The Dissertation Committee for Mauro Lorenzo Mugnai Certifies that this is the approved version of the following dissertation:

COMPUTATIONAL STATISTICAL MECHANICS OF PROTEIN FUNCTION

Committee:

Ron Elber, Supervisor

Adrian T. Keatinge-Clay

Dmitrii E. Makarov

Pengyu Ren

Peter J. Rossky

COMPUTATIONAL STATISTICAL MECHANICS OF PROTEIN FUNCTION

by

Mauro Lorenzo Mugnai, B.S.; M.S.

Dissertation

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

Doctor of Philosophy

The University of Texas at Austin

August 2014

Acknowledgements

First and foremost, I would like to thank my advisor Dr. Elber. In the past 6 years he has provided numerous ideas that inspired me in developing my own. He supported my interests, but at the same time encouraged me to work in many different fields, and be exposed to multiple different disciplines. The variety of the things that I learned would have been possible only by working with someone with his diverse and unique background of knowledge.

I would like to thank the members of my committee, Dr. Keatinge-Clay, Dr. Makarov, Dr. Ren, and Dr. Rossky. In their respective fields I have received advice that helped me overcome problems and view projects from a different perspective.

I thank my former advisor Prof. Ciccotti, who has continued to influence my development as a scientist.

I would also like to thank the current and former members of CLSB lab, including visiting scientist Grazia. Past group members have helped me get started with the software. Research discussions with all of them have been very stimulating. From each one, I learned something.

Ruth and Betsy helped me with the bureaucratic side of the development of this dissertation, and I am really grateful for their help.

Finally, I would like to thank my parents. Without their support I would have never been where I am now.

Computational Statistical Mechanics of Protein Function

Mauro Lorenzo Mugnai, PhD

The University of Texas at Austin, 2014

Supervisor: Ron Elber

Molecular dynamics (MD) provides an atomically detailed description of the dynamics of a system of atoms. It is a useful tool to understand how protein function arises from the dynamics of the atoms of the protein and of its environment. When the MD model is accurate, analyzing a MD trajectory unveils features of the proteins that are not available from a single snapshot or a static structure. When the sampling of the accessible configurations is accurate, we can employ statistical mechanics (SM) to connect the trajectory generated by MD to experimentally measurable kinetic and thermodynamic quantities that are related to function.

In this dissertation I describe three applications of MD and SM in the field of biochemistry. First, I discuss the theory of alchemical methods to compute free energy differences. In these methods a fragment of a system is computationally modified by removing its interactions with the environment and creating the interactions of the environment with the new species. This theory provides a numerical scheme to efficiently compute protein-ligand affinity, solvation free energies, and the effect of mutations on protein structure. I investigated the theory and stability of the numerical algorithm.

The second research topic that I discuss considers a model of the dynamics of a set of coarse variables. The dynamics in coarse space is modeled by the Smoluchowski equation. To employ this description it is necessary to have the correct potential of mean

force and diffusion tensor in the space of coarse variables. I describe a new method that I developed to extract the diffusion tensor from a MD simulation.

Finally, I employed MD simulations to explain at a microscopic level the stereospecificity of the enzyme ketoreductase. To do so, I ran multiple simulations of the enzyme bound with the correct ligand and its enantiomer in a reactive configuration. The simulations showed that the enzyme retained the correct stereoisomer closer to the reactive configuration, and highlighted which interactions are responsible for the specificity. These weak physical interactions enhance binding with the correct ligand even prior to the steps of chemical modification.

Table of Contents

Chapter 1: Introduction[1]

Proteins, nucleic acids, carbohydrates, lipids and all the other building blocks of life are made up of atoms. The description of the dynamics of these atoms provides the ultimate understanding of how biological macromolecules perform their function.[1] The dynamics of the atoms follow the laws of physics. In problems of biological relevance, the number of atoms that one needs to follow to describe the dynamics of macromolecules exceeds the capacity of the human brain alone, and demands computational aid.

Molecular dynamics (MD)[2] is a computational method whose aim is the description of the structure and dynamics of macromolecules at an atomic level. Frequently, a MD simulation begins with the experimentally known structure of a macromolecule and with an energy function that defines the interactions between the different components of the system. The classical equations of motion are then solved numerically to yield a trajectory, i.e. coordinates of the macromolecule as a function of time. MD is the only experimental or theoretical tool capable of producing such a comprehensive atomic-level description of the dynamics of a macromolecule.

Once the MD trajectory is known, we need to find a way to extract information that can be compared to experimental measurements. To do so, we need a theoretical tool that connects the dynamics of the atoms to macroscopic, measurable quantities. The theory aimed at this task is called statistical mechanics (SM).[3] Given the proper theoretical framework, the comparison with experiments provides a tool to check the

---

[1] Part of the material of this paragraph is taken from a work published in collaboration with my advisor, Prof. Elber, who supervised the work. Reproduced in part with permission from *J. Chem. Theory Comput.*, 2012, 8 (9), pp 3022–3033. Copyright 2012 American Chemical Society.
http://pubs.acs.org/doi/abs/10.1021/ct3003817

accuracy of the numerical calculation and, if the accuracy is sufficient, a valuable tool to highlight the microscopic reasons of the macroscopic behavior.

Despite its usefulness, a number of shortcomings affect the description provided by MD. First of all, MD is a method based on classical grounds, i.e. the dynamics is obtained as a numerical solution of classical equations of motion (Newton's law), and it exploits a classical energy function (among which is OPLS,[4] the one that was used in all my simulations). The classical nature of MD is necessarily inaccurate whenever there are chemical reactions, or electronic wave function re-arrangements. Many efforts have been made to introduce some quantum mechanics in MD, but I will not discuss them, as the work that I have done belongs entirely to the world of classical physical chemistry. Secondly, even if we believe the classical description of macromolecular dynamics, the limited capability of computers does not allow us to follow protein dynamics for time spans relevant in biology, and sometimes not even to gather enough statistics to get converged averages to compare with experiments. A number of methods have been introduced to enhance the sampling,[5] or to provide indirect, but computationally efficient, methods of calculating correct free energy differences.[6] I will discuss one of such techniques. Whenever the force field is accurate enough to give a reasonable description of the dynamics, just a few nanoseconds of simulation can unveil features of a macromolecule. This is true even if the simulations did not converge in the true sense of the ergodic hypothesis. These features are hard to capture from static pictures obtained with experimental methods (X-ray scattering and NMR mostly).[7]

The amount of information that MD provides is often too large to be intelligible. We need a way to coarse such a fine description into its essential features by minimizing the amount of relevant information lost. To do that, we need a theory that extracts these features and provide us with a dynamics in a smaller dimensionality.[8]

In the introduction of my dissertation, I describe a few of the many ways in which MD and SM are necessary in the field of biochemistry, paying particular attention to those fields in which, during the course of my studies, I have tried to contribute.

ALCHEMICAL METHODS

Protein-ligand non-covalent binding is a central problem in biochemistry. Enzyme function and signaling are among the processes that occur via the interaction of proteins and small molecules. Drug discovery is of course a related problem.

Binding is a dynamical process. The lock-and-key scheme, in which the ligand fits in a pre-existing cavity in the protein, needs to be integrated with a more dynamical picture.[9] The ligand induces modification of the structure of the protein by triggering the adjustment of the amino acids around the binding pocket (induced fit), or by selectively binding one of the many conformations that a protein explores in solution (conformation selection).[10] The adjustment of the structure of the binding pocket to accommodate the ligand is hard to predict. With a polar ligand, hydrogen bonds between the protein and the ligand may form in the binding pocket. This helps to compensate the de-solvation free energy, but comes at the price that the amino acids in the binding pocket lose conformational flexibility. This is the so-called enthalpy/entropy compensation.[11] On the other hand, when a hydrophobic ligand is bound, there is evidence of an opposite phenomenon: free energy reduction might arise from an increased backbone flexibility of the protein (which increases the entropy).[11-12] Answering these questions may facilitate the lead phase of drug discovery.[13]

MD holds the promise to provide this dynamical picture. For example, a study of acetylcholinesterase showed that the catalytic triad is formed exclusively in the presence of the ligand, an example of induced fit.[14] The dynamics of the protein might show

features of the binding site that are not visible from crystal structures. For instance, MD simulation showed a trench in the binding pocket of HIV Integrase that was not available from static structures. This trench was later used as a drug target.[7]

Even though the idea of "mimicking" a real experiment with a MD simulation is extremely appealing, biological phenomena, such as protein-ligand binding, happen on time scales that are not accessible to routine MD simulations. Only recently, and just using the fastest architecture available (the special-purpose machine Anton[15]), it was possible to observe the entire process of ligand binding.[16] Two different ligands were shown to bind to Src kinase with X-ray structure accuracy. In doing so, they wandered around the protein for multiple microseconds before finally finding the correct binding pocket. While searching for the binding site, the ligands were trapped at other sites on the protein. This holds the promise of becoming a powerful method to identify new allosteric sites of proteins. On the other hand, the length of the process makes it impossible to conduct routine MD simulations to see the binding event. Even more difficult is gathering sufficient statistics to draw conclusions on rates and free energy of the process that could be compared with experiments. If we want to do so, another approach has to be found.

An intriguing feature of MD for the computations of free energy differences is that we can modify the nature of the components of the system. This is the idea behind alchemical methods in MD.[17] Instead of simulating multiple binding and unbinding events, it is more convenient to make the ligand disappear from water and appear in the binding pocket. To study the free energy difference between two ligands in a protein, we can directly mutate one into the other.

Even though these calculations do not follow the experimental thermodynamic pathways, we can extract from them the correct free energy difference by using alternative routes. Indeed we can present alchemical and physical processes together in a

complete thermodynamic cycle.[6] Exploiting the fact that the free energy is a state function, the physical free energy difference is then obtained as a linear combination of the difference of the alchemical free energies. The computation of alchemical transformations is less expensive than the straightforward simulation of the experimental processes.

In alchemical processes, a molecule (or a fragment of a molecule) is mutated into another one, or decoupled from the environment, by modifying the interactions of the atoms of the mutated species. This modification can be performed according to two protocols: single or dual topology.[18] In the single topology protocol the geometry of the molecule is progressively changed into the new moiety. In a dual topology protocol the native and mutant molecules coexists. The interactions of the native moiety are progressively annihilated, while the interactions of the mutant moiety are turned on. In my work I mainly used a dual topology approach, so I will only discuss that. In a dual topology approach the alchemical modification is performed by multiplying the interactions by a switching parameter (or order parameter). Along the transformation, the switching parameter changes from 0 to 1 for the new fragment, and from 1 to 0 for the old fragment that needs to be substituted.

Alchemical methods were proven to be useful in many fields of biochemistry, studying the effect of point mutations on protein stability,[19] of protein-ligand binding,[20] to compute solvation free energies,[21] and particularly in the context of drug discovery.[13, 22]

To improve the efficiency and accuracy of the alchemical calculations, many of methodological studies were performed. During my PhD, I wrote a code for computing alchemical substitution that is part of the MOIL[23] and MOIL-OPT[24] MD packages. While writing these codes I faced a number of questions. Often the answer was found in the

literature, sometimes I tried to give my contribution addressing problems whose solution I could not find.

The main questions that I faced were the following:

1. What protocol should I use to change the switching parameter?
2. What is the most accurate and efficient alchemical pathway?

Regarding the first question, there are multiple methods to carry out an alchemical MD simulation. The alchemical transformation can be performed modifying the switching parameter infinitely slowly, so performing equilibrium simulations. The results of the simulations performed with different values of the switching parameter can be combined using different techniques (Thermodynamic Integration,[25] Free Energy Perturbation,[26] and Bennett's Acceptance Ratio[27]). It is possible to drive the system from the native to the mutant state using a predefined time-dependent protocol for switching between thermodynamic states. This requires the Jarzinski equality[28] to get free energy differences from non-equilibrium trajectories. The switching parameter can be modified according to a dynamical scheme.[29] A potential can be defined for the switching parameter, and according to that potential the alchemical state is dynamically changed.

During my PhD I did not explore non-equilibrium calculations, or dynamics of the switching parameter. The code that I wrote instead is suited to perform equilibrium simulations.[30] I tried to explore different ways of combining the results, and I found out that, in agreement with what suggested in the literature,[31] Bennett Acceptance Ratio is the most efficient among these methods.

Regarding the choice of the alchemical pathway, the first question to address is what is the minimal number of interactions that we need to create/annihilate to get an accurate result? Particularly, should we remove the interactions within the mutated molecule (self-interactions) to get correct results? Can we keep some of the interactions

between the alchemically annihilated/created fragments and the environment (external interactions)? The idea is that, if some interactions are such that their contribution to the free energy difference cancels out within the thermodynamic cycle, we do not need to create/annihilate them, but we can keep them as they are. Keeping interactions simplifies the alchemical pathway.

In the multitude of applications of alchemical methods the idea of retaining the self-interactions of the substituted entity has been used inconsistently. Whenever an entire molecule is decoupled from the environment all the self-interactions are retained. This is the common practice to compute the standard free energy of binding of a ligand to an enzyme [32] or to compute the solvation free energy of a solute. [33] A different approach is followed when only a molecular fragment of a larger scaffold is modified to create a new, mutant molecule starting from the original, native chemical species. Such an approach was used to compute, for instance, relative binding free energy between two ligands,[20a, 34] or the change in stability of a protein conformation upon a mutation of a residue.[19b-d] In all these relative free energy calculations the retention of the self-interactions of the fragments has been vigorously debated. Different authors reported significant[19b, 35] or negligible[19c] contribution from the self-interactions of fragments. Keeping the self-interactions is equivalent to a change in the "end-state" from atomic to molecular ideal gas.[36] It was suggested that retaining the bonded self-interactions does not affect relative free energy differences [36] and a proof of this assertion followed.[37] A general proof that did not distinguish between bonded and non-bonded interactions was provided as well.[38] In the dissertation, I revisit the problem, and I provide a simple and general proof that all the self-interactions can be retained without affecting the relative free energy difference in the context of a thermodynamic cycle.

The annihilation/creation of the external non-bonded interactions is required to compute the proper free energy difference. However, it is not necessary to annihilate/create all the external bonded interactions connecting a fragment to a scaffold. Since in the initial (final) state the mutant (native) fragment is decoupled from the rest of the environment, it is possible to fix its external degrees of freedom to the scaffold without affecting the internal degrees of freedom of the scaffold or of the fragment. In this way, at most six bonded interactions constraining the six external degrees of freedom are retained throughout the simulation. All the other external bonded interactions are annihilated/created in the MD simulation. This concept was introduced as the Virtual Bond Algorithm (VBA), [39] which was proposed in the context of protein-ligand binding, and is exact. Six bonded interactions between the protein and the ligand are added to restrain the overall motions of the ligand and then the non-bonded external interactions are removed during the alchemical substitution. The same VBA interactions may be retained in the process of substituting a fragment bonded to a scaffold.

Once the list of interactions to create/annihilate is provided, there is a second question to address: what is the best way to perform this annihilation/creation of interactions? As long as the initial and final states are correct, the protocol does not affect the correctness of the result. On the other hand, the stability of the numerical method can change significantly. The straightforward approach is to multiply the interactions to remove (create) by a linear function of the switching parameter that is equal to one (zero) at the beginning of the alchemical substitution, and zero (one) at the end. We can also use a quadratic protocol, or any other power, or even a different function. For instance, when the free energy is computed using Themodynamic Integration, it was noted that the numerical accuracy of the integration was enhanced if the van der Waals interactions

were created/annihilated using a switching parameter to the fourth power or higher.[40] It is also possible that the functional form of some of the terms in the potential energy function introduces issues when annihilated/created. This can be fixed by modifying the functional form during the alchemical substitution. For instance, it was observed that the simulation is more stable if the van der Waals interactions were created/removed as soft-core interactions.[40] Again, as long as they keep the correct functional form in the end states, no further adjustment is needed to obtain the exact free energy difference. It is also known that to enhance the stability of the simulation the electrostatic interactions should be removed before and created after the van der Waals interactions are established.[41] This order of creation/annihilation avoids overlaps of particles with the same charges, and consequent numerical instability.

While testing the code, I observed that the creation/annihilation of other terms in the energy function created numerical instabilities. When alchemically removed, configurations can be found with bond-angles and torsions leading to singular energy and discontinuous forces. To avoid this, I proposed to modify the angular functional form from a bond-angle in polar coordinates, to an extra bond in Cartesian space (the so-called Urey-Bradley potential).[30] The creation/removal of these bonds was numerically more convenient when performed using a quadratic switching parameter. I also proposed to remove (create) torsions before (after) the angular potential, as this avoided numerical instability associated with the torsional functional form.[30]

The details of the study on alchemical substitutions are in Chapter 2.

EXTRACTING DIFFUSION TENSOR FROM MD SIMULATIONS

Many experimental techniques follow the time evolution of few microscopic quantities to investigate kinetic and thermodynamic properties of biopolymers.

Techniques such as FRET[42] or force spectroscopy[43] can be used to monitor in time the end-to-end distance of proteins or nucleic acids. The information extracted from these techniques can be used to study the free energy and kinetics of folding,[42-44] and the dynamics of the protein in the unfolded state.[45] Let's refer to quantities such as the end-to-end distance as coarse variables. So, these experiments follow the time evolution of one coarse variable. In theoretical studies, together with this coarse variable of interest, we can consider a few more. More detailed (but still reduced) description can help understanding what are the interactions affecting the dynamics of the measured coarse variable. For instance, the dynamics of an unfolded polymer is affected by two types of general interactions that are sometimes modeled by effective frictional forces: one force is a result of interaction with the solvent, and another force is called internal.[45a, 46] Experimentally this is studied by following the end-to-end distance of the polymer at different concentrations of a viscogen (such as glycerol) to modify the solvent viscosity. In a simulation study, we follow many more degrees of freedom using atomically detailed simulations. However it is useful to reduce the comprehensive representation to include only a few coarse variables to have a better understanding of the problem and compare to the experimental measurement. Therefore, other than the end-to-end distance, some other coarse variable describing interactions within the polymer and between the polymer and the solvent can be modeled theoretically.

