

Towards grid-based models for molecular association

Hana Zupan and Bettina G. Keller^{a)}

*Freie Universität Berlin, Department of Biology, Chemistry and Pharmacy,
Arnimallee 22, 14195 Berlin, Germany*

(Dated: 1 October 2024)

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 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.

Keywords: Markov state models, molecular binding, rigid body, square-root approximation, spherical Voronoi, quaternions

I. 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 timescales 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. They thus give insight into the mechanism of multi-state molecular dynamics. One of the fields in which MSM

^{a)}Electronic mail: bettina.keller@fu-berlin.de

have been particularly useful is in modeling molecular association, such as protein-ligand and protein-protein binding [7–9].

However, MSMs, in particular MSMs of molecular association processes, are very sensitive to statistical uncertainties [10, 11] 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 [12, 13], variational and core-set Markov models [4, 14, 15], adaptive sampling algorithms [16], and enhanced sampling combined with dynamical reweighting [17]. 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 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 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 [18–21] 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 [18, 19, 21, 22]. A proof of principle for alanine-dipeptide has been reported in Ref. [19]. 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 the association of two molecules A and B in solution. The formation of bi-molecular complexes typically consists of two stages: 1) diffusion-based association and 2) interaction-based completion of binding [23]. 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. 24, 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 [25–27], 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.

II. THEORY

A. Square-root approximation

The square-root approximation has been derived and tested in Refs. [18–21]. In this section, we summarize the most important equations. We additionally provide a more detailed derivation in the supplementary material.

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: $x_i : \mathbb{R}^{3N} \mapsto \mathbb{R}$. We assume that in a low dimensional collective variable space $\mathbf{x} = (x_1, x_2 \dots x_m) \in \Omega \subset \mathbb{R}^m$, where $m \ll 3N$, the dynamics of the system can be modelled by overdamped Langevin dynamics

$$d\mathbf{x}(t) = \boldsymbol{\mu}(\mathbf{x}(t))dt + \sigma d\mathbf{B}(t). \quad (1)$$

$\mathbf{B}(t) = (B_1(t) \dots B_m(t))$ is an m -dimensional Wiener process, $\boldsymbol{\mu}$ is the m -dimensional drift vector $\boldsymbol{\mu}(\mathbf{x}(t)) = -\xi^{-1}M^{-1}\nabla V_{\text{eff}}(\mathbf{x}(t))$. $\sigma = \sqrt{2k_B T \xi^{-1} M^{-1}}$ scales the Wiener process and is linked to the diffusion of the system in the collective variables space. We assume that the diffusion is isotropic in the collective variable space, and hence σ is simply a scalar. For non-isotropic 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, $V_{\text{eff}} : \Omega \mapsto \mathbb{R}$ 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 $f : \mathbb{R}^m \rightarrow \mathbb{R}$.

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

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

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

$$\pi(\mathbf{x}) = Z^{-1} \exp\left(-\frac{1}{k_B T} V_{\text{eff}}(\mathbf{x})\right) \quad (3)$$

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

The collective variable space Ω is discretized into N_d non-overlapping grid cells $\Omega_1, \dots, \Omega_{N_d}$, where \mathbf{x}_{α} denotes the center of cell Ω_{α} . On this grid, eq. 2 can be approximated by a matrix-vector equation

$$\frac{d}{dt} \boldsymbol{\rho}^{\top}(t) = \boldsymbol{\rho}^{\top}(t) \mathbf{Q} \quad (4)$$

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

$$\mathbf{Q} : Q_{\alpha\beta} = \begin{cases} Q_{\alpha\beta, \text{adjacent}} & \alpha \sim \beta \\ 0 & \alpha \not\sim \beta \\ -\sum_{\beta=1, \beta \neq \alpha}^{N_d} Q_{\alpha\beta, \text{adjacent}} & \alpha = \beta. \end{cases} \quad (5)$$

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 [17]

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

where τ_{MSM} is the MSM lag time.

The square-root approximation of \mathcal{Q} [18, 19, 21] provides an analytical expression for the transition rate constant between adjacent cells

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

where $h_{\alpha\beta} = |\mathbf{x}_\beta - \mathbf{x}_\alpha|$ is the Euclidean distance between two cell centers. $\mathcal{V}_\alpha = \int_{\Omega_\alpha} d\mathbf{x} 1$ is the volume of Ω_α , and $\mathcal{S}_{\alpha\beta} = \oint_{\partial\Omega_\alpha \partial\Omega_\beta} d\mathbf{x} 1$ 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 $\sigma = \text{const.}$

The significance of eq. 7 is that, given the geometric parameters of the grid $h_{\alpha\beta}$, $\mathcal{S}_{\alpha\beta}$ and \mathcal{V}_α , 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 [18, 19, 21].

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 [17]. This yields the transpose of \mathbf{Q} .

B. 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 re-conceptualize 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 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

$$\mathbf{r}^{(k)} = (\mathbf{r}_1^{(k)}, \mathbf{r}_2^{(k)}, \dots, \mathbf{r}_{N_k}^{(k)}), \quad k = A, B, \quad (8)$$

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 i th atom in the respective molecule. The vector $\mathbf{r}^{(k)}$ can also be represented by translational, rotational and internal coordinates

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

The 3-dimensional center-of-mass coordinate is

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

where $m_i^{(k)}$ is the mass of the i th 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, these radius and angles cover the 3-dimensional space: $\mathbb{R}_+ \times S^2 = \mathbb{R}^3$

$\mathbf{q}^{(k)} \in SO(3)$ 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. [28] for a review on different representations of the rotational group $SO(3)$ and Ref. [29] 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 [30]. 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 [31]:

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

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 $q_0 > 0$. For $q_0 = 0$, quaternions with $q_1 > 0$ are included. If $q_1 = 0$, the decision is based on the third component.)

