD method.



Supplementary Figure 5: Variations in Atomic Partial Charges Calculated with Different Charge Redistribution Schemes. The figure depicts all possible combinations of link atom treatments currently available in NAMD (see **Supplementary Fig. 4**). The vertical axis shows the variation in the calculated charge for hydrogen atoms 1 and 2 (see **Supplementary Fig. 3**). The violins indicate the variation of charge in a mirrored vertical density plot, while the dots indicate the mean value over the simulation. For each link atom treatment the various electrostatic embedding schemes were applied, as described in the **Online Methods**. “Original” indicates no alteration are made on the surrounding classical partial charges; “Shift” indicates the electrostatic smoothing function is applied to the partial charges; “Switch” indicates only the partial charges beyond the “switch distance” are scaled down. The total sum of partial charges can be rounded to the nearest integer (indicated by “Shift + Round” or “Switch + Round”), or it can be rounded to match the charge of the QM region, so their sum is zero (indicated by “Shift + Zero” or “Switch + Zero”). All calculation were done using ORCA and the TriAla system in a water sphere, as described for the benchmarks and tests (see **Online Methods**). The charge of each hydrogen was acquired during 35 different simulations, one for each combination of point charge treatment and Link Atom treatment. All simulations ran for 2.5ps with 0.5 fs timestep. A total of $n=5,000$ timesteps were used to calculate each mean charge, and violin plot. **a)** The HF-3C method was used for the QM portion of the simulation, and Mulliken charges were calculated for each atom. All charges were normalized by subtracting from their value the mean charge calculated using the CS charge redistribution and the “Original” scheme. **b)** The B3LYP functional and 6-31G** basis set was used for the QM portion of the simulation, and ChelpG charges were calculated for each atom. The charges of hydrogen atoms 1 and 2 were not normalized.



Supplementary Figure 6: **QM Energy Conservation.** QM energy from QM/MM MD simulations, for different systems and conditions, as calculated by NAMD (left column), or Amber (right column). In all plots, the system is indicated in top right hand corner, followed by the method used to calculate the QM energy. NAMD calculations used MOPAC for all RM1 simulations and ORCA for all PM3 calculations. The “Ala” system (**a** and **h**) is composed of a single alanine residue treated entirely with quantum mechanics. The N-methyl acetamide (NMA) system (**b** and **i**) is similarly composed of a single NMA molecule treated entirely with QM. The TriAla system (**c** and **j**) is composed of a peptide of three alanine residues, where only the central alanine is treated quantum mechanically (see **Online Methods**). The NMA + H₂O systems (**d**, **e**, **k** and **l**) has an NMA molecule treated quantum mechanically solvated by a water sphere made up of classical water molecules. The TriAla + H₂O systems (**f**, **g**, **m** and **n**) has a TriAla molecule treated quantum mechanically solvated by a water sphere made up of classical water molecules. All energy plots have been normalized for better comparison of fluctuations. Energy conservation for the total system, and other details, can be found in **Supplementary Table 1**.

QM/MM available in the Advanced Run Tab

A system can be prepared from scratch using a PDB code or browsing for a local PDB file. Alternatively, a previously prepared system can be loaded after a Classical MD run, in order to start a QM/MM simulation.

The Protocol section provides default configuration options for major simulation steps.

The QM Options section allows one to set relevant options related to electrostatic embedding, and to define how the QM calculation will be carried out by ORCA or MOPAC.

If no topology is available for a residue included in a QM region, QwikMD can create an artificial topology file so NAMD can load the system and execute the QM/MM simulation.

One must confirm the elements of all atoms in the residue, as well as the total charge of the residue.

The Structure Manipulation window allows one to perform mutations and other modifications to one's system, as well as selecting or creating topologies for new molecules, such as drugs and metabolites.

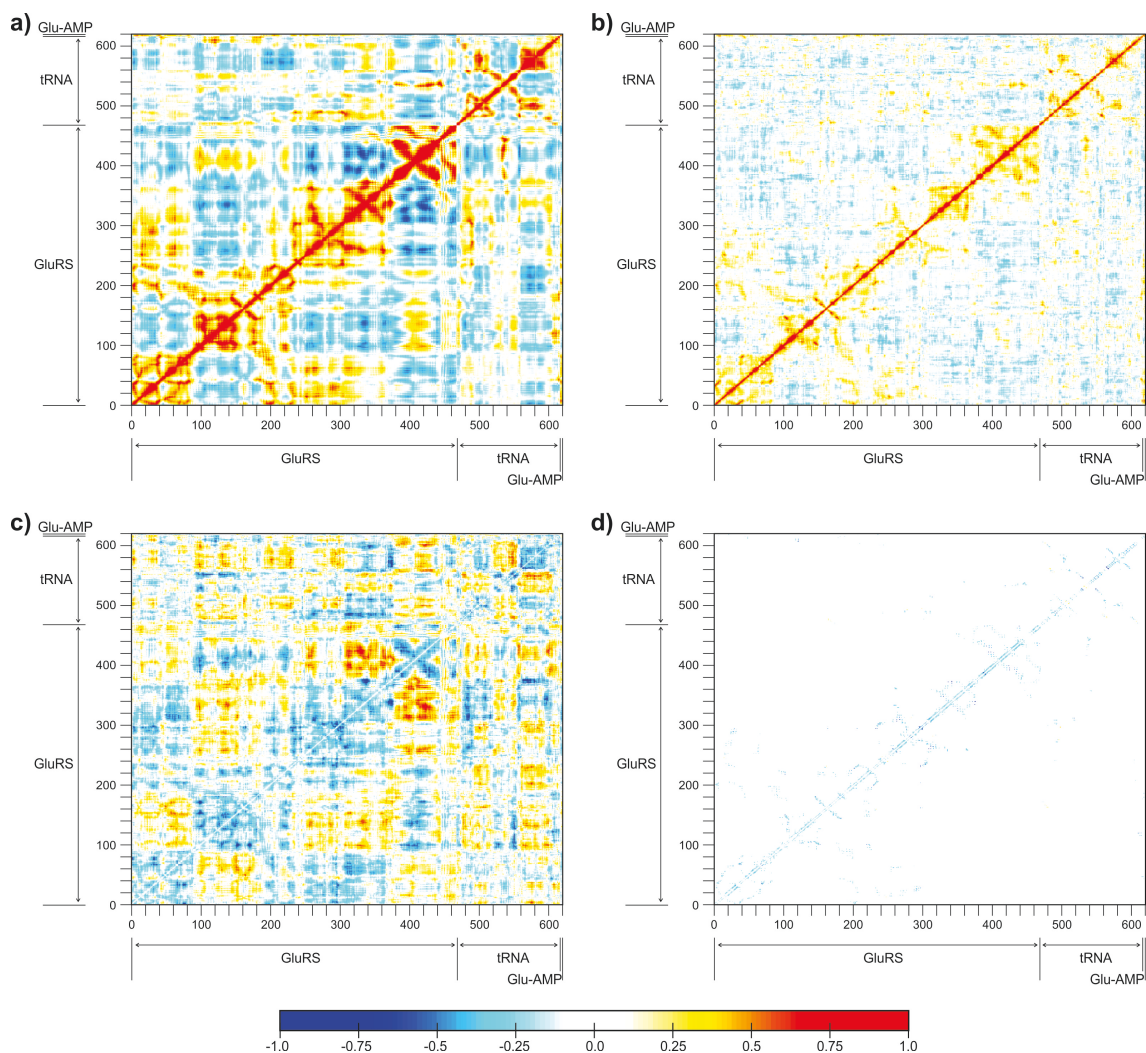
A QM region can be selected in any combination of three basic methods: a) Using the atomselect language and allowing QwikMD to select residues within a cutoff of your selection; b) Selecting residues from the list presented in the "Region Selection" window; c) Clicking on the residues themselves in the main VMD window, as one examines the structure being simulated.