The starting point of such a theoretical study is the development of a model for the dynamics in the space of coarse variables. Ideally, such model will employ an exact projection of the entire dynamics onto a few degrees of freedom.[8] Exact projection theories are the Generalized Langevin Equation (GLE),[8a] which describes the dynamics of the coarse variables, and the Generalized Master Equation (GME), [47] which describes the evolution of probability density in coarse space. In the GLE, the variables of interest

evolve according to a potential of mean force (free energy). The other degrees of freedom behave as a "bath" able to exchange energy with the system in a balanced way (i.e. following the fluctuation-dissipation theorem[48]). Their contribution appears in a random noise term and in the friction kernel, which is proportional to the autocorrelation function of this random noise. To describe the dynamics of the system with a Langevin equation we need to know the friction kernel. Extracting the exact friction kernel is a hard task, because it involves the solution of a set of equations of motion that are evolved according to the so-called orthogonal dynamics. There are a few possible ways to extract the friction kernel. One way is to assume that the orthogonal dynamics is fast compared to the relevant degrees of freedom. In this case the friction kernel can be obtained from the integral of the force-force autocorrelation function.[46] Furthermore, if we assume that the friction kernel decays to zero on a time scale that is much shorter than the time scale for the dynamics in the relevant coordinates, a static friction can be obtained as the time integral of the force-force autocorrelation function.[49] So it becomes a time-independent, space-dependent friction tensor. This is the so-called Markovian approximation.[46] Another possibility is to make use of the velocity autocorrelation function (VACF). The time derivative of the VACF is equal to the time convolution of the VACF with the friction kernel. This equation can be solved in Laplace space, by assuming a simple functional form (delta function and exponential decay, which is physically sound) for the friction kernel.[48] Again, it is possible to get a static friction from the Markovian approximation as before.[50] Recently another interesting solution was found for the case of the linear generalized Langevin equation.[51] A new equation was proposed describing the exact projected dynamics. This equation can be solved once the entire MD trajectory is at hand. Once the projected dynamics is known, it is also possible to derive the friction kernel.[52]

Here we consider a different route based on the GME. In the GME the time evolution of the probability density in coarse space depends on the transition probabilities per unit time (rates) between positions in coarse space. The transition probability per unit time is a complicated function of the orthogonal dynamics.[47] One possible way to obtain from MD the information needed to formulate the GME is to use Milestoning.[53] Milestoning is a method to map atomically detailed dynamics to a set of coarse variables of interest. The coarse variable space is partitioned into cells, whose separators are defined milestones. MD trajectories are computed between different milestones. The probability of going from one milestone to another is estimated, and the time in which this transition happens is recorded. Milestoning can also be considered an analysis tool of one long simulation, which can be divided into individual transitions between milestones. The Milestoning equations provide a non-Markovian description of the dynamics in coarse space, which was shown to be equivalent to the GME.[53a] While it is already known how to extract important thermodynamic information (the free energy at the milestones[53b]) and kinetic information (mean first passage time[53b]), a method to extract from Milestoning the friction kernel was not available. The friction kernel does not appear explicitly in the GME, so the connection between the transition rates per unit time and the friction kernel is not straightforward. During my PhD, I developed a method to extract the diffusion tensor, which is related to the friction tensor, for the case in which the dynamics in coarse space is Markovian. In this case the GME becomes the Master Equation (ME). To derive the diffusion tensor, I exploited the so-called Kramers-Moyal (KM) expansion.[54] KM expansion provides a method to derive the Fokker-Planck (FP) equation[54a] from the ME. The FP equation is a differential equation that described the time evolution of the probability density in coarse space as a function of a potential of mean force and a coordinate-dependent diffusion tensor. The Einstein relationship states

the connection between diffusion and friction tensor,[55] so computing one is equivalent to computing the other. To derive the FP equation from the ME, KM exploits an expansion of the transition probability per unit time (i.e. the rates) in the ME. The truncation of this expansion at the second order yields the FP equation. This truncation is exact in the case in which the underlying dynamics in coarse space is overdamped.[56] In this case, the diffusion tensor is exactly described by a function of the rates in the ME. In the Milestoning language, with the rates provided by the Milestoning analysis[53b] it is possible to compute the space-dependent diffusion tensor on each milestone.

Other methods are available to determine the space-dependent diffusion tensor from MD simulations. One possibility[57] is to compute the zero time limit of the average of the position autocorrelation function, which gives the diffusion in the overdamped limit: $D(q^*) = \lim_{t\to 0} \left\langle [q(t) - q(0)]^2 \mid q(0) = q^* \right\rangle / t$. If the dynamics is overdamped, the space dependent diffusion coefficient in q^* can be obtained computing the first passage time of going from q^* to q' and backwards.[58] If instead one assumes a Langevin equation for the variable of interest, it is possible to obtain the local diffusion coefficient from the integral of the velocity autocorrelation function,[59] or from the integral of the position autocorrelation function.[60] Hummer[60] proposed a method that shares some similarities with the one that I worked on. He started from the ME as well, and derived the rates of transiting between different states in the system with a Bayesian approach. This means that the probability of the trajectory given the rates was computed using MD simulations, and the Bayes' rule was used to determine the rates from the trajectory. This was done with a uniform prior. In this case the approach corresponds to maximizing the likelihood function. The optimization can be done with different priors to enhance the smoothness of the estimate of the space-dependent diffusion. A similar Bayesian approach with uniform prior was used to derive a formula for the rates in a Markovian

formulation of Milestoning that is analogous to the one that I use.[61] Hummer derives the rates using the assumption of detailed balance.[62] This approach does not necessarily hold in Milestoning, where the system can be simulated in steady state flux conditions, or when non-equilibrium methods are used.[63] The formula that we used is not affected by this assumption and it is therefore more general.

Others have used Bayesian inference to determine the diffusion tensor. The likelihood used was not the probability of the trajectory given the rates, but the probability of the displacement in the space of coarse variables assuming an overdamped dynamics.[64] Others have adapted the method to the case of simulations performed using the adaptive biasing force technique (ABF), which enhances the sampling by allowing to cross high free energy barriers,[17] so to have a better sampling in coarse space and improve the estimate of the diffusion tensor.[63] This method was recently used in biological applications. It was used to study the permeability of a lipid bilayer,[65] which is a function of the diffusivity across the membrane. Another study considered a small solute permeation through a lipid bilayer using ABF.[66] In this case a two-dimensional reaction coordinate was used: the position and the orientation of the permeant were considered simultaneously. Interestingly, a similar application in Milestoning was already performed, where the Milestoning equations were used to describe the permeation of a small peptide molecule (NATA, N-acetyl-L-tryptophanamide) across a biological membrane.[67] Using Milestoning it is possible to follow events on time scales (millisecond or more) that are not accessible by straightforward MD simulations. This is done by combining multiple independent short MD trajectories that can run simultaneously. Since the method that I developed to extract the diffusion tensor is based on Milestoning, it fully benefits from all the numerical advantages of the Milestoning method. Finally, Milestoning in principle is equivalent to GME, not ME. Working in the framework of

Milestoning holds the promise to be able to extend our method to a non-Markovian case, if an equivalent of the KM expansion for the GME will be derived.

The details of the study on the method to extract the diffusion equation with Milestoning are described in Chapter 3.

ENZYME SPECIFICITY

Understanding how enzymes catalyze reactions is a fundamental problem in biochemistry. A typical reaction is described by the following series of steps (see scheme 1 in [68]):

1. The enzyme weakly binds the substrate in the open conformation;
2. The enzyme undergoes a conformational transition from the open to the closed state (induced fit), tightening the binding;
3. The chemical reaction is carried out;
4. The product is released to solution.

The cellular milieu contains a large number of potential substrates for the enzyme. It is therefore important that the enzyme is specific to the correct substrate. The specificity of an enzyme can be measured as the second order rate constant for substrate binding, multiplied by the probability of forming the product once the complex is formed (this is a definition of k_{cat}/K_M).[68]

One could think that enzymes show their specificity in the binding step, by selecting the ligand that properly fits the binding pocket, or in the chemistry step, by the alignment of the reactive amino acids. But there are also examples in which the conformational transition selects the proper ligand.[68] Overall, it is reasonable to believe that any of the steps discussed before can carry a contribution to the specificity.[68]

The microscopic reasons for specificity may go beyond the static fitting of the ligand in the binding pocket. Therefore, MD holds the promise of a more comprehensive understanding of the atomically detailed mechanism of specificity. The main limitation of MD in this case is its classical nature. It is difficult with classical simulations to describe the chemical step, since specific functional form for the Born-Oppenheimer energy surface must be developed. In typical simulations of biological molecules the potential used in MD simulations retains specific bonded topology. Within the limitations of these approximations, MD can shed light on the microscopic reasons for specificity. To do so, a possible approach is to run simulations of the enzyme with the correct and incorrect substrate, and to look at the differences in structure of the binding site, and at the free energy of the two complexes. Using an approach of this type, it was argued that the specificity of lactate dehydrogenase for the pro-R compared to the pro-S hydrogen of the nicotinamide ring of NADH was due to favorable electrostatic interactions of the carboxamide group of the NADH ring when the pro-R hydrogen faces the pyruvate.[69] For a DNA polymerase, a successful comparison of kinetic data from MD simulations and experiments for the complex with the correct and an incorrect (mismatch) nucleotide lead to microscopic insights on role of induced fit in selecting the correct substrate.[70] Of course, alchemical methods can be of used as well, both to compute the free energy difference of binding between two ligands, or between two protein sequences, if a less-specific mutant is known, as in the case of cytochrome P450.[71]

In my PhD, I studied the case of the enzyme ketoreductase (KR). This enzyme is part of polyketide synthase (PK). PKs are large multi-domain, multi-functional enzymes that generate polyketides, complex secondary metabolite. These molecules are involved in multiple functions in the cell, among which anti-bacterial functions.[72] The enzyme is composed of multiple subunits, some of which are involved in the extension of the

growing polyketide, others instead process the intermediate by modifying its chemical structure.[72] The polyketide is bound to a large prosthetic group (18Å) of a domain of PK called acyl carrier protein (ACP). ACP carries the intermediate between the different components of the enzyme. The structure of the PK at different stages of the growth/processing of an intermediate was recently resolved by cryo-EM,[73] highlighting the large displacements undergone by ACP.

KR is a processing domain of PK. KR binds to an intermediate and reduces the β-keto oxygen in a reaction that involves the oxidation of the coenzyme NADPH. Different KRs yield hydroxyl groups with different spatial arrangements; the A-type generates L hydroxyl groups, the B-type D hydroxyl groups. The work presented here is on an A-type. A-type KRs are divided in two groups: A1 enzymes, which stereoselectively reduce polyketide intermediates with D-α substituent, and A2 enzymes, which reduce only the stereoisomer (L orientation of the α substituent). The question that we want to address is: what are the microscopic reasons for the stereospecificity of the A1 enzyme? A number of experimental studies on mutants of the A1 enzyme give interesting hints. It was found that the stereospecificity is lost upon mutation of glutamine to histidine at position -3 from the reactive tyrosine (this glutamine is numbered 368 in the simulations).[74] It was also found that a second mutation from glycine to threonine at position -12 from the reactive tyrosine switches completely the stereospecificity.[75] Upon this mutation, a tryptophan that is considered responsible for the proper alignment of the ligand in the binding pocket was found significantly displaced.[75]

In the simulations that I carried out, I studied an A1 enzyme (molecule B from PDB structure 3MJS). To form the ternary complex (A1, NADPH, and substrate) the substrate 2-methyl-3-oxopentanoate-S-N acetyl cysteamine was added to the binary complex (A1 with NADPH) taken from the crystal structure. From now on the substrate

will be addressed as diketide (dk). The ligand has a chiral center at the α-carbon. The α-methyl substituent can be found in two optically distinguishable orientations, defining two enantiomers of the ligand, dkD and dkL. The A1 enzyme is stereospecific for the D enantiomer.

The MD simulations of the complexes A1-dkD and A1-dkL have to overcome a number of challenges. First of all, the force field for the dk and NADPH ligands was not available, so it was generated combining force fields of similar chemical species and filling the gaps with quantum mechanical simulations. The force field was not tested against experimental measurements, therefore the simulations of the A1-dk complexes are a test of the accuracy of the force field. Secondly, the A1 enzyme is very large (475 amino acids), which makes the simulations slow, and so it is computationally challenging to relax the slow modes of the enzyme. Third, as mentioned before, only the binary and not the ternary structures were available. Finally, and maybe more importantly, the ligand is extremely flexible, and the binding cleft from the crystal structure is not tight enough to allow for a single binding configuration. For these reasons it was not possible to perform a quantitative study. Nevertheless, from the set of simulations that were performed it is clear that the correct ligand (dkD) is retained closer to the reactive configuration compared to the incorrect ligand (dkL). Also, it was found that the glutamine 368 has an important role in the specificity. The simulations showed that Q368 tends to obstacle the reactive alignment of the ligand by offering a hydrogen bond to the ligand that competes with the reactive one. This competition is much more significant in the incorrect complex, while in the correct complex the D-α-methyl substituent seems to displace Q368 away from the reaction site. Interestingly, the experiment mentioned before showed that a mutation of this glutamine to histidine reduces dramatically the stereospecificity of the enzyme.[75] The details of this study are in Chapter. 4.

Chapter 2: Free Energy Calculation: Thermodynamic Cycles

INTRODUCTION

In this chapter[2] I discuss a code that I developed to perform alchemical free energy calculations.[17] The code is now part of the MOIL[23] and MOIL-OPT[24] software packages.

There is an extensive literature on methods for alchemical calculation.[17, 41] The theoretical foundation of the method was developed long ago.[25-26] The numerical implementation became popular in more recent years, particularly within the context of Thermodynamic Cycles.[6] Even in their early more straightforward numerical implementations, alchemical simulations shed light on problems such as the effect of point mutations on protein stability,[19] protein-ligand binding,[20] and make it possible to compute solvation free energies.[21] Alchemical free energy calculations have then become popular, particularly in the context of drug discovery.[13, 22] Many methodological studies have also been carried out. These studies highlighted the pitfalls of a straightforward implementation of the theoretical procedures,[36-40] making the alchemical methods more accurate.

In what follows, I describe the theory of alchemical free energy calculations, I use some of the major results that improve the accuracy of the numerical implementation, and I describe what has been my contribution to the field. Most of the results discussed in this chapter have been published.[30] Some of the most recent (and unpublished) developments are discussed as well.

---

[2] Part of the material of this chapter is taken from a work published in collaboration with my advisor, Prof.Elber, who supervised the work. Reproduced in part with permission from *J. Chem. Theory Comput.*, 2012, 8 (9), pp 3022–3033. Copyright 2012 American Chemical Society. http://pubs.acs.org/doi/abs/10.1021/ct3003817

THEORY

Free Energy Differences

We consider the free energy difference between two physical systems, A and B. These two systems share a certain number of particles, which we call P, as in "protein". Some particles exist exclusively in the system A, and we call them N, as in "native". Other particles exist exclusively in the system B, and we will refer to them as M, as in "mutant". The systems A and B have two different Hamiltonians:

$$\begin{aligned} H_A(P,N) &= K(P,N)+U(P)+U(N)+U(P,N) \\ H_B(P,M) &= K(P,M)+U(P)+U(M)+U(P,M) \end{aligned} \tag{2.1}$$

Here K is the kinetic energy and U the potential energy. The term $U(X)$ includes interactions within X (self-interactions), and $U(X,Y)$ only interactions between X and Y (external interactions). The phase space volume element is $d\Gamma$ (e.g. $d\Gamma_{PN} = d\Pi_P dQ_P d\Pi_N dQ_N$ were Π_X denotes momentum, and Q_X coordinate vectors of atoms of species X). The volume element of coordinate space is $d\Gamma'$ (e.g. $d\Gamma'_{PN} = dQ_P dQ_N$). Finally, we denote the number of X particles as N_X, where X can be any combination of P, M and N. The desired free energy difference $\Delta F_{A,B}$ is given by

$$\begin{aligned} &\Delta F_{A,B} = F_B - F_A = \\ &= -\beta^{-1}\ln\left(\frac{\frac{1}{C_{PM}h^{3N_{PM}}}\int d\Gamma_{PM}\exp\left\{-\beta\left[K(P,M)+U(P)+U(P,M)+U(M)\right]\right\}}{\frac{1}{C_{PN}h^{3N_{PN}}}\int d\Gamma_{PN}\exp\left\{-\beta\left[K(P,N)+U(P)+U(P,N)+U(N)\right]\right\}} \right) \end{aligned} \tag{2.2}$$

The coefficients C_{PM} ad C_{PN} are corrections for permutation of identical particles,[3] and h is the Planck constant. Note that the number of particles of the two states can be different. The free energies are functions of the temperature, the volume, and the number of particles (e.g. N_{PN} or N_{PM}). We can then integrate the momenta and obtain:

$$\Delta F_{A,B} = F_B - F_A =$$

$$= -\beta^{-1} \ln \left[\frac{C_{PN}}{C_{PM}} \left(\frac{\prod_{i \in M} \frac{2\pi m_i}{h^2 \beta}}{\prod_{i \in N} \frac{2\pi m_i}{h^2 \beta}} \right)^{\frac{3}{2}} \frac{\int d\Gamma'_{PM} \exp\left\{ -\beta \left[U(P) + U(P,M) + U(M) \right] \right\}}{\int d\Gamma'_{PN} \exp\left\{ -\beta \left[U(P) + U(P,N) + U(N) \right] \right\}} \right] \quad (2.3)$$

This equation represents the free energy difference between the two physical systems A and B. A result very close to this one can be obtained using alchemical methods.