Finally, $\mathbf{v}^{(k)} = (v_1^{(k)}, v_2^{(k)}, \dots, v_{3N_k-6}^{(k)})$ are the $3N_k - 6$ internal degrees of freedom. Within the rigid-body approximation, we assume that the internal degrees of freedom are constant: $\mathbf{v}^{(k)} = \text{const.}$ 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

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

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.

C. 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 [29, 32–34]. In Ref. [24], 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 , $i = 1, 2 \dots N_r$, are equidistantly spaced between selected r_{\min} and r_{\max} . The polyhedron approach to discretize S^2 is illustrated in Fig. 1.a. First, an icosahedron is inscribed into a 2-sphere. The twelve vertices of this icosahedron yield the grid points (ϕ_i, θ_i) for the coarsest grid on S^2 (black dots in Fig. 1.a). Each face of the icosahedron is an equilateral triangle. Further grid points are created at mid-points of the icosahedron edges and then scaled to lie on the 2-sphere (red dots in Fig. 1.a). 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 Fig. 1.a). 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

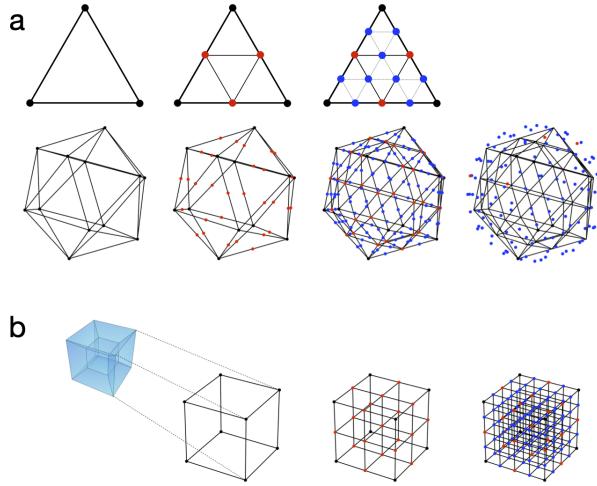


FIG. 1. Illustration of grid construction for S^2 (a) and S^3 (b). For S^3 , only one of hypercube cells is shown, as the projection onto the 3-sphere cannot be depicted.

largest grid and removing an appropriate number of points, a topic we also discussed in our previous publication [24]. 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 to 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 Fig. 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 $\mathbf{q}_j = \frac{1}{2}(\pm 1, \pm 1, \pm 1, \pm 1)$. 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 (Fig 1.b). The 16 vertices of the 4-cube yield the grid points \mathbf{q}_j for the coarsest grid on S^3 (black dots in Fig. 1.b). 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 Fig. 1.b). The new points are scaled to unit length to ensure that they lie on S^3 . As in Fig. 1.a, this process can be repeated iteratively to achieve finer and finer grids (e.g. blue dots in Fig. 1.b). 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 , $\mathbf{r}_{ij}^{(B)}$, 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 $\mathbf{r}_{\text{ref}}^{(B)} = (\mathbf{r}_{\text{COM}}^{(B)} = (0, 0, 0), \mathbf{q}_{\text{ref}}^{(B)}, \mathbf{v}^{(B)})$. That is, B is placed at the origin of the coordinate system and in a specific reference rotation $\mathbf{q}_{\text{ref}}^{(B)}$. 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

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

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

D. Energy of grid cells

The energy associated to a grid cell (ij) is given by the effective potential $V_{\text{eff}}(\mathbf{x}_{ij}) = V_{\text{eff}}(\mathbf{r}_{\text{COM},i}^{(B)}, \mathbf{q}_j^{(B)})$. Obtaining $V_{\text{eff}}(\mathbf{x})$ requires a free-energy calculation [35] 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 bi-molecular system

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

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 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.

E. Distances, surfaces and volumes of grid cells

The grid points \mathbf{x}_{ij} induce a Voronoi-like tessellation [36] of the six-dimensional translation and rotation space, where each point in this space is assigned to its closest grid point forming non-overlapping 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 tessellation in Cartesian space are discussed in section IV D.

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}_α 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 $\mathbf{x}_\beta = \mathbf{x}_{kl} = (\mathbf{t}_k, \mathbf{q}_l)$.

1. Translation grid

The translation grid is constructed from the radial and spherical sub-grids. This leads to two types of adjacency relations: i) radial neighbours (orange and blue cells in Fig. 2d and 2e), and ii) angular neighbours (orange and blue cells in Fig. 2b). 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 $\mathbf{t}_k = (r_{k=i \pm 1}, \phi_{k=i}, \theta_{k=i})$. Angular neighbors have the same radius but are neighbors on the spherical grid: $\mathbf{t}_i = (r_i, \phi_i, \theta_i)$ 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 $R_1, R_2 \dots R_{N_r-1}$ at mid-points between grid points r_i as shown in Fig. 2a. The distance metric on the radial grid is

$$h_{ik,\text{radial}} = |r_k - r_i|. \quad (15)$$

The intersecting surface is calculated as the area of the n -sided spherical polygon with interior angles $\alpha_1 \dots \alpha_N$ [37] (Fig. 2b).