Generate Missing QM Region Topology

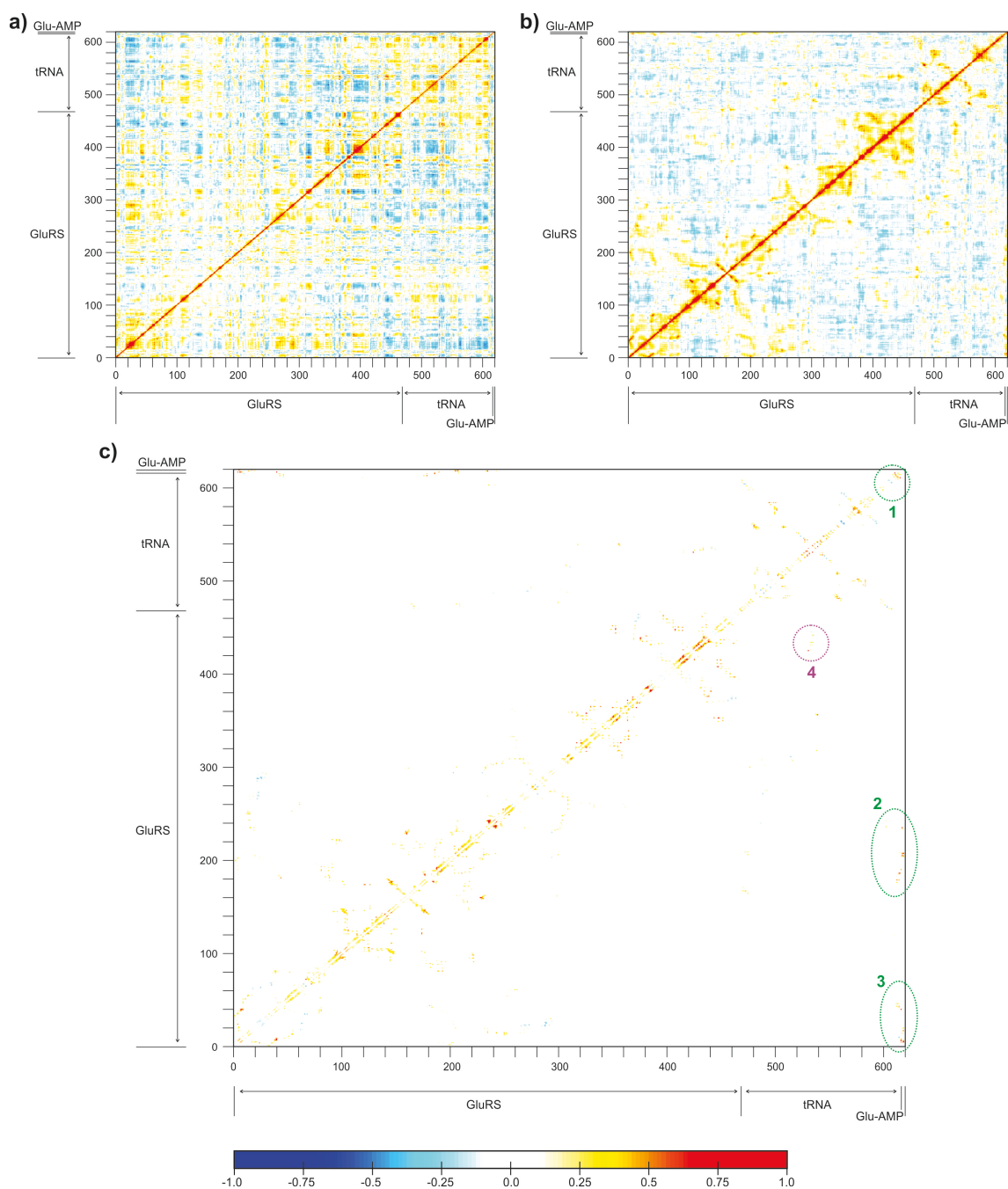
Index	Rename	Res ID	Chain ID	Atom Name	Element	Charge	Type	RES1	NMA1
1	NMA1	1	X	O	O	0.00	hetero		
2	NMA1	1	X	N	N	0.00	hetero		
3	NMA1	1	X	C	C	0.00	hetero	ATOM D	O 0.00
4	NMA1	1	X	C	C	0.00	hetero	ATOM N	N 0.00
5	NMA1	1	X	C	C	0.00	hetero	ATOM C	C 0.00
6	NMA1	1	X	H	H	0.00	hetero	ATOM C	C 0.00
7	NMA1	1	X	H	H	0.00	hetero	ATOM C	C 0.00
8	NMA1	1	X	H	H	0.00	hetero	ATOM H	H 0.00
9	NMA1	1	X	H	H	0.00	hetero	ATOM H	H 0.00
10	NMA1	1	X	H	H	0.00	hetero	ATOM H	H 0.00
11	NMA1	1	X	H	H	0.00	hetero	ATOM H	H 0.00
12	NMA1	1	X	H	H	0.00	hetero	ATOM H	H 0.00

Generate Topology
Select Residue: 1 X File Name: NMA1.rtf
Total Charge: 0.00
Generate Topology
Save Topology Delete Ok Cancel

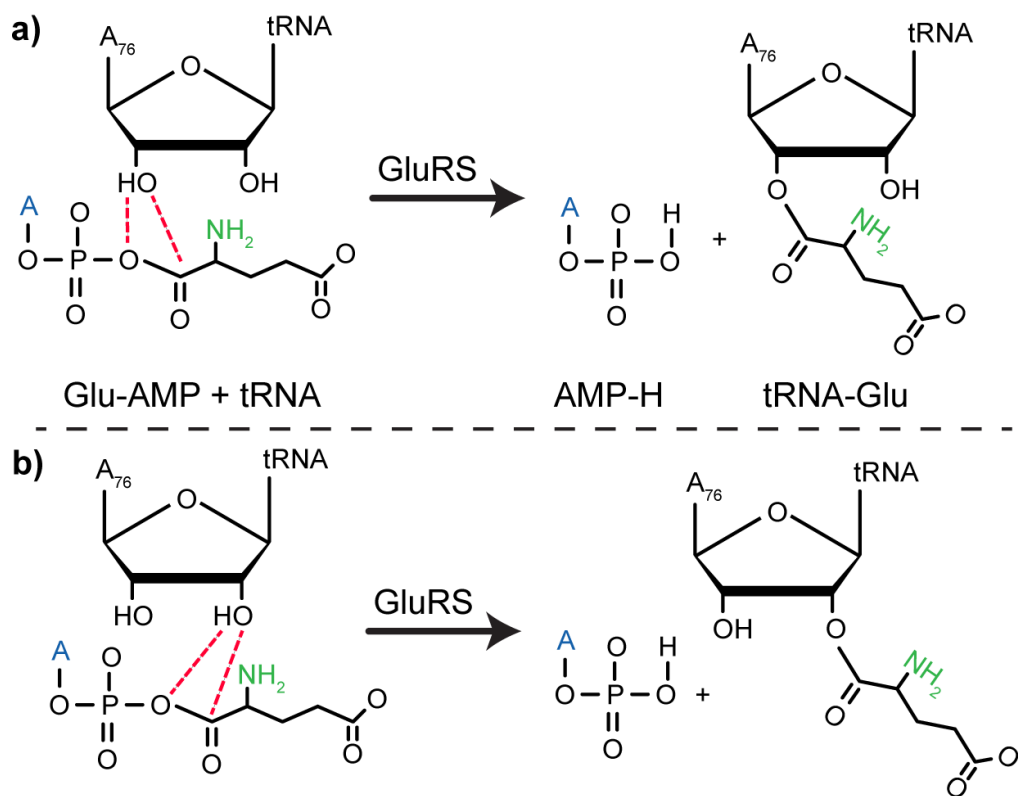
Supplementary Figure 7: **Overview.** Example of QwikMD interface being used to build a minimalistic example of QM/MM simulation. N-methyl acetamide (NMA) is the only molecule in a QM region, and it is solvated by a classical water box.



