JCTC Journal of Chemical Theory and Computation

pubs.acs.org/JCTC

Article

Toward Grid-Based Models for Molecular Association

Published as part of Journal of Chemical Theory and Computation special issue "Markov State Modeling of Conformational Dynamics".

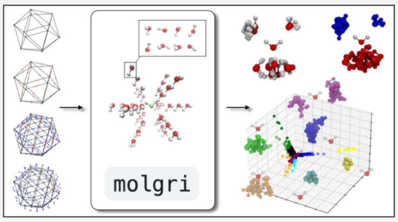
Hana Zupan and Bettina G. Keller*

Cite This: *J. Chem. Theory Comput.* 2025, 21, 614−628

Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

ABSTRACT: This paper presents a grid-based approach to model molecular association processes as an alternative to sampling-based Markov models. Our method discretizes the six-dimensional space of relative translation and orientation into grid cells. By discretizing the Fokker−Planck operator governing the system dynamics via the square-root approximation, we derive analytical expressions for the transition rate constants between grid cells. These expressions depend on geometric properties of the grid, such as the cell surface area and volume, which we provide. In addition, one needs only the molecular energy at the grid cell center, circumventing the need for extensive MD simulations and reducing the number of energy evaluations to the number of grid cells. The resulting rate matrix is closely related to the Markov state model transition matrix, offering insights into metastable states and association kinetics. We validate the accuracy of the model in identifying metastable states and binding mechanisms, though improvements are necessary to address limitations like ignoring bulk transitions and anisotropic rotational diffusion. The flexibility of this grid-based method makes it applicable to a variety of molecular systems and energy functions, including those derived from quantum mechanical calculations. The software package MolGri, which implements this approach, offers a systematic and computationally efficient tool for studying molecular association processes.



1. INTRODUCTION

The vastness of configuration space is a fundamental problem in molecular dynamics (MD) simulations. Due to its immense dimensionality and the fact that only narrow regions are significantly populated, often separated by large barriers, obtaining a comprehensive view of likely configurations and their transition time scales is clearly challenging. MD simulations explore this space by taking steps (determined by Newtonian forces and the choice of a thermostat) in configuration space that involve small changes in all degrees of freedom (DoF) at once. If the simulation is ergodic, every region of space will be visited proportionally to its Boltzmann weight, given infinite simulation time. However, even if ergodicity is formally fulfilled, there is no guarantee that all regions of interest have been sufficiently sampled within the finite time of a simulation. It is difficult to even determine whether all low-energy states have been reached.[1,2]

Markov state models (MSMs)[3−6] are a powerful tool to analyze complex molecular dynamics. They reduce the complexity of the high-dimensional, continuous dynamics by discretizing the configurational space into grid cells. The system's dynamics are then modeled as transitions between these grid cells, where the transition probability is estimated from MD simulations. The resulting transition probability matrix allows for a quantitative analysis of the molecular dynamics in terms of metastable states, mean-first passage times and pathways between different regions of configurational space. MSMs thus give insight into the mechanism of multistate molecular dynamics. One of the fields in which MSM have been particularly useful is in modeling molecular association, such as protein−ligand and protein−protein binding.[7−9] With recent advances in electronic structure methods[10,11] and the advent of neural network potentials,[12,13] it is likely that MSMs will be applied to more diverse molecular associations processes, such as adsorption on surfaces, the formation of nanoparticles or encounter complexes of chemical reactions.

However, MSMs, in particular MSMs of molecular association processes, are very sensitive to statistical uncertainties[14,15] and therefore often require extensive MD simulations. Approaches to improve the statistical efficiency of MSM estimations include improved feature selection for the definition of the underlying grid,[16,17] variational and core-set Markov models,[4,18,19] adaptive sampling algorithms,[20] and enhanced sampling combined with dynamical reweighting.[21] Despite these advances, MSM studies remain subject to the assumption that statistical noise does not distort the results.

An alternative is a generative grid-based approach, which we are pursuing in this contribution. The idea is to systematically

Received: September 30, 2024
Revised: November 27, 2024
Accepted: December 27, 2024
Published: January 13, 2025

produce structures at selected grid points in configuration space, calculate the point energies of generated structures and use this information along with the geometrical properties of the grid cells to obtain a probability flow across the cell boundaries. From these probability flows, one can calculate the transition rate matrix, a close analogue to the MSM transition probability matrix. Thus, instead of extensive MD simulations, only a single energy calculations per grid-point is needed. In addition, this approach guarantees that all regions of space are taken into account up to the boundaries and the resolution of the grid.

The grid-based approach, including the formula for the pairwise transition-rate constants, is derived[22−25] by assuming that the system evolves according to overdamped Langevin dynamics in a collective variable space and by discretizing the associated Fokker−Planck operator, leading to the square-root approximation of the Fokker−Planck operator (SqRA). The method has shown excellent replication of sampling-based MSMs for low-dimensional Cartesian spaces.[22,23,25,26] A proof of principle for alanine-dipeptide has been reported in ref 23. However, a crucial assumption in the SqRA is that the grid cells are so small that the potential within each grid cell is essentially constant. Thus, grids with high resolution are required, effectively limiting the grid-based approach to low-dimensional collective variable spaces.

Here, we consider the association of two molecules A and B in solution. The formation of bimolecular complexes typically consists of two stages: 1) diffusion-based association and 2) interaction-based completion of binding.[27] Comprehensive sampling of the diffusion-based association is almost intractable with standard molecular simulation as the simulation time needed to explore all possible relative translations and relative orientations of the two molecules is immense. However, within the rigid-body approximation, this process reduces to diffusion in the six-dimensional space of translation and rotation of molecule B relative to molecule A.

There are two major challenges in constructing translational and rotational grids for a SqRA-Markov model. First, the grid must be uniform, meaning that all grid cells should have approximately the same size. Second, it is necessary to calculate both the six-dimensional volume of each grid cell and the five-dimensional hypersurface area that represents the boundary between neighboring cells. In ref[28] we benchmarked methods for constructing uniform grids in translational space. Here, we extend the discretization to the full six-dimensional translation and rotation space. Drawing inspiration from the robotics community,[29−31] we employ regular Voronoi tessellation of the rotational space using quaternions, and we provide equations for the corresponding grid cell volumes and surfaces. We have developed a Python package, `MolGri`, which generates grids for the six-dimensional translation and orientation space, calculates the geometric parameters of the grid, and interfaces with MD programs to obtain the grid energies and compute the rate matrix. At the current stage, the package does not yet account for transitions into the bulk and for anisotropic rotational diffusion. We discuss the remaining steps needed to achieve an accurate SqRA-Markov model for molecular association processes.

2. THEORY

2.1. Square-Root Approximation.

The square-root approximation has been derived and tested in refs.[22−25] In this section, we summarize the most important equations. We additionally provide a more detailed derivation in the Supporting Information.

Consider a molecular system with N atoms and $3N$ translational degrees of freedom. A collective variable x_i is a function that maps the $3N$ translational degrees of freedom to a real number: x_i: $\mathbb{R}^{3N} \to \mathbb{R}$. We assume that in a low dimensional collective variable space $\mathbf{x} = (x_1, x_2, ..., x_m) \in \Omega \subset \mathbb{R}^m$, where $m \ll 3N$, the dynamics of the system can be modeled by overdamped Langevin dynamics:

$$\mathrm{d}\mathbf{x}(t) = \boldsymbol{\mu}(\mathbf{x}(t))\,\mathrm{d}t + \sigma\,\mathrm{d}\mathbf{B}(t) \tag{1}$$

where $\mathbf{B}(t) = (B_1(t) \ldots B_\mathrm{m}(t))$ is an m-dimensional Wiener process, $\boldsymbol{\mu}(\mathbf{x}(t)) = -\xi^{-1}M^{-1}\nabla V_{eff}(\mathbf{x}(t))$ is the m-dimensional drift vector, and $\sigma = \sqrt{2k_\mathrm{B}T\xi^{-1}M^{-1}}$ scales the Wiener process and is linked to the diffusion of the system in the collective variable space. We assume that the diffusion is isotropic in the collective variable space, and hence σ is simply a scalar. For nonisotropic diffusion, σ has to be replaced by an $(m \times m)$-matrix. ξ is a friction parameter with units s^{-1}, M is the effective mass, V_eff: $\Omega \to \mathbb{R}$ is the effective potential in the collective variable space, k_B is the Boltzmann constant, T is the temperature, and$\nabla f(\mathbf{x})$ denotes the gradient of a function f: $\mathbb{R}^m \to \mathbb{R}$.

$\rho(\mathbf{x}, t)$ is a probability density in the space of collective variables, whose time-evolution is governed by the Fokker−Planck equation associated with eq 1,

$$\begin{aligned}\frac{\partial}{\partial t}\rho(\mathbf{x}, t) &= -\nabla\cdot[\boldsymbol{\mu}(\mathbf{x}(t))\cdot\rho(\mathbf{x}, t)] + D\nabla\cdot\nabla\rho(\mathbf{x}, t)\\ &= \mathcal{Q}\rho(\mathbf{x}, t)\end{aligned} \tag{2}$$

where $\mathcal{Q}$ is the Fokker−Planck operator. For a vector field $\mathbf{f}$: $\Omega \to \mathbb{R}^m$, $\nabla \cdot \mathbf{f}(\mathbf{x})$ denotes the divergence of the vector field. $D = \sigma^2/2 = k_\mathrm{B}T\xi^{-1}M^{-1}$ is the diffusion constant. The stationary density associated with eq 2 is the Boltzmann density

$$\pi(\mathbf{x}) = Z^{-1}\exp\left(-\frac{1}{k_\mathrm{B}T}V_\mathrm{eff}(\mathbf{x})\right) \tag{3}$$

where $Z = \int_\Omega \mathrm{d}\mathbf{x}\exp\left(-\frac{1}{k_\mathrm{B}T}V_\mathrm{eff}(\mathbf{x})\right)$ is the configurational partition function.

The collective variable space Ω is discretized into N_d nonoverlapping grid cells Ω_1, ... Ω_{N_d}, where $\mathbf{x}_\alpha$ denotes the center of cell Ω_α. On this grid, eq 2 can be approximated by a matrix-vector equation

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\rho}^\mathrm{T}(t) = \boldsymbol{\rho}^\mathrm{T}(t)\mathbf{Q} \tag{4}$$

where the N_d-dimensional vector $\boldsymbol{\rho}(t)$: $\rho_\alpha(t) = \int_{\Omega_\alpha}\mathrm{d}\mathbf{x}\rho(\mathbf{x}, t)$ contains the time-dependent probabilities to find the system within each grid cell, and

$$\mathbf{Q}\text{: } Q_{\alpha\beta} = \begin{cases} Q_{\alpha\beta,\mathrm{adjacent}} & \alpha \sim \beta \\ 0 & \alpha \neq \beta \\ -\displaystyle\sum_{\beta=1,\beta\neq\alpha}^{N_\mathrm{d}} Q_{\alpha\beta,\mathrm{adjacent}} & \alpha = \beta \end{cases} \tag{5}$$

is a rate matrix and the discretized version of the Fokker−Planck operator $\mathcal{Q}$. $\alpha \sim \beta$ indicates that Ω_α and Ω_β are adjacent. The rate matrix $\mathbf{Q}$ is related to the MSM transition matrix $\mathbf{P}(\tau_{\text{MSM}})$ by[21]

$$\mathbf{P}(\tau_{\text{MSM}}) = \exp(\mathbf{Q}\tau_{\text{MSM}}) \tag{6}$$

where τ_{MSM} is the MSM lag time.

The square-root approximation of Q,[22,23,25] provides an analytical expression for the transition rate constant between adjacent cells

$$Q_{\alpha\beta\ \text{adjacent}} = \frac{\sigma^2}{2}\frac{1}{h_{\alpha\beta}}\frac{\mathcal{S}_{\alpha\beta}}{\mathcal{V}_\alpha}\sqrt{\frac{\pi(\mathbf{x}_\beta)}{\pi(\mathbf{x}_\alpha)}} \tag{7}$$

where $\text{h}_{\alpha\beta} = |\mathbf{x}_\beta - \mathbf{x}_\alpha|$ is the Euclidean distance between two cell centers. $\mathcal{V}_\alpha = \int_{\Omega_\alpha} \text{d}\mathbf{x}1$ is the volume of Ω_α, and $\mathcal{S}_{\alpha\beta} = \oint_{\partial\Omega_\alpha\partial\Omega_\beta} \text{d}\mathbf{x}1$ is the surface area of the intersecting (hyper-)surface between adjacent cells Ω_α and Ω_β. eq 7 relies on the following assumptions:

1 The grid is a Voronoi grid.
2 The grid cells are small, so that $V_{\text{eff}}(\mathbf{x})$, $\pi(\mathbf{x})$, and $\rho(\mathbf{x})$ are approximately constant within a grid cell.
3 Diffusion is isotropic, so that σ = const.