In an alchemical process a molecule, or a small part of it, is "mutated" into another by modifying its external and self-interactions. Such a "mutation" is usually performed according to two protocols:[18] a "single topology" protocol, in which the geometry of the native molecule/moiety is progressively changed into the geometry of the mutant, and a "dual topology", in which the two molecules/moieties coexist. Along the substitution pathway the interactions of the native part are gradually annihilated while those of the mutant are growing to their full strength.

We follow a dual topology approach. In this case, the two Hamiltonians presented in Eq.(2.1) are mixed into a single Hamiltonian, which is function of the coordinates of all the species P, M, N, and a switching parameter (or order parameter) λ :

$$\begin{aligned} H(P,M,N;\lambda) &= K(P,M,N) + U(P) + \\ &+ (1-\lambda)\left[U(P,N) + U(N) \right] + \lambda \left[U(P,M) + U(M) \right] \end{aligned} \quad (2.4)$$

The order parameter λ varies between 0 and 1. When the order parameter is equal to 0, the Hamiltonian in Eq.(2.4) describes a system equivalent to system A (P and N particles) plus N_M ideal gas particles. When the order parameter is equal to 1, the system described by Eq.(2.4) is system B (P and M particles) plus N_N ideal gas particles. When the order parameter is $0 < \lambda < 1$ the system is a mix of the initial and final state, whose free energy is:

$$f(\lambda) = -\beta^{-1} \ln \int d\Gamma_{P,N,M} \exp\left\{-\beta H(P,M,N;\lambda)\right\} = -\beta^{-1} \ln Q(\lambda) \tag{2.5}$$

The free energy difference computed upon switching the order parameter from 0 to 1 is:

$$f(\lambda=1) - f(\lambda=0) = \Delta f =$$
$$= -\beta^{-1} \ln \frac{\int d\Gamma_{P,N,M} \exp\left\{-\beta\left[K(P,M,N) + U(P) + U(P,M) + U(M)\right]\right\}}{\int d\Gamma_{P,N,M} \exp\left\{-\beta\left[K(P,M,N) + U(P) + U(P,N) + U(N)\right]\right\}} \tag{2.6}$$

The integration over the momenta yields the same free energy contribution in the initial and final states, and cancels out. In the initial (final) state the M (N) particles do not interact with any other particle, so each of their configuration integrals is the volume V:

$$\Delta f = -\beta^{-1} \ln \frac{V^{N_N}}{V^{N_M}} - \beta^{-1} \ln \frac{\int d\Gamma'_{P,M} \exp\left[-\beta\left(U(P) + U(P,M) + U(M)\right)\right]}{\int d\Gamma'_{P,N} \exp\left[-\beta\left(U(P) + U(P,N) + U(N)\right)\right]} \tag{2.7}$$

The alchemical Hamiltonian preserves the number of particles of the reactant and the product. The difference between the alchemical free energy difference and the free energy of mutations (in which the number of particles changes) is summarized in the formula below

$$\Delta F_{A,B} = \Delta f - \beta^{-1} \ln \left[\frac{C_{PN}}{C_{PM}} \left(\frac{\prod_{i \in M} \frac{2\pi m_i}{h^2 \beta} V^{\frac{2}{3}}}{\prod_{i \in N} \frac{2\pi m_i}{h^2 \beta} V^{\frac{2}{3}}} \right)^{\frac{3}{2}} \right] = \Delta f - \Delta F_{corr} \tag{2.8}$$

Computing a single free energy difference between the native and the mutant using alchemical pathway requires the calculation of ΔF_{corr}.

Thermodynamic Cycles

Alchemical methods are often used in thermodynamic cycles (TC)[6] to compare an experimental measurement to an equivalent quantity that can be computed but is not accessible to direct experimental measurements. As an example, we refer to Figure 1.

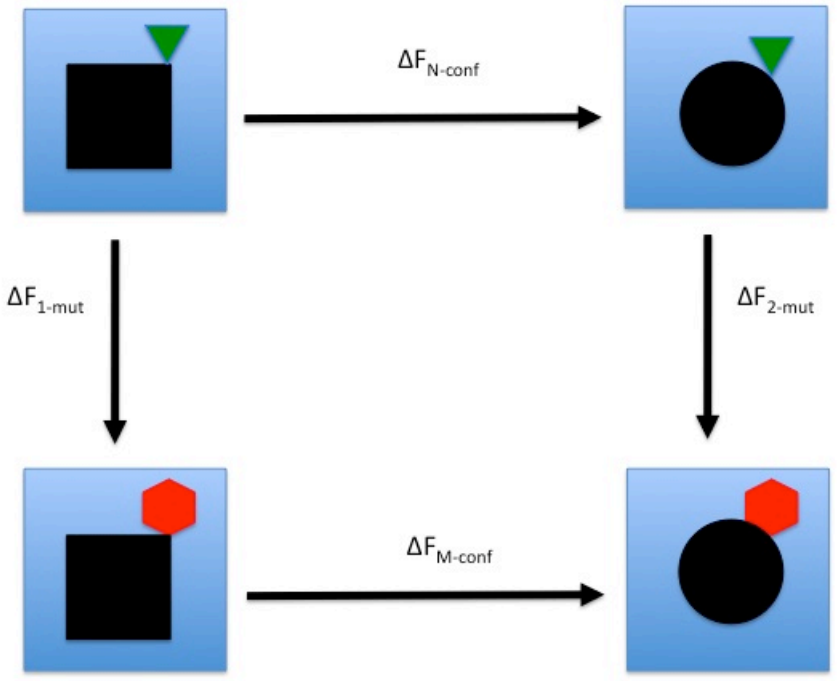


Figure 1: An example of a thermodynamic cycle for a conformational transition in a protein and a protein mutant. The black shapes represent the protein in two conformations (the square and the circle). The green triangle and the red hexagon represent native and mutant residues, respectively. The blue background represents solution. The horizontal arrows indicate conformation transitions. The vertical arrows indicate mutations. This cycle compares experimentally measured free energy differences (those that are related to the conformation transitions $\Delta F_{N\text{-}conf}$ and $\Delta F_{M\text{-}conf}$) to computed free energy differences (the mutations, $\Delta F_{1\text{-}mut}$ and $\Delta F_{2\text{-}mut}$).

The two different black shapes correspond to two different conformations of the protein, P1 and P2. The green triangle is a native residue N, and the red hexagon represents a mutant residue M. The free energy differences associated with transitions between the four states of the system are of conformation transitions (horizontal arrows), and of mutations of one residue (vertical arrows).

We are interested in the relative stability of the two conformational states (P1 and P2) for the two mutants (PN and PM), i.e.:

$$\Delta\Delta F_{\text{exp}} = \Delta F_{\text{N-conf}} - \Delta F_{\text{M-conf}} \tag{2.9}$$

which is measured experimentally. The computation of the transition between the two conformations may be long and the convergence of free energy difference by such simulation may be difficult to obtain and/or expensive. For a complete TC the free energy change is zero, i.e. $\Delta F_{\text{N-conf}} + \Delta F_{\text{2-mut}} - \Delta F_{\text{M-conf}} - \Delta F_{\text{1-mut}} = 0$, which we exploit to write

$$\Delta F_{\text{exp}} \equiv \Delta F_{\text{N-conf}} - \Delta F_{\text{M-conf}} = \Delta F_{\text{1-mut}} - \Delta F_{\text{2-mut}} \equiv \Delta\Delta F_{\text{mut}} \tag{2.10}$$

We can therefore compute $\Delta\Delta F_{\text{mut}}$ instead of $\Delta\Delta F_{\text{exp}}$, which is less costly, since it does not require waiting for multiple conformational transitions, which might happen on time scales not accessible to MD.

The free energy differences of mutation computed by alchemical methods (see Eq. (2.7)) are:

$$\begin{aligned}
\Delta f_{\text{1-mut}} &= -\beta^{-1}\ln\frac{V^{N_N}}{V^{N_M}} - \beta^{-1}\ln\frac{\int d\Gamma'_{P,M}\,\Theta_{P1}\exp\left[-\beta\left(U(P)+U(P,M)+U(M)\right)\right]}{\int d\Gamma'_{P,N}\,\Theta_{P1}\exp\left[-\beta\left(U(P)+U(P,N)+U(N)\right)\right]} \\
\Delta f_{\text{2-mut}} &= -\beta^{-1}\ln\frac{V^{N_N}}{V^{N_M}} - \beta^{-1}\ln\frac{\int d\Gamma'_{P,M}\,\Theta_{P2}\exp\left[-\beta\left(U(P)+U(P,M)+U(M)\right)\right]}{\int d\Gamma'_{P,N}\,\Theta_{P2}\exp\left[-\beta\left(U(P)+U(P,N)+U(N)\right)\right]}
\end{aligned} \tag{2.11}$$

where Θ_X is a Heaviside function which is equal to one if the coordinate vector belongs to conformation *X*, and it is zero otherwise. To adjust the free energy differences

of the dual topology to the free energy of mutation the correction must be added to each of these terms (Eq.(2.8)). We have

$$\begin{aligned} \Delta F_{\text{1-mut}} &= \Delta f_{\text{1-mut}} - \Delta F_{\text{corr}} \\ \Delta F_{\text{2-mut}} &= \Delta f_{\text{2-mut}} - \Delta F_{\text{corr}} \end{aligned} \tag{2.12}$$

The correction to the free energy differences is independent of the protein conformation. Therefore, the relative free energy difference computed by alchemical methods ($\Delta\Delta f_{\text{mut}}$) is the same as the actual relative free energy ($\Delta\Delta F_{\text{mut}}$):

$$\Delta\Delta F_{\text{mut}} = \Delta F_{\text{1-mut}} - \Delta F_{\text{2-mut}} = \Delta f_{\text{1-mut}} - \Delta F_{\text{corr}} - \Delta f_{\text{2-mut}} + \Delta F_{\text{corr}} = \Delta\Delta f_{\text{mut}} \tag{2.13}$$

The elimination of the correction terms suggests that the end points of the calculations can be manipulated to our advantage (as first suggested in [36a]) to minimize the cost of the calculations. As long as they cancel out the correct relative free energy difference is obtained.

Retaining all the self-interactions

We illustrate an alchemical pathway that retains the self-interactions of the substituted fragment and provides the correct relative free energy difference. This pathway avoids the scaling of the self-interactions by the order parameter λ in the interpolating Hamiltonian, and therefore suggests a shorter, more efficient path of mutation:

$$\begin{aligned} H'(P,M,N;\lambda) &= K(P,M,N) + U(P) + (1-\lambda)U(P,N) + \\ &+ \lambda U(P,M) + U(N) + U(M) \end{aligned} \tag{2.14}$$

If λ=0 the system described by the Hamiltonian is composed by particles P and N, and by a molecule M in the ideal gas state. When λ=1, the P and M particles make the mutant protein, while N is a molecule in ideal gas state.

The free energy difference between the initial and final state computed using this Hamiltonian is:

$$\Delta f' = -\beta^{-1} \ln \frac{\int d\Gamma'_{P,M,N} \exp\left\{-\beta\left[U(P)+U(P,M)+U(M)+U(N)\right]\right\}}{\int d\Gamma'_{P,M,N} \exp\left\{-\beta\left[U(P)+U(P,N)+U(M)+U(N)\right]\right\}} \qquad (2.15)$$

The key observation is that the N particles in the numerator and M particles in the denominator do not interact with the protein or the solution; therefore their contribution to the free energy difference can be isolated:

$$\Delta f' = -\beta^{-1} \ln \frac{\int d\Gamma'_{P,M} \exp\left\{-\beta\left[U(P)+U(P,M)+U(M)\right]\right\}}{\int d\Gamma'_{P,N} \exp\left\{-\beta\left[U(P)+U(P,N)+U(M)\right]\right\}} + \\ -\beta^{-1} \ln \frac{\int dQ_N \exp\left[-\beta U(N)\right]}{\int dQ_M \exp\left[-\beta U(M)\right]} \qquad (2.16)$$

The deviation of this free energy difference and the one obtained with alchemical methods that scale the self-interactions (Eq.(2.7)) is:

$$\Delta f = \Delta f' + \beta^{-1} \ln \frac{V^{N_M} \int dQ_N \exp(-\beta U(N))}{V^{N_N} \int dQ_M \exp(-\beta U(M))} \qquad (2.17)$$

Similar to our previous argument about the difference between ΔF and Δf, the difference in Eq.(2.17) is independent of the protein conformations and we therefore have

$$\Delta\Delta f_{\text{mut}} = \Delta f_{\text{1-mut}} - \Delta f_{\text{2-mut}} = \Delta f'_{\text{1-mut}} - \Delta f'_{\text{2-mut}} = \Delta\Delta f'_{\text{mut}} \qquad (2.18)$$

This proves that the $\Delta\Delta F$ computed along this pathway that avoids the annihilation/creation of self-interactions of the substituted fragment is exact. This proof is similar to the one reported in[37], where only the bonded interactions are scaled. Our algorithm is more general; all the interactions involving only M or N particles are left "as are". This may be useful since the gas phase molecule will have more restricted conformational space that can be sampled more efficiently. There is also another general proof, which is different from the one that I just presented.[38]

Scaling the P-N(M) Bonded Interactions

The path discussed in the previous section requires that all the external interactions (e.g. interactions involving P, and N or M particles) are annihilated/created to bring the system to the correct end-states. If the annihilated chemical group is decoupled from the rest of the environment, it explores the whole simulation box, which makes statistical convergence difficult.[36a] A solution is to restrain the overall relative translations and rotations of the fragment with respect to the scaffold. An algorithm capable of doing this was presented for free energy calculation of binding of a ligand to an enzyme, and is called Virtual Bond Algorithm (VBA). [39]

According to VBA, it is possible to retain a few bonded P-N(M) interactions by "cross linking" the six external degrees of freedom of the annihilated particles to the P atoms. If there are no interactions between the annihilated particles (say M) and the P atoms, the partition function of the overall system is:

$$Z(P,M) = Z(P)Z(M) = Z(P)Z_{\text{int}}(M)8\pi^2 V \tag{2.19}$$

In the right hand side of Eq.(2.19) the partition function of the M molecule/fragment $Z(M)$ is separated into the partition function of its internal degrees of freedom $Z_{int}(M)$, and the partition function of its external degrees of freedom, which is $8\pi^2 V$.[38]

The VBA algorithm makes it possible to find a transformation of coordinates such that we can isolate the six external degrees of freedom of the M species and restrain them to the P particles. This yields:

$$Z'(P,M) = Z(P)Z_{\text{int}}(M)Z_{P\text{-}M} \tag{2.20}$$

Here, $Z_{P\text{-}M}$ is the "cross linking" partition function, i.e. the partition function of the six restraints on the external degrees of freedom of the M molecule/fragment that

restrain its relative distance and orientation with respect to the P molecule/scaffold. It is important to highlight that $Z_{P\text{-}M}$ does not depend on the coordinates of the P molecule/scaffold. The free energies of the system with the free M molecule/fragment (Eq.(2.19)) and the one with the "cross linked" M molecule/fragment (Eq.(2.20)) are different. Their difference is:

$$\Delta F_{P-M} = -\beta^{-1} \ln Z'(P,M) + \beta^{-1} \ln Z(P,M) = -\beta^{-1} \ln \frac{Z_{P-M}}{8\pi^2 V} \qquad (2.21)$$

Since the six restraints in Z_{P-M} are independent, Z_{P-M} can be written as the product of six one-dimensional integrals. These integrals may be solved analytically or numerically, but no further MD simulations are required.

According to VBA, it is possible to retain one bond, two angles and three torsions, chosen in such a way that they involve 3 P particles, and 3 M or N particles.[39] Since we often have more bonded interactions between P, and M or N particles, we still have to deal with the alchemical annihilation/creation of a few bonded interactions. In the next section we propose a route to remove such interactions avoiding problems due to the periodicity of angle and improper torsion terms.

In a thermodynamic cycle these cross-linking restraints do not affect the relative free energy difference, i.e. $\Delta\Delta F$ computed with a cross-linked system is identical to $\Delta\Delta F$ computed without the cross-linking interactions. To illustrate this, let us refer to Figure 1. The restraining potential appears only in the two mutations (vertical arrows). In the first mutation (right vertical arrow), the simulation is run with the cross-links "on" for both the mutant and the native fragments. To correct for this bias we need to add the free energy contribution of restraining the mutant to the protein (ΔF_{P-M}) and remove the free energy difference of restraining the native ($-\Delta F_{P-N}$). In the second mutation (left vertical arrow), the simulations are run again with the cross-links "on" for both the mutant and

the native. However, in this case we remove the free energy contribution of restraining the mutant ($-\Delta F_{P-M}$), and add the free energy difference of restraining the native (ΔF_{P-N}). The restraining partition function is decoupled from the P partition function (see Eq. (2.20)), therefore it is not affected by the conformation of the protein, and the contribution to the relative free energy difference due to the cross-linking is 0.

A similar argument is reported elsewhere, [37] where the authors did not use VBA, but consider examples to explain which bonded interactions between P-M(N) may be retained without introducing "spurious correlations" within the protein. The conclusions that they drew are analogous to VBA.