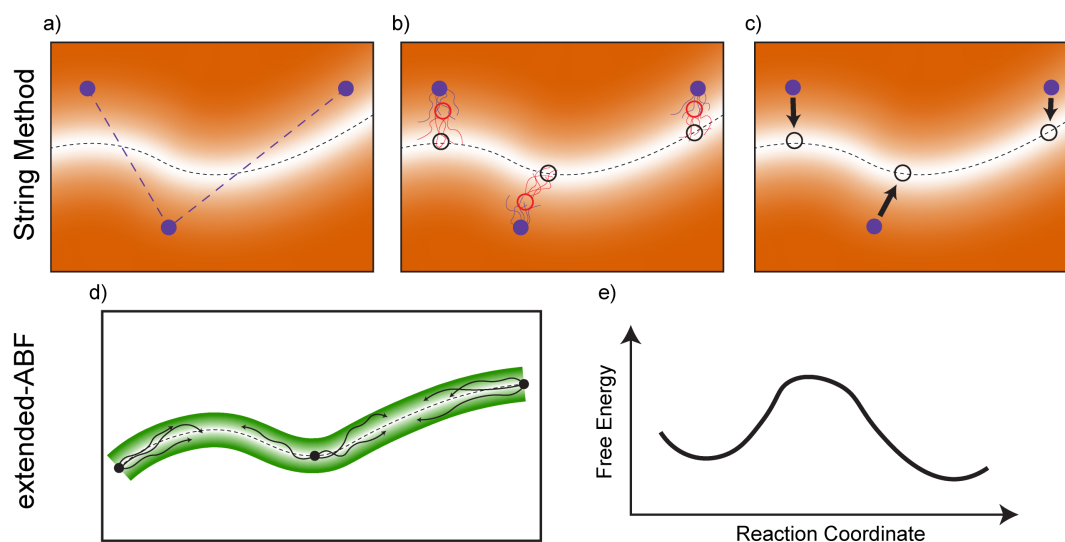
Supplementary Figure 8: Cross-correlation Matrix - MM 20ns vs 10x100ps. **a)** Cross-correlation matrix calculated from a single ($n=1$) 20ns classical MD simulation of the GluRS:tRNA^{Glu}:Glu-AMP complex. **b)** Cross-correlation matrix calculated from the union of 10 independent 100ps classical MD simulations of the GluRS:tRNA^{Glu}:Glu-AMP complex. **c)** Difference between the cross-correlation matrices in **a)** and **b)**. **d)** Same data as the cross-correlation matrix in **c)** after applying a mask that only retains correlations used in network analysis, that is, only retains correlations between nodes whose heavy-atoms are less than 4.5 Å apart for at least 75% of the simulated time. From **d)**, we can conclude that the network obtained from both protocols will represent similar behavior, which validates the protocol used for the QM/MM simulations (see **Supplementary Fig. 9**).



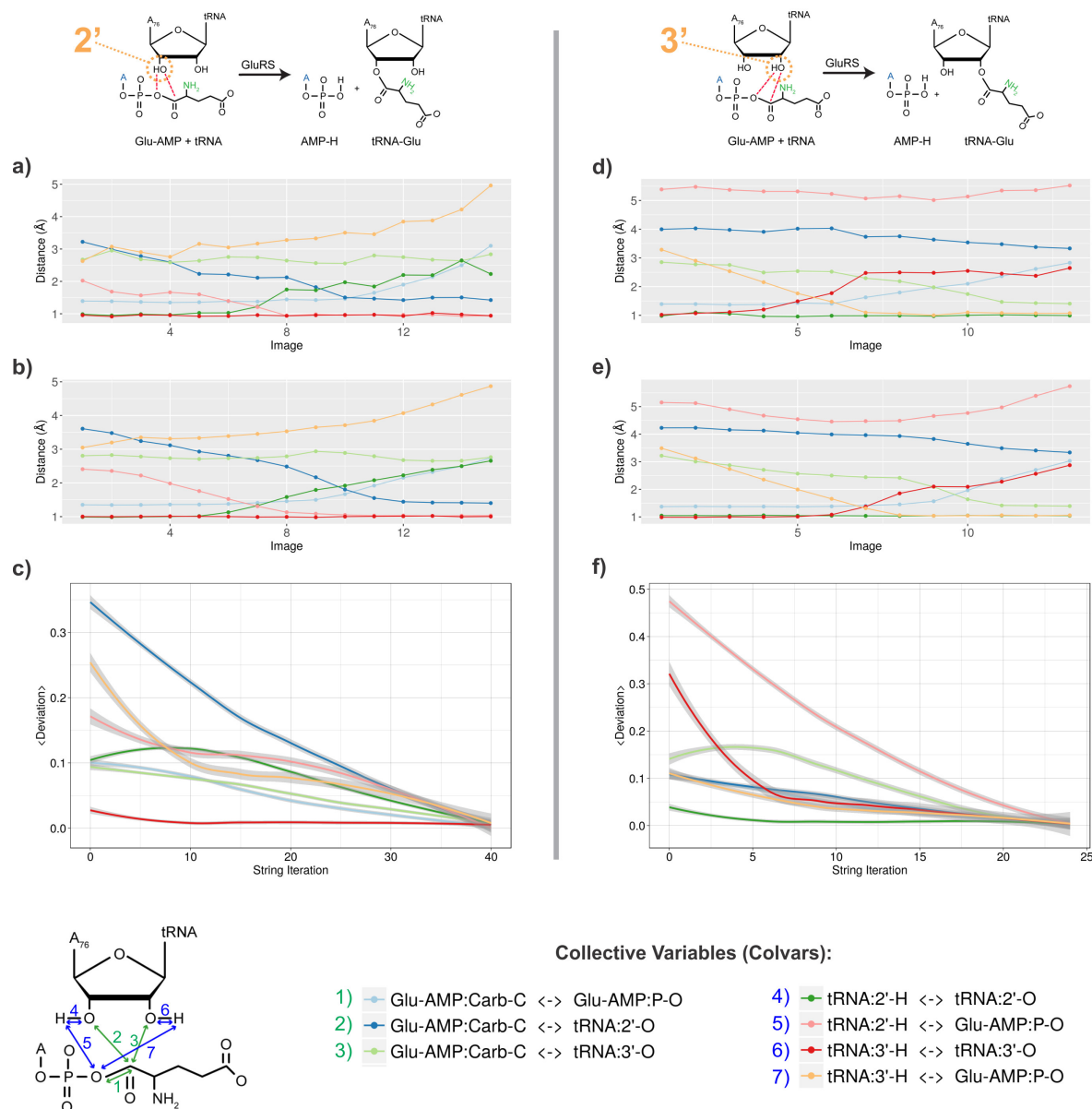
Supplementary Figure 9: **Cross-correlation Matrix - MM vs QMMM.** **a)** Cross-correlation matrix calculated from the union of 10 independent 100ps classical QM/MM MD simulations of the GluRS:tRNA^{Glu}:Glu-AMP complex. **b)** Cross-correlation matrix calculated from the union of 10 independent 100ps classical MD simulations of the GluRS:tRNA^{Glu}:Glu-AMP complex. **c)** Difference between the cross-correlation matrices in **a)** and **b)**, combined with a mask that only retains correlations used in network analysis. Analogous to **Supplementary Fig. 8d**. Four regions have been circled to indicate the differences between interactions calculated with pure MM versus QM/MM simulations. Regions 1, 2 and 3 are in the active site, while 4 is in the anticodon region.