The significance of eq 7 is that, given the geometric parameters of the grid $h_{\alpha\beta}$, $\mathcal{S}_{\alpha\beta}$ and $\mathcal{V}_\alpha$, along with the effective potential energy at the grid cell centers $V_{\text{eff}}(\mathbf{x}_\alpha)$, one can construct an MSM without the need for MD simulations.[22,23,25]

Note that, in eq 5, we ensured that the row-sum of the rate matrix is zero. This convention is consistent with the MSM convention, in which the transition matrix is usually row-normalized to one. However, in communities that work with rate matrices rather than transition matrices, by convention, the columns of the rate matrices are normalized to zero.[21] This yields the transpose of $\mathbf{Q}$.

2.2. Rigid Body Approximation and Coordinate System.

To model molecular association, we consider a molecular system with two molecules A and B in the absence of any external potential. Since the total energy is independent of the overall translation and rotation of the system, we can reconceptualize it as the molecule A completely fixed at origin and the molecule B free to translate (3 DoF) and rotate (3 DoF) as a rigid body. We therefore choose the molecular frame of molecule A as a our coordinate system, i.e., the Cartesian coordinate frame whose origin is at the center of mass of molecule A and whose three axes are aligned with the principal axes of inertia of molecule A. The Cartesian coordinates of the two molecules in this coordinate system are denoted as

$$\mathbf{r}^{(k)} = \left(\mathbf{r}_1^{(k)}, \mathbf{r}_2^{(k)}, ..., \mathbf{r}_{N_k}^{(k)}\right), \qquad k = A, B \tag{8}$$

where N_A and N_B are the respective numbers of atoms, and $\mathbf{r}_i^{(k)} = (x_i^{(k)}, y_i^{(k)}, z_i^{(k)})$ is the position of the ith atom in the respective molecule. The vector $\mathbf{r}^{(k)}$ can also be represented by translational, rotational and internal coordinates

$$\mathbf{r}^{(k)} = \left(\mathbf{r}_{\text{COM}}^{(k)}, \mathbf{q}^{(k)}, \mathbf{v}^{(k)}\right), \qquad k = A, B \tag{9}$$

The 3-dimensional center-of-mass coordinate is

$$\mathbf{r}_{\text{COM}}^{(k)} = \frac{\sum_{i=1}^{N_k} m_i^{(k)}\mathbf{r}_i^{(k)}}{M^{(k)}}, \qquad k = A, B \tag{10}$$

where $m_i^{(k)}$ is the mass of the ith atom, and $M^{(k)} = \sum_{i=1}^{N_k} m_i^{(k)}$ is the total mass of the respective molecule. It describes the translation of the respective molecule with respect to the origin of the coordinate system. Hence, $\mathbf{r}_{\text{COM}}^{(A)} = (0, 0, 0)$. To construct the translation grid, we describe the center-mass-coordinate of B in spherical coordinates $\mathbf{r}_{\text{COM}} = (r, \theta, \phi) \in \mathbb{R}^3$, where $r \in \mathbb{R}_+$ is the radius, $\theta \in [0, \pi]$ is the polar angle, and $\phi \in [0, 2\pi]$ is the azimuth angle. The angles $(\theta, \phi) \in S^2$ define a point on the three-dimensional unit sphere (2-sphere). As a product, the radius and two angles cover the 3-dimensional space: $\mathbb{R}_+ \times S^2 = \mathbb{R}^3$.

$\mathbf{q}^{(\text{k})} \in SO\ (3)$ represents the three rotational degrees of molecule $k = A, B$ with respect to a reference rotation $\mathbf{q}_{\text{ref}}^{(k)}$. $SO(3)$ is the rotational group and $R(\mathbf{q}^{(k)})$ is the rotation matrix that transforms $\mathbf{q}_{\text{ref}}^{(k)}$ into $\mathbf{q}^{(k)}$. We use unit quaternions $\mathbf{q} = (q_0, q_1, q_2, q_3)$, $\|\mathbf{q}\|_2 = 1$, to describe the rotation of the molecule. See ref 32 for a review on different representations of the rotational group $SO(3)$ and ref 33 for more information on quaternions. Unit quaternions cover a 3-sphere (unit hypersphere) $\mathbf{q} \in S^3$. Each quaternion $\mathbf{q}$ corresponds to a rotation $R(\mathbf{q})$ in three-dimensional space.[34] However, each rotation in three-dimensional space is represented by exactly two quaternions, because the rotation induced by $\mathbf{q}$ equals that of $-\mathbf{q}$: $R(\mathbf{q}) = R(-\mathbf{q})$. The relationship of quaternion $\mathbf{q}$ to a 3 × 3-rotation matrix $R(\mathbf{q})$ can be expressed as[35]

$$R(\mathbf{q}_j) = \begin{bmatrix} 1 - 2q_2^2 - 2q_3^2 & 2q_1q_2 - 2q_3q_0 & 2q_1q_3 + 2q_2q_0 \\ 2q_1q_2 + 2q_3q_0 & 1 - 2q_1^2 - 2q_3^2 & 2q_2q_3 - 2q_1q_0 \\ 2q_1q_3 - 2q_2q_0 & 2q_2q_3 + 2q_1q_0 & 1 - 2q_1^2 - 2q_2^2 \end{bmatrix} \tag{11}$$

To avoid the double coverage $R(\mathbf{q}) = R(-\mathbf{q})$ we always select one out of the quaternion pair by limiting ourselves to $\mathbf{q} \in S_+^3$, where S_+^3 denotes the "upper half" of the hypersphere to describe a rotation. (Select quaternions with $\text{q}_0 > 0$. For $q_0 = 0$, quaternions with $\text{q}_1 > 0$ are included. If $q_1 = 0$, the decision is based on the third component.)

Finally, $\mathbf{v}^{(k)} = (v_1^{(k)}, v_2^{(k)}, ... v_{3N_k-6}^{(k)})$ are the $3N_k - 6$ internal degrees of freedom. Within the rigid-body approximation, we assume that the internal degrees of freedom are constant: $\mathbf{v}^{(k)}$ = const with $k = A,B$. Because we aligned the coordinate system with the molecular frame of molecule A, its translation and rotation also remain constant. Thus, within this model, the dynamics of the system is given by changes in $\mathbf{r}_{\text{COM}}^{(B)} \in \mathbb{R}^3$ and $\mathbf{q}^{(B)} \in S_+^3$. The collective variable vector in eq 1 then is

$$\mathbf{x} = \left(\mathbf{r}_{\text{COM}}^{(B)}, \mathbf{q}^{(B)}\right) \in SE(3) \tag{12}$$

where $\mathbb{R}^3 \times SO(3) = SE(3)$ is the special Euclidean group and describes the complete configuration space of the rigid body motion.

2.3. Translation and Rotation Grid.

To systematically generate configurations of B relative to A, we discretize $SE(3)$ by constructing grids for the translation and rotation subspaces $\mathbb{R}^3$ and $SO(3)$. This involves constructing uniform grids on $\mathbb{R}_+$ as well as (hyper)spheres S^2 (translation space) and S^3 (rotation space), both of which are closely related rotation group $SO(3)$. This is a challenging task.[33,36−38] In ref 28 we compared several algorithms and concluded that a polyhedron/polytope approach for grids on S^n fits our needs best.

For the translation grid, we discretize radius $r \in \mathbb{R}_+$ and angles $(\theta, \varphi) \in S^2$ separately. A uniform grid on $\mathbb{R}_+$ is straightforward: radial grid points r_i, $i = 1,2...N_r$, are equidistantly spaced between selected $r_{\min}$ and $r_{\max}$. The polyhedron approach to discretize S^2 is illustrated in Figure 1a.

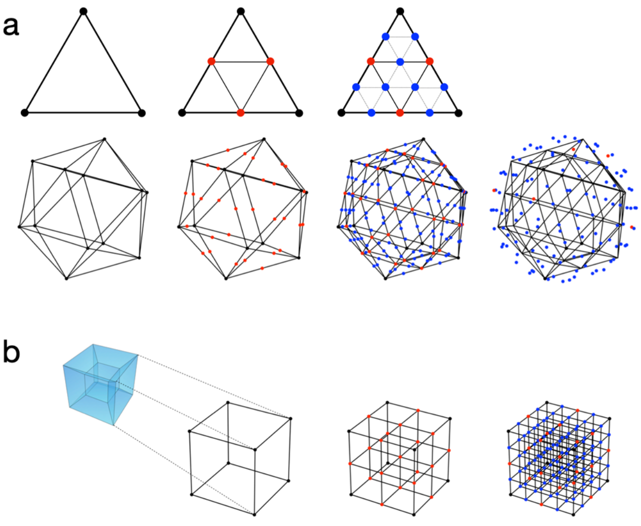


Figure 1. Illustration of grid construction for S^2 (a) and S^3 (b). For S^3, only one of hypercube cells is shown, as the projection onto the 3-sphere cannot be depicted.

First, an icosahedron is inscribed into a 2-sphere. The 12 vertices of this icosahedron yield the grid points (ϕ_i, θ_i) for the coarsest grid on S^2 (black dots in Figure 1a). Each face of the icosahedron is an equilateral triangle. Further grid points are created at midpoints of the icosahedron edges and then scaled to lie on the 2-sphere (red dots in Figure 1a). The resulting grid has 42 grid points. The mid points of the icosahedron edges discretize each icosahedron face into four smaller equilateral triangles. The next finer grid is generated by creating mid points on their edges and scaling them to lie on the 2-sphere (blue dots in Figure 1a). The process can be iteratively repeated to obtain finer and finer grids. Obviously, this process directly generates only specific sets of grid points: 12, 42, 80, ... However, an arbitrary number of grid points can be obtained by creating the next largest grid and removing an appropriate number of points, a topic we also discussed in our previous publication.[28] N_s denotes the number of grid points on S^2. By combining the S^2 grid with the radial grid we obtain the translation grid with $N_r \times N_s$ grid points. Each grid point i is associated with a translation vector $\mathbf{t}_i = (r_i, \phi_i, \theta_i)$. The grid points form N_s rays, each with N_r points, as depicted in Figure 2c).

We construct the rotation grid by systematically generating quaternions $\mathbf{q}_j$ using the polytope approach (polytope is an equivalent to polyhedron in higher-dimensional spaces). First, a 4-cube (tesseract) is inscribed in S^3. A 4-cube is an four-dimensional analogue of a three-dimensional cube and is defined by 16 vertices $\mathbf{q}_j = \frac{1}{2}(\pm1, \pm1, \pm1, \pm1)$. The prefactor 1/2 ensures that $\mathbf{q}_j$ is normalized to 1 and thus lies on S^3. One can visualize a 4-cube as an object consisting of eight cubic cells (Figure 1b). The 16 vertices of the 4-cube yield the grid points $\mathbf{q}_j$ for the coarsest grid on S^3 (black dots in Figure 1b). Further grid points are created by adding a point along each edge, face and center of the 4-cube, thereby subdividing each cubic cell into eight smaller cubes (red dots in Figure 1b). The new points are scaled to unit length to ensure that they lie on S^3. As in Figure 1a, this process can be repeated iteratively to achieve finer and finer grids (e.g., blue dots in Figure 1b). In the last step, the orientation grid is truncated to the "upper half" of S^3. N_o denotes the number of grid points on S^3_+.