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

To define the Voronoi tessellation of the spherical grid [38, 39], we use the angular distance

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

where $|\mathbf{t}_i| = r_i$ 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 (Fig. 2b). The intersecting area is a part of the corresponding circular sector (Fig. 2f) and can be calculated by subtracting the area of the circular sector with the smaller radius from the one with the larger radius

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

where β_{ik} is given by eq. 17. In summary

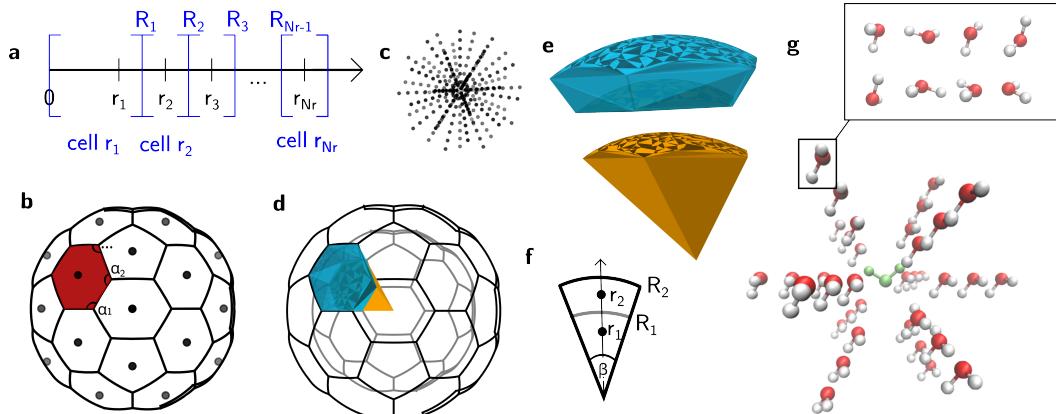


FIG. 2. Translation grid. **a** radial grid with blue lines showing the cell boundaries. **b** angular grid, with spherical Voronoi division of the unit sphere (example of 42-point icosahedron grid), area shaded in red denotes the cell assigned to this grid point. **c** Example of translation grid with $N_s = 42$ 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.

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

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 fsurface of the grid cell surface on this sphere:

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

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

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

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

2. Rotation grid

The distance metric in rotation space [40] is based on the angle between the unit quaternions \mathbf{q}_j and \mathbf{q}_l of two adjacent rotation grid cells

$$h_{jl}(\mathbf{q}_j, \mathbf{q}_l) = \min \{d_{\text{ang}}(\mathbf{q}_j, \mathbf{q}_l), \pi - d_{\text{ang}}(\mathbf{q}_j, \mathbf{q}_l)\} \quad (22)$$

with

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

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 Fig. 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 performing the following algorithm.

Let $v_1 \dots v_s$ be Cartesian coordinates of Voronoi vertices shared between neighbouring hypersphere cells \mathbf{q}_j and \mathbf{q}_l (as a condition of neighbourhood, cells must share at least three vertices). As they all share the property of equal distance to \mathbf{q}_j and \mathbf{q}_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 options imply that there is no intersecting hyper-surface between \mathbf{q}_j and \mathbf{q}_l and will not occur if \mathbf{q}_j and \mathbf{q}_l are adjacent. The third option tells us that the intersecting hyper-surface between \mathbf{q}_j and \mathbf{q}_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 $v_1 \dots v_s$ that define 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 volumes¹ 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

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.

of surface triangulation that is shown in Fig. 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 [41] 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:

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

3. Configurational grid

To construct the configuration grid on $SE(3)$, we combine translation and rotation grid. For two centers on the configurational 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

$$\begin{aligned} (i) \quad & \mathbf{x}_\alpha \sim \mathbf{x}_\beta && \text{if } \mathbf{t}_i \sim \mathbf{t}_k \text{ and } \mathbf{q}_j = \mathbf{q}_l \\ (ii) \quad & \mathbf{x}_\alpha \sim \mathbf{x}_\beta && \text{if } \mathbf{t}_i = \mathbf{t}_k \text{ and } \mathbf{q}_j \sim \mathbf{q}_l \\ (iii) \quad & \mathbf{x}_\alpha \not\sim \mathbf{x}_\beta && \text{otherwise.} \end{aligned} \quad (25)$$

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

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

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

$$\mathcal{V}_\alpha = \mathcal{V}_i \times \alpha^3 \mathcal{V}_j \quad (27)$$

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 [42]. We currently set $\alpha = 1$.

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

F. 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]

$$\begin{aligned} \mathbf{Q}\psi_i &= \kappa_i \psi_i \Leftrightarrow \mathbf{P}(\tau_{\text{MSM}})\psi_i = \exp(\kappa_i \tau_{\text{MSM}})\psi_i \\ \phi_i^\top \mathbf{Q} &= \kappa_i \phi_i^\top \Leftrightarrow \phi_i^\top \mathbf{P}(\tau_{\text{MSM}}) = \exp(\kappa_i \tau_{\text{MSM}})\phi_i^\top , \end{aligned} \quad (28)$$

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

$$\boldsymbol{\pi}_\alpha Q_{\alpha\beta} = \boldsymbol{\pi}_\beta Q_{\beta\alpha} . \quad (29)$$

Consequently, right and left eigenvectors are linked by $\text{diag}(\boldsymbol{\pi})\psi_i = \phi_i$, 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 $P_{\alpha\beta}(\tau_{\text{MSM}})$ from an MD simulation. In the grid-based SqRA approach, they are obtained by evaluating the energy and geometric properties of each grid cell. Fig. 3 compares the two approaches.

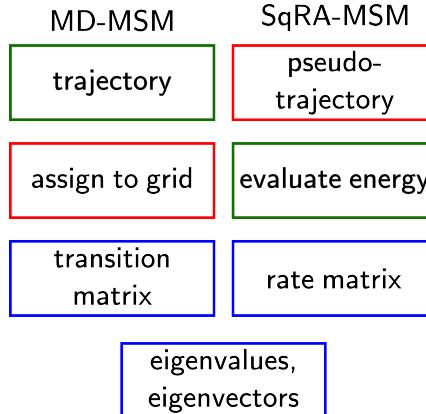


FIG. 3. Sampling-based and grid-based approach to construct a Markov model. The fields outlined in red denote steps that involve force field evaluation, those in green connect the molecular structures to the grid points.