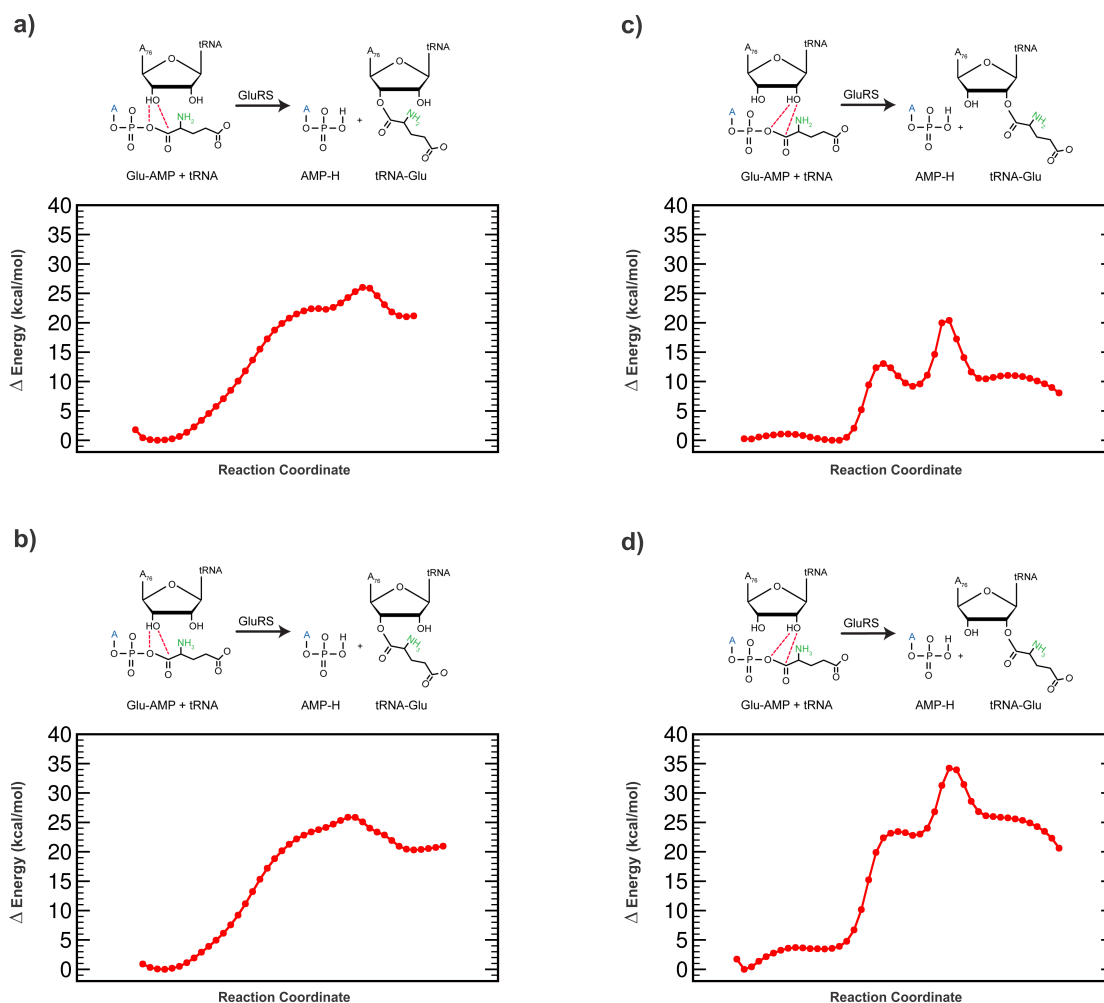
Supplementary Figure 10: **Investigated Reaction Mechanisms for Charging of the tRNA.** Illustration of investigated concerted reaction mechanism for binding at **a)** 2'-OH group of tRNA or **b)** 3'-OH group of tRNA (see text). The mechanism is based on concerted proton transfer and attack on carbonyl carbon of Glu-AMP (red dashed lines). Simulations were performed with either a protonated or deprotonated amino group of GluAMP (in green), leading to four possible mechanisms.