The full grid for the configuration space $SE(3)$ is obtained as all possible combinations of translation and rotation grid points $\mathbf{x}_{ij} = (\mathbf{t}_i, \mathbf{q}_j)$. The total number of grid points is $N_d = N_r \cdot N_s \cdot N_o$. Then the configurations of molecule B, $\mathbf{r}_{ij}^{(B)}$,

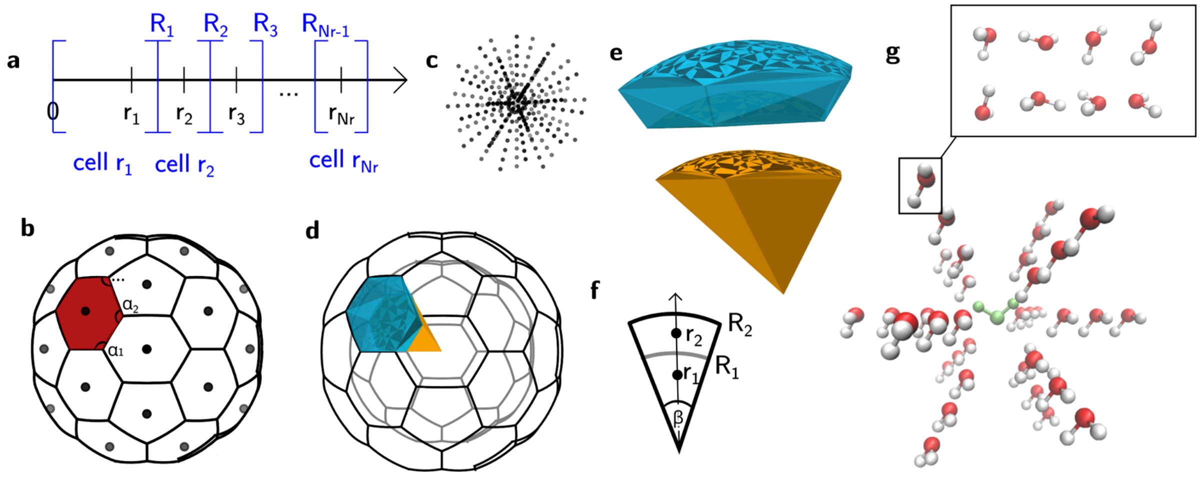


Figure 2. Translation grid. a radial grid with blue lines showing the cell boundaries. b angular grid, with spherical Voronoi division of the unit sphere (example of 42-point icosahedron grid), area shaded in red denotes the cell assigned to this grid point. c Example of translation grid with N_s = 42 and N_r = 7. d Partition of translation space into cells (only two radial layers are shown for clarity). The volumes of two cells are shown in color. e Close-up of the two colored cells of translation grid. f Schematic view of side borders between cells. g Grid with N_r = 3, N_s = 12 and N_o = 8 points applied to the system of two water molecules. All molecular translations generated with this grid are shown and the eight orientations are shown just for one example. The stationary molecule A is shown in green.

corresponding to grid points $(\mathbf{t}_i, \mathbf{q}_j)$ are constructed in a two-step process. First, molecule B is placed in the reference configuration $\mathbf{r}^{(B)}_{\text{ref}} = (\mathbf{r}^{(B)}_{\text{COM}} = (0, 0, 0), \mathbf{q}^{(B)}_{\text{ref}}, \mathbf{v}^{(B)})$. That is, B is placed at the origin of the coordinate system and in a specific reference rotation $\mathbf{q}^{(B)}_{\text{ref}}$. This reference rotation can have the axes of inertia aligned with the axes of the coordinate system, but this is not necessary. In the second step, the molecule B is first rotated by $R(\mathbf{q}_j)$ and then translated by $\mathbf{t}_i$, where the transformation is applied to each atom l individually

$$\mathbf{r}^{(B)}_{l,ij} = R(\mathbf{q}_j)\mathbf{r}^{(B)}_{l,\text{ref}} + \mathbf{t}_i \quad l = 1 \ldots N_B \tag{13}$$

The resulting configuration of molecule B can be represented in Cartesian coordinates $\mathbf{r}^{(B)}_{ij} = (\mathbf{r}^{(B)}_{1,ij}, \mathbf{r}^{(B)}_{2,ij}, \ldots \mathbf{r}^{(B)}_{N_B,ij})$ or in translational, rotational and internal coordinates $\mathbf{r}^{(B)}_{ij} = (\mathbf{r}^{(B)}_{\text{COM},i}, \mathbf{q}^{(B)}_j, \mathbf{v}^{(B)})$.

2.4. Energy of Grid Cells.

The energy associated with a grid cell (ij) is given by the effective potential $V_{\text{eff}}(\mathbf{x}_{ij}) = V_{\text{eff}}(\mathbf{r}^{(B)}_{\text{COM},i}, \mathbf{q}^{(B)}_j)$. Obtaining $V_{\text{eff}}(\mathbf{x})$ requires a free-energy calculation[39] for the six translational and rotational DoF and is computationally very costly. However, within the rigid-body approximation, the energy of the internal degrees of freedom is constant, and one can therefore replace the effective potential by the full $(N_A + N_B)$-atom potential of the bimolecular system

$$V_{\text{eff}}(\mathbf{x}_{ij}) = V\left(\mathbf{r}^{(A)}_{\text{COM}}, \mathbf{q}^{(A)}, \mathbf{v}^{(A)}, \mathbf{r}^{(B)}_{\text{COM},i}, \mathbf{q}^{(B)}_j, \mathbf{v}^{(B)}\right) \tag{14}$$

Thus, in principle, a single energy evaluation per grid point is sufficient. In practice, one might want to slightly improve this energy approximation using two strategies. First, to account for steric clashes, $\mathbf{v}_A$ and $\mathbf{v}_B$ can be relaxed while keeping the translational and rotational degrees of freedom of both molecules constrained. Second, to account for the fact that the energy is not entirely constant throughout the grid cell, $V_{\text{eff}}(\mathbf{x}_{ij})$ can be calculated as a an average over a short simulation, where translational and rotational degrees of freedom of both molecules are restrained to remain close but not exactly equal to the set translation and orientation. Having obtained a valid expression of $V_{\text{eff}}(\mathbf{x}_{ij})$ for each grid cell, the Boltzmann ratio in eq 7 can be evaluated according to eq 3.

2.5. Distances, Surfaces and Volumes of Grid Cells.

The grid points $\mathbf{x}_{ij}$ induce a Voronoi-like tessellation[40] of the six-dimensional translation and rotation space, where each point in this space is assigned to its closest grid point forming nonoverlapping grid cells. We defined these distances in terms of spherical coordinates (r, θ, ϕ) and in terms of angles between quaternions. The deviation from a Voronoi tessallation in Cartesian space are discussed in Section 4.4.

To calculate the distance between two adjacent grid points $h_{\alpha\beta}$, the area of the intersecting surface of their grid cells $\mathcal{S}_{\alpha\beta}$ and the volume of a grid cell $\mathcal{V}_\alpha$ in eq 7, we need to define a distance metric for the translation and orientation space. We will first discuss distance, surface and volume for translation and rotation space separately, before forming their product to discretize the SE(3) space. Throughout the discussion, we consider two adjacent grid points $\mathbf{x}_\alpha = \mathbf{x}_{ij} = (\mathbf{t}_i, \mathbf{q}_j)$ and $\mathbf{x}_\beta = \mathbf{x}_{kl} = (\mathbf{t}_k, \mathbf{q}_l)$.

2.5.1. Translation Grid.

The translation grid is constructed from the radial and spherical subgrids. This leads to two types of adjacency relations: (i) radial neighbors (orange and blue cells in Figure 2d,e, and (ii) angular neighbors. Radial neighbors are stacked along one of the N_s radial rays in the translation grid. Their grid points have the same angular coordinates, but differ by one in the radial index: $\mathbf{t}_i = (r_i, \phi_i, \theta_i)$ and $\mathbf{t}_k = (r_{k=i\pm1}, \phi_{k=i}, \theta_{k=i})$. Angular neighbors have the same radius but are neighbors on the spherical grid: $\mathbf{t}_i$= (r_i, ϕ_i, θ_i) and $\mathbf{t}_k = (r_{k=i}, \phi_k, \theta_k)$, where $(\phi_i, \theta_i) \sim (\phi_k, \theta_k)$.

The cells of the radial grid are separated by radii $R_1, R_2 \ldots R_{N_r-1}$ at midpoints between grid points r_i as shown in Figure 2a. The distance metric on the radial grid is

$$h_{ik,\text{radial}} = |r_k - r_i| \tag{15}$$

The intersecting surface is calculated as the area of the n-sided spherical polygon with interior angles $\alpha_1 \ldots \alpha_N$[41] (Figure 2b).

$$\mathcal{S}_{ik,\text{radial}} = R_i^2\left[\left(\sum_{m=1}^{n}\alpha_m\right) - (n-2)\pi\right] \tag{16}$$

To define the Voronoi tesselation of the spherical grid,[42,43] we use the angular distance

$$h_{ik,\text{angular}} = r_i \cdot \beta_{ik} = r_i \cdot \arccos\left(\frac{\mathbf{t}_i \cdot \mathbf{t}_k}{|\mathbf{t}_i||\mathbf{t}_k|}\right) \tag{17}$$

where $|\mathbf{t}_i| = \mathrm{r}_i$ is the Euclidean length of the translation vector $\mathbf{t}_i$, and β_{ij} is the angle between $\mathbf{t}_i$ and $\mathbf{t}_k$. Within this distance metric, the points on a sphere that are closest to the coordinate pair (ϕ_i, θ_i) have the geometrical form of a spherical polygon (Figure 2b). The intersecting area is a part of the corresponding circular sector (Figure 2f) and can be calculated by subtracting the area of the circular sector with the smaller radius from the one with the larger radius

$$\mathcal{S}_{ik,\text{angular}} = \frac{\beta_{ik}R_i^2}{2} - \frac{\beta_{ik}R_{i-1}^2}{2} \tag{18}$$

where β_{ik} is given by eq 17.

In summary

$$h_{ik}, \mathcal{S}_{ik} = \begin{cases} h_{ik,\text{radial}}, \mathcal{S}_{ik,\text{radial}} & \text{if } r_i \sim r_k \text{ and} (\theta_i, \phi_i) \\ & \quad = (\theta_k, \phi_k) \\ h_{ik,\text{angular}}, \mathcal{S}_{ik,\text{angular}} & \text{if } r_i = r_k \text{ and} (\theta_i, \phi_i) \\ & \quad \sim (\theta_k, \phi_k) \end{cases} \tag{19}$$

The volume of translation grid cells $\mathbf{r}_i$ is calculated from the corresponding sector of the sphere with radius R_i, where sector volume is proportional to the surface of the grid cell:

$$\mathcal{V}_{i,\text{sector}} = \frac{\mathcal{S}_{\alpha\beta,\text{radial}}}{4\pi R_i^2} \cdot \frac{4}{3}\pi R_i^3 = \frac{1}{3}\left[\left(\sum_{m=1}^{n_i}\alpha_{i,m}\right) - (n-2)\pi\right]R_i \tag{20}$$

The cell volume is obtained by subtracting the area of the next smaller sector $\mathcal{V}_{i-1,\text{sector}}$ from $\mathcal{V}_{i,\text{sector}}$

$$\mathcal{V}_i = \frac{1}{3}\left[\left(\sum_{m=1}^{n_i}\alpha_{i,m}\right) - (n-2)\pi\right](R_i - R_{i-1}) \tag{21}$$

where $R_0 = 0$ and angles $\alpha_{i,1} \ldots \alpha_{i,n_i}$ are schematically shown in Figure 1b.

2.5.2. Rotation Grid. The distance metric in rotation space[44] is based on the angle between the unit quaternions $\mathbf{q}_j$ and $\mathbf{q}_l$ of two adjacent rotation grid cells

$$h_{jl}(\mathbf{q}_{\mathrm{j}}, \mathbf{q}_{\mathrm{l}}) = \min\{\mathrm{d}_{\mathrm{ang}}(\mathbf{q}_{\mathrm{j}}, \mathbf{q}_{\mathrm{l}}), \pi - \mathrm{d}_{\mathrm{ang}}(\mathbf{q}_{\mathrm{j}}, \mathbf{q}_{\mathrm{l}})\} \tag{22}$$

with

$$d_{\mathrm{angular}}(\mathbf{q}_j, \mathbf{q}_l) = \arccos\left(\frac{\mathbf{q}_j \cdot \mathbf{q}_l}{|\mathbf{q}_j||\mathbf{q}_l|}\right) \tag{23}$$

Due to double coverage of the S^3-hypersphere, the distance is defined as the minimum of the two values in eq 22.

The geometrical properties of 3-sphere Voronoi cells are difficult to picture directly, but an intuition can be built on analogy with the 2-sphere tessellation displayed in Figure 2b. In the S^2 example, cells are spherical polygons and borders are spherical arcs between them, i.e., sections of S^1. Intuition suggests that borders between cells in S^3 could have the form of a section of S^2, i.e., spherical polygons. We confirm this intuition by the following consideration.