ALCHEMICAL METHODS IN MD

There are multiple techniques to compute alchemical free energy differences.[17, 41] Some methods are based upon equilibrium calculations, i.e. the switching parameter is changed infinitely slowly along the reaction, allowing the system to relax and sample equilibrium configurations at every value of the switching parameter. An example of these methods is the so-called multiconfiguration[76] Thermodynamic Integration (TI),[25] which is among those that I will discuss in this section. Some other methods instead are non-equilibrium techniques, in which the switching parameter is changed in time following a protocol that leads the system from the initial A state to the final B state.[77] These methods make use of Jarzynski equality[28] to recover the correct free energy difference from non-equilibrium trajectories. I will not discuss these techniques further, because the code that I implemented is based on equilibrium calculations. Other methods are based upon infinitely fast transitions between different values of the switching parameter from an initial to a final state. This means that an equilibrium simulation is carried out in the initial state and/or in the final state. The configurations sampled in the

initial state are then used to compute the energy of the initial and final state, and the same is done with the configurations sampled in the final state. These initial and final states are not necessarily the A and B states, but can also be a series of intermediates. I will discuss two methods of this type: Free Energy Perturbation (FEP)[26] and Bennett Acceptance Ratio (BAR).[27] Finally, other methods introduce a dynamics on the switching parameter that does not follow a special pre-defined protocol, but changes in time according to a dynamics that is coupled with the configurations of the system.[29] This method was not implemented in the code, so I will not discuss it.

Thermodynamic Integration (TI)

Thermodynamic integration was introduced by Kirkwood[25] as a method to compute the chemical potential. Given an alchemical free energy that depends on λ, like the one in Eq.(2.5), we differentiate it with respect to the order parameter λ to get:

$$\frac{df(\lambda)}{d\lambda} = \frac{\int d\Gamma_{P,M,N} \frac{\partial H(P,M,N;\lambda)}{\partial \lambda} e^{-\beta H(P,M,N;\lambda)}}{\int d\Gamma_{P,M,N} e^{-\beta H(P,M,N;\lambda)}} = \left\langle \frac{\partial H(P,M,N;\lambda)}{\partial \lambda} \right\rangle_{\lambda} \qquad (2.22)$$

Upon integration of Eq.(2.22) over the switching parameter from 0 to 1, we get the free energy difference of Eq.(2.6):

$$\Delta f = f(\lambda = 1) - f(\lambda = 0) = \int_0^1 d\lambda \left\langle \frac{\partial H(\lambda)}{\partial \lambda} \right\rangle_{\lambda} \qquad (2.23)$$

In MD we can run multiple simulations at a fixed value of the switching parameter, and compute the average of the derivative of the Hamiltonian with respect to λ (the so-called multiconfigurational thermodynamic integration[76]). Using a numerical integration scheme, such as the trapezoidal rule, we get the desired free energy difference.

Free Energy Perturbation (FEP)

FEP was derived by Zwanzig[26] as the starting point of a perturbation expansion on the free energy. It is another exact method, and it is derived from the alchemical free energy difference (2.6). We can manipulate the integrand at the numerator of this free energy difference in the following way:

$$\Delta f = \frac{\int d\Gamma_{P,M,N}\left[e^{-\beta H(P,M,N;\lambda=0)}e^{\beta H(P,M,N;\lambda=0)}\right]e^{-\beta H(P,M,N;\lambda=1)}}{\int d\Gamma_{P,M,N}e^{-\beta H(P,M,N;\lambda=0)}}$$

Note that the term in brackets is equal to 1. This equation can be rewritten as an average of the exponent of the difference between the Hamiltonians $\Delta H(P,M,N) = H(P,M,N;\lambda=1) - H(P,M,N;\lambda=0)$:

$$\Delta f = \frac{\int d\Gamma_{P,M,N}e^{-\beta\Delta H(P,M,N)}e^{-\beta H(P,M,N;\lambda=0)}}{\int d\Gamma_{P,M,N}e^{-\beta H(P,M,N;\lambda=0)}} = \left\langle e^{-\beta\Delta H}\right\rangle_{\lambda=0} \qquad (2.24)$$

In MD, this means that we can run the simulation at the initial value of λ, sample the configurations and average the exponential of the difference of the Hamiltonians. It is possible to divide this calculation into multiple steps: instead of changing the order parameter from 0 to 1 in one calculation we can perform a series of small changes of the order parameter and then combine them to get the final result. This can be numerically advantageous, since when the difference between the two Hamiltonians is very large the weight of the point (the exponential of the difference in energy expressed in k_BT) has a very broad distribution and so it is hard to converge.[17]

Bennett Acceptance Ratio (BAR)

BAR[27] is based upon the idea of generating an estimator of the free energy that minimizes the statistical error given a sample of independent configurations. The idea is to use a weighting function $w(\Gamma)$ whose value does not influence the free energy, but it is chosen in such a way that the statistical error for the free energy calculation is

minimized. The starting point is the free energy difference in Eq.(2.6), which can be rewritten as:

$$\Delta f = -\beta^{-1}\ln\frac{Q_1}{Q_0} = -\beta^{-1}\ln\frac{Q_1\int d\Gamma w e^{-\beta[H_1+H_0]}}{Q_0\int d\Gamma w e^{-\beta[H_1+H_0]}} = -\beta^{-1}\ln\frac{\left\langle we^{-\beta H_1}\right\rangle_0}{\left\langle we^{-\beta H_0}\right\rangle_1} \tag{2.25}$$

To make the notation simpler I removed from the integrand function the explicit dependence on phase space, and the subscripts indicating the types of particles (all the particles P, M and N are included in the integration). Furthermore, the subscript to the partition function, the Hamiltonian and the average symbols is the value of the switching parameter. Note that in Eq.(2.25) the free energy is independent of the function w. On the other hand, the weighting function affects the variance. To understand why, let's consider the statistical error of a function $f(x,y)$ depending on two independent variables (i.e. with zero covariance), x and y, each of which is subject to statistical error. Using the propagation error formula, the variance of $f(x,y)$ is[3]:

$$\sigma^2(f) = \left(\frac{\partial f}{\partial x}\right)^2\sigma^2(x) + \left(\frac{\partial f}{\partial y}\right)^2\sigma^2(y) = \left(\frac{\partial f}{\partial x}\right)^2\frac{\left\langle x^2\right\rangle - \left\langle x\right\rangle^2}{n_x} + \left(\frac{\partial f}{\partial y}\right)^2\frac{\left\langle y^2\right\rangle - \left\langle y\right\rangle^2}{n_y} \tag{2.26}$$

Here, we defined n_x and n_y to be the number of independent configurations sampled for the variables x and y, respectively. Let's consider that the ensemble

[3] The error propagation formula comes from a Taylor expansion $f(x,y) \approx f(x_0,y_0) + \partial_x f(x-x_0) + \partial_y f(y-y_0)$. If we consider that the initial point is not affected by the average, and that the varialbes x and y are independent (so their covariance is zero) then we get $\left\langle \left(f - \langle f\rangle\right)^2\right\rangle = \left(\partial_x f\right)^2\left\langle \left(x-\langle x\rangle\right)^2\right\rangle + \left(\partial_y f\right)^2\left\langle \left(y-\langle y\rangle\right)^2\right\rangle$. If we then recall that the error on the average of an observable is equal to the observable divided by the number of measurements (central limit theorem) we divide each side by the number of measurements and get the propagation of error for the average. Eq.(2.26) is thus justified, but note that in this equation the number of measurements may be different for x and y.

averages are performed over a set of n_0 and n_1 independent configurations. Using Eq.(2.26) the statistical error associated with the estimate of Δf from Eq.(2.25) is:

$$\sigma^2(\beta\Delta f) = \frac{\left\langle w^2 e^{-2\beta H_0}\right\rangle_1 - \left\langle we^{-\beta H_0}\right\rangle_1^2}{n_1\left\langle we^{-\beta H_0}\right\rangle_1^2} + \frac{\left\langle w^2 e^{-2\beta H_1}\right\rangle_0 - \left\langle we^{-\beta H_1}\right\rangle_0^2}{n_0\left\langle we^{-\beta H_1}\right\rangle_0^2} \tag{2.27}$$

This equation can be reorganized with some algebra to yield:

$$\sigma^2(\beta\Delta f) = \frac{\int d\Gamma w^2\left[\frac{Q_1}{n_1}e^{-\beta H_0} + \frac{Q_0}{n_0}e^{-\beta H_1}\right]e^{-\beta(H_1+H_0)}}{\left[\int d\Gamma we^{-\beta(H_1+H_0)}\right]^2} - \frac{1}{n_1} - \frac{1}{n_0} \tag{2.28}$$

The goal here is to choose the function w that minimizes this variance. To do that, we look for the stationary point of the numerator of Eq.(2.28), while keeping its denominator fixed using a Lagrange multiplier (this prevents us from finding solutions in which the denominator goes to zero or infinity). This means that we will look for the stationary point by setting to zero the functional derivative:[4]

$$\delta_w\left\{\int d\Gamma w^2\left[\frac{Q_1}{n_1}e^{-\beta H_0} + \frac{Q_0}{n_0}e^{-\beta H_1}\right]e^{-\beta(H_1+H_0)} + \mu\left[\int d\Gamma we^{-\beta(H_1+H_0)} - C\right]\right\} = 0 \tag{2.29}$$

[4] Note the following. We seek the minimum of a function $f(x,y,z)$ with the constraint that $g(x,y,z) = C$. Let's suppose that such minimum exists and is at the point (x_0,y_0,z_0). Let's define a curve $\vec{r}(t)$ lying on the plane $g(x,y,z) = C$, and passing through (x_0,y_0,z_0). The gradient of $g(x,y,z)$ on the plane is orthogonal to the tangent to the curve $d\vec{r}(t)/dt$ by definition. Since the curve $\vec{r}(t)$ passes on the minimum (x_0,y_0,z_0), we have that $df/dt = \nabla_{\vec{r}} f \cdot d\vec{r}/dt = 0$ at the minimum, so in that point $f(x,y,z)$ is perpendicular to the tangent of the line $\vec{r}(t)$ and parallel to the gradient of $g(x,y,z)$, so $\nabla_{\vec{r}} f(x,y,z) = \mu\nabla_{\vec{r}} g(x,y,z)$. This is the equation for the Lagrange multipliers. If the function that we wanted to maximize on the plane $g(x,y,z) = C$ is $h(x,y,z) = f(x,y,z)/g(x,y,z)$, the maximum on the plane would be the same as before, and following the same procedure we would find $dh/dt = 1/g\nabla_{\vec{r}} f \cdot d\vec{r}/dt - f/g^2\nabla_{\vec{r}} g \cdot d\vec{r}/dt$. By definition, on the constrained plane $\nabla_{\vec{r}} g \cdot d\vec{r}/dt = 0$, so the equation for the Lagrange multiplier is exactly the same. This explains why we can find a stationary point of Eq.(2.28) using Eq.(2.29).

This is true for any variation if:

$$w = \mu \frac{1}{\frac{Q_1}{n_1} e^{-\beta H_0} + \frac{Q_0}{n_0} e^{-\beta H_1}} \tag{2.30}$$

The constant μ is irrelevant for the free energy estimate because it cancels out in Eq.(2.25), therefore it will be neglected. If we plug back Eq.(2.30) into Eq.(2.25) we get:

$$\beta\Delta f = \ln \frac{\left\langle \frac{e^{-\beta H_0}}{\frac{Q_1}{n_1} e^{-\beta H_0} + \frac{Q_0}{n_0} e^{-\beta H_1}} \right\rangle_1}{\left\langle \frac{e^{-\beta H_1}}{\frac{Q_1}{n_1} e^{-\beta H_0} + \frac{Q_0}{n_0} e^{-\beta H_1}} \right\rangle_0} = \ln \frac{\frac{n_1}{Q_1} \left\langle \frac{1}{1 + \frac{Q_0}{n_0}\frac{n_1}{Q_1} e^{-\beta(H_1 - H_0)}} \right\rangle_1}{\frac{n_0}{Q_0} \left\langle \frac{1}{1 + \frac{Q_1}{n_1}\frac{n_0}{Q_0} e^{-\beta(H_0 - H_1)}} \right\rangle_0} =$$

$$\ln \frac{Q_0}{n_0}\frac{n_1}{Q_1} + \ln \frac{\left\langle \frac{1}{1 + e^{-\beta\left(H_1 - H_0 - \beta^{-1}\ln\frac{Q_0}{n_0}\frac{n_1}{Q_1}\right)}} \right\rangle_1}{\left\langle \frac{1}{1 + e^{-\beta\left(H_0 - H_1 + \beta^{-1}\ln\frac{Q_0}{n_0}\frac{n_1}{Q_1}\right)}} \right\rangle_0}$$

Now, if we define:

$$C = -\beta^{-1} \ln \frac{Q_1 n_0}{Q_0 n_1} = \Delta f - \beta \ln \frac{n_0}{n_1} \tag{2.31}$$

we end up with the following pair of equations:

$$\begin{aligned} \Delta f &= C - \beta \ln \frac{\left\langle \frac{1}{1 + e^{-\beta(H_1 - H_0 - C)}} \right\rangle_1}{\left\langle \frac{1}{1 + e^{\beta(H_1 - H_0 - C)}} \right\rangle_0} \\ C &= \Delta f + \beta \ln \frac{n_1}{n_0} \end{aligned} \tag{2.32}$$

These two equations can be used iteratively to get the correct free energy difference. One can start from a guess for C, used the sample data to get the free energy difference, then update C and re-iterate until convergence.

As FEP, also BAR can be performed for small changes of the switching parameter, then the results may be combined to get the overall free energy difference.

Comparison between TI, FEP and BAR

The efficiency of the three methods that I just discussed was extensively studied in the past.[31] Without the intent of being exhaustive on the topic, in this paragraph I discuss some of the observations that I made studying running simulations. These observations are in agreement with some of the conclusions of this previous study.[31]

Among the three methods, the least efficient seems to be FEP. The average of the exponent of the difference in the energies (see Eq.(2.24)) is difficult to converge.

TI seems to have good convergence properties. The main difficulties with TI are:

1. The need of computing the derivative of the Hamiltonian with respect to the order parameter. Even though it can be done analytically it requires extra calculations compared to methods that require only energies.
2. The numerical integration in Eq.(2.23) may present difficulties if the function has regions of large curvature.

To mitigate the second difficulty, λ path is chosen that flattens these high curvature regions (see the discussion on how to remove angular potentials) or integrable singularities (see the discussion on how to deal with Lennard-Jones interactions[40]).

On the other hand TI does not require significant post-processing, which is an advantage.

Inspired by the success of BAR,[31] I decided to test its efficiency against TI. As a test of the efficiency I used TI and BAR to estimate the free energy difference using a different sets of intermediates (see the end of the results section). BAR does not change its accuracy when the number of intermediates is reduced, while the accuracy of TI rapidly decreases (see Figure 12).

ALCHEMICAL PATHWAY

In the Theory section we described a Hamiltonian for the alchemical substitution that depends linearly on λ (see Eq.(2.4) and Eq.(2.14)). This is the most straightforward choice, but not necessarily the best one from a numerical perspective. Indeed, there are a number of numerical issues that are encountered when some interactions are progressively created/annihilated. Here, we discuss how to solve them.

Bonded Interactions:

While we retain (of course) all the internal covalent interactions in N and M, some of the external covalent interactions need to be annihilated/created. The annihilation of some of these bonded interaction terms is problematic. In particular, we found that the creation/annihilation of bond-angles and torsions can introduce numerical instabilities, and therefore needs to be treated somehow differently.

Bond-Angles

First of all, to explain what is the problem that occurs when a regular bond-angle is removed, let's consider the angle θ between three particles i, j and k. The bond – angle potential restraining this angle around its equilibrium value θ_{eq} is:

$$U_{ang}(\theta) = k_{ang}\left(\theta - \theta_{eq}\right)^2 \tag{2.33}$$

where k_{ang} is the spring stiffness. In MD, we define the angle θ_{MD} from the Cartesian coordinates of the three particles i, j and k using the definition of scalar product:

$$\theta_{MD} = \arccos\left[\frac{\vec{r}_{ij}\cdot\vec{r}_{jk}}{r_{ij}r_{jk}}\right] = \arccos\left[\cos(\theta)\right] \tag{2.34}$$

where the vectors $\vec{r}_{ij}$ and $\vec{r}_{jk}$ connect particle j to the other two particles. This means that the bond-angle potential that we are actually using is slightly different from the one in Eq.(2.33):

$$U_{ang}(\theta_{MD}) = k_{ang}(\theta_{MD}-\theta_{eq})^2 = k_{ang}\left\{\arccos\left[\cos(\theta)\right]-\theta_{eq}\right\}^2 \tag{2.35}$$

The inverse function arccosine is defined in the set $[0,\pi]$. Therefore, only in this set are the potentials in Eq.(2.33) and in Eq.(2.35) the same. Figure 1 shows the bond-angle potential (Eq.(2.33)) in blue and the one that we use in MD (Eq.(2.35)) in red.