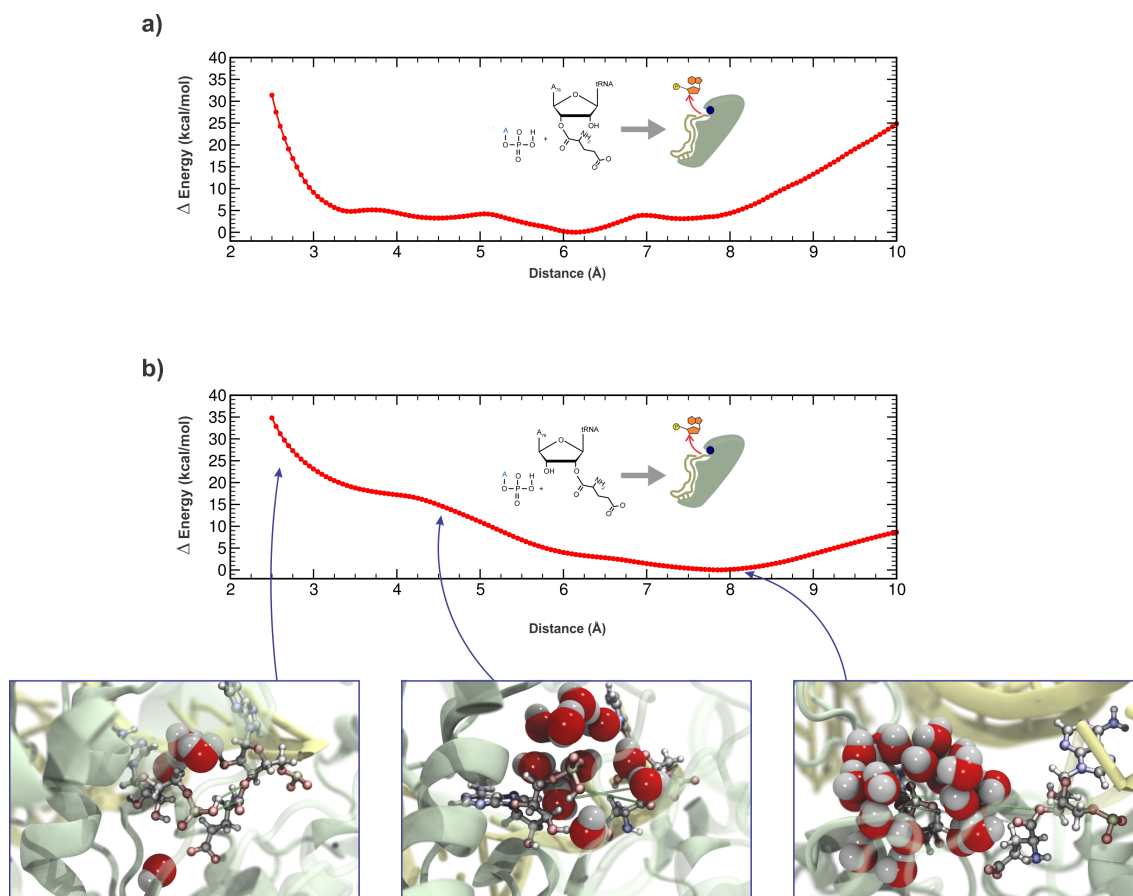
Supplementary Figure 11: **Overview of Protocol for String Method and eABF.** **a)** The example depicts a 2D energy surface governed by two collective variables, where the orange-to-white gradient indicates a high-to-low energy landscape. The solid purple circles (which we call “images”) connected by a long-dashed lines indicate different states along a transformation. The simplified example has, from left to right, an initial, an intermediate and a final states. **b)** During the string optimization, multiple independent MD simulations are initiated from each image (indicated by the groups of five solid lines), and after a small number of steps the average values of the collective variables (“colvars”) are calculated and used to update the position of the image (indicated by the overlapping open red circle). The iterative process begins again from the new centers and creates new updated images (represented by the red solid lines stemming from the red circles, culminating on the black open circles), and is repeated until no significant change is produced in image positions. **c)** The converged calculation guides the images towards energy minima. **d)** For eABF, the optimized images are then used to define a path using path collective variables, which constrain MD simulations to follow a “tunnel” in colvar space. Groups of independent biased MD simulations are initialized along the path and sample the path optimized by the string method. **e)** After several rounds of simulations, the sampling generates a Potential of Mean Force that shows the free energy change along the path.



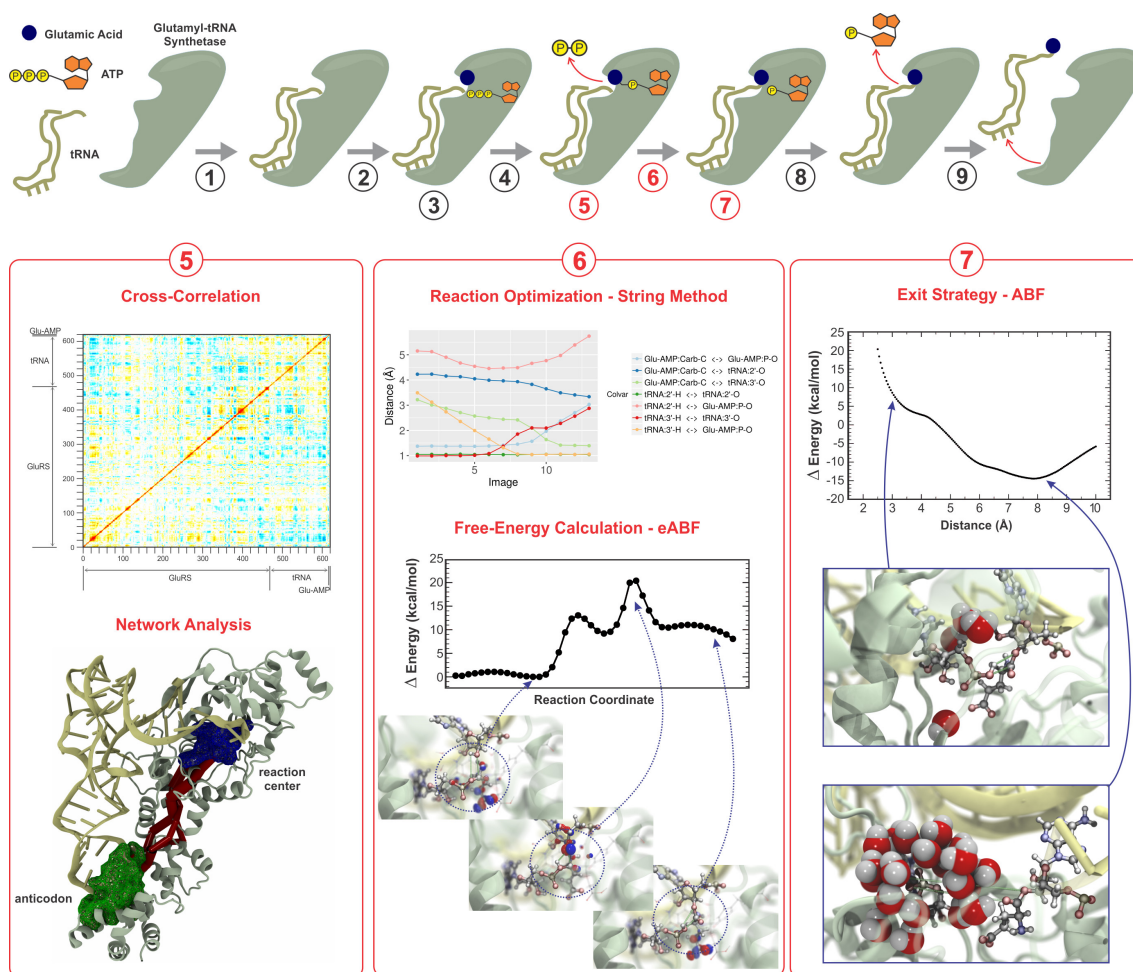
Supplementary Figure 12: String Convergence. String optimization of 7 collective variables that controlled atom distances for key atoms involved in the different reaction mechanisms. See caption and **Supplementary Fig. 10**. All colvars are measured in Ångstroms. Figures **a)** and **d)** represent the initial (pre-optimization) values of the collective variables for all images along the strings for the 2' reaction and 3' reaction, respectively. Figures **b)** and **e)** represent the final optimized values of the colvars for the 2' reaction and 3' reaction, respectively. Figures **c)** and **f)** show the mean deviation of the colvars along the iterative string optimization. The deviation for a colvar was defined as the difference between the average colvar value across all 15 walkers at each iteration and the average value at the last iteration. String optimizations were run until average deviations were below 0.2 Å for at least 10 iterations. The same colors were used to identify the different colvars across all images, and the legend indicates the pairs of atoms that are used to measure the distance in each colvar.