Let $v_1 \ldots v_s$ be Cartesian coordinates of Voronoi vertices shared between neighboring hypersphere cells $\mathbf{q}_{\mathrm{j}}$ and $\mathbf{q}_{\mathrm{l}}$ (as a condition of neighborhood, cells must share at least three vertices). As they all share the property of equal distance to $\mathbf{q}_{\mathrm{j}}$ and $\mathbf{q}_{\mathrm{l}}$, they must lie on a hyperplane. However, as they are Voronoi vertices of hypersphere tessellation, they must also lie on a hypersphere. Thus, they lie on an intersection of a hyperplane and hypersphere, which can be an empty set, a point or a 2-sphere. The first two option imply that there is no intersecting hyper-surface between $\mathbf{q}_{\mathrm{j}}$ and $\mathbf{q}_{\mathrm{l}}$ and will not occur if $\mathbf{q}_{\mathrm{j}}$ and $\mathbf{q}_{\mathrm{l}}$ are adjacent. The third option tells us that the intersecting hyper-surface between $\mathbf{q}_{\mathrm{j}}$ and $\mathbf{q}_{\mathrm{l}}$ has the form of a sphere in three-dimensional space and we take advantage of this property to visualize and calculate its areas.

We devised the following algorithm to determine the area of the intersecting surface. If we stack the vertices $v_1 \ldots v_s$ that defined the intersecting hyper-surface, we obtain a $4 \times s$ matrix $\mathbf{V}_4$. However, because we know they belong to a three-dimensional subspace (a sphere), there must exist a rotation rendering the fourth coordinate of all points equal zero. We find this rotation with singular value decomposition (SVD). Now, the rotated matrix can be interpreted as a $3\times\ s$ matrix $\mathbf{V}_3$ and the vertices as points on a unit sphere that divide the spherical surface into two spherical polygons, the smaller of which is the border area we are looking for. This means that we can again use the formula given by eq 16 for $R_i = 1$ for analytical calculation of spherical polygon areas $\mathcal{S}_{jl}$.

Finally, the volumes[1] of hyperspherical Voronoi cells must be determined. To the best of our knowledge, there is no general analytic solution for this problem. To perform a numerical approximation, (higher-dimensional) triangulation can be performed by analogy of surface triangulation that is shown in Figure 2e. Additional 5000 points are selected at random on the hypersurface of a hypersphere and assigned to their corresponding Voronoi cells. On 2-spheres, Delaunay triangles[45] are constructed from a dense set of points and their combined area approximates the area of a spherical section. Similarly, on 3-spheres, Delaunay triangulation leads to small tetrahedra filling a cell and their combined volumes are an approximation of a cell volume.

To confirm that the volumes of hypersphere Voronoi cells are reasonable, we compare them to the analytical value of unit hypersphere hyper-surface (what we call volume) π^2 equally divided into $2N_o$ sections:

$$V_{\mathrm{ideal}}[o_k] = \frac{A_{\mathrm{hypersphere}}}{2N_o} = \frac{\pi^2}{N_o} \text{ for } o_k \in (1, 2 \ldots N_o) \tag{24}$$

2.5.3. Configurational Grid. To construct the configuration grid on $SE(3)$, we combine translation and rotation grid. For two centers on the configuration grid, $\mathbf{x}_\alpha = \mathbf{x}_{ij} = (\mathbf{t}_i, \mathbf{q}_j)$ and $\mathbf{x}_\beta = \mathbf{x}_{kl} = (\mathbf{t}_k, \mathbf{q}_l)$, to be adjacent they must share a point in one of the subgrids and must be adjacent in the other subgrid. That is

$$\begin{array}{lll} (i) & \mathbf{x}_\alpha \sim \mathbf{x}_\beta & \text{if } \mathbf{t}_i \sim \mathbf{t}_k \text{ and } \mathbf{q}_j = \mathbf{q}_l \\ (ii) & \mathbf{x}_\alpha \sim \mathbf{x}_\beta & \text{if } \mathbf{t}_i = \mathbf{t}_k \text{ and } \mathbf{q}_j \sim \mathbf{q}_l \\ (iii) & \mathbf{x}_\alpha \neq \mathbf{x}_\beta & \text{otherwise} \end{array} \tag{25}$$

The first case represents a transition in translation space, whereas the second case represents a transition in rotation space. The distance and surface between adjacent cells then are

$$h_{\alpha\beta}, \mathcal{S}_{\alpha\beta} = \begin{cases} h_{ik}, \mathcal{S}_{ik} & \text{if } \mathbf{t}_i \sim \mathbf{t}_k \text{ and } \mathbf{q}_j = \mathbf{q}_l \\ \alpha h_{jl}, \alpha^2 \mathcal{S}_{jl} & \text{if } \mathbf{t}_i = \mathbf{t}_k \text{ and } \mathbf{q}_j \sim \mathbf{q}_l \end{cases} \tag{26}$$

where h_{ik} and $\mathcal{S}_{ik}$ are given by eq 19, and h_{jl} and $\mathcal{S}_{jl}$ are given by eqs 22 and 18. The grid cell volume is

$$\mathcal{V}_\alpha = \mathcal{V}_i \times \alpha^3 \mathcal{V}_j \tag{27}$$

where $\mathcal{V}_i$ is given by eq 21, and $\mathcal{V}_j$ is determined numerically.

In the combined space $SE(3) = \mathbb{R}^3 \times SO(3)$, the factor α represents the weight of the rotation space $SO(3)$ relative to the translation space $\mathbb{R}^3$, and has been introduced in discussions of $SE(3)$ robot manipulator spaces.[46] We currently set $\alpha = 1$.

$h_{\alpha\beta}$, $\mathcal{S}_{\alpha\beta}$ and $\mathcal{V}_\alpha$ can then be inserted into eq 7 to calculate $Q_{\alpha\beta,\mathrm{adjacent}}$.

2.6. Relation to Markov Models.

The rate matrix $\mathbf{Q}$ and MSM transition matrix $\mathbf{T}(\tau)$ are related by eq 6, and therefore share the same left and right eigenvectors[5]

$$\begin{aligned} \mathbf{Q}\boldsymbol{\psi}_i = \kappa_i \boldsymbol{\psi}_i \quad &\Leftrightarrow \quad \mathbf{P}(\tau_{\mathrm{MSM}})\boldsymbol{\psi}_i = \exp(\kappa_i \tau_{\mathrm{MSM}})\boldsymbol{\psi}_i \\ \boldsymbol{\phi}_i^{\mathrm{T}}\mathbf{Q} = \kappa_i \boldsymbol{\phi}_i^{\mathrm{T}} \quad &\Leftrightarrow \quad \boldsymbol{\phi}_i^{\mathrm{T}}\mathbf{P}(\tau_{\mathrm{MSM}}) = \exp(\kappa_i \tau_{\mathrm{MSM}})\boldsymbol{\phi}_i^{\mathrm{T}} \end{aligned} \tag{28}$$

where $\boldsymbol{\psi}_i$ are the right eigenvectors, and $\boldsymbol{\phi}_i$ are the left eigenvectors, and $\lambda_i(\tau_{\mathrm{MSM}})$ =exp$(\kappa_i \tau_{\mathrm{MSM}})$ are the associated MSM eigenvalues. The definition of the rate matrix within the square-root approximation (eqs 5 and 7) enforces detailed balance

$$\boldsymbol{\pi}_\alpha Q_{\alpha\beta} = \boldsymbol{\pi}_\beta Q_{\beta\alpha} \tag{29}$$

Consequently, right and left eigenvectors are linked by diag$(\boldsymbol{\pi})\psi_i = \phi_i$, where $\boldsymbol{\pi}$ is the stationary distribution and is equal to the first left eigenvector.

The dominant MSM eigenvectors contain a wealth of information on the metastable states and slow molecular processes. In a sampling-based MSM approach, they are obtained by estimating the elements of the MSM transition matrix $P_{\alpha\beta}(\tau_{\mathrm{MSM}})$ from an MD simulation. In the grid-based SqRA approach, they are obtained by evaluating the energy and geometric properties of each grid cell. Figure 3 compares the two approaches.

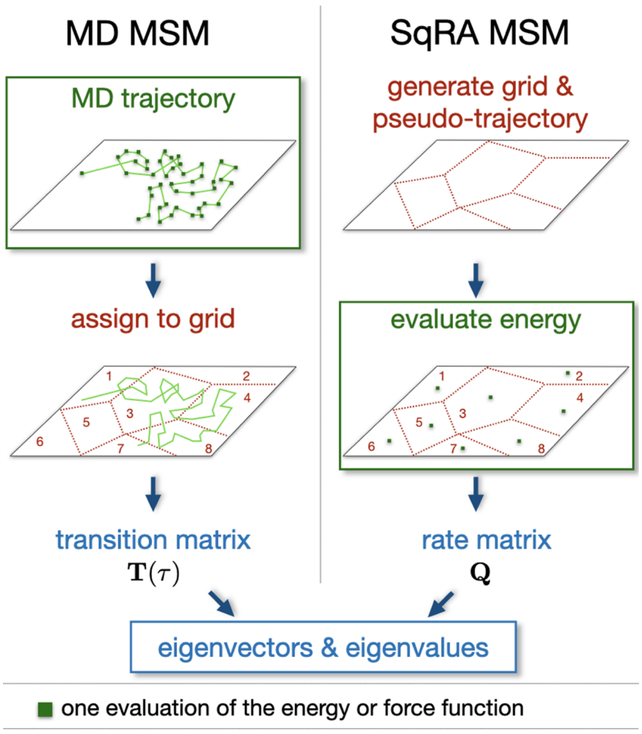


Figure 3. Comparison of the workflows for sampling-based and SqRA MSMs.

3. COMPUTATIONAL METHODS

3.1. Grid-Based Models. For the molecular association of two water molecules A and B, we constructed a grid based on $N_r = 10$ radial grid points, which were equally spaced between 0.2 and 0.4 nm, $N_s = 80$ angular grid points, and $N_o = 80$ rotational grid points. This yielded a grid with $N_d = N_o \times N_s \times N_r = 64{,}000$ grid cells. For the association of two $C_{60}F_2$ molecules, we used a radial grid with $N_r = 10$ points equally spaced between 0.8 and 1.3 nm, and the same angular and rotational grid as for the water dimer. For the association of bovine pancreatic trypsin inhibitor (BPTI) with trypsin, we used a radial grid with $N_r = 10$ points equally spaced between 3.5 and 4.5 nm, and the same angular and rotational grid as for the water dimer. Other settings for the radial grid are reported in the Supporting Information.

We calculated the volume $\mathcal{V}_\alpha$ of each grid cell. We constructed the adjacency matrix of the grid and calculated the distances between adjacent cells $h_{\alpha\beta}$ and the areas of the intersecting surfaces $\mathcal{S}_{\alpha\beta}$. Then, molecule A was placed at the origin of the coordinate system, and molecule B was translated and rotated to each of the grid cells as described in Section 2. The resulting 64,000 configurations were sequentially written to a `.trr` file (GROMACS trajectory format). The potential energy of each of these configurations was evaluated using GROMACS's rerun command.The water molecules were modeled using the TIP3P water model.[47] The $C_{60}F_2$ molecules were created in Avodagro[48] by modifying the template for the C_{60} fullerene. The topology was created with the GROMOS-Topology-Builder.[49,50] For the BPTI and Trypsine molecules the starting structure was PDB structure 4Y0Y[51] with the two components separated and prepared as described in a previous publication of our group.[52] From this information, we calculated the factor $\sqrt{\pi(\mathbf{x}_\alpha)/\pi(\mathbf{x}_\beta)} = \exp[(V_{\mathrm{eff}}(\mathbf{x}_\beta) - V_{\mathrm{eff}}(\mathbf{x}_\alpha))/(2k_{\mathrm{B}}T)]$ for all pairs of adjacent cells. To avoid integer overflow, the energy difference $V_{\mathrm{eff}}(\mathbf{x}_\beta) - V_{\mathrm{eff}}(\mathbf{x}_\alpha)$ was capped at 500 kJ/mol. Energy differences of ≥500 kJ/mol correspond to transition rates that are numerically zero and can thus be safely set to zero. The $(N_{\mathrm{d}} \times N_{\mathrm{d}})$-rate matrix $\mathbf{Q}$ was calculated using eq 7. The left and right eigenvectors and associated eigenvalues of the rate matrix were calculated with `scipy`'s eigenvalue solver.

The code we wrote to construct and evaluate grid-based models is formatted as a python 3.12.4[53] package called `molgri` (for **mol**ecular **gri**ds) and the computational experiments formatted as snakemake 8.14.0[54] pipelines for better reproducibility. Major Phython dependencies used in this project are numpy 1.26.4,[55] scipy 1.13.1,[56] networkx 3.3[57] and mdanalysis 2.7.0.[58,59]

3.2. Spectral Clustering of Q. Spectral clustering was performed on the first six right eigenvectors of the rate matrix $\mathbf{Q}$. The clustering algorithm used was KMeans[60] as implemented in scikit-learn 1.5[61] and the choice of 12 clusters was made.