III. COMPUTATIONAL METHODS

A. 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, $N_s = 80$ rotational grid points, and $N_o = 80$ orientational grid points. This yielded a grid with $N_d = N_r \times N_r \times N_r = 64'000$ grid cells. We calculated the volume \mathcal{V}_α 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, water molecule A was placed at the origin of the coordinate system, and water molecule B was translated and rotated to each of the grid cells as described in section II. 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, where the water molecules were modelled using the TIP3P water model [43]. From this information, we calculated the factor $\sqrt{\pi(\mathbf{x}_\alpha)/\pi(\mathbf{x}_\beta)}$ for all pairs of adjacent cells. The $(N_d \times N_d)$ -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 [44] package called **molgri** (for **molecular grids**) and the computational experiments formatted as snakemake 8.14.0 [45] pipelines for better reproducibility. Major Python dependencies used in this project are numpy 1.26.4 [46], scipy 1.13.1 [47], networkx 3.3 [48] and mdanalysis 2.7.0 [49, 50].

B. Spectral clustering of \mathbf{Q}

Spectral clustering was performed on the first six right eigenvectors of the \mathbf{Q} . The clustering algorithm used was KMeans [51] as implemented in scikit-learn 1.5 [52] and the choice of 12 clusters was made.

C. Molecular dynamics simulations

Molecular dynamics simulations were conducted with GROMACS 2022 [53–56].

For the vacuum simulations, two water molecules were placed in a cubic box with 3 nm edge length. The interactions were modelled using the TIP3P water model [43]. O-H bond lengths were constrained using LINCS algorithm [57]. The dynamics were propagated using the built-in stochastic integrator for Langevin dynamics [58, 59] (setting `sd`) with a timestep of $\Delta t = 1$ 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 $4 \cdot 10^7$ timesteps, corresponding to 40 ns simulation time. Long range interactions were cut-off 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 (Fig. 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 nm²) for $0.5 \text{ nm} \leq r < 0.7$ nm, and a linear restraining potential with force constant $k = 500$ kJ/(mol nm²) for $r \geq 0.7$ nm. No pressure coupling was applied. translations 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 modelled using the TIP3P water model [43]. 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 modelling a helium atom. The simulation parameters are the same as for the vacuum simulations with the exception of the long-range interactions, where we used Ewald summation with a long-range cut-off of 1.4 nm and time step Δt which was increased to 2 fs.

D. 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 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 ps and 1.0 ps but always show $\tau_{\text{MSM}} = 0.1$ 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 timescales were calculated as $t_{\text{its},i} = -\tau_{\text{MSM}} / \ln(\lambda_i(\tau_{\text{MSM}}))$, where $\lambda_i(\tau_{\text{MSM}})$ is the i th MSM eigenvalue.

IV. RESULTS AND DISCUSSION

A. 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 $6.4 \cdot 10^4$ molgrid cells. Fig. 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 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*.

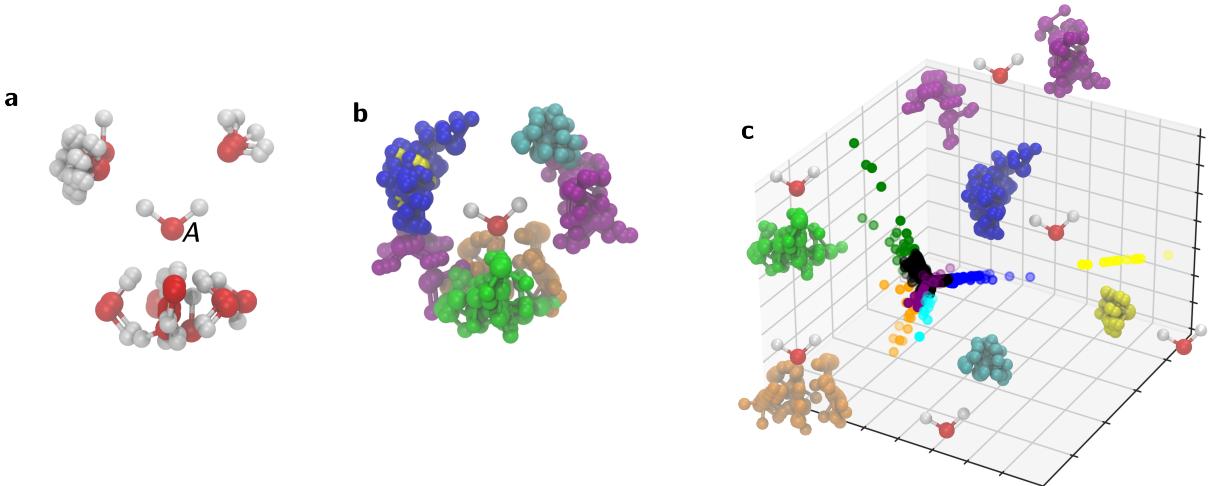


FIG. 4. **a** Most probable configurations of the bi-molecular system according to the stationary distribution calculated with the SqRA-Markov model. Molecule *A* is labeled, the rest of the structures represent molecule *B*. **b** K-Means clustering performed on the space of the first six eigenvectors 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.

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. Fig. 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 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 behaviour 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 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 timescales 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 ps and 1.0 ps, the implied timescale of the slowest process decreases from 3646 ps to 3.65 ps (see SI Fig. 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.

B. Comparison to MD

Fig. 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 $6.4 \cdot 10^4$ grid cells as the SqRA-Markov model. We sampled extensively (40 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 bi-molecular 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. While the sampling-based MSMs do identify the four regions of hydrogen bonding (see eg. the third eigenvector in the second row of Fig. 5), there is significantly more noise in the transition matrix decomposition than in the comparable rate matrix eigenvectors despite the fact that the trajectories used in MSM creation involved orders of magnitude more force field evaluations than the grid-based approach.