Supplementary Figure 13: **Free Energy Profiles of the Analyzed Reaction Mechanisms for Charging of the tRNA.** The GluRS:tRNA^{Glu}:Glu-AMP complex comprised 263,577 atoms, and 226 atoms were treated using the semiempirical PM7 method for the active site. Free energy profiles were calculated from collective variables optimized by the String Method, and the string images were used to define Path Collective Variables for the eABF calculation. **a)** Free energy profile for the 2' reaction with de-protonated amino group. **b)** Free energy profile for the 2' reaction with protonated amino group. **c)** Free energy profile for the 3' reaction with de-protonated amino group. **d)** Free energy profile for the 3' reaction with protonated amino group. For all systems, the free energy for the products is higher than the one for the reactants. This is due to the complex exit strategy of AMP, which involves solvating its PO₄ group. This process was investigated using classical MD ABF calculations (see **Supplementary Fig. 14**).



Supplementary Figure 14: **Mechanism of AMP-H exit.** The free energy of solvation of the AMP-H, was investigated using adaptive biasing force (ABF). The free energy profile over the distance between Glu-AMP:Carb-C and Glu-AMP:P-O (see **Supplementary Fig. 12**) was used to investigate this mechanism of solvation. Solvation of AMP-H is a first step in the exit strategy of this ligand. **a)** Free energy profile for the products of the 2' reaction. **b)** Free energy profile for the products of the 3' reaction. Only products with an uncharged amino group were investigated, as the exit mechanism is known to be facilitated by NH₂.



Supplementary Figure 15: **Overview of the Interaction and Charging Mechanism for the GluRS:tRNA^{Glu} Complex.** The top illustration presents the overall mechanisms of charging of the tRNA by GluRS. The individual steps indicate (1) the binding of the tRNA to the protein, (2) the binding of ATP and Glutamate, (3) the adenylation of the amino acid, (4) the un-binding of the pyrophosphate, (5) the interaction between tRNA and GluRS where the allosteric signal from the anticodon induces changes in the active site, (6) the aminoacylation of the tRNA, (7) the rearrangements of the active site after the reaction has occurred and solvation of the protonated AMP, (8) the exit process of the protonated AMP and the release of the charged tRNA 3' end, and (9) the unbinding of the tRNA and GluRS. **Box 5:** From the cross-correlation matrix of interacting nodes within the protein-tRNA complex, the sub-optimal paths between the two QM regions are determined. **Box 6:** Indicates the optimized collective variables for the reaction of charging the tRNA on the 3' hydroxyl group, and the Free Energy profile calculated using eABF along the optimized path. The snapshots illustrate the three key stages prior to the reaction, during the reactions and after the reaction has occurred. **Box 7:** Free Energy profile for the release of the AMP group. The snapshots depict how the solvation drastically changes as the Phosphate group distances itself from the charged tRNA.

Supplementary Table 1: Energy conservation comparison for multiple systems and conditions

System	NAMD		Amber 16	
	<Total> (kcal/mol)	Drift (kcal/mol/ps)	<Total> (kcal/mol)	Drift (kcal/mol/ps)
Ala (RM1)	-77.17 $\pm$ 0.07	0.003	-84.53 $\pm$ 0.35	0.004
NMA (RM1)	-37.40 $\pm$ 0.04	0.002	-36.08 $\pm$ 0.09	0.0007
TriAla (RM1)	-166.81 $\pm$ 0.03	0.00001	-89.00 $\pm$ 0.20	0.002
NMA + H ₂ O (RM1)	-3233.27 $\pm$ 0.05	0.0007	-3243.34 $\pm$ 0.29	0.001
NMA + H ₂ O (PM3)	-24131.45 $\pm$ 0.05	0.0002	-3236.10 $\pm$ 0.27	0.001
TriAla + H ₂ O (RM1)	-4934.12 $\pm$ 0.06	0.0005	-13334.66 $\pm$ 0.59	0.005
TriAla + H ₂ O (PM3)	-26154.72 $\pm$ 0.06	0.0009	-13327.59 $\pm$ 0.59	0.004

Energy conservation results for the same systems and methods as described in **Supplementary Fig. 6**. The values in the table are derived from a linear regression calculated on the Total Energy of each system during a 50 ps NVE QM/MM MD simulation, except for the solvated TriAla simulations, where 15 ps were simulated (n=1 independent simulation was used per system). The <Total> column indicates the mean of the Total Energy over n=100,000 steps of simulation, except for the solvated TriAla simulations, where an n=30,000 simulation steps were used (the energy was saved at every step). The error was calculated as the standard deviation of the Total Energy. The "Drift" indicates the angular coefficient of a linear fit over the Total Energy. All systems were equilibrated with a 50 ps NVT QM/MM MD simulation. It is important to highlight that NAMD used MOPAC for the RM1 simulations and ORCA for the PM3 simulations, which lead to significant differences in the reported energy. That is because MOPAC reports the Heat of Formation for the calculated QM system while ORCA reports the Total Energy. Despite the difference, the connection with ORCA and MOPAC guarantees energy conservation, as shown by the low energy drifts.

Supplementary Table 2: Energy conservation using PME

	PME		Cutoff (12 Å)	
Method	<Total> (kcal/mol)	Drift (kcal/mol/ps)	<Total> (kcal/mol)	Drift (kcal/mol/ps)
PM3	-31099.06 $\pm$ 3.55	1.580	-31763.75 $\pm$ 0.73	0.392
HF-3C	-164026.20 $\pm$ 4.08	1.274	-164675 $\pm$ 1.41	0.879
B3LYP 6-311G**	-165904.5 $\pm$ 6.07	2.899	-166552.1 $\pm$ 1.83	1.099

All simulations consisted of an N-methyl acetamide (NMA) molecule solvated in a water box with no ions. The QM region consisted of the entire NMA molecule (12 atoms), calculated with different methods available in the ORCA package. The <Total> column indicates the mean of the Total Energy over n=10,000 steps of simulation (the energy was saved at every step). The error was calculated as the standard deviation of the Total Energy. The "Drift" indicates the angular coefficient of a linear fit over the Total Energy.

Supplementary Table 3: Benchmarks

System	Number of atoms		Average Performance (ps/day)				
	Full System	QM Region	PM7	PM7 (GPU)	PM7 (MOZ.)	HF (3-21G)	B3LYP (6-31G*)
NMA	12	12	9,600	104	10,164	96	29
NMA + H ₂ O	3648	12	4,800	104	4,800	77	27
TriAla	33	10+2	9,600	104	10,164	90	30
TriAla + H ₂ O	4115	10+2	8,228	104	8,228	87	28
Active Site	263577	213+13	51	21	69	—	—
TrpCage	8360	284	28	16	64	0.06	—
BR [†]	80594	3663	—	—	0.89	—	—
GluRS [‡]	263577	7684	—	—	0.45	—	—

Benchmarks comparing average time for a NAMD timestep, using different theory levels, software implementations and system sizes. The Full System column indicates the total number of atoms in the simulated system, while the QM Region column indicates the number of atoms sent to the QM code for computation of forces, charges and energy. PM7, PM7 (GPU) and PM7 (MOZ.) columns reflect performance when NAMD used MOPAC with, respectively, multithreaded, multithreaded with GPU, and MOZYME implementations. HF and DFT columns indicate performance while using ORCA, and varying parameters. Both the NMA and TriAla systems were simulated with and without a water box (solvated systems are indicated in the table by "+ H₂O"). The GluRS system (composed of the GluRS:tRNA:Glu-AMP complex), TrpCage and the Bacteriorhodopsin system (which was embedded in a lipid bi-layer) were simulated in a water box. For GluRS, TrpCage and Bacteriorhodopsin systems, all amino acid residues were calculated quantum mechanically. The "Active Site" results represent the catalytic region of the GluRS:tRNA:Glu-AMP complex, containing the adenylate, the A₇₆ residue of the tRNA and several amino acids. The number of atoms in the QM region for the TriAla systems indicates the number of QM atoms in the peptide (only the inner residue) plus the number of link atoms (one per QM/MM bond). Analogously, for the Active Site system, QM Region column indicates the number of atoms from the protein, tRNA and Glu-AMP plus the number of link atoms. [†] Bacteriorhodopsin. [‡] GluRS:tRNA^{Glu} system with the entire protein and ligand in a single QM region.