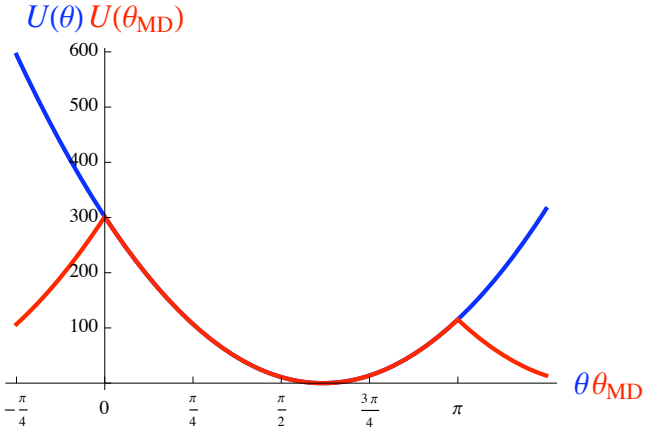


Figure 2: Expected bond-angle potential (blue, see Eq.(2.33)) and bond-angle potential computed in MD (red, see Eq.(2.35)). The equilibrium value ($\theta_0 = 1.941$) and the spring constant ($k_{ang} = 80$kcal/mol) are those for the N-Cα-C backbone angle in OPLS-AAL force field.[4]

It is clear that if the angle θ exits the range $[0,\pi]$ in a MD simulation, we will encounter numerical problems due to inconsistency between the potential and the force. The derivative of the potential in Eq.(2.33) is:

$$\frac{dU_{\text{ang}}(\theta)}{d\theta} = 2k_{\text{ang}}\left(\theta - \theta_{\text{eq}}\right) \tag{2.36}$$

while the derivative of the bond-angle potential computed in MD is:

$$\left.\frac{dU_{\text{ang}}(\theta)}{d\theta}\right|_{\theta=\theta_{\text{MD}}} = 2k_{\text{ang}}\left(\theta_{\text{MD}} - \theta_{\text{eq}}\right) = 2k_{\text{ang}}\left\{\arccos\left[\cos(\theta)\right] - \theta_{\text{eq}}\right\} \tag{2.37}$$

In Figure 3 we show the difference between these two derivatives:

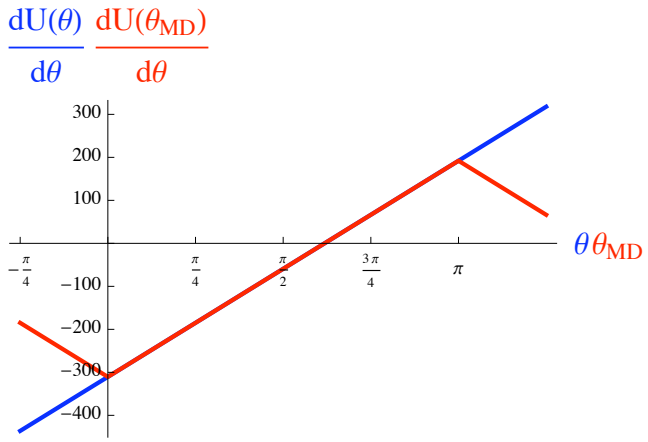


Figure 3: Derivative of the expected bond-angle potential (blue, see Eq.(2.33)) and of the bond-angle potential computed in MD (red, see Eq.(2.35)). The equilibrium value ($\theta_0 = 1.941$) and the spring constant ($k_{\text{ang}} = 80\text{kcal/mol}$) are those for the N-Cα-C backbone angle in OPLS-AAL force field.[4]

This function is continuous in $\theta = 0$ and in $\theta = \pi$, but it has the wrong sign for $\theta < 0$ and $\theta > \pi$. Indeed, the derivative of the computed potential (red line in Figure 2) for $\theta < 0$ should be positive, while the computed one (red line in Figure 3) is negative. The opposite is true for $\theta > \pi$. This means that in these regions the energy (and so free

energy) calculation is inconsistent with the forces governing the time evolution (and so the sampling) of configurations.

In a regular MD simulation at 300K, the spring constant is stiff enough to restrain the fluctuations of the angle within the range of a few of degrees. The problem just highlighted is never encountered, since the energy of the configurations $\theta = 0$ and $\theta = \pi$ is too high to be sampled.

In an alchemical substitution, the bond-angle potential is multiplied by the switching parameter λ. The effective spring constant is then weakened, and the lower the value of λ the larger is the angular space that is sampled. Eventually, the energy barrier to reach the configurations $\theta<0$ and $\theta>\pi$ will be so low that such configurations may actually be sampled. Therefore, we need to find a way to restrain the angle between three particles such that if annihilated by alchemical methods it does not introduce the numerical instability associated with the configurations $\theta<0$ and $\theta>\pi$.

The solution that we propose is to use Urey-Bradley bonds to restrain the angle near the equilibrium value for all those angles that are going to be annihilated/created along the alchemical pathway (e.g. the angles that connect the P with M or N parts of the system). Removal of the Urey-Bradley bond is simpler. Given the three particles above *i*, *j* and *k*, the Urey-Bradley bond restrains the fluctuations of θ around the equilibrium angle by adding a spring in Cartesian space between particles *i* and *k*:

$$U_{UB}\left(r_{ik}\right) = k_{UB}\left(r_{ik} - r_{ik,eq}\right)^2 \tag{2.38}$$

Here, of course

$$r_{ik} = \sqrt{r_{jk}^2 + r_{ij}^2 - 2r_{jk}r_{ij}\cos\left(\theta\right)} \tag{2.39}$$

The two parameters $r_{ik,\mathrm{eq}}$ and θ_{eq} are chosen such that the minimum of the potential corresponds to the equilibrium position according to bond-angle potential and the small fluctuations around the minimum are the same. Therefore, we obtain:

$$r_{ik,\mathrm{eq}} = \sqrt{r_{jk,\mathrm{eq}}^2 + r_{ij,\mathrm{eq}}^2 - 2r_{jk,\mathrm{eq}}r_{ij,\mathrm{eq}}\cos\left(\theta_{\mathrm{eq}}\right)} \tag{2.40}$$

$$k_{\mathrm{UB}} = \frac{1+2\dfrac{r_{ij,\mathrm{eq}}^2 r_{jk,\mathrm{eq}}^2}{r_{ij,\mathrm{eq}}^2 + r_{jk,\mathrm{eq}}^2}\cos\left(\theta_{\mathrm{eq}}\right)}{\dfrac{r_{ij,\mathrm{eq}}^2 r_{jk,\mathrm{eq}}^2}{r_{ij,\mathrm{eq}}^2 + r_{jk,\mathrm{eq}}^2}\sin^2\left(\theta_{\mathrm{eq}}\right)} k_{\mathrm{ang}} \tag{2.41}$$

The choice of the alchemical pathway to annihilate/create the Urey-Bradley bonds (as for any other potential) is arbitrary, provided that the end states are correct. Therefore, we select a scheme (a particular λ-scaling) such that the free energy difference is computed numerically with higher accuracy. For this purpose, we consider a "toy" problem in which we can evaluate the free energy "almost" analytically. We compute the free energy difference due to the annihilation of a Urey-Bradley bond for a triatomic molecule. The alchemical potential of such simple system is given by two chemical bonds and a Urey-Bradley bond scaled by the switching parameter. The exponent of λ, the parameter α, is the target for optimization:

$$U\left(r,r',r_{\mathrm{UB}};\lambda\right) = k\left(r - r_{\mathrm{eq}}\right)^2 + k'\left(r' - r'_{\mathrm{eq}}\right)^2 + \lambda^{\alpha}k_{\mathrm{UB}}\left(r_{\mathrm{UB}} - r_{\mathrm{UB,eq}}\right)^2 \tag{2.42}$$

The configurational partition function for this potential is:[78]

$$Z\left(N,V,T;\lambda\right) = 8\pi^2 V\int r^2 r'^2 \sin\left(\theta\right)\exp\left[-\beta U\left(r,r',r_{\mathrm{UB}};\lambda\right)\right]dr\,dr'\,d\theta \tag{2.43}$$

Here, we changed the coordinates to a polar system and integrated the external degrees of freedom. Let us make the assumption that the spring constants for the chemical bonds are so stiff that in the Jacobian in Eq.(2.43) and in the Urey-Bradley bond the bond lengths *r* and *r'* can be substituted by their equilibrium values. This yields:

$$Z\left(N,V,T;\lambda\right) = 8\pi^2 V r_{\mathrm{eq}}^2 r'^2_{\mathrm{eq}}\sqrt{\frac{2\pi}{k\beta}}\sqrt{\frac{2\pi}{k'\beta}}\int \sin\left(\theta\right)\exp\left[-\beta\lambda^{\alpha}\left(r_{\mathrm{UB}} - r_{\mathrm{UB,eq}}\right)^2\right]d\theta \tag{2.44}$$

Let us further assume that $k=k'$, $r_{eq}=r'_{eq}$ and that $\theta_{eq}=0$. In this case the Urey-Bradley potential is (see Eq.(2.38) -(2.41)):

$$U_{\mathrm{UB}}\left(\theta\right) = 2k_{\mathrm{UB}}r_{\mathrm{eq}}^2\left[1-\cos\left(\theta\right)\right] \tag{2.45}$$

The free energy difference computed by TI is:

$$\frac{\partial F}{\partial \lambda} = \left\langle \frac{\partial U}{\partial \lambda} \right\rangle_{\lambda} = 2\alpha\lambda^{\alpha-1} k_{UB} r_{eq}^2 \frac{\int \sin(\theta)\left[1-\cos(\theta)\right]\exp\left\{-2\beta\lambda^{\alpha} k_{UB} r_{eq}^2\left[1-\cos(\theta)\right]\right\} d\theta}{\int \sin(\theta)\exp\left\{-2\beta\lambda^{\alpha} k_{UB} r_{eq}^2\left[1-\cos(\theta)\right]\right\} d\theta} \quad (2.46)$$

The integral can be solved numerically as a function of λ using Mathematica.[79] With $\alpha=1$ we get the blue line in Figure 4, with $\alpha=2$ the red line.

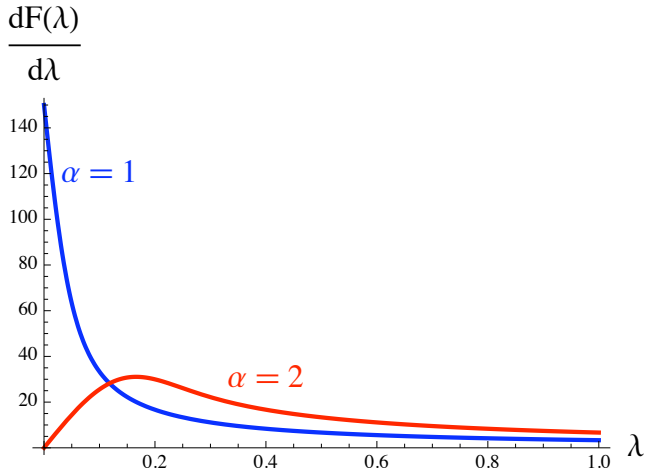


Figure 4: The derivative of the alchemical free energy with respect to λ (Eq. (2.46)) is reported. In blue we show α=1, and in red α=2. The rest of the parameters are: $r_{eq}=1$Å, $k_{UB}=75$ kcal/mol/ Å² and $\beta=0.6$ kcal/mol.

We noticed (unpublished results) that the larger relative errors in the evaluation of $dF/d\lambda$ are found when λ approaches zero. A possible rationale is that the closer the system is to the decoupling point, the larger is the configurational space to sample. To reduce the impact on the overall free energy difference of these terms, we would like $dF/d\lambda$ to be small at $\lambda\cong 0$. Therefore, we adopted $\alpha=2$ in our simulations.

The use of Urey-Bradley bonds instead of regular bond-angle potential comes at a price. In a Thermodynamic Cycle the Urey-Bradley potential is used only when an angular potential is annihilated/created, whenever instead it is left as is, the standard

bond-angle potential is used. This inconsistency introduces a correction to the free energy (the difference between the UB and the usual harmonic angular terms) that we need to compute explicitly. This correction can be computed exactly, as shown in Appendix A. In the numerical example that I used here, the contribution to the free energy of the cycle turned out to be negligible.

Torsions

Given four particles (say, i, j, k and l), the torsion potential restrains the dihedral angle φ between the plane identified by the particles i, j and k and the plane identified by the particles j, k and l (see Figure 5). The functional form used in molecular mechanics force fields is periodic in the angle and does not suffer the same problems as the angular potential. On the other hand, if the weakening of the angular interaction allows particle l to collapse on the line that connects particle j to particle k, the plane identified by the particles j, k and l is not defined, and so the dihedral angle itself is not defined. To avoid this issue it is enough to remove torsions and improper torsions before the angles.

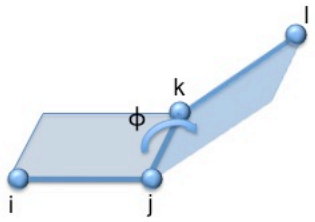


Figure 5 An example of torsion dihedral φ is reported

Van der Waals Interactions

The creation and the annihilation of particles with a finite Van der Waals radius present (integrable) singularities.[40] Such singularities are removed if the Van der Waals interactions are created/annihilated with scaling parameter λ^{α}, with $\alpha \geq 4$.[40] Moreover, the use of a soft core Lennard Jones potential avoids hard clashes during the creation/annihilation of interactions between particles.[40] The potential used is then:

$$\lambda^4 U(r) = 4\varepsilon\lambda^4 \left\{ \frac{\sigma^{12}}{\left[r^6 + \alpha_{LJ}(1-\lambda)^2 \sigma^6 \right]^2} - \frac{\sigma^6}{\left[r^6 + \alpha_{LJ}(1-\lambda)^2 \sigma^6 \right]} \right\} \qquad (2.47)$$

It is clear that for $\lambda = 0$ this potential disappears, for $\lambda = 1$ we recover the usual Lennard Jones potential (in this way the correct end-state is recovered), for $0 < \lambda < 1$ the hard core is softened, i.e. there is no divergence in $r = 0$. The larger is the value of α_{LJ} the softer is the potential. In the numerical example discussed in the next section we used

$\alpha_{LJ} = 0.3$, in agreement with what was used in the original work where this issue was presented and solved.[40]

To avoid the collapse of particles with opposite charges to the same point in space while the Van der Waals repulsion between them is reduced, the free energy calculation is conducted in two steps: in the first step the electrostatic interactions of the N particles are turned off, while the Van der Waals interactions of the M particles are created. In the second step the Van der Waals interactions of the N particles are removed, while the electrostatic interactions of the M particles are turned on.[41] For this reason, we break the free energy calculation into two parts: MutationPHase1 (MPH1) and MutationPHase2 (MPH2). In MPH1 we remove electrostatic interactions, torsions and improper torsions between P and N particles and create Van der Waals interactions and angles between P and M particles. In MPH2 we do the opposite. We remove Van der Waals interactions and angles between P and N particles and create electrostatic interactions, torsions and improper torsions between P and M particles. Note that we do not create or remove the bond between particles P and M, or the bond between P and N. Note also that all the self-interactions, bonded and non-bonded, are never created nor annihilated.

The arguments presented so far are summarized in the following two tables, which define the alchemical pathway used in the numerical illustration.

	MPH1									
	scaling					correction				
	P-P	P-M	P-N	M-M	N-N	P-P	P-M	P-N	M-M	N-N
Vdw	1	λ^4	1	1	1		Soft core			
Ele	1	0	$(1-\lambda)$	1	1					
Bond	1	1	1	1	1					
Angle	1	λ^2	1	1	1		Urey-Bradley			
Tors	1	0	$(1-\lambda)^2$	1	1					
ImpT	1	0	$(1-\lambda)^2$	1	1					

Table 1 The alchemical pathway for the first phase of the mutation. The angular and Van der Waals interactions between P and M particles are created, while torsions, improper torsions and electrostatic interactions between P and N are progressively switched off. The right hand side of the table reports the corrections to the regular OPLS-AAL [4] potential that were used. Notice that the self-interactions, as well as the bonds, are never scaled (see details in the text).

	MPH2									
	scaling					correction				
	P-P	P-M	P-N	M-M	N-N	P-P	P-M	P-N	M-M	N-N
Vdw	1	1	$(1-\lambda)^4$	1	1			Soft core		
Ele	1	λ	0	1	1					
Bond	1	1	1	1	1					
Angle	1	1	$(1-\lambda)^2$	1	1			Urey-Bradley		
Tors	1	λ^2	0	1	1					
ImpT	1	λ^2	0	1	1					

Table 2 The alchemical pathway for the second phase of the mutation. The torsions, improper torsions and electrostatic interactions between P and M particles are created, while angular and Van der Waals interactions between P and N are progressively switched off. The right hand side of the table reports the corrections to the regular OPLS-AAL [4] potential that were used. Notice that the self-interactions, as well as the bonds, are never scaled (see details in the text).

Electrostatic Interactions

To ensure accurate estimates of the electrostatic interaction we use Particle Mesh Ewald.[80] The drawback of this method with respect to the alchemical substitution is that the reciprocal space contribution to the energy cannot be separated into the specific contributions of each pair of interacting particles because the reciprocal space energy of a particle cannot be separated into single pairwise contributions. Therefore, it is not possible to selectively annihilate/create some interactions involving a particle, while keeping the others fixed.

Recall the PME theory.[80] We define $\vec{n}$ as the lattice vectors, and $C(i,j)$ are interaction parameters for pair of particles. A potential energy which is the sum of the interactions of all the pair of particles with the exception of self (as the prime in the summation symbol below reminds us) is:

$$E(\vec{r}_1,...,\vec{r}_N) = \frac{1}{2}\sum_{\vec{n}}{}'\sum_i\sum_j \frac{C(i,j)}{\left|\vec{r}_i - \vec{r}_j + \vec{n}\right|^p} \tag{2.48}$$

the PME methodology is applicable if we can write (see Appendix of [80]):

$$C(i,j) = \pm C(i)C(j) \tag{2.49}$$

If we want to retain the "self" electrostatic interactions, we cannot use Eq.(2.49). Indeed, let us suppose that particle *i* is a M particle, then we have:

$$C(i,j) = \begin{cases} \lambda \dfrac{q_i q_j}{4\pi\varepsilon_0} & j \in P \\ 0 & j \in N \\ \dfrac{q_i q_j}{4\pi\varepsilon_0} & j \in M \end{cases} \tag{2.50}$$

where q_i are charges and ε_0 is the dielectric constant. Since the P particles interact according to the usual Coulomb law ($q_i q_j / 4\pi\varepsilon_0$, without λ) and the N particles may carry a charge different from zero, we cannot write Eq.(2.50) as Eq.(2.49), i.e. the

functional form of the potential cannot be reduced to individual properties of the particles, but depends on the pair of interacting particles.