As in the SqRA-Markov models, the eigenvectors are largely unaffected by the magnitude of τ_c (SI Fig. S2-S5). In the vacuum simulations (SI Fig. S2 and S3), the implied timescales decrease slightly with decreasing friction, but not orders of magnitude as in the SqRA-Markov model. In the simulations with explicit solvent (SI Fig. S2 and S3), the implied timescales do not change 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 (eg. second eigenvector in row 2 in Fig. 5). This prompted us investigate boundary conditions further, see the following section.

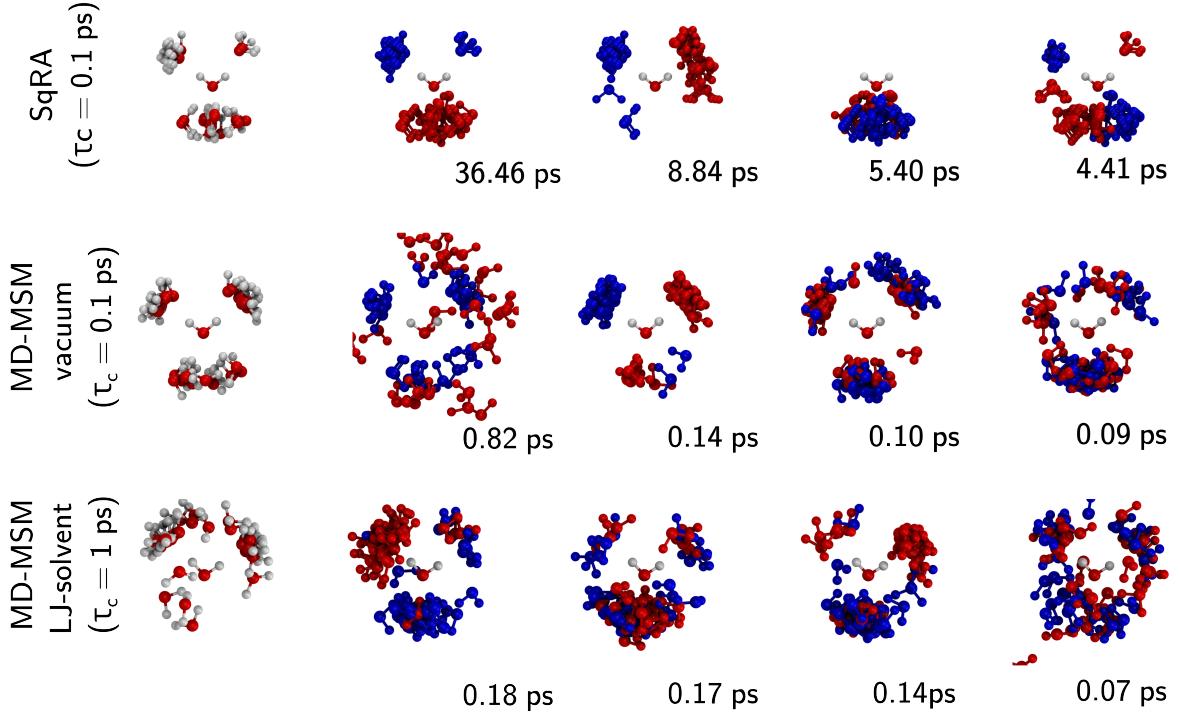


FIG. 5. First five eigenvectors of water dimer system at three different simulation conditions. Top: SqRA model with $\tau_c = 0.1 \text{ ps}$, middle: 40 ns MD trajectory in vacuum with $\tau_c = 0.1 \text{ ps}$, bottom: 80 ns MD trajectory in a box of LJ particles with $\tau_c = 1 \text{ ps}$. In **blue**: 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 timescales are noted below the corresponding eigenvector.

C. 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 neighbouring 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 Fig. 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 Fig. 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

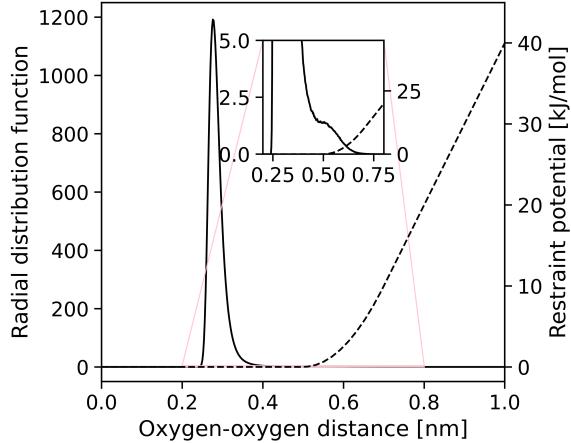


FIG. 6. Radial distribution function (full line) and restraint potential (dashed line) for a MD run of two water molecules in a box of Lennard-Jones particles.

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 to 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.

Fig. 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. The radial transitions appear to mix with transitions between metastable states in eigenvectors 4 and 5. 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. [60].

D. Non-linear 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 $\frac{1}{h_{\alpha\beta}} \frac{S_{\alpha\beta}}{V_\alpha}$ 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. Fig. 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.