Supplementary Table 4: CPU C_{60} benchmark with 516x519x507 grid

POWER8 160 Threads		Xeon E5-2660v3 40 Threads		Xeon Phi 7250 68 Threads	
C++	VSX	C++	SSE	C++	AVX-512ER
17.57 s	8.03 s	59.18 s	7.14 s	53.04 s	1.47 s
1.0×	2.18×	1.0×	8.28×	1.0×	36.06×

VMD C_{60} molecular orbital kernel performance with varying CPU thread counts and SMT utilization, and vectorization approach. Speedups are shown normalized to the performance of pure C++ implementations. The Xeon Phi results show a tremendous performance advantage for a kernel hand-written using AVX-512ER intrinsics, as compared with plain C++. The hand-written AVX-512ER kernel benefits significantly from the use of the fast `_mm512_exp2a23_ps()` exponential approximation intrinsic, accounting for roughly 30% of the execution time. The new AVX-512ER kernel performs best with 68 threads (one thread per physical CPU core) on Xeon Phi since it is arithmetic-bound. The AVX-512ER kernel running on Xeon Phi is the highest performance CPU result we have measured to date. Our previously published results on POWER8 and Xeon CPUs are provided for comparison [1].

Supplementary Table 5: GPU-accelerated C_{60} benchmark with 516x519x507 grid

POWER8-PCIe + Tesla K40m			POWER8-NVLink + Tesla P100			Xeon E5-2697Av4 + Tesla V100	
1	2	4	1	2	4	1	2
3.49 s	1.76 s	0.91 s	1.09 s	0.56 s	0.30 s	0.610 s	0.294 s
1.0×	1.98×	3.83×	1.0×	1.95×	3.64×	1.0×	2.0×

GPU-accelerated VMD C_{60} molecular orbital performance with multiple GPU hardware types, and varying GPU counts. Speedups are normalized to the performance a single GPU of each type. The Tesla K40m results are representative of the performance obtainable by previous-generation “Kepler” family of NVIDIA GPUs. The Tesla P100 results are representative of a variety of GPUs that are based on the “Pascal” GPU architecture. The Tesla-V100 result represents state-of-the-art GPU performance using the “Volta” GPU architecture. Volta GPU performance is roughly twice that of previous GPU hardware, and is the highest performance result we have measured on any platform to date.

Supplementary Table 6: GPU-accelerated glutamyl-tRNA synthetase trajectory animation performance

Grid Size	Orbitals	Cartoon	DynamicBonds	Total	Frames/sec
92x81x75	0.19 s	0.1 s	0.01 s	0.30 s	3.3 FPS
144x128x118	0.37 s	0.1 s	0.01 s	0.48 s	2.1 FPS
200x178x165	0.79 s	0.1 s	0.01 s	0.90 s	1.1 FPS
336x296x275	3.54 s	0.1 s	0.01 s	3.65 s	0.27 FPS

VMD trajectory animation display rates and runtimes for graphical representations used in the visualization of the glutamyl-tRNA synthetase simulation shown in Fig. 3. The test scene contained a total of 10 molecular orbital representations, 4 “cartoon” secondary structure ribbon representations, 4 dynamic bond representations, and 2 Van der Waals representations for the QM/MM simulation trajectory. The timings reported here represent the average time per frame while animating a series of trajectory frames, requiring recomputation of molecular orbital grids and associated isosurfaces, cartoon secondary structure ribbons, and dynamic bonds for each frame. The visualization performance shown for the graphical representations associated with the atomic structure is effectively constant due to a fixed set of selected atoms, but performance for the molecular orbital representations varies substantially according to the spatial resolution of the orbital grids computed and the orbital surface meshes that are ultimately displayed. All tests were performed on a workstation with dual NVIDIA Quadro-GP100 GPUs (Pascal GPU architecture).

Supplementary Notes

Orbital Visualization

VMD allows for easy and straight-forward implementation in C/C++ of plugins for novel file type read-in through the *molfile plugin* interface. Several other molfile plugins to read output of QM packages have been previously developed, including GAMESS-UK and Gaussian, allowing for fast representation of molecular orbitals [2].

To extend the analysis capabilities for the QM/MM suite and its supported QM packages, we implemented a molfile plugin for both ORCA and MOPAC. The plugin is able to read a plethora of job types, such as single point calculations, geometry optimizations, and gradient calculations. Geometries, orbitals and atomic charges are parsed from the output files and directly loaded into VMD for easy analysis and visual inspection. A major feature is the capability to view a trajectory of molecular orbitals (MOs) generated by a QM/MM simulation. Trajectories can be viewed by loading the appended output files for energy and gradient calculations performed by either ORCA or MOPAC during the QM/MM simulation without any workaround. VMD’s user-friendly interface further provides great simplicity in orbital visualization, allowing for quick selection of the orbital of interest (e.g. HOMO or LUMO). The orbital rendering implemented in VMD is extremely fast and outperforms visualizations performed using the so-called cube files produced by most QM packages. The VMD implementation only requires a plain ORCA or MOPAC output file, and allows rendering of orbitals at arbitrarily high resolutions. For advanced analysis, other properties as point charges, gradients, Hessians, SCF energies are accessible through the VMD Tcl scripting interface.

Orbital visualization in theory In quantum chemistry ab initio or DFT methods, molecular orbitals (MOs) are expressed in a discrete basis set expansion of Gaussian-type orbitals (GTOs). The GTOs are given by the basis set definition, where multiple GTOs are contracted to a single GTO basis function.

Let $\{\varphi_j\}$ be a set of molecular orbitals and $\{\chi_i^{\alpha\beta\gamma}\}$ a set of contracted GTOs, given by

$$\chi_i^{\alpha\beta\gamma} = N_i x^\alpha y^\beta z^\gamma \exp(-\zeta_i r^2), \quad (6)$$

with the normalization factor N_i , the exponent ζ_i and angular momenta α, β, γ , where the total angular momentum of the orbital is $l = \alpha + \beta + \gamma$.

The MOs are given by linear combinations of GTOs through

$$\varphi_j = \sum_i^{\text{nbas}} c_i \chi_i^{\alpha\beta\gamma} \quad (7)$$

$$\varphi_j = \sum_i^{\text{nbas}} c_i N_i x^\alpha y^\beta z^\gamma \exp(-\zeta_i r^2) \quad (8)$$

The matrix c_{ij} is printed out by QM ab initio or DFT methods as the MO coefficients corresponding to the respective basis functions.

A complication for loading MO coefficients into VMD and using them for visualization is the fact that there are two different representations of wavefunctions of higher angular momentum, namely Cartesian or pure spherical harmonics representation. For angular momenta $l \geq 2$, the degeneracy of the Cartesian and pure representation differs, together with the mathematical formulation of the wavefunctions as such. As ORCA can only print out the MO coefficients in pure representation and VMD needs coefficients in Cartesian representation for orbital rendering, the plugin needs to convert the coefficients during read-in. We achieve this by a simple matrix multiplication of the coefficient block of the respective orbital in order to create new linear combinations from pure MO coefficients.