3.3. Molecular Dynamics Simulations. Molecular dynamics simulations were conducted with GROMACS 2022[62−65] and performed only for the system of two water molecules.

For the vacuum simulations, two water molecules were placed in a cubic box with 3 nm edge length. The interactions were modeled using the TIP3P water model.[47] O−H bond lengths were constrained using LINCS algorithm.[66] The dynamics were propagated using the built-in stochastic integrator for Langevin dynamics[67,68] (setting `sd`) with a time step of $\Delta t = 2$ fs. The reference temperature was set to 300 K, and we varied the coupling time in across different simulations runs: $\tau_c = 0.001$ ps, 0.010 ps, 0.100 ps, 1.000 ps. Each simulation run was conducted for 4×10^7 timesteps, corresponding to 80 ns simulation time. Long range interactions were cutoff at 1.4 nm. To prevent that the two molecules diffuse far beyond the maximum radius of the SqRA grid $R_{N_r} = 0.4$nm, we applied a flat-bottomed distance restraint (Figure 6) along the oxygen−oxygen distance r: no restraining potential for $0 \leq r < 0.5$ nm, harmonic potential with force constant $k = 500$ kJ/(mol nm^2) for 0.5 nm $\leq r < 0.7$ nm, and a linear restraining potential with force constant $k = 500$ kJ/(mol nm^2) for $r \geq 0.7$ nm. No pressure coupling was applied. Coordinates of the water molecules were written to a file every 5 timesteps.

For the simulations in explicit solvent, two water molecules were solvated with 2033 Lennard-Jones particles in cubic box with 3.6 nm edge length. The water molecules were modeled using the TIP3P water model.[47] The Lennard-Jones particles had a mass of 4 atomic units, no charge and the following Lennard-Jones parameters: $\epsilon = 0.8202$ kJ/mol and $\sigma = 0.253$ nm, approximately modeling a helium atom. The simulation parameters were the same as for the vacuum simulations with the exception of the long-range interactions, where we used Ewald summation with a long-range cutoff of 1.4 nm.

3.4. Markov State Models. From the MD simulation trajectories, we constructed Markov state models using the same grid as for the SqRA models. The trajectories were aligned to the reference translation and orientation of the first

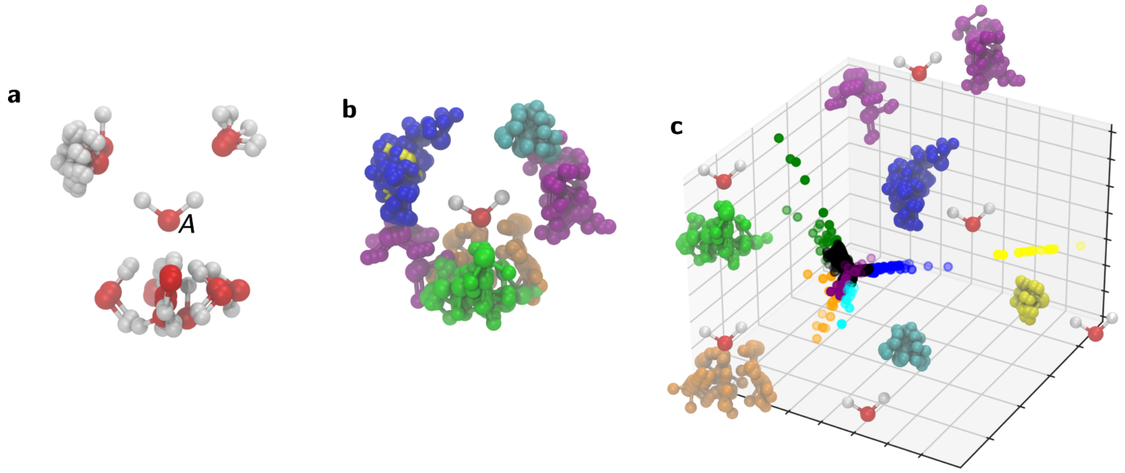


Figure 4. (a) Most probable configurations of the bimolecular system according to the stationary distribution calculated with the SqRA-Markov model. Molecule *A* is labeled, the rest of the structures represent molecule *B*. (b) K-Means clustering performed on the space of the first six eigenvectors of the rate matrix. Configurations that are part one cluster are shown in the same color. (The largest cluster with population 63834 and five tiny clusters with population <10 are not shown.) (c) A dot for each grid point is shown in the space of 1st, 2nd and 3rd eigenvector and colored according to the clustering in part b.

water molecule, and the translation and orientation of the second molecule was assigned to a grid cell using the following procedure: (1) the center-of-mass distance between the two molecules is calculated and the nearest cell center in the distance grid is selected, (2) the center-of-mass distance vector is scaled to unit length and the nearest cell center in the direction grid is selected, and (3) the rotation matrix between orientation of the reference structure and the orientation of molecule *B* is calculated and the nearest quaternion is selected from the rotation grid. The combination of the three assignments yields a grid cell index for each trajectory frame, and thus a microstate trajectory. The assignment is also implemented in our Python package `molgri`. To construct a Markov model from the microstate trajectory, we followed standard procedures.[5] We constructed a MSM count matrix $\mathbf{C}(\tau)$ by counting state-to-state transitions within lag time τ_{MSM}. The resulting $(N_s \times N_s)$-matrix was stored in a sparse data format. We varied τ_{MSM} between 0.01 and 1.0 ps but always show $\tau_{\mathrm{MSM}} = 0.1$ ps in the results section. Detailed balance was enforced. The count matrix was row-normalized to obtain the MSM transition matrix $\mathbf{T}(\tau)$. The left and right eigenvectors and associated eigenvalues were again calculated with `scipy`. Implied time scales were calculated as $t_{\mathrm{its},i} = -\tau_{\mathrm{MSM}}/\ln(\lambda_i(\tau_{\mathrm{MSM}}))$, where $\lambda_i(\tau_{\mathrm{MSM}})$ is the ith MSM eigenvalue.

4. RESULTS AND DISCUSSION

4.1. SqRA Model of the Water Dimer.

To illustrate the grid-based approach to molecular association, we consider two water molecules in vacuum and construct the SqRA-Markov model of the water dimer association on a configuration grid with 6.4×10^4 `molgri` grid cells. Figure 4a shows the highest-probability configurations of water molecule *B* relative to water molecule *A*, which were extracted from the stationary probability vector (first left eigenvector) of the SqRA rate matrix $\mathbf{Q}$. The model correctly identifies configurations in which water *A* acts as a hydrogen-bond donor and configurations in which it acts as hydrogen-bond acceptor. Remarkably, in the hydrogen-bond-donor configurations, the rotation of water *B* is very restricted, whereas is its free to rotate in the hydrogen-bond-acceptor configuration. This free rotation, although at variance with the water-dimers of actual water molecules, is likely correct for the TIP3P water model, which does not account for the oxygen electron lone pairs. Further note that there is a difference in the number of left and right side hydrogen-bond donor configurations, which might be due to slight asymmetry in the discretization of the translation grid relative to the mirror plane of *A*.

To further analyze the dominant eigenspace of $\mathbf{Q}$, we projected each grid center into the space of six dominant right eigenvectors and clustered in this six-dimensional space using the KMeans algorithm. Figure 4c shows that the eigenvectors of the rate matrix clearly separate the subspaces expected by chemical intuition: four clusters are found that correspond to the four possible hydrogen bonds with molecule *A* (shown in yellow, cyan, green and orange), where KMeans separates the large set of hydrogen-bond-acceptor configurations in two separate clusters (orange and green). The violet cluster represents the transition region between the two types of hydrogen bonding. There is an additional cluster (dark blue) that shows a broader region around one of the potential minima and five small clusters of 2−7 structures that seem to be artifacts of the choice of the number of clusters (not shown).

Also not shown is the most populated cluster containing over 99.7% of all generated poses which can be regarded as the set of all structures that have no particular importance to the slow processes of the system. This is a big contrast to the usual statistics of sampling-based methods, where a almost all sampled structures are found in the vicinity of the (few) deepest potential minima. The fact that a large majority of generated structures is not relevant to the binding of the two molecules might first seem like disappointing performance, but it is expected behavior for a grid that uniformly fills the configuration space. It is even desirable for two reasons: first, it allows us to identify transition states between low-energy configurations, such as the violet cluster; second, for a grid-based approach it is sufficient to reveal a single pose that lies inside a particular potential minimum, because the ensemble of structures within that minimum can be easily obtained in a

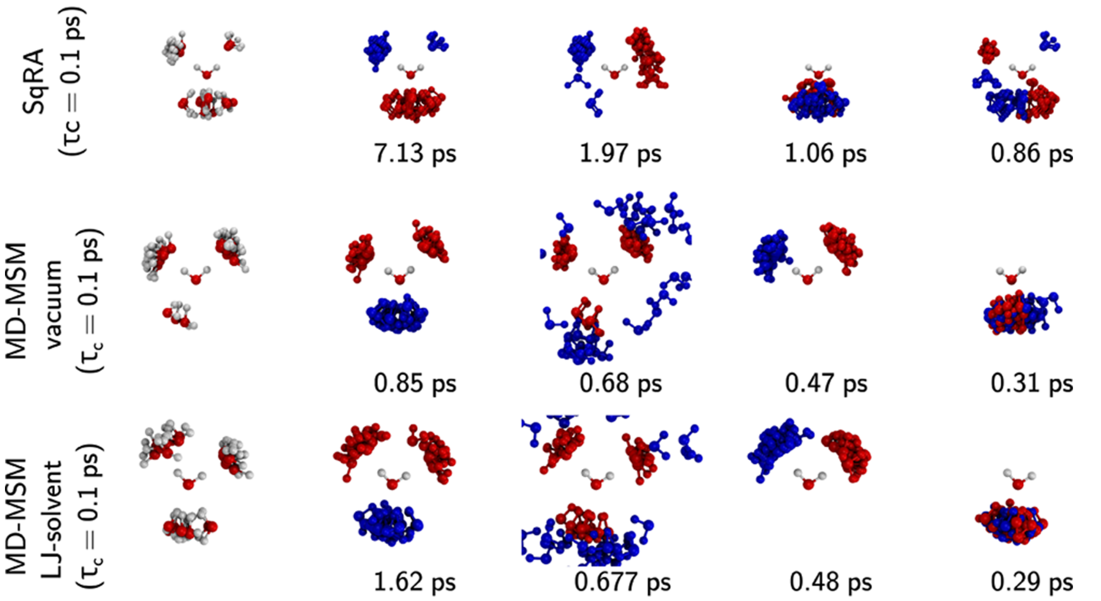


Figure 5. First five eigenvectors of water dimer system at three different simulation conditions. Top: SqRA model with $\tau_C = 0.1$ ps, middle: 40 ns MD trajectory in vacuum with $\tau_C = 0.1$ ps, bottom: 80 ns MD trajectory in a box of LJ particles with $\tau_C = 1$ ps. In **blue**: structures corresponding to the 30 most negative values in the eigenvector; in **red**: structures corresponding to the 30 most positive values in the eigenvector. For the 0th eigenvector, structures corresponding to 30 eigenvector entries with the largest absolute value are shown. The implied time scales are noted below the corresponding eigenvector.

subsequent step, either by applying a denser grid in the region of interest or by performing a short simulation starting from the identified structure.

The implied time scales of the SqRA-Markov model depend linearly on $\tau_c = \xi^{-1}$, ξ is the friction coefficient of the overdamped Langevin dynamics (eq 1). This friction coefficient is usually implemented as a thermostat coupling time τ_c. By varying τ_c between 0.001 and 1.0 ps, the implied time scale of the slowest process decreases from 712 to 0.71 ps (see SI Figure S1). Since τ_c is an arbitrary parameter at this point, these values should not be assigned any chemical significance. Importantly, the eigenvectors remain largely unaffected by changes in the magnitude of the friction coefficient.