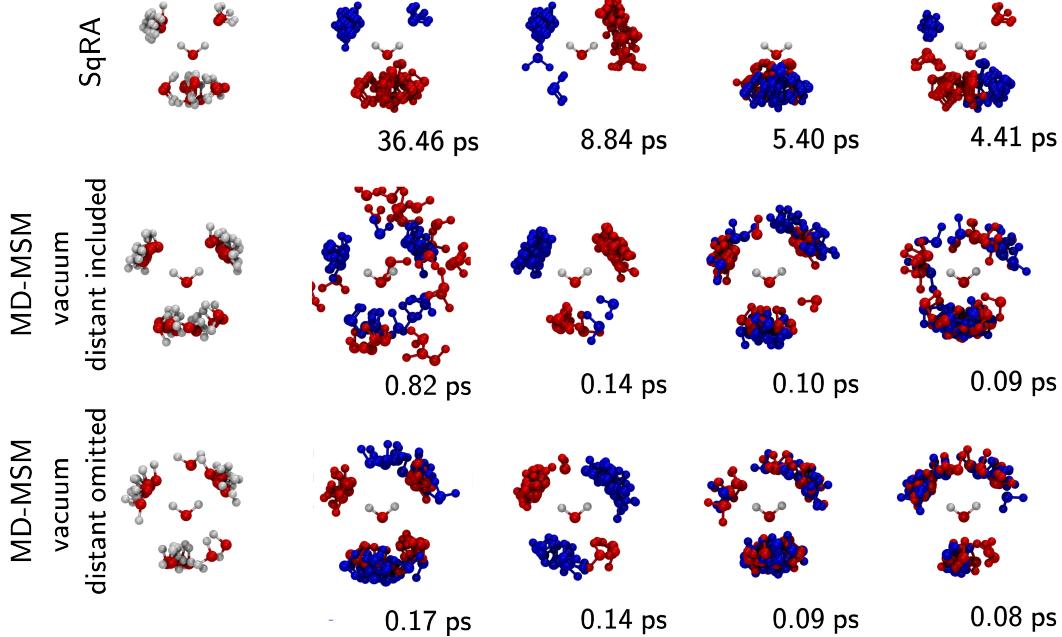


FIG. 7. First five eigenvectors of the water dimer system at three different treatments of distant structures. Top: SqRA model, distant structures do not exist by construction, transition into this region is assumed impossible; middle: MD trajectory in vacuum, distant structures are assigned to the best available cell; bottom: MD trajectory in vacuum, distant structures are not assigned to any cell. Coupling time to a thermostat is $\tau_C = 0.1$ 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 Fig. 5.

The situation is more complicated for the rotation grid, because rotational diffusion occurs in inherently non-linear coordinates. This non-linearity 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 $\mathbf{D} \in \mathbb{R}^{3 \times 3}$. This matrix depends on the moments of inertia of the rotating molecule. The further the molecule's shape deviates from spherical symmetry, the more \mathbf{D} will deviate from $D\text{Id}$, 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. [61]. To incorporate anisotropic diffusion into the SqRA-Markov model, we need an analytical expression for the rotational diffusion tensor and must re-derive the prefactor in eq. 7 for anisotropic diffusion.

E. Computational cost

A limit on the feasibility of SqRA is set by the number of discretization cells N_d . As the rate matrix is of shape $N_d \times N_d$, memory constraints quickly become an issue even if sparse matrices are used; for most discretization schemes, the ceiling of N_d seems to be around $\mathcal{O}(10^5)$ [21]. This why we limit ourselves to a reduced set of coordinates with 6 DoF, 3 corresponding to translational motion of the centre of mass and 3 to rotation around principal axes.

It should be noted that the entire discussion of grids and their geometric properties is entirely independent from specific molecular systems; very detailed grids can be pre-

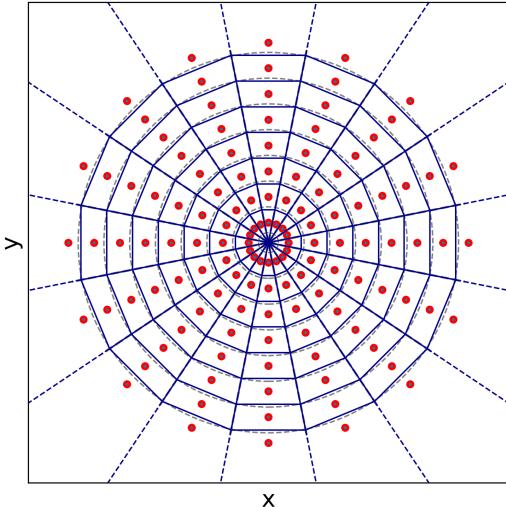


FIG. 8. Comparison between Voronoi tessellation in polar coordinates (dashed gray lines) and Voronoi tessellation in Cartesian coordinates (blue grid) for a regular grid in polar coordinates (r, θ) (red dots)

calculated to high precision in advance and applied repeatedly to various systems, bringing the cost of geometry evaluation in calculations to essentially zero.

V. 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 $\mathcal{O}(10^4)$ to $\mathcal{O}(10^5)$ 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.

Another concern is that the rigid-body assumption overlooks induced fit and solvent effects, which we know to be major contributors to molecular association. We anticipate that these effects could be incorporated by including explicit solvent molecules to the energy calculation and 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 [62], although this is computationally expensive in six dimensions.