To solve this problem we compute the reciprocal space energy term between selected types of particles and scale the interactions with the proper switching parameter. A linear combination of the terms obtained in this way gives us the desired overall energy dependence on the switching parameter (i.e. the "self interactions" are not scaled).

The recipe is summarized in the two following tables that explain how to carry out the Ewald sums for MPH1 and MPH2.

MPH1			
#	Type of particle interacting	Scaling	Result
1	P	λ	$\lambda U(P)$
2	P N	$(1-\lambda)$	$(1-\lambda)[U(P)+U(P,N)+U(N)]$
3	M	1	U(M)
4	N	λ	$\lambda U(N)$
OVERALL: $U(P) + (1-\lambda)U(P,N) + U(N) + U(M)$			

Table 3 The numerical recipe to properly scale only the reciprocal space external molecular electrostatic interactions computed by Ewald sums in MPH1. The reciprocal space is computed four times: first only with the electrostatic interactions between the P particles turned on and scaled by λ. Then the electrostatic interactions between P and N particles are turned on and the result is scaled by $1-\lambda$. Then only the M particles are interacting and the result is not scaled. Finally, only the N particles are interacting, and the result is scaled by λ. The sum of these four contributions is shown at the bottom of the table. The self-interactions are not scaled. The external interactions are instead scaled by $1-\lambda$.

MPH2			
#	Type of particle interacting	Scaling	Result
1	P	(1-λ)	(1-λ)U(P)
2	P M	λ	λ[U(P)+U(P,M)+U(M)]
3	M	(1-λ)	(1-λ)U(M)
4	N	1	U(N)
OVERALL: U(P) + λU(P,M) + U(M) + U(N)			

Table 4 The numerical recipe to properly scale only the reciprocal space external molecular electrostatic interactions computed by Ewald sums in MPH2. The reciprocal space is computed four times: first only with the electrostatic interactions between the P particles turned on and scaled by 1-λ. Then the electrostatic interactions between P and M particles are turned on and the result is scaled by λ. Then only the M particles are interacting and the result is scaled by λ. Finally, only the N particles are interacting, and the result is not scaled. The sum of these four contributions is shown at the bottom of the table. The self- interactions are not scaled. The external interactions are instead scaled by λ.

This solution has two major disadvantages. First, using PME four times per integration step is an expensive procedure. Second, the final state of the decoupled fragments is not a fragment, but an infinite periodic lattice of the fragment, which in theory interacts with all the copies of the system. This is a problem if simulations in vacuum are needed to close the Thermodynamic Cycle: instead of just sampling the conformations of the decoupled fragment in vacuum, one has to take into account the interactions with the copies as well. An illustration of this issue can be found in Figure 6, in which the fully coupled particles are represented by filled circles, while the decoupled particles are represented by circles filled with a pattern of dots (blue represents the P particles, green the N particles, and red the M particles).

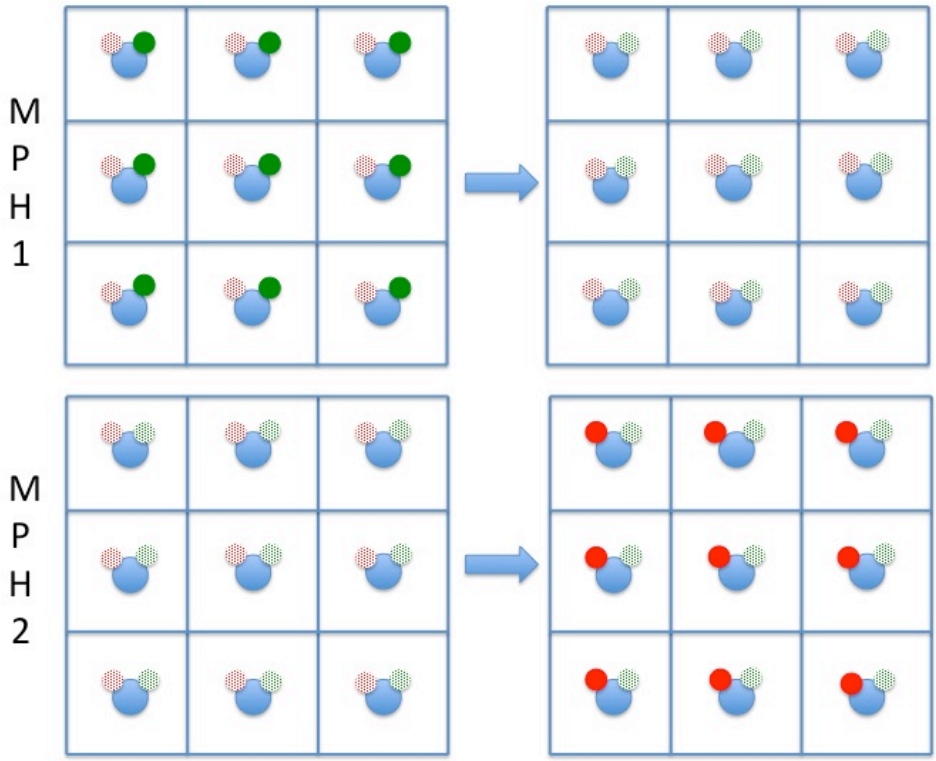


Figure 6: Pictorial illustration of the alchemical scheme for electrostatic interactions. The different quadrants represent multiple copies of the system due to periodic boundary conditions. The blue circle represents the set of P particles, the red circle the M particles and the green circle the N particles. A fully colored circle represents a species (P, M or N) that has all its interactions with the other copies and the other species "on". When the coloring of the circle is instead dotted, that means that the species (M or N) is interacting only with particles of the same species. In MPH1 the system at the beginning of the simulations is characterized by fully interacting P (blue) and N (green) particles, while the M particles (red, dotted) only see each other. At the end of MPH1, also the green particles (now green and dotted) are decoupled from the system and interact only with N particles. This is also the starting point for MPH2. During MPH2, the M particles (red, dotted) are progressively coupled to the system and the end state has P (blue) and M (red) particles fully interacting, while the N particles (green, dotted) only see each other.

While this is not wrong in principle, an end state made of one fragment, and not a lattice of fragments, would be more intuitive.

The calculations presented in the Results section use the recipe hereby introduced to account for the Ewald sums. A new solution was implemented in the latest version of MOIL (MOIL-OPT[24]), which was aimed at solving this problem. An illustration of this new solution is left at the end of this chapter.

NUMERICAL EXAMPLE

We study an alchemical substitution that retains all the self-interactions. The difference in hydration free energies of ILE and GLN side chain analogs is computed. The cycle that we consider is presented in Figure 7.

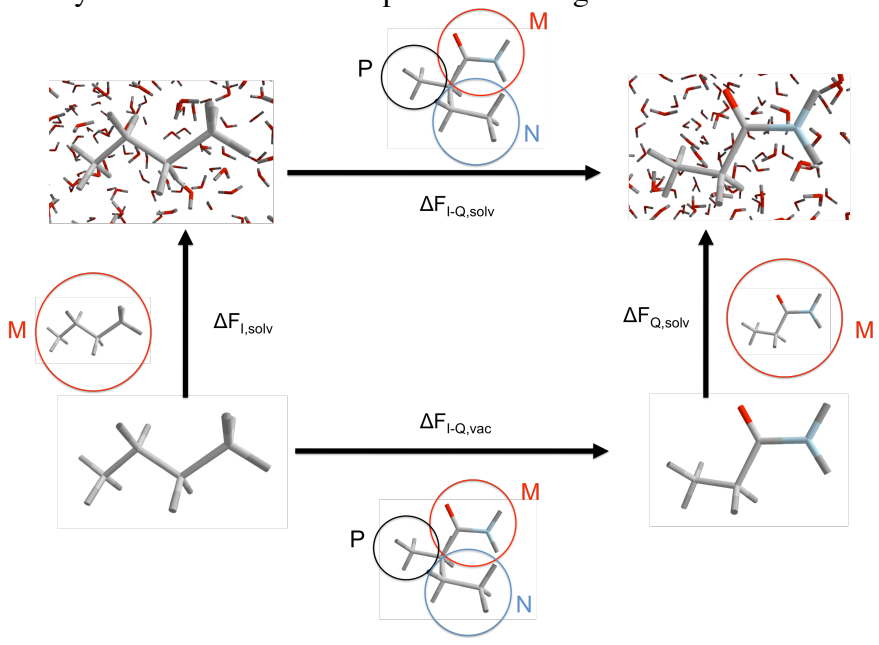


Figure 7 The thermodynamic cycle that we used in these calculations. The horizontal arrows represent free energy differences upon mutation of the solute (ILE side chain analog and GLN side chain analog). The vertical arrows represent the solvation process, i.e. the solute is brought from vacuum into solution. For each free energy calculation the colored circles highlight which atoms are considered P (black), N (blue) or M (red).

$\Delta F_{I\text{-}Q,solv}$ is the free energy difference of mutating ILE to GLN side chain analog in solution, while $\Delta F_{I\text{-}Q,vac}$ is the same quantity computed in vacuum (gas phase). $\Delta F_{I,solv}$ and $\Delta F_{Q,solv}$ are the free energy differences of inserting ILE and GLN side chain analogs into water from gas phase, respectively.

The free energy difference of the complete cycle is:

$$\oint dF = \Delta F_{I-Q,solv} - \Delta F_{Q,solv} - \Delta F_{I-Q} + \Delta F_{I,solv} = \\ = -\beta^{-1} \ln\left[\frac{Q_{solv,Q}}{Q_{solv,I}} \frac{Q_{solv} Q_Q}{Q_{solv,Q}} \frac{Q_I}{Q_Q} \frac{Q_{solv,I}}{Q_{solv} Q_I}\right] = 0 \tag{2.51}$$

as it should be. The numerical task and test is to reproduce the zero in simulations.

Mutations

To compute the free energy differences $\Delta F_{I\text{-}Q,solv}$ and $\Delta F_{I\text{-}Q}$ we used the system shown in Figure 8.

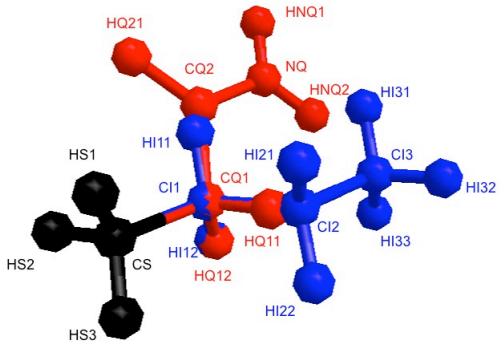


Figure 8 The alchemical intermediate that was used. The black methyl group is shared between the two side chain analogs. The blue atoms are the rest of the ILE side chain analog. The red atoms are the remaining of the GLN side chain analog. While the black atoms interact with all the other atoms, blue and red particles do not interact with each other.

The black methyl group is considered the P part of the system, i.e. the part that interacts with both mutants during the alchemical calculation. The blue atoms are the native N, i.e. the ILE side chain without its methyl group that is considered P. The red atoms are the mutant M, i.e. the GLN side chain analog without the methyl group which is part of P. To illustrate our procedure, the simulations are performed while retaining the full strength of the self-interactions. More details about the simulations follow.

Solvation

To compute the free energy difference of the two side chains in solution and in gas phase the non-bonded interactions between the side chain analog and the solvent around it are turned off.[33] Following the nomenclature introduced in the previous sections, we consider water as P molecules, the side chain analog (ILE or GLN) as M particles. No N particles are involved in this simulation.

Simulation details

The OPLS-AAL force field[4] was used with a modification on the charges of the C_β of the residues analogs.[33] Also, the Lennard-Jones parameters for polar hydrogens belonging to the solutes were set to σ=0.3Å and ε=0.0498kcal/mol. All the systems were solvated in a periodic cubic box of volume 34.45Å^3 with 1355 TIP3P water molecules[81] at 300K. The size of the water box was chosen in such a way to have an average water density at corners of the box near 0.998g/cm^3 with the ILE sidechain analog solvated in the center of the water box. The rationale of this choice is that at the corners of the box we are far enough from the solute to be in "bulk".

The cutoff distance was 10Å for Van der Waals interactions and 14Å for electrostatic forces. These real space cutoff distances gave very good energy conservation throughout the simulation on a 2ns test case conducted in the NVE ensemble. Particle

Meshed Ewald (PME)[80] was used in all simulations with a grid of 64x64x64. PME was used also in the simulations carried out in vacuum. The use of PME in the entire cycle includes interactions between molecules in different periodic cells. The straightforward choice of using a finite cutoff for the electrostatic calculations in vacuum introduces a systematic error (results not shown). Indeed, the use of PME in all the parts of the Thermodynamic Cycle is necessary to ensure consistency if the scheme for PME is the one presented in Table 3 and Table 4. The contributions of the Van der Waals interactions between particles that are beyond the cutoff distance are not considered explicitly in this simulation. Other authors have introduced an analytical correction for the free energy of solvation of small molecules.[33] The solvent around each of the alchemically substituted particles of the solute is assumed homogeneous and isotropic. Moreover, the pair correlation function between the alchemically substituted solute and the solvent particles is assumed independent of λ and is equal to 1 (the cutoff is large enough that there are no correlations). The formula that was used is the following:[33, 82]

$$\Delta F_{LRC} = 4\pi \sum_{i \in M} \rho_O \int_{r_c}^{\infty} dr_{i,O} r_{i,O}^2 U_{VdW}\left(r_{i,O}\right) - 4\pi \sum_{i \in N} \rho_O \int_{r_c}^{\infty} dr_{i,O} r_{i,O}^2 U_{VdW}\left(r_{i,O}\right) \qquad (2.52)$$

Here, the density ρ_O is the number density of oxygen in water molecules. The Van der Waals interactions considered are only between solute and oxygen, according to the TIP3P water model.

The results for this correction are reported in Table 6

Table 6, together with all the other results. This correction is not affecting the overall free energy cycle. In Figure 7, we notice that each correction is added (at the end of the solvation simulation for ILE, and at the end of the mutation for GLN) and then

removed (at the beginning of the mutation simulation for ILE and the end of the solvation simulation for GLN) per each solute. The net effect is zero.

In the two mutations, the geometric center of the P particles was restrained to the center of the box by a harmonic potential with spring constant $k = 1\text{kcal/mol}$. In the two solvated calculations, the geometric center of the analog was restrained to the center of the box by a harmonic potential with the same spring constant k. In both of the cases, the free energy contribution from this harmonic potential restrains an external degree of freedom, so it is separable. Since the two spring constants are the same, the free energy of the system is not affected by the choice of the subset of particles to restrain. Therefore, there is no overall contribution from this constraint to the total free energy difference.

The solvated simulations were performed in the NVT ensemble by rescaling the velocities at every time step to maintain the temperature of the thermal bath.[83] The vacuum simulations were performed in the NVT ensemble using a Langevin thermostat[82] to enhance coupling between different degrees of freedom and ergodicity.

The calculation of the free energy difference was performed using TI in the "multiconfiguration" approach[76]. The numbers of intermediates λ values at which the free energy differences were sampled and computed are listed in the following table. They were chosen to allow a better description of the regions in $d\text{F}/d\lambda$ characterized by a larger curvature.

	MPH1	MPH2	Total simulation time
$\Delta F_{I\text{-}Q,solv}$	33	33	132ns
$\Delta F_{I\text{-}Q,vac}$	33	33	132ns
$\Delta F_{I,solv}$	35	23	116ns
$\Delta F_{Q,solv}$	35	23	116ns

Table 5 The number of λ points used in the evaluation of the free energy differences of Figure 7 for each of the mutation phases MPH1 and MPH2. The last column shows the total simulation time for each stage in the cycle.

For the solvated simulations, at each λ value two equilibrations of 100ps were performed: one with the solute frozen and the other with the solute free to move. Then a 2ns long sampling of configurations was carried out for each λ value. For the simulations carried out in vacuum, we sampled configurations for 2ns using Langevin dynamics with friction coefficient γ=60/ps. One configuration per picosecond was used to compute the average and the variance of $dH/d\lambda$. The configurations sampled in this way may be correlated. To account for this correlation we computed the variance of $dH/d\lambda$ using the following formula:[82, 84]

$$\sigma^2\left(\left\langle\frac{dH}{d\lambda}\right\rangle_T\right)=\frac{2}{T}\int_0^T d\tau\left(1-\frac{\tau}{T}\right)\left\langle\left(\frac{dH}{d\lambda}(\tau)-\left\langle\frac{dH}{d\lambda}\right\rangle_T\right)\left(\frac{dH}{d\lambda}(0)-\left\langle\frac{dH}{d\lambda}\right\rangle_T\right)\right\rangle_T \quad (2.53)$$

In this equation, the symbol "$\langle\ldots\rangle_T$" refers to a time average over the sampled configurations carried out for all the length of the simulation T. The argument of the integral contains the correlation function of $dH/d\lambda$. The correlation function decays to *0* within the first few picoseconds (see Figure 9). Therefore, it was computed for 20ps and then set to *0* to avoid integration over a noisy tail that may introduce unphysical contributions to the integral. The integral in Eq.(2.53) was evaluated using the trapezoidal rule. Example of correlation functions are reported in Figure 9.