4.2. Comparison to MD.

Figure 5 compares the SqRA-Markov model to MSMs built from MD simulations of two water molecules in vacuum. To prevent that differences in the discretization distort the results, we built the sampling-based MSMs on the grid with 6.4×10^4 grid cells as the SqRA-Markov model. We sampled extensively (80 ns) to minimize the statistical error. Since the SqRA derivation assumes overdamped Langevin dynamics but typical molecular dynamics simulations are performed under underdamped conditions, we must enforce that the translation and rotation of the molecular system are in the overdamped regime. We try two simulation set-ups to fulfill this requirement: (i) simulating the bimolecular system in vacuum but with large friction constant or (ii) augmenting the thermostat noise with explicit solvent particles smaller than the solvate, in our case helium-like Lennard-Jones particles while setting τ_c to values conventionally used in MD. We expect the SqRA-MSM to align more closely with the vacuum simulation than with the Lennard-Jones solvent simulation. But relying on the thermostat as the main source of noise is a somewhat artificial setup, as friction and random forces in molecular systems naturally arise from the surrounding solvent. Thus, we include the second setup for comparison. However, the helium-like Lennard-Jones particles are comparatively large for the water dimer, so deviations from the ideal overdamped Langevin dynamics are expected.

The sampling-based MSM in vacuum identifies the same metastable states as the SqRA (first column in Figure 5), corresponding to two distinct states in which molecule *A* acts as a hydrogen-bond donor and one broad state in which molecule *A* acts as a hydrogen bond acceptor. In accordance with the SqRA-MSM, the sampling-based MSMs identify the exchange between hydrogen-bond acceptor state and hydrogen-bond donor states as a slow process (second column in Figure 5) and the exchange between the two hydrogen-bond donor states as a slightly faster process (fourth column in Figure 5).

As in the SqRA-Markov models, the eigenvectors are largely unaffected by the magnitude of τ_c (Figures S2−S5). In the vacuum simulations (Figures S2 and S3), the implied time scales decrease slightly with decreasing friction, but not orders of magnitude as in the SqRA-Markov model. In the simulations with explicit solvent (Figures S2 and S3), the implied time scales change minimally when τ_c is varied, because the friction predominantly arises from interactions with Lennard-Jones particles.

The sampling-based MSMs yield additional processes that are not part of the dominant eigenspace of the SqRA-Markov model. Specifically, we find processes that represent the exchange between inner and outer regions of the configuration grid (e.g., third eigenvector in row 2 in Figure 5). This prompted us investigate boundary conditions further, see the following section.

4.3. Boundary Conditions.

In our current SqRA model, transitions out of the grid into the bulk are not accounted for. A molecule in a boundary cell (one of the grid cells with the largest radial distance between molecules A and B) can diffuse to neighboring grid cells but its probability of diffusing through the outer surface between the current cell and the bulk is zero. Implementing these reflecting boundary conditions in an MD simulation is difficult, as a hard reflecting boundary or a strong restraining potential at the grid boundary will distort the dynamics in, at least, the outer grid cells. We therefore permitted unbinding and transitions across the grid boundary into the bulk. However, to prevent water molecule *B* from diffusing away from molecule *A*, we added a restraining potential to our system that starts increasing when oxygen−oxygen distance reaches 0.5 nm. This distance must be large enough to not disturb the bound structures of the system, which we confirmed in Figure 6 where the restraining potential is plotted alongside oxygen−oxygen radial distribution function from one of the MD simulations in explicit solvent. The radial distribution function of water in that figure might seem unusual, but the absence of second maximum and bulk limit is simply the consequence of simulating only two water molecules, where only the first solvation shell is present. Because of the restraining potential, we observe some would-be transitions into the bulk as bounces off the restraining potential. There is a small peak at around 0.5 nm in the distribution in Figure 6 that can be attributed to this bounce. To enforce the reflecting boundary conditions in our sampling-based MSMs, we considered two approaches to treat transitions out of the grid. In the first approach we include all distant out-of-grid configurations in the MSM estimation and assign them to the closest grid cell (closest orientation, closest direction and largest radius). This effectively extends the boundary cells of the translation grid indefinitely to $R = \infty$. In the second approach, distant out-of-grid configurations are assigned to none of the cells. When building the MS count matrix, transitions that either start or end with an unassigned structure are then omitted.

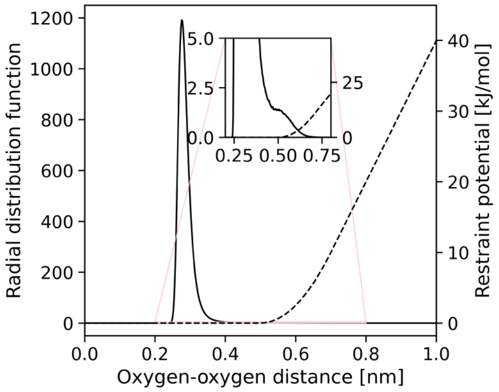


Figure 6. Radial distribution function (full line) and restraint potential (dashed line) for a MD run of two water molecules in a box of Lennard-Jones particles.

Figure 7 illustrates the impact of the boundary treatment on sampling-based MSMs. The eigenvectors and eigenvalues of the two MSMs are nearly identical, except for the second eigenvector which represents the exchange between close and distant structures. As expected, including out-of-grid configurations in the MSM estimation causes this process to represent the exchange across the grid boundary into the bulk. When these configurations are omitted, this process still reflects a radial transition but is now confined within the grid. In the MD-MSM in which distant structures are omitted, the radial transitions appear to mix with transitions between metastable states in eigenvectors 4. It is important to note that the reflective boundary is highly artificial. We plan to extend the SqRA-Markov model to more accurately model transitions into the bulk, potentially by employing approaches such as those in ref 69.

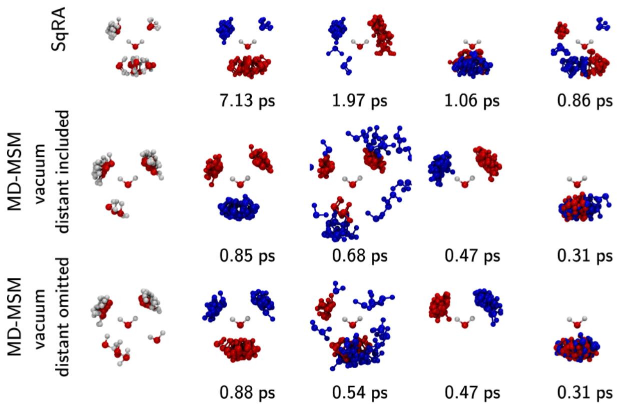


Figure 7. First five eigenvectors of the water dimer system at three different treatments of distant structures. Top: SqRA model, distant structures do not exist by construction, transition into this region is assumed impossible; middle: MD trajectory in vacuum, distant structures are assigned to the best available cell; bottom: MD trajectory in vacuum, distant structures are not assigned to any cell. Coupling time to a thermostat is τ_C = 0.1 ps in all three cases. For more information on distant structures, assigned and omitted approaches see text. For explanation of red/blue regions see description of Figure 5. The MD MSM in which distant structures were omitted featured a transition into a state with low population as a slow process. This is a known numerical artifact of MSMs and the eigenvector is therefore not shown.

4.4. Nonlinear Coordinates and Anisotropic Diffusion.

We model translational diffusion in Cartesian coordinates in eq 2, but construct the translational grid in spherical coordinates. Specifically, we create the translational grid as a Voronoi tessellation in spherical coordinates and calculate the corresponding grid cell volumes, surfaces and distances accordingly. This approach induces a slight error in the prefactor $\frac{1}{h_{\alpha\beta}}\frac{\mathcal{S}_{\alpha\beta}}{\mathcal{V}_{\alpha}}$ in eq 7. The derivation of this prefactor that the grid is a Voronoi tessellation in the same coordinate system as the Fokker−Planck operator, which, in this case, is the 3-dimensional Cartesian space. However, this deviation is likely minor because for a dense, regular spherical grid, the Voronoi tessellation in spherical coordinates closely resembles the equivalent in Cartesian coordinates, resulting in nearly identical grid cells. Figure 8 illustrates this effect in the 2-dimensional space for polar coordinates: red dots define a regular polar grid, and the Voronoi tessellation in polar coordinates (gray lines) almost perfectly overlaps with the Voronoi tessellation in the Cartesian coordinates (blue lines). For the translation grid in our model, it is however possible to replace the Voronoi tessellation in spherical coordinates by a Voronoi tessellation in Cartesian coordinates. We report the corresponding equations in the Supporting Information.

The situation is more complicated for the rotation grid, because rotational diffusion occurs in inherently nonlinear coordinates. This nonlinearity introduces an anisotropic diffusion tensor in the Fokker−Planck operator, meaning that *D* in eq 2 is no longer a constant but becomes a matrix $\mathbf{D} \in \mathbb{R}^{3\times3}$. This matrix depends on the moments of inertia of

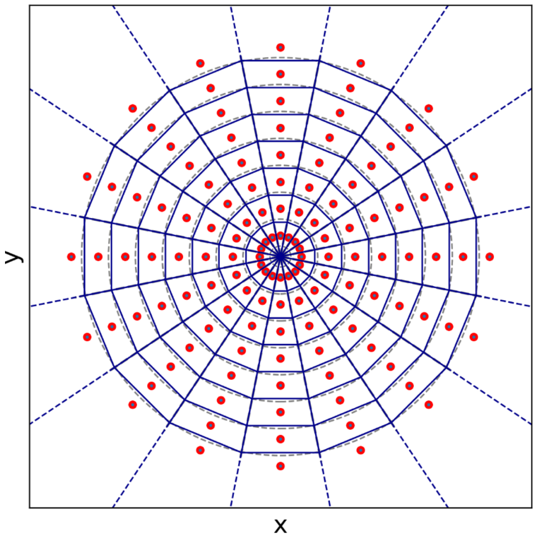


Figure 8. Comparison between Voronoi tessellation in polar coordinates (dashed gray lines) and Voronoi tessellation in Cartesian coordinates (blue grid) for a regular grid in polar coordinates (r, θ) (red dots).

the rotating molecule. The further the molecule's shape deviates from spherical symmetry, the more **D** will deviated from DId, where Id is the identity matrix. These deviations lead to inaccuracies in the current version of the SqRA-Markov model, which essentially assumes a spherical particle. A method for estimating the rotational diffusion tensor from MD simulations has been proposed in ref 70. To incorporate anisotropic diffusion into the SqRA-Markov model, we need an analytical expression for the rotational diffusion tensor and must rederive the prefactor in eq 7 for for anisotropic diffusion.

4.5. Computational Cost.

Our grid-based approach scales to molecular systems that are considerably larger than a water dimer. Specifically, we tested a dimer of two fluorinated fullerenes $C_{60}F_2-C_{60}F_2$, and the protein−protein complex of bovine pancreatic trypsin inhibitor (BPTI) with trypsin.[52,71] SqRA-MSM eigenvectors for these systems are reported in the Supporting Information in Figures SI 6 and SI 7, but should be interpreted with caution.

The computational cost of the grid-based models of our three test systems are shown in Figure 9, broken down by the workflow steps that were introduced in Figure 3. While all steps are orchestrated through the molgri software, only the first step is truly dependent on our algorithms, the energy calculation is handled by GROMACS[62−65] and the decomposition is performed by the python package scipy.[56] The total wall-clock time to calculate each of the three models on an Intel Xeon processor is about an hour, but the individual workflow steps scale differently with system size and merit a more detailed discussion.

The combined task of generating the grid and the pseudotrajectory increases in cost with the number of atoms N, but the scaling remains highly sublinear. The increase is caused by the generation of the pseudotrajectory. The sublinear scaling is expected, as transformations affect the center of mass and principal axes, and consequently the transformation does not need to be recomputed for each atom individually. Additionally, part of the system size effect arises from the increased cost of writing larger files as the number of atoms grows. The computational cost of generating the underlying grid is entirely independent of the number of atoms N and only depends on the number of grid points N_d. In general, the cost of generating the grid scales superlinearly with the number of grid points. However, we implemented an efficient network-based representation of polygons, ensuring that grid generation remains a matter of seconds. The combined cost for grid and pseudotrajectory generation for $N_d = 64{,}000$ is thus less than 10 min.