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

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

ACKNOWLEDGEMENTS

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

REFERENCES

- ¹J. D. Chodera, “A simple method for automated equilibration detection in molecular simulations,” *Journal of chemical theory and computation* **12**, 1799–1805 (2016).
- ²A. Grossfield, P. N. Patrone, D. R. Roe, A. J. Schultz, D. W. Siderius, and D. M. Zuckerman, “Best practices for quantification of uncertainty and sampling quality in molecular simulations,” *Living journal of computational molecular science* **1** (2018).
- ³W. C. Swope, J. W. Pitera, and F. Suits, “Describing protein folding kinetics by molecular dynamics simulations. 1. theory,” *The Journal of Physical Chemistry B* **108**, 6571–6581 (2004).
- ⁴N.-V. Buchete and G. Hummer, “Coarse master equations for peptide folding dynamics,” *The Journal of Physical Chemistry B* **112**, 6057–6069 (2008).
- ⁵J.-H. Prinz, H. Wu, M. Sarich, B. Keller, M. Senne, M. Held, J. D. Chodera, C. Schütte, and F. Noé, “Markov models of molecular kinetics: Generation and validation,” *The Journal of chemical physics* **134** (2011).
- ⁶B. E. Husic and V. S. Pande, “Markov state models: From an art to a science,” *Journal of the American Chemical Society* **140**, 2386–2396 (2018).
- ⁷Y. Ge and V. A. Voelz, “Markov state models to elucidate ligand binding mechanism,” *Protein-Ligand Interactions and Drug Design* , 239–259 (2021).
- ⁸N. Plattner and F. Noé, “Protein conformational plasticity and complex ligand-binding kinetics explored by atomistic simulations and markov models,” *Nature communications* **6**, 7653 (2015).
- ⁹S. Olsson, “Markov state models of protein–protein encounters,” *Protein Interactions: The Molecular Basis of Interactomics* , 163–193 (2022).
- ¹⁰N. Kozlowski and H. Grubmüller, “Uncertainties in markov state models of small proteins,” *Journal of Chemical Theory and Computation* **19**, 5516–5524 (2023).
- ¹¹Z. He, F. Paul, and B. Roux, “A critical perspective on markov state model treatments of protein–protein association using coarse-grained simulations,” *The Journal of Chemical Physics* **154** (2021).
- ¹²C. R. Schwantes and V. S. Pande, “Modeling molecular kinetics with tica and the kernel trick,” *Journal of chemical theory and computation* **11**, 600–608 (2015).
- ¹³D. Nagel, S. Sartore, and G. Stock, “Selecting features for markov modeling: a case study on hp35,” *Journal of Chemical Theory and Computation* **19**, 3391–3405 (2023).
- ¹⁴F. Nuske, B. G. Keller, G. Pérez-Hernández, A. S. Mey, and F. Noé, “Variational approach to molecular kinetics,” *Journal of chemical theory and computation* **10**, 1739–1752 (2014).
- ¹⁵C. Schütte, F. Noé, J. Lu, M. Sarich, and E. Vanden-Eijnden, “Markov state models based on milestoning,” *The Journal of chemical physics* **134** (2011).
- ¹⁶G. R. Bowman, D. L. Ensign, and V. S. Pande, “Enhanced modeling via network theory: Adaptive sampling of markov state models,” *Journal of chemical theory and computation* **6**, 787–794 (2010).

- ¹⁷B. G. Keller and P. G. Bolhuis, "Dynamical reweighting for biased rare event simulations," *Annual Review of Physical Chemistry* **75**, 137–162 (2024).
- ¹⁸H. C. Lie, K. Fackeldey, and M. Weber, "A square root approximation of transition rates for a markov state model," *SIAM Journal on Matrix Analysis and Applications* **34**, 738–756 (2013).
- ¹⁹L. Donati, M. Heida, B. G. Keller, and M. Weber, "Estimation of the infinitesimal generator by square-root approximation," *Journal of Physics: Condensed Matter* **30**, 425201 (2018).
- ²⁰M. Heida, M. Kantner, and A. Stephan, "Consistency and convergence for a family of finite volume discretizations of the Fokker–Planck operator," *ESAIM: Mathematical Modelling and Numerical Analysis* **55**, 3017–3042 (2021).
- ²¹L. Donati, M. Weber, and B. G. Keller, "Markov models from the square root approximation of the Fokker–Planck equation: Calculating the grid-dependent flux," *Journal of Physics: Condensed Matter* **33**, 115902 (2021).
- ²²L. Donati, M. Weber, and B. G. Keller, "A review of girsanov reweighting and of square root approximation for building molecular markov state models," *Journal of Mathematical Physics* **63** (2022).
- ²³R. R. Gabdoulline and R. C. Wade, "Biomolecular diffusional association," *Current opinion in structural biology* **12**, 204–213 (2002).
- ²⁴H. Zupan, F. Heinz, and B. G. Keller, "Grid-based state space exploration for molecular binding," *Canadian Journal of Chemistry* (2022).
- ²⁵J. J. Kuffner, "Effective sampling and distance metrics for 3D rigid body path planning," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004*, Vol. 4 (IEEE, 2004) pp. 3993–3998.
- ²⁶S. R. Lindemann, A. Yershova, and S. M. LaValle, "Incremental grid sampling strategies in robotics," in *Algorithmic Foundations of Robotics VI* (Springer, 2005) pp. 313–328.
- ²⁷J. Ichnowski and R. Alterovitz, "Fast nearest neighbor search in SE (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.
- ²⁸J. Diebel *et al.*, "Representing attitude: Euler angles, unit quaternions, and rotation vectors," *Matrix* **58**, 1–35 (2006).
- ²⁹C. F. Karney, "Quaternions in molecular modeling," *Journal of Molecular Graphics and Modelling* **25**, 595–604 (2007).
- ³⁰W. R. Hamilton, "Xxxi. on quaternions; or on a new system of imaginaries in algebra," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* **26**, 220–224 (1845).
- ³¹D. Kirk, *Graphics Gems III (IBM Version)* (Elsevier, 2012).
- ³²A. Yershova, S. Jain, S. M. LaValle, and J. C. Mitchell, "Generating uniform incremental grids on SO (3) using the hopf fibration," *The International journal of robotics research* **29**, 801–812 (2010).
- ³³A. Yershova and S. M. LaValle, "Deterministic sampling methods for spheres and SO (3)," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004*, Vol. 4 (IEEE, 2004) pp. 3974–3980.
- ³⁴D. Roșca, A. Morawiec, and M. De Graef, "A new method of constructing a grid in the space of 3D rotations and its applications to texture analysis," *Modelling and Simulation in Materials Science and Engineering* **22**, 075013 (2014).
- ³⁵F. Pietrucci, "Strategies for the exploration of free energy landscapes: Unity in diversity and challenges ahead," *Reviews in Physics* **2**, 32–45 (2017).
- ³⁶Wikipedia, "Voronoi diagram," (2024), https://en.wikipedia.org/wiki/Voronoi_diagram [Accessed: 10 Sep 2024].
- ³⁷I. Todhunter, *Spherical trigonometry, for the use of colleges and schools: with numerous examples* (Macmillan, 1863).
- ³⁸M. Caroli, P. M. M. de Castro, S. Loriot, O. Rouiller, M. Teillaud, and C. Wormser, *Robust and Efficient Delaunay triangulations of points on or close to a sphere*, Ph.D. thesis, INRIA (2009).
- ³⁹A. Van Oosterom and J. Strackee, "The solid angle of a plane triangle," *IEEE transactions on Biomedical Engineering* , 125–126 (1983).
- ⁴⁰D. Q. Huynh, "Metrics for 3D rotations: Comparison and analysis," *Journal of Mathematical Imaging and Vision* **35**, 155–164 (2009).
- ⁴¹B. Delaunay *et al.*, "Sur la sphère vide," *Izv. Akad. Nauk SSSR, Otdelenie Matematicheskii i Estestvennyka Nauk* **7**, 1–2 (1934).
- ⁴²R. M. Murray, Z. Li, and S. S. Sastry, *A mathematical introduction to robotic manipulation* (CRC press, 2017).
- ⁴³W. L. Jorgensen, J. Chandrasekhar, J. D. Madura, R. W. Impey, and M. L. Klein, "Comparison of simple potential functions for simulating liquid water," *The Journal of chemical physics* **79**, 926–935 (1983).
- ⁴⁴G. Van Rossum and F. L. Drake, *Python 3 Reference Manual* (CreateSpace, Scotts Valley, CA, 2009).
- ⁴⁵F. Mölder, K. P. Jablonski, B. Letcher, M. B. Hall, C. H. Tomkins-Tinch, V. Sochat, J. Forster, S. Lee, S. O. Twardziok, A. Kanitz, *et al.*, "Sustainable data analysis with snakemake," *F1000Research* **10** (2021).
- ⁴⁶C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, "Array programming with NumPy," *Nature* **585**, 357–362 (2020).