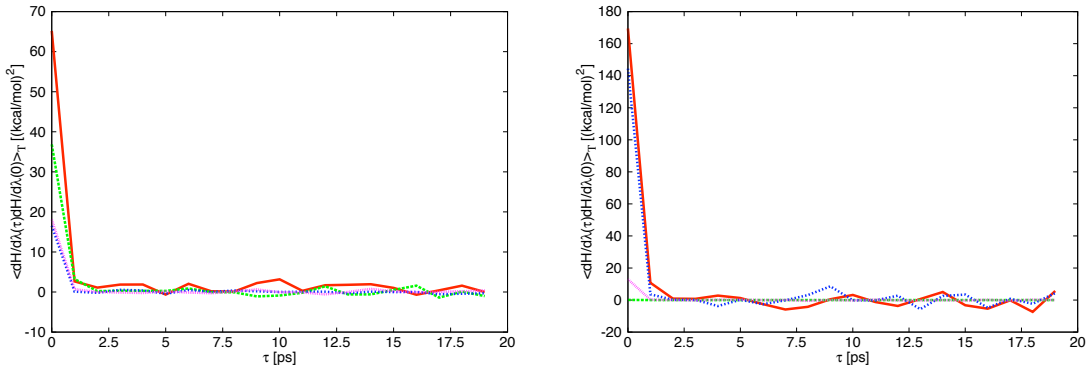


Figure 9: Correlation function of $dH/d\lambda$ as a function of time (in ps). On the left the figure reports calculations at $\lambda=0.5$ of the MPH1 (red) and MPH2 (green) phases of the mutation from ILE side chain analog to GLN sidechain analog in solution. In the same figure the MPH1 (blue) and MPH2 (pink) phases of the same mutations in vacuum are reported. The figure on the right reports the calculations at $\lambda=0.5$ of the MPH1 (red) and MPH2 (green) of the solvation of ILE side chain. In the same figure the MPH1 (blue) and the MPH2 (pink) phases of the solvation of GLN side chain analog are also reported.

The variance of the integral was estimated from the variances of each point of $dF/d\lambda$ using the propagation formula.

Results

The profiles for $dF/d\lambda$ as a function of λ are reported in Fig. 5.

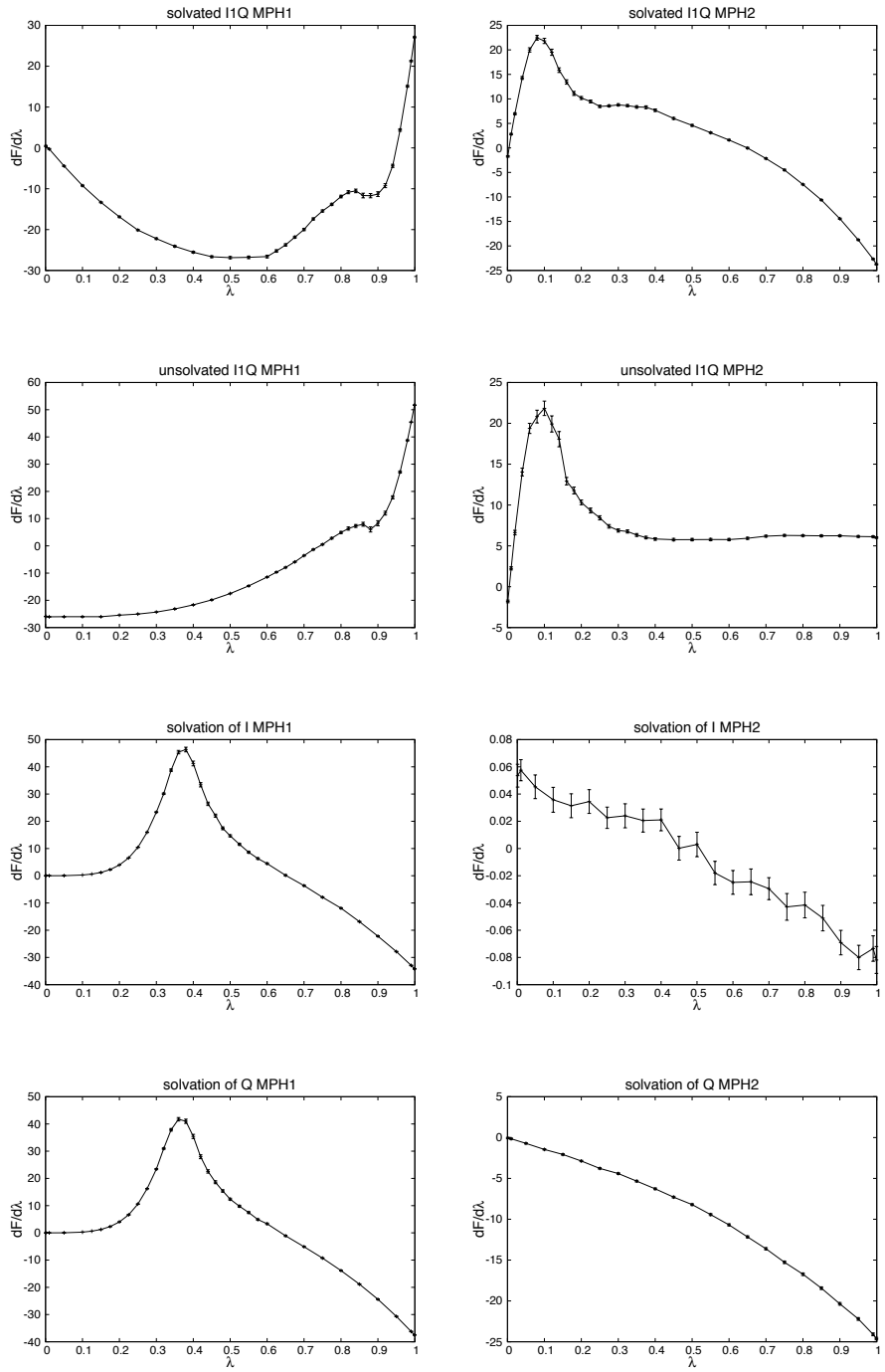

solvated I1Q MPH1
solvated I1Q MPH2
unsolvated I1Q MPH1
unsolvated I1Q MPH2
solvation of I MPH1
solvation of I MPH2
solvation of Q MPH1
solvation of Q MPH2
dF/dλ
λ

Figure 10

Figure 10: dF/dλ profiles (in kcal/mol) as a function of λ of both of the phases MPH1 and MPH2 of the free energy differences of each part of the cycle of Figure 7. On the left the MPH1 phase is reported, on the right the MPH2. From top to bottom there are: mutation from ILE to GLN in solution, mutation from ILE to GLN in vacuum, solvation of ILE and solvation of GLN. The error bars are shown in all figures.

In Table 6A the results are reported for each part of the calculation.

(A)

	TOTAL	MPH1	MPH2	LRC
$\Delta F_{I\text{-}Q,solv}$	-14.073±0.066	-16.218±0.050	2.199±0.043	-0.051
$\Delta F_{I\text{-}Q,vac}$	-2.847±0.068	-10.931±0.047	8.084±0.049	
$\Delta F_{I,solv}$	2.891±0.050	3.379±0.050	-0.0075±0.0019	-0.481
$\Delta F_{Q,solv}$	-8.354±0.054	1.858±0.047	-9.680±0.027	-0.532

(B)

	Total[33]	Vdw[33]	Ele[33]	LRC[33]
$\Delta F_{I\text{-}Q,solv}$				
$\Delta F_{I\text{-}Q,vac}$				
$\Delta F_{I,solv}$	2.73±0.03	3.10±0.03	-0.01±0.001	-0.37
$\Delta F_{Q,solv}$	-8.40±0.04	1.80±0.03	-9.79±0.02	-0.41

(C)

	Experiment[33, 85]
$\Delta F_{I\text{-}Q,solv}$	
$\Delta F_{I\text{-}Q,vac}$	
$\Delta F_{I,solv}$	2.15
$\Delta F_{Q,solv}$	-9.38

Table 6

Table 6: The results of the free energy calculations and the statistical errors (all the values are in kcal/mol). (A) The values reported are the result of our simulations. The second column reports the total, the third the free energy difference in MPH1, the fourth the free energy difference in MPH2, the fifth the long-range correction (LRC). (B) This table shows the values for the calculation of the solvation free energy of the same side chain analogs that we considered as reported in ref.[33] The second column is total, the third the value of the solvation related only with creation of Van der Waals interactions between the solute and the solvent with the electrostatic interactions off (analogous to our MPH1), the fourth column reports the free energy contribution associated with creation of electrostatic interactions when the Van der Waals are already fully coupled (analogous to our MPH2). The last column is the long-range correcion (LRC). The protocol used to carry out these calculations in ref. [33] is significantly different from ours. The authors in ref. [33] carried out NPT simulations and their cutoff scheme is different from ours (no Ewald, group-based and tapered interactions). This is reflected mostly in the LRC term, which is, of course, dependent on the scheme used for the Van der Waals cutoff. (B) Experimental results for the solvation free energies of the two side chain analogs.[85] These results were used as a comparison with the simulations in ref.[33]

In the second column of Table 6A we report the total of the free energy calculation. The total is then broken in three parts: the result for the first phase (MPH1, third column), the result for second phase (MPH2, fourth column) and the Long Range Correction (LRC, fifth column) approximately correcting for the finite Van der Waals cutoff used in simulations.

In a previous computation of the solvation of side chain analogs,[33] the authors computed the solvation free energies of these two side chain analogs. The free energy calculation reported by them[33] was performed following an alchemical pathway similar to ours: i.e. splitting the decoupling in two steps. First the Van der Waals interactions are turned on, while the electrostatic interactions are kept off. Secondly, the electrostatic interactions are created, while keeping the Van der Waals interactions at their full strength. The first step is analogous to our MPH1, and it is reported in the third column of Table 6B. The second step is analogous to our MPH2, and it is reported in the fourth column of Table 6B. Our results are very close to those reported in ref.[33] This is particularly interesting since we used a significantly different sampling protocol. First, they sampled from the NPT ensemble, while we sampled configurations from the NVT ensemble. Second, they did not use Ewald sum to compute long-range electrostatics. They adopt a neutral group based, finite cutoff distance for both electrostatics and Van der Waals interactions. The different cutoff scheme for Van der Waals interactions results in a significant difference between our long-range Van der Waals corrections (fifth column in Table 6A) and theirs (fifth column in Table 6B). Overall, our agreement with the simulations previously reported (second column of Table 6B)[33] and the experimental results to which they compare with[33, 85] (Table 6C) are in the expected range.

The expected result for the free energy difference over the cycle is, of course, 0kcal/mol. The free energy computed numerically over the complete cycle is:

$$\Delta F_{I-Q,solv} - \Delta F_{Q,solv} - \Delta F_{I-Q,vac} + \Delta F_{I,solv} = (0.02 \pm 0.12)\text{kcal/mol} \tag{2.54}$$

The correction for the angular term over the cycle is negligible (see Appendix A), therefore our final result is:

$$\oint dF = (0.02 \pm 0.12)\text{kcal/mol} \tag{2.55}$$

To show that the result is converged I report the cumulative running average of the cycle for the last 1ns of simulation in Figure 5.

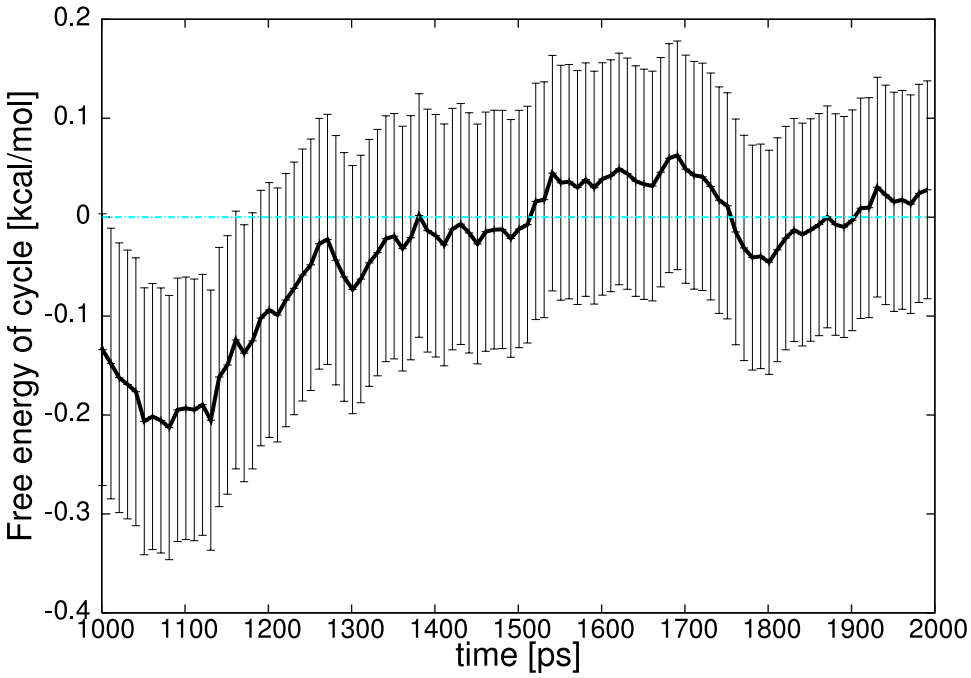


Figure 11: Running free energy as a function of time of the last 1ns of simulation. The black thick line reports free energy, the black thin lines report the error bar, which is shown only for one point out of ten. The light blue the expected value.

The numerical result is consistent with the analytical prediction of zero and is well within the statistical error bars. This calculation illustrates numerically the use of a thermodynamic cycle that retains the full strength of all the self-interactions.

Here, simulations at constant volume were carried out, while the experiment was done at constant pressure. The focus in the present investigation is to show that the complete cycle yields a free energy change of zero, not comparison to experiment, which

may also depend on the force field. Indeed the deviations of the calculations performed at constant pressure[33] from experiments are comparable to those reported here.

Influence of the size of the system

A possible dependence of the accuracy on the size of the periodic system was discovered while running the simulations in a smaller water box. The result of the cycle in this smaller box (29.35Å size) turned out to be different from zero and of the order of 0.2 ± 0.1kcal/mol. Further analysis is needed to detect the observed "small-size" effect, and was not yet carried out.

TI vs BAR

To test efficiency of BAR compared to TI, I analyzed the data of the MPH1 part of the mutation from ILE to GLN side chain analog in water. To test the efficiency, I measured the free energy difference as a function of the number of intermediate λ values used in the calculation. The largest number of intermediates corresponded to the whole set. Then, keeping the first and the last, I removed every other intermediate λ value, and I repeated this procedure three times. The results are in the following figure

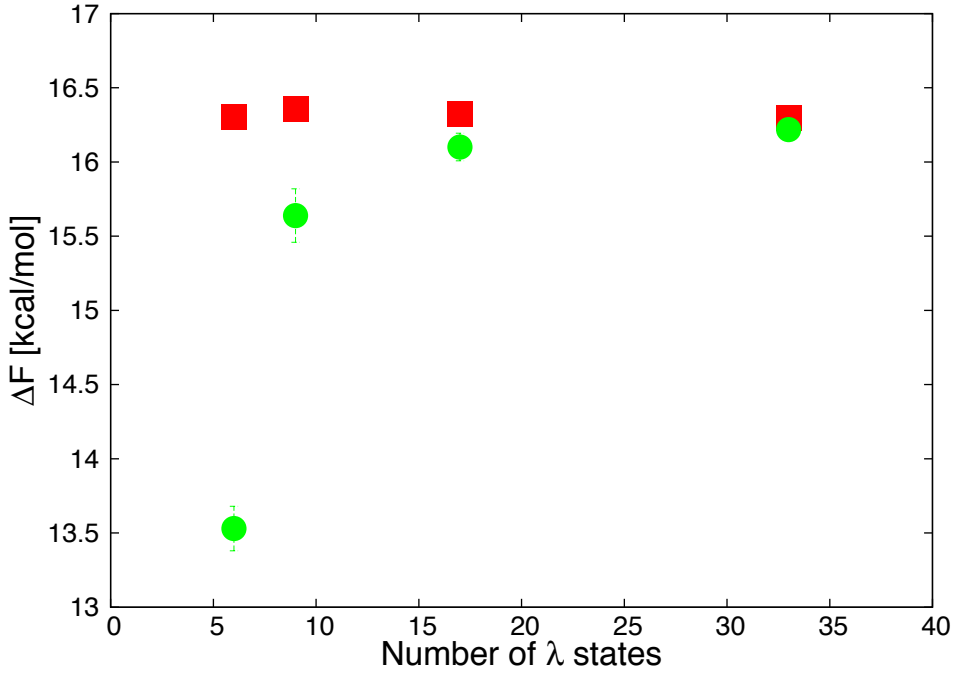


Figure 12: Value of the free energy difference measure for the MPH1 part of the ILE to GLN mutation in water as a function of the number of intermediate states used. In red the results for BAR are reported, in green the results for TI.

The value of the estimated free energy difference for TI (green) and BAR (red) is reported here. When the number of intermediates is decreased, the value of the free energy computed with BAR does not change significantly, while the value computed with TI rapidly looses accuracy. This is an evidence that, in agreement with ref.,[31] BAR gives a more efficient estimation of the free energy difference.