Molecular orbital visualization performance To visualize the shape of a certain MO φ_j , a so-called isosurface of the MO is needed. Therefore, a surface is rendered through points where the MO has a certain value, i.e. $\varphi_j(x, y, z) = \text{isovalue}$. The isovalue can be set in the graphical interface of VMD. Interactive computation and display of MOs is achieved by high performance parallel MO algorithms for state-of-the-art CPUs and GPUs [3, 2, 4, 5, 1]. The MO grid calculation step is the most computationally demanding, and is something we have continued to optimize to enable high quality interactive visualizations to be performed. We have developed new MO visualization algorithms utilizing Intel x86 AVX2 (8-way) and AVX-512 (16-way) vector instructions supported by the latest Intel Xeon Phi KNL (Knights Landing), and Skylake generation CPUs, reducing the performance gap between CPUs and GPU accelerators for the most time consuming MO visualization calculations.

Below we summarize MO grid calculation performance results for a challenging high resolution C_{60} visualization test case, shown for CPUs in **Supplementary Table 4**, and for GPUs in **Supplementary Table 5**. The work associated with MO grid calculation scales linearly with the total number of grid points (grid point counts grow cubically when halving the uniform grid spacing). Smaller grids in the range of $172 \times 173 \times 169$ run very fast, enabling trajectory animation at 10 to 30 frames per

second, thereby supporting interactive animation on-the-fly from a live in-progress simulation.

Overall QM/MM visualization performance was evaluated using the glutamyl-tRNA synthetase simulation shown in **Fig. 3**, for varying molecular orbital resolution settings, summarized in **Supplementary Table 6**. Performance results were measured using a workstation with dual NVIDIA Quadro GP100 GPUs (Pascal GPU architecture). Visualization timings included all of the individual steps involved in evaluating time-varying bond connectivity, molecular orbital grid calculation, isosurface extraction, calculation of protein secondary structure “cartoon” ribbon representations, and final display with OpenGL rasterization. The tRNA visualization test case included 10 molecular orbital representations, which add up to a significant amount of per-frame computation, both for the computation of molecular orbital grids and for the isosurface extraction step that produces the final molecular orbital surface. A longer video exemplifying VMD’s orbital rendering capabilities can be seen in **Supplementary Video 1**, which depicts the aminoacylation reaction.

In order to achieve higher performance for challenging QM/MM trajectory animation tasks, the molecular orbital representation implementation in VMD was improved to automatically identify and reuse molecular orbital grids common to multiple graphical representations, so that they do not need to be redundantly recomputed. This approach provides a tremendous performance increase compared to the previous algorithms implemented in VMD for common visualization cases when animating trajectories that contain large QM regions when molecular orbital grid calculation dominates the per-frame execution time. The VMD trajectory animation performance results shown in **Supplementary Table 6** were produced with this optimization, yielding a roughly 30% performance increase on the small grid sizes, and up to a 2 $\times$ performance gain for the largest grid size.

With the new molecular orbital grid reuse optimization implemented in VMD, the primary inhibitor to higher trajectory animation frame rates for the small and moderate size molecular orbital grids is the molecular orbital isosurface extraction step, which is presently performed on the host CPU [3]. We have previously reported performance benefits obtainable by performing isosurface extraction in-place on a GPU [5], but our previous work was limited to a single-GPU implementation and will require some significant changes to best accommodate state-of-the-art multi-GPU platforms. We expect that implementing in-place isosurface extraction on the GPUs will provide roughly a factor of 2 $\times$ trajectory animation performance increase, perhaps more, for molecular orbital representations that use up to mid-sized molecular orbital grids. The VMD trajectory animation performance results presented in **Supplementary Table 6** are more than fast enough to support on-the-fly animation for live QM/MM simulations, and the user’s ability to control grid spacing parameters

and other details permits a wide range of performance and visual quality trade-offs to be made when performing interactive animation or when producing still images for publication. The combination of new QM/MM visualization features with VMD's powerful built-in ray tracing engines permits QM/MM simulations to be rendered in high fidelity for manuscript figures and conventional movies [6], and for immersive omnidirectional stereoscopic virtual reality movies for display using commodity head mounted displays [7].

Benchmarks

Execution Details All systems that had a solvent box (and membrane, as in the case of Bacteriorhodopsin) were simulated using electrostatic embedding, which lead to differences in performance for NMA vs NMA + Solvent and TriAla vs TriAla + Solvent. A timestep of 2fs was used in all simulations, along with rigid bonds for all hydrogens. MD integration was done using the same parameters as described in the Methods section. The average wall-clock length of time spent computing each timestep was calculated from 100-step simulations for the PM7, PM7 (GPU) and PM7 (Mozyme) QM implementations, and 10-step simulations for HF and DFT QM implementations.

Software and Hardware All benchmark results shown in **Supplementary Table 3** were done using NAMD 2.12, in connection with MOPAC 2016 and ORCA 4.0. MOPAC can be used in QM/MM simulations using its CPU-only version, CPU+GPU version and its MOZYME module, as exemplified in the benchmark results. All benchmarks were carried out using an Intel Core i7-7700K CPU (4.2 GHz), with 64GB of RAM, and two GeForce GTX 1080 Ti.

References

- [1] Stone, J. E., Hynninen, A.-P., Phillips, J. C. & Schulten, K. Early experiences porting the NAMD and VMD molecular simulation and analysis software to GPU-accelerated OpenPOWER platforms. *Lect. Notes Comput. Sc.* **9945**, 188–206 (2016).
- [2] Stone, J. E., Hardy, D. J., Ufimtsev, I. S. & Schulten, K. GPU-accelerated molecular modeling coming of age. *J. Mol. Graph. Model.* **29**, 116–125 (2010).
- [3] Stone, J. E. *et al.* High performance computation and interactive display of molecular orbitals on GPUs and multi-core CPUs. In *Proceedings of the 2nd Workshop on General-Purpose Processing on Graphics Processing Units, ACM International Conference Proceeding Series*, vol. 383, 9–18 (ACM, New York, NY, USA, 2009).

- [4] Stone, J. E., Hardy, D. J., Saam, J., Vandivort, K. L. & Schulten, K. GPU-accelerated computation and interactive display of molecular orbitals. In Hwu, W. (ed.) *GPU Computing Gems*, chap. 1, 5–18 (Morgan Kaufmann Publishers, 2011).
- [5] Stone, J. E. *et al.* Evaluation of emerging energy-efficient heterogeneous computing platforms for biomolecular and cellular simulation workloads. *2016 IEEE International Parallel and Distributed Processing Symposium Workshop (IPDPSW)* 89–100 (2016).
- [6] Stone, J. E., Vandivort, K. L. & Schulten, K. GPU-accelerated molecular visualization on petascale supercomputing platforms. In *Proceedings of the 8th International Workshop on Ultrascale Visualization*, UltraVis '13, 6:1–6:8 (ACM, New York, NY, USA, 2013).
- [7] Stone, J. E. *et al.* Atomic detail visualization of photosynthetic membranes with GPU-accelerated ray tracing. *Parallel Comput.* **55**, 17–27 (2016).