- ⁴⁷P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, “SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python,” *Nature Methods* **17**, 261–272 (2020).
- ⁴⁸A. A. Hagberg, D. A. Schult, and P. J. Swart, “Exploring network structure, dynamics, and function using networkx,” in *Proceedings of the 7th Python in Science Conference* (2008) pp. 11 – 15.
- ⁴⁹R. J. Gowers, M. Linke, J. Barnoud, T. J. Reddy, M. N. Melo, S. L. Seyler, J. Domanski, D. L. Dotson, S. Buchoux, I. M. Kenney, *et al.*, “MDAnalysis: a python package for the rapid analysis of molecular dynamics simulations,” in *Proceedings of the 15th python in science conference*, Vol. 98 (SciPy Austin, TX, 2016) p. 105.
- ⁵⁰N. Michaud-Agrawal, E. J. Denning, T. B. Woolf, and O. Beckstein, “MDAnalysis: a toolkit for the analysis of molecular dynamics simulations,” *Journal of computational chemistry* **32**, 2319–2327 (2011).
- ⁵¹M. Ahmed, R. Seraj, and S. M. S. Islam, “The k-means algorithm: A comprehensive survey and performance evaluation,” *Electronics* **9**, 1295 (2020).
- ⁵²F. Pedregosa, “Scikit-learn: Machine learning in python,” *Journal of machine learning research* **12**, 2825 (2011).
- ⁵³D. Van Der Spoel, E. Lindahl, B. Hess, G. Groenhof, A. E. Mark, and H. J. Berendsen, “GROMACS: fast, flexible, and free,” *Journal of computational chemistry* **26**, 1701–1718 (2005).
- ⁵⁴M. J. Abraham, T. Murtola, R. Schulz, S. Páll, J. C. Smith, B. Hess, and E. Lindahl, “GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers,” *SoftwareX* **1**, 19–25 (2015).
- ⁵⁵E. Lindahl, B. Hess, and D. Van Der Spoel, “GROMACS 3.0: a package for molecular simulation and trajectory analysis,” *Molecular modeling annual* **7**, 306–317 (2001).
- ⁵⁶H. J. Berendsen, D. van der Spoel, and R. van Drunen, “GROMACS: A message-passing parallel molecular dynamics implementation,” *Computer physics communications* **91**, 43–56 (1995).
- ⁵⁷B. Hess, H. Bekker, H. J. Berendsen, and J. G. Fraaije, “LINCS: a linear constraint solver for molecular simulations,” *Journal of computational chemistry* **18**, 1463–1472 (1997).
- ⁵⁸N. Goga, A. Rzepiela, A. De Vries, S. Marrink, and H. Berendsen, “Efficient algorithms for Langevin and DPD dynamics,” *Journal of chemical theory and computation* **8**, 3637–3649 (2012).
- ⁵⁹S. Kieninger and B. G. Keller, “GROMACS stochastic dynamics and BAOAB are equivalent configurational sampling algorithms,” *Journal of Chemical Theory and Computation* **18**, 5792–5798 (2022).
- ⁶⁰D. Beglov and B. Roux, “Finite representation of an infinite bulk system: Solvent boundary potential for computer simulations,” *The Journal of chemical physics* **100**, 9050–9063 (1994).
- ⁶¹M. Linke, J. Köfinger, and G. Hummer, “Fully anisotropic rotational diffusion tensor from molecular dynamics simulations,” *The Journal of Physical Chemistry B* **122**, 5630–5639 (2018).
- ⁶²M. Badaoui, A. Kells, C. Molteni, C. J. Dickson, V. Hornak, and E. Rosta, “Calculating kinetic rates and membrane permeability from biased simulations,” *The Journal of Physical Chemistry B* **122**, 11571–11578 (2018).

TABLE-OF-CONTENT GRAPHIC