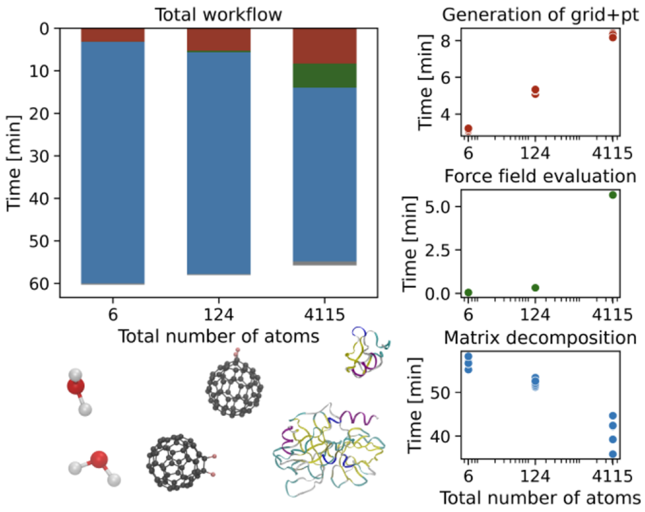


Figure 9. Comparison of time needed for the total molgri+SQRA workflow for systems spanning three different orders of magnitude in size: water dimer (6 atoms), fluorinated fullerene dimer (124 atoms) and Trypsine-BPTI system (4115 atoms). A 64,000-cell grid is used in all of the examples, only the radial distances are modified to account for different sizes of the molecules. The breakdown of the time needed for major steps of the process is shown on the right side. In addition to the separate steps, total time of the process includes some additional steps like file management and plot generation shown in gray on the total time plot. Runtime was measured five times for calculations running on up to ten cores of an Intel(R) Xeon(R) processor (2.20 GHz, 22 cores, 56 MB cache).

The computational cost of the energy evaluation rapidly increases with system size, where the exact scaling depends on the type of molecular energy function and how this function is implemented. Empirical force fields, as used in this study, typically scale as $N \ln N$ and can be calculated so efficiently that the 64'000 energy evaluations for each system are completed within seconds for the water and the fullerene system, and within slightly more than 5 min for the protein-dimer. Thus, despite the steep scaling, energy evaluation constitutes the smallest contribution to the overall computational cost in our test systems. However, the grid-based approach is not restricted to empirical force fields. More expensive energy functions, such as energies based on quantum chemical calculations, can be used in lieu of a force field.

In our current setup, the largest contribution to the computational cost is the eigendecomposition of the rate matrix. The time needed and the success of eigendecomposition of a large sparse matrix **Q** depends strongly on how well-conditioned the matrix is. Therefore, we observe changes in speed that are not directly related to the system size N or the grid-size N_d. The decreasing trend in Figure 9 is probably coincidental. Another critical resource in matrix decompositions is the computer memory, which currently limits our approach to $O(10^5)$ grid points.[25]

4.6. Comparison with Other Grid-Based Methods.

Although our method is not primarily designed for protein−ligand binding, it is informative to compare it to molecular docking approaches[72−74] and sampling-based methods for protein−ligand binding, such as MSMs[7−9] and SEEKR2.[75,76]

The objective of molecular docking is to generate potential configurations of a protein−ligand system, often using grids in translational and rotational space, and to identify low-energy configurations through a single energy calculation per grid point. In these two aspects, our method resembles molecular docking. Specifically, because SqRA-MSMs also require only a single energy calculation per grid point, our grid-based approach is almost as computationally efficient as docking, potentially enabling high-throughput screening of binding partners.

Molecular docking cannot provide kinetic information because, to accurately calculate the probability flux between neighboring grid cells, the grid must meet specific requirements. First, it needs to be a Voronoi grid. Second, the grid cells should be small and should be ideally of equal size. Third, the geometric parameters of the grid cell must be known. Our work has therefore focused on generating uniform Voronoi grids[28] and on deriving analytical expressions for their geometric parameters (Section 2). With these two requirements in place, one can construct SqRA-MSMs for molecular association, which give access to metastable states and competing binding pathways, along with the associated time scales. Since SqRA-MSMs require small grid cells, we generate around $O(10^4)$ configurations for a molecular association process, approximately 10 to 100 times more than a typical docking run.

Sampling-based approaches for modeling molecular association also frequently use grids to describe the kinetics, estimating the probability flux between neighboring cells by monitoring the cell-to-cell transitions in simulations. This approach has two major drawbacks compared to the SqRA-MSMs. First, accurately estimating the probability flux requires multiple crossings of cell boundaries, demanding thousands of simulation time steps and force evaluations—far more computationally intensive than the single force evaluation per grid cell in SqRA-MSMs. Second, it is challenging to ensure convergence, as these simulations must thoroughly sample both translational and rotational space. Additionally, the rates and metastable states in sampling-based MSMs are sensitive to statistical noise in flux estimates, making accuracy difficult to achieve and control. By contrast, the accuracy of a SqRA-MSM can be fully controlled by adjusting the space covered by the grid and the grid resolution.

A major advantage of sampling-based approaches compared to SqRA-MSMs is that simulations do not (usually) rely on the rigid body assumption. Therefore, sampling-based approaches naturally account for conformational flexibility of the two molecules and for solvent effects, which we know to be major contributors to molecular association. We anticipate that these effects could be incorporated into our method by including explicit solvent molecules to the energy calculation and by performing a short energy minimization while restraining the system to the grid cell. An alternative approach would be to average the results over a brief MD simulation. Achieving fully accurate energy values for each grid cell would ideally require constructing a free-energy surface,[77] although this is computationally expensive in six dimensions. In SEEKR2,[75,76] each individual grid cell can use a distinct simulation technique. This approach allows data from multiple sources, each with different computational demands, to be seamlessly integrated into a unified multiscale kinetic model. A similar strategy could potentially be applied to link SqRA-MSM with sampling-based approaches.

Overall, our grid-based approach to molecular association, in its current version, is best suited for molecular systems which require a computationally demanding energy function and are well approximated by assuming rigid-body behavior.

5. CONCLUSION AND OUTLOOK

The grid-based approach to molecular association offers significant computational advantages, as it requires only a single energy evaluation per grid cell, making it highly efficient. Notably, the number of grid points does not increase with system size, allowing the method to be applied to large molecular systems. With $O(10^4)$ to $O(10^5)$ grid points, this approach is compatible with computationally expensive energy functions, including energies based on electronic structure calculations.

We implemented the grid-based approach to molecular association in the python package MolGri. In its current version, the MolGri package offers a systematic approach to generating configurations for molecular association processes and analyzing their energies. This functionality makes it immediately valuable for studies of molecular association and for producing input structures for electronic structure calculations. MolGri can also be used to generate cluster or solvation shell configurations by first constructing a dimer grid, extracting the low-energy configurations, and then iteratively adding more molecules, using the extracted configurations as molecule A in subsequent steps.

MolGri can also be employed to construct SqRA-Markov models, which we have demonstrated to accurately identify the metastable states of molecular association processes. These models provide insight into the long-range interactions that steer molecular association and the underlying binding mechanism. However, the current implementation does not yet yield dynamically accurate results. The primary limitations are the neglect of transitions into the bulk and the omission of anisotropic rotational diffusion, which we aim to address in the next version of the model.

In summary, this grid-based method significantly reduces the number of energy evaluations required compared to MD simulations of molecular associations processes, while still offering a comprehensive view of the configuration space and estimates of key transition kinetics. Its potential applications span a range of fields, including dimer formation, nanoparticle growth, molecular self-assembly, protein−ligand binding, host−guest systems, and chemical reactions.

■ ASSOCIATED CONTENT

Data Availability Statement

The python package `MolGri` can be installed from PyPi (`pip install molgri`) or from the development repository on GitHub: (https://github.com/bkellerlab/molecularRotationalGrids).

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.jctc.4c01293.

Square-root approximation, water models, larger systems, Voronoi tessellation in Cartesian coordinates for the translational grid, and difference between Voronoi grids in spherical and Cartesian coordinates (PDF)

AUTHOR INFORMATION

Corresponding Author

Bettina G. Keller − *Department of Biology, Chemistry and Pharmacy, Freie Universität Berlin, 14195 Berlin, Germany;* orcid.org/0000-0002-7051-0888; Email: bettina.keller@fu-berlin.de

Author

Hana Zupan − *Department of Biology, Chemistry and Pharmacy, Freie Universität Berlin, 14195 Berlin, Germany;* orcid.org/0000-0002-5501-3883

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.jctc.4c01293

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) through grant CRC 1349 Fluorine-Specific Interactions, project id 387284271, and through grand CRC 1114 Scaling Cascades in Complex Systems, project id 235221301.

ADDITIONAL NOTE

[1]The difference between the mathematical objects *sphere* and *ball* is important here. We are interested in the three-dimensional object (embedded in 4D) of a 3-sphere, which is the surface of the 3-ball. The volumes of our hypersphere cells are sections of the 3-sphere. We are at no point interested in the 4D volume of a 3-ball.

REFERENCES

(1) Chodera, J. D. A simple method for automated equilibration detection in molecular simulations. *J. Chem. Theory Comput.* **2016**, *12*, 1799−1805.

(2) Grossfield, A.; Patrone, P. N.; Roe, D. R.; Schultz, A. J.; Siderius, D. W.; Zuckerman, D. M. Best Practices for Quantification of Uncertainty and Sampling Quality in Molecular Simulations [Article v1.0]. *Living J. Comput. Mol. Sci.* **2019**, *1*, 5067.

(3) Swope, W. C.; Pitera, J. W.; Suits, F. Describing protein folding kinetics by molecular dynamics simulations. 1. theory. *J. Phys. Chem. B* **2004**, *108*, 6571−6581.

(4) Buchete, N.-V.; Hummer, G. Coarse master equations for peptide folding dynamics. *J. Phys. Chem. B* **2008**, *112*, 6057−6069.

(5) Prinz, J.-H.; Wu, H.; Sarich, M.; Keller, B.; Senne, M.; Held, M.; Chodera, J. D.; Schütte, C.; Noé, F. Markov models of molecular kinetics: Generation and validation. *J. Chem. Phys.* **2011**, *134*, 174105.

(6) Husic, B. E.; Pande, V. S. Markov state models: From an art to a science. *J. Am. Chem. Soc.* **2018**, *140*, 2386−2396.

(7) Ge, Y.; Voelz, V. A. Markov state models to elucidate ligand binding mechanism. In *Protein-Ligand Interactions And Drug Design*; Humana: New York, NY, 2021, Vol. *2266*, pp 239−259. DOI: .

(8) Plattner, N.; Noé, F. Protein conformational plasticity and complex ligand-binding kinetics explored by atomistic simulations and markov models. *Nat. Commun.* **2015**, *6*, 7653.

(9) Olsson, S.; Markov state models of protein−protein encounters. *Protein Interactions: The Molecular Basis Of Interactomics*; Wiley: 2022, pp 163−193. DOI: .

(10) Schade, R.; Kenter, T.; Elgabarty, H.; Lass, M.; Kühne, T. D.; Plessl, C. Breaking the exascale barrier for the electronic structure problem in ab-initio molecular dynamics. *Int. J. High Perform. Comput. Appl.* **2023**, *37*, 530−538.

(11) Gaus, M.; Cui, Q.; Elstner, M. Dftb3: Extension of the self-consistent-charge density-functional tight-binding method (scc-dftb. *J. Chem. Theory Comput.* **2011**, *7*, 931−948.

(12) Kocer, E.; Ko, T. W.; Behler, J. Neural network potentials: A concise overview of methods. *Annu. Rev. Phys. Chem.* **2022**, *73*, 163−186.

(13) Noé, F.; Tkatchenko, A.; Müller, K.-R.; Clementi, C. Machine learning for molecular simulation. *Annu. Rev. Phys. Chem.* **2020**, *71*, 361−390.

(14) Kozlowski, N.; Grubmüller, H. Uncertainties in markov state models of small proteins. *J. Chem. Theory Comput.* **2023**, *19*, 5516−5524.

(15) He, Z.; Paul, F.; Roux, B. A critical perspective on Markov state model treatments of protein−protein association using coarse-grained simulations. *J. Chem. Phys.* **2021**, *154*, 084101.

(16) Schwantes, C. R.; Pande, V. S. Modeling molecular kinetics with tica and the kernel trick. *J. Chem. Theory Comput.* **2015**, *11*, 600−608.

(17) Nagel, D.; Sartore, S.; Stock, G. Selecting features for markov modeling: a case study on hp35. *J. Chem. Theory Comput.* **2023**, *19*, 3391−3405.

(18) Nuske, F.; Keller, B. G.; Pérez-Hernández, G.; Mey, A. S.; Noé, F. Variational approach to molecular kinetics. *J. Chem. Theory Comput.* **2014**, *10*, 1739−1752.

(19) Schütte, C.; Noé, F.; Lu, J.; Sarich, M.; Vanden-Eijnden, E. Markov state models based on milestoning. *J. Chem. Phys.* **2011**, *134*, 204105.

(20) Bowman, G. R.; Ensign, D. L.; Pande, V. S. Enhanced modeling via network theory: Adaptive sampling of markov state models. *J. Chem. Theory Comput.* **2010**, *6*, 787−794.

(21) Keller, B. G.; Bolhuis, P. G. Dynamical reweighting for biased rare event simulations. *Annu. Rev. Phys. Chem.* **2024**, *75*, 137−162.

(22) Lie, H. C.; Fackeldey, K.; Weber, M. A square root approximation of transition rates for a markov state model. *SIAM J. Matrix Anal. Appl.* **2013**, *34*, 738−756.

(23) Donati, L.; Heida, M.; Keller, B. G.; Weber, M. Estimation of the infinitesimal generator by square-root approximation. *J. Phys.: Condens. Matter* **2018**, *30*, 425201.

(24) Heida, M.; Kantner, M.; Stephan, A. Consistency and convergence for a family of finite volume discretizations of the Fokker−Planck operator. *ESAIM: Math. Modell. Numer. Anal.* **2021**, *55*, 3017−3042.

(25) Donati, L.; Weber, M.; Keller, B. G. Markov models from the square root approximation of the Fokker−Planck equation: Calculating the grid-dependent flux. *J. Phys.: Condens. Matter* **2021**, 33, 115902.

(26) Donati, L.; Weber, M.; Keller, B. G. A review of girsanov reweighting and of square root approximation for building molecular markov state models. *J. Math. Phys.* **2022**, *63*, 123306.

(27) Gabdoulline, R. R.; Wade, R. C. Biomolecular diffusional association. *Curr. Opin. Struct. Biol.* **2002**, *12*, 204−213.

(28) Zupan, H.; Heinz, F.; Keller, B. G. Grid-based state space exploration for molecular binding. *Can. J. Chem.* **2023**, *101*, 710−724.

(29) Kuffner, J. J.Effective sampling and distance metrics for 3D rigid body path planning, In *IEEE International Conference on Robotics and Automation*, IEEE, 2004, Vol. *4*, pp 3993−3998.

(30) Lindemann, S. R.; Yershova, A.; LaValle, S. M., Incremental grid sampling strategies in robotics, In *Algorithmic Foundations of Robotics VI*, Springer, 2005, pp 313−328.

(31) Ichnowski, J.; Alterovitz, R., Fast nearest neighbor search inSE (3) for sampling-based motion planning, In *Algorithmic Foundations of Robotics XI: Selected Contributions of the Eleventh International Workshop on the Algorithmic Foundations of Robotics*, Springer, 2015, pp 197−214.

(32) Diebel, J. Representing attitude: Euler angles, unit quaternions, and rotation vectors. *Matrix* **2006**, *58*, 1−35.

(33) Karney, C. F. Quaternions in molecular modeling. *J. Mol. Graphics Modell.* **2007**, *25*, 595−604.

(34) Hamilton, W. R. Xxxi. on quaternions; or on a new system of imaginaries in algebra. *London, Edinburgh Dublin Philos. Mag. J. Sci.* **1845**, *26*, 220−224.

(35) Kirk, D. *Graphics Gems III (IBM Version)*. Elsevier, 2012.

(36) Yershova, A.; Jain, S.; Lavalle, S. M.; Mitchell, J. C. Generating uniform incremental grids onSO (3) using the hopf fibration. *Int. J. Robot. Res.* **2010**, *29*, 801−812.

(37) Yershova, A.; LaValle, S. M.Deterministic sampling methods for spheres and SO (3), In *IEEE International Conference on Robotics and Automation*, IEEE, 2004, Vol. *4*, pp 3974−3980.

(38) Roşca, D.; Morawiec, A.; De Graef, M. A new method of constructing a grid in the space of 3D rotations and its applications to texture analysis. *Modell. Simul. Mater. Sci. Eng.* **2014**, *22*, 075013.

(39) Pietrucci, F. Strategies for the exploration of free energy landscapes: Unity in diversity and challenges ahead. *Phys. Rev.* **2017**, *2*, 32−45.

(40) Wikipedia. *Voronoi diagram*. 2024, https://en.wikipedia.org/wiki/Voronoi_diagram. (accessed 10 September 2024).

(41) Todhunter, I. *Spherical trigonometry, for the use of colleges and schools: with numerous examples*; Macmillan, 1863.

(42) Caroli, M.; de Castro, P. M. M.; Loriot, S.; Rouiller, O.; Teillaud, M.; Wormser, C. *Robust and Efficient Delaunay triangulations of points on or close to a sphere*. Ph.D. thesis, School INRIA, 2009.

(43) Van Oosterom, A.; Strackee, J. The solid angle of a plane triangle. *IEEE Trans. Biomed. Eng.* **1983**, *BME-30*, 125−126.

(44) Huynh, D. Q. Metrics for 3D rotations: Comparison and analysis. *J. Math. Imaging Vis.* **2009**, *35*, 155−164.

(45) Delaunay, B. Sur la sphère vide, izvestiya akademii nauk sssr. *Otdelenie Matematicheskii I Estestvennyka Nauk* **1934**, *7*, 793−800.

(46) Murray, R. M.; Li, Z.; Sastry, S. S. *A mathematical introduction to robotic manipulation*; CRC press, 2017.

(47) 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.

(48) Hanwell, M. D.; Curtis, D. E.; Lonie, D. C.; Vandermeersch, T.; Zurek, E.; Hutchison, G. R. Avogadro: an advanced semantic chemical editor, visualization, and analysis platform. *J. Cheminf.* **2012**, *4*, 17.

(49) Koziara, K. B.; Stroet, M.; Malde, A. K.; Mark, A. E. Testing and validation of the automated topology builder (atb) version 2.0: prediction of hydration free enthalpies. *J. Comput.-Aided Mol. Des.* **2014**, *28*, 221−233.

(50) Canzar, S.; El-Kebir, M.; Pool, R.; Elbassioni, K.; Malde, A. K.; Mark, A. E.; Geerke, D. P.; Stougie, L.; Klau, G. W. Charge group partitioning in biomolecular simulation. *J. Comput. Biol.* **2013**, *20*, 188−198.

(51) Ye, S.; Loll, B.; Berger, A.; Mülow, U.; Alings, C.; Wahl, M.; Koksch, B. Fluorine teams up with water to restore inhibitor activity to mutant BPTI. *Chem. Sci.* **2015**, *6*, 5246−5254.

(52) Wehrhan, L.; Keller, B. G. Prebound State Discovered in the Unbinding Pathway of Fluorinated Variants of the Trypsin−BPTI Complex Using Random Acceleration Molecular Dynamics Simulations. *J. Chem. Inf. Model.* **2024**, *64*, 5194−5206.

(53) Van Rossum, G.; Drake, F. L. *Python 3 Reference Manual*; CreateSpace, 2009.

(54) Mölder, F.; Jablonski, K. P.; Letcher, B.; Hall, M. B.; Tomkins-Tinch, C. H.; Sochat, V.; Forster, J.; Lee, S.; Twardziok, S. O.; Kanitz, A.; et al. Sustainable data analysis with snakemake. *F1000Research* **2021**, *10*, 33.

(55) Harris, C. R.; Millman, K. J.; van der Walt, S. J.; Gommers, R.; Virtanen, P.; Cournapeau, D.; Wieser, E.; Taylor, J.; Berg, S.; Smith, N. J.; Kern, R.; Picus, M.; Hoyer, S.; van Kerkwijk, M. H.; Brett, M.; Haldane, A.; Del Río, J. F.; Wiebe, M.; Peterson, P.; Gérard-Marchant, P.; Sheppard, K.; Reddy, T.; Weckesser, W.; Abbasi, H.; Gohlke, C.; Oliphant, T. E. Array programming withNumPy. *Nature* **2020**, *585*, 357−362.

(56) Virtanen, P.; Gommers, R.; Oliphant, T. E.; Haberland, M.; Reddy, T.; Cournapeau, D.; Burovski, E.; Peterson, P.; Weckesser, W.; Bright, J.; et al. SciPy 1.0 Contributors, SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nat. Methods* **2020**, *17*, 261−272.

(57) Hagberg, A. A.; Schult, D. A.; Swart, P. J.Exploring network structure, dynamics, and function using networkx. In *Proceedings of the 7th Python in Science Conference*, Los Alamos National Laboratory (LANL), 2008, pp 11−15.

(58) Gowers, R. J.; Linke, M.; Barnoud, J.; Reddy, T. J.; Melo, M. N.; Seyler, S. L.; Domanski, J.; Dotson, D. L.; Buchoux, S.; Kenney, I. M., et al.MDAnalysis: a python package for the rapid analysis of molecular dynamics simulations, In *Proceedings of the 15th python in science conference*, SciPy, 2016, Vol. *98*, p 105.

(59) Michaud-Agrawal, N.; Denning, E. J.; Woolf, T. B.; Beckstein, O. MDAnalysis: a toolkit for the analysis of molecular dynamics simulations. *J. Comput. Chem.* **2011**, *32*, 2319−2327.

(60) Ahmed, M.; Seraj, R.; Islam, S. M. S. The k-means algorithm: A comprehensive survey and performance evaluation. *Electronics* **2020**, *9*, 1295.

(61) Pedregosa, F. Scikit-learn: Machine learning in python. *J. Mach. Learn. Res.* **2011**, *12*, 2825.

(62) Van Der Spoel, D.; Lindahl, E.; Hess, B.; Groenhof, G.; Mark, A. E.; Berendsen, H. J. GROMACS: fast, flexible, and free. *J. Comput. Chem.* **2005**, *26*, 1701−1718.

(63) 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*, 19−25.

(64) Lindahl, E.; Hess, B.; Van Der Spoel, D. GROMACS 3.0: a package for molecular simulation and trajectory analysis. *J. Mol. Model.* **2001**, *7*, 306−317.

(65) Berendsen, H. J.; van der Spoel, D.; van Drunen, R. GROMACS: A message-passing parallel molecular dynamics implementation. *Comput. Phys. Commun.* **1995**, *91*, 43−56.

(66) Hess, B.; Bekker, H.; Berendsen, H. J.; Fraaije, J. G. LINCS: a linear constraint solver for molecular simulations. *J. Comput. Chem.* **1997**, *18*, 1463−1472.

(67) Goga, N.; Rzepiela, A.; De Vries, A.; Marrink, S.; Berendsen, H. Efficient algorithms for Langevin and DPD dynamics. *J. Chem. Theory Comput.* **2012**, *8*, 3637−3649.

(68) Kieninger, S.; Keller, B. G. GROMACS stochastic dynamics and BAOAB are equivalent configurational sampling algorithms. *J. Chem. Theory Comput.* **2022**, *18*, 5792−5798.

(69) Beglov, D.; Roux, B. Finite representation of an infinite bulk system: Solvent boundary potential for computer simulations. *J. Chem. Phys.* **1994**, *100*, 9050−9063.

(70) Linke, M.; Köfinger, J.; Hummer, G. Fully anisotropic rotational diffusion tensor from molecular dynamics simulations. *J. Phys. Chem. B* **2018**, *122*, 5630−5639.

(71) Wehrhan, L.; Keller, B. G. Fluorinated Protein−Ligand Complexes: A Computational Perspective. *J. Phys. Chem. B* **2024**, *128*, 5925−5934.

(72) Jiang, F.; Kim, S.-H. "soft docking": matching of molecular surface cubes. *J. Mol. Biol.* **1991**, *219*, 79−102.

(73) Katchalski-Katzir, E.; Shariv, I.; Eisenstein, M.; Friesem, A. A.; Aflalo, C.; Vakser, I. A. Molecular surface recognition: determination of geometric fit between proteins and their ligands by correlation techniques. *Proc. Natl. Acad. Sci. U. S. A.* **1992**, *89*, 2195−2199.

(74) Eisenstein, M.; Katchalski-Katzir, E. On proteins, grids, correlations, and docking. *C. R. Biol.* **2004**, *327*, 409−420.

(75) Votapka, L. W.; Jagger, B. R.; Heyneman, A. L.; Amaro, R. E. Seekr: simulation enabled estimation of kinetic rates, a computational tool to estimate molecular kinetics and its application to trypsin−benzamidine binding. *J. Phys. Chem. B* **2017**, *121*, 3597−3606.

(76) Votapka, L. W.; Stokely, A. M.; Ojha, A. A.; Amaro, R. E. Seekr2: Versatile multiscale milestoning utilizing the openmm molecular dynamics engine. *J. Chem. Inf. Model.* **2022**, *62*, 3253−3262.

(77) Badaoui, M.; Kells, A.; Molteni, C.; Dickson, C. J.; Hornak, V.; Rosta, E. Calculating kinetic rates and membrane permeability from biased simulations. *J. Phys. Chem. B* **2018**, *122*, 11571−11578.