A NEW APPROACH FOR PME

A Faster Scheme

As I stated before in the discussion of the alchemical pathway used in the simulations presented so far, the scheme used for PME calculation (see Table 3 and Table 4) had a number of issues. First of all, we needed to perform the Ewald calculations 4 times per time step. Secondly the end states (a periodic lattice of the decoupled fragment, see Figure 6) do not sound very intuitive. To attempt to solve these issues, I devised a different PME scheme for the new version of the alchemical code in the latest issue of the MOIL code, named MOIL-OPT.[24] First, the alchemical code in MOIL-OPT works with the RESPA[86] scheme implemented in MOIL-OPT. RESPA allows the computation of the long-range interactions only once every few steps, typically 4 in MOIL-OPT. In this way, at least the reciprocal-space part of the code, which turns out to be the most time consuming, is performed once every 4 steps. This helps particularly with the alchemical code, which requires multiple calls per time step.

Secondly, I devised a different scheme for the calculation of the electrostatic interactions, which is described in the following Tables.

MPH1		
Step #1	Ewald YES	$(1-\lambda)[U_{EW}(P)+U_{EW}(N)+U_{EW}(P,N)]$
Step #2	Ewald YES	$\lambda U_{EW}(P)$
Step #3	NO Ewald	$\lambda U_{COU}(N)+U_{COU}(M)$
TOTAL	$U_{EW}(P) + (1-\lambda)U_{EW}(P,N) + (1-\lambda)U_{EW}(N) + \lambda U_{COU}(N)+U_{COU}(M)$	

Table 7: New scheme for electrostatic calculations in alchemical simulations, MPH1 stage of the alchemical substitution. Only 3 steps are now required, and only the first two involve a PME calculation, the last is a direct electrostatic calculation employing Coulomb law and involving only a few particles.

MPH2		
Step #1	Ewald YES	$(1-\lambda)U_{EW}(P)$
Step #2	Ewald YES	$\lambda[U_{EW}(P)+U_{EW}(M)+U_{EW}(P,M)]$
Step #3	NO Ewald	$(1-\lambda)U_{COU}(M)+U_{COU}(N)$
TOTAL	$U_{EW}(P) + \lambda U_{EW}(P,M) + \lambda U_{EW}(M) + (1-\lambda)U_{COU}(M)+U_{COU}(N)$	

Table 8: New scheme for electrostatic calculations in alchemical simulations, MPH2 stage of the alchemical substitution. Only 3 steps are now required, and only the first two involve a PME calculation, the last is a direct electrostatic calculation employing Coulomb law and involving only a few particles.

This new scheme requires only two calls of the PME routine, so it is much faster then before. It also requires a third step in which the electrostatic interactions involving exclusively the already decoupled fragment (M in MPH1, N in MPH2), and the fragment in the process of being annihilated (N in MPH1) or created (M in MPH2) are computed employing the standard Coulomb law instead of using PME. This makes this third step extremely fast. Also, it changes the end-states for the decoupled fragment, as pictorially illustrated in the next figure.

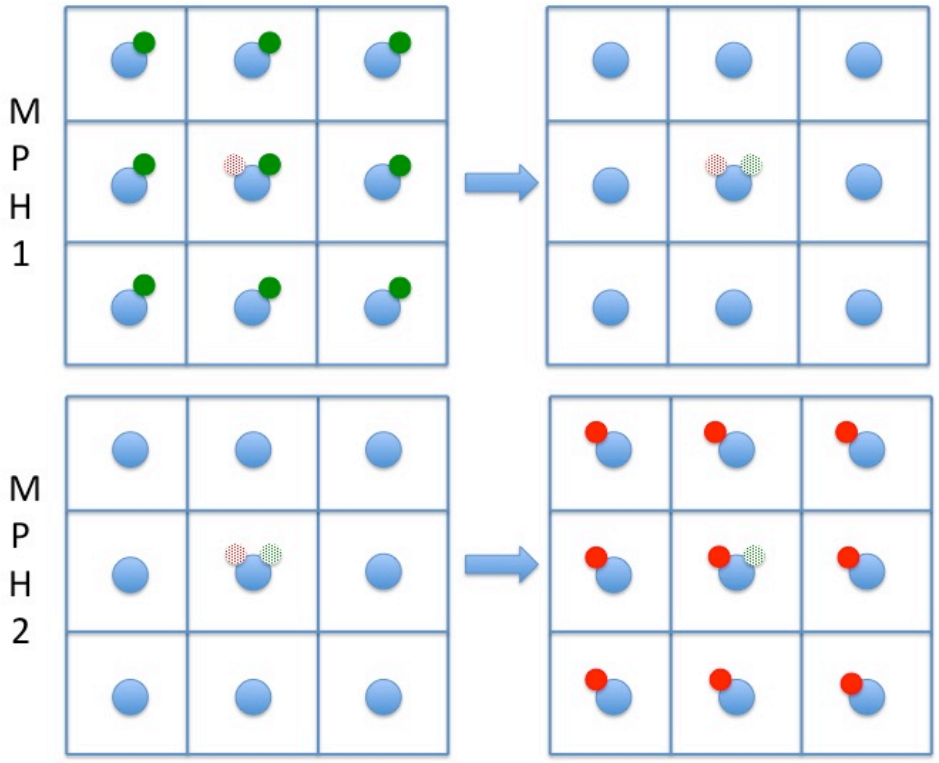


Figure 13: Pictorial illustration of the new scheme for electrostatic calculations in alchemical simulations. The different quadrants represent multiple copies of the system due to periodic boundary conditions. The blue circle represents the set of P particles, the red circle the M particles, and the green circle the N particles. A fully colored circle represents a species (P, M or N) that has all its interactions with the other copies and the other species "on". When the coloring of the circle is instead dotted, that means that the species (M or N) is interacting only with particles of the same species. In MPH1 the system at the beginning of the simulations is characterized by fully interacting P (blue) and N (green) particles, while the M particles (red, dotted) only see each other, and exclusively in the main copy of the system. At the end of MPH1, also the N particles (now green and dotted) are decoupled from the system and interact only with N particles, and only in the main copy of the system. This is also the starting point for MPH2. During MPH2, the M particles (red, dotted) are progressively coupled to the system and the end state has P (blue) and M (red) particles fully interacting, while the N particles (green, dotted) only see each other in the central copy.

The free energies for the solvation of ILEA and GLNA computed with this new algorithm are reported in the next table.

	PREVIOUS TOTAL (Table 6 Table 6)	TOTAL	MPH1	MPH2	LRC
$\Delta F_{I,solv}$	2.891±0.050	3.019±0.056	3.512±0.056	-0.0118±0.0022	-0.481
$\Delta F_{Q,solv}$	-8.354±0.054	-8.108±0.063	2.138±0.054	-9.714±0.032	-0.532

Table 9: Results for the solvation of ILEA and GLNA with the new scheme for PME (all the numbers are given in kcal/mol). In the first column the data from Table 6 is reported. The second column shows the total obtained with the new method. The third and fourth columns are the results of the MPH1 and MPH2 stages of the calculation. The last column reports the long-range correction (see Eq.(2.52)).

The results are not identical but very close (the differences in the totals are below 0.25kcal/mol).

Chapter 3: Derivation of The Smoluchovski Equation From MD using Milestoning

In this chapter[5] I discuss a method to derive the potential of mean force and the diffusion tensor in the space of coarse variables from MD data using Milestoning.[53] The Milestoning analysis of a MD trajectory yields a model in which the dynamics in coarse space is described as transitions between milestones. Once the positions of these milestones have been chosen, the Milestoning analysis yields the rate constants for transition between nearby milestones. The determination of the rates provides a Master Equation.

From the Master equation, we extract the potential of mean force and the diffusion tensor that appear in the Smoluchowski equation.[54a] This extraction follows the Kramers-Moyal expansion of the Master Equation.[54] This expansion is exact if the dynamics in the space of coarse variables can be described by Brownian equations of motion.

I used these expressions and extracted the potential of mean force and the diffusion tensor from a Milestoning analysis of three model systems.

FROM THE MASTER EQUATION TO THE FOKKER-PLANCK EQUATION

Consider an N-dimensional vector $\vec{x}$ of N coarse variables of interest. We assume that the time evolution of $\vec{x}$ is Markovian and that it can be described by an overdamped equation

$$d\vec{x} = \vec{a}(\vec{x})dt + \sqrt{2}\hat{b}(\vec{x})d\vec{W}(t) \tag{3.1}$$

[5] The work in this chapter was carried with my advisor Dr.Elber, who supervised the project.

where $\vec{W}(t)$ is a vector of N independent Wiener processes,[87] i.e. $\langle W_i(t)\rangle = 0$ and $\langle [W_i(t)-W_i(s)][W_j(t)-W_j(s)]\rangle = \delta_{ij}(t-s)$ with $t > s$, δ_{ij} the usual Kronecker delta, and the product $\hat{b}(\vec{x})d\vec{W}(t)$ should be integrated according to Ito calculus.[56]

The probability of finding the system in position $\vec{x}$ at time t is $p(\vec{x},t)$. The time evolution of the probability is described by a master equation:

$$\frac{\partial p(\vec{x},t)}{\partial t} = \int d\vec{y}\left[W(\vec{x}\,|\,\vec{y})\,p(\vec{y},t) - W(\vec{y}\,|\,\vec{x})\,p(\vec{x},t)\right] \tag{3.2}$$

The function $W(\vec{y}\,|\,\vec{x})$ is the transition probability per unit time, and expresses the rate at which the system goes from state $\vec{x}$ to state $\vec{y}$. It is always non-negative, and it is connected to the conditional probability by [88]

$$p(\vec{y},t+\tau\,|\,\vec{x},t) = \left[1 - D^{(0)}(\vec{x})\tau\right]\delta(\vec{x}-\vec{y}) + \tau W(\vec{y}\,|\,\vec{x}) + O(\tau^2) \tag{3.3}$$

where

$$D^{(0)}(\vec{x}) = \int d\vec{x}'\,W(\vec{x}'\,|\,\vec{x}) \tag{3.4}$$

Here, we also assumed that the system is time homogeneous, so the rate does not depend on the absolute time t. If we expand $W(\vec{x}\,|\,\vec{y})$ following the Kramers-Moyal (KM)[54b, 54c] technique, we obtain the following partial differential equation (see Appendix B):

$$\frac{\partial p(\vec{x},t)}{\partial t} = \sum_{n=1}^{\infty}\sum_{k_1=1}^{N}\cdots\sum_{k_n=1}^{N}(-1)^n\frac{\partial^n}{\partial x_{k_1}\cdots\partial x_{k_n}}D^{(n)}_{k_1\cdots k_n}(\vec{x})\,p(\vec{x},t) \tag{3.5}$$

where $D^{(n)}_{k_1\ldots k_n}(\vec{x})$ are the KM coefficients:

$$D^{(n)}_{k_1\cdots k_n}(\vec{x}) = \frac{1}{n!}\int d\vec{y}\left(y_{k_1}-x_{k_1}\right)\cdots\left(y_{k_n}-x_{k_n}\right)W(\vec{y}\,|\,\vec{x}) \tag{3.6}$$

Note that for the n-th KM coefficient there are n indexes k_i, each of which runs from 1 to N. This means that the first KM coefficient is a vector of size N, the second is a matrix of size NxN etc.

In general, this expansion cannot be truncated, but in some special cases it can be terminated at the second order. Pawula's theorem[54a, 54c] states that if any of the KM

coefficients of order higher than the second is equal to zero, then all the coefficients but the first and the second must be equal to zero. With this truncation we obtain the Fokker-Plank equation:

$$\frac{\partial p(\vec{x},t)}{\partial t} = -\sum_{k_1=1}^{N} \frac{\partial}{\partial x_{k_1}} D_{k_1}^{(1)}(\vec{x})\, p(\vec{x},t) + \sum_{k_1=1}^{N}\sum_{k_2=1}^{N} \frac{\partial^2}{\partial x_{k_1}\, \partial x_{k_2}} D_{k_1 k_2}^{(2)}(\vec{x})\, p(\vec{x},t) \tag{3.7}$$

The vector of the first order KM coefficients is defined

$$\vec{A}(\vec{x}) = \begin{pmatrix} D_1^{(1)}(\vec{x}) \\ . \\ . \\ . \\ D_N^{(1)}(\vec{x}) \end{pmatrix} \tag{3.8}$$

and the matrix of the second order KM coefficients is

$$\hat{D}(\vec{x}) = \begin{pmatrix} D_{1,1}^{(2)}(\vec{x}) & D_{1,2}^{(2)}(\vec{x}) & . & . & . & D_{1,N}^{(2)}(\vec{x}) \\ D_{2,1}^{(2)}(\vec{x}) & D_{2,2}^{(2)}(\vec{x}) & . & . & . & D_{2,N}^{(2)}(\vec{x}) \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ D_{N,1}^{(2)}(\vec{x}) & . & . & . & . & D_{N,N}^{(2)}(\vec{x}) \end{pmatrix} \tag{3.9}$$

The N-dimensional Fokker-Plank equation becomes

$$\frac{\partial p(\vec{x},t)}{\partial t} = -\vec{\nabla}\cdot\left[\vec{A}(\vec{x})\, p(\vec{x},t)\right] + \vec{\nabla}\vec{\nabla}^{T} : \left[\hat{D}(\vec{x})\, p(\vec{x},t)\right] \tag{3.10}$$

The vector $\vec{A}(\vec{x})$ is associated with the drift of the probability distribution, while the matrix $\hat{D}(\vec{x})$ with its spread (diffusion).

We require that the diffusion matrix $\hat{D}(\vec{x})$, other than being symmetric (see definition in equation (3.6)), is positive definite. In this case, there exists a symmetric matrix $\hat{b}(\vec{x})$ such that:

$$D_{ij}(\vec{x}) = \sum_{k=1}^{N} b_{ik}(\vec{x})\, b_{jk}(\vec{x}) \tag{3.11}$$

If the time evolution of the N variables of interest is given by Eq.(3.1), it is possible to show that only the first two KM coefficients are different from zero [54a, 56] (see Appendix C). Therefore the truncation of the Master Equation expansion at the second order is exact, and equation (3.10) describes the evolution of the probability distribution, with $\vec{a}(\vec{x}) = \vec{A}(\vec{x})$ and where the relationship between $\hat{b}(\vec{x})$ and $\hat{D}(\vec{x})$ is given in (3.11).

Following a statistical mechanical recipe for a system in contact with a thermal reservoir, the equilibrium probability distribution of the N variables of interest is given by:

$$p_{eq}(\vec{x}) = \frac{e^{-\beta U(\vec{x})}}{\int e^{-\beta U(\vec{x})} d\vec{x}} \tag{3.12}$$

where $U(\vec{x})$ is the potential of mean force (averaged over the degrees of freedom that are not described by the N variables of interest), and $\beta = 1/k_B T$, with T temperature and k_B the Boltzmann constant. The Fokker-Planck equation that meets these conditions is:

$$\begin{aligned} \frac{\partial p(\vec{x},t)}{\partial t} = & -\sum_{i=1}^{N} \frac{\partial}{\partial x_i} \left[-\beta \sum_{j=1}^{N} D_{ij}(\vec{x}) \frac{\partial}{\partial x_j} U(\vec{x}) + \sum_{j=1}^{N} \frac{\partial}{\partial x_j} D_{ij}(\vec{x}) \right] p(\vec{x},t) \\ & + \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{\partial^2}{\partial x_i \partial x_j} D_{ij}(\vec{x}) p(\vec{x},t) \end{aligned} \tag{3.13}$$

Using the KM coefficients (3.6), we can compute

$$D_{ij}(\vec{x}) = \frac{1}{2} \int d\vec{x}' (x'_i - x_i)(x'_j - x_j) W(\vec{x}' \mid \vec{x}) \tag{3.14}$$

$$\sum_{j=1}^{N} \beta D_{ij}(\vec{x}) F_j(\vec{x}) + \sum_{j=1}^{N} \frac{\partial}{\partial x_j} D_{ij}(\vec{x}) = \int d\vec{x}' (x'_i - x_i) W(\vec{x}' \mid \vec{x}) \tag{3.15}$$

So far, the only assumption that we made is that the dynamics of the variables of interest is overdamped.

MILESTONING AND THE KRAMERS-MOYAL COEFFICIENTS

The Milestoning equations[53] describe the dynamics in the space of coarse variables using three quantities. The first quantity is $p_\alpha(t)$, the probability of being at milestone α at time t. The second quantity is the flux across milestone α at time t, $q_\alpha(t)$, which is the probability of crossing that milestone exactly at time t. Finally, we consider the kernel $K_{\alpha\beta}(t)$, which defines the probability that a trajectory that crosses milestone α then crosses milestone β exactly after time t. These three quantities can be arranged in the two defining equations for Milestoning:

$$p_\alpha(t) = \int_0^t q_\alpha(t') \left[1 - \sum_\beta \int_0^{t-t'} K_{\alpha\beta}(\tau) d\tau \right] dt' \tag{3.16}$$

$$q_\alpha(t) = p_\alpha(0)\delta(t-\varepsilon) + \sum_\beta \int_0^t q_\beta(t') K_{\beta\alpha}(t-t') dt' \tag{3.17}$$

Equation (3.16) states that the probability that at time t milestone α was the last milestone to be crossed is equal to the probability of crossing milestone α at some previous time t', multiplied by the probability of staying in α for the remaining time $t-t'$ (the term is square brackets). Equation (3.17) instead states that the probability of crossing milestone α at time t is equal to the probability of crossing some other milestone β at an earlier time t', multiplied the probability of crossing milestone α exactly after $t-t'$ from the previous crossing. The initial condition has to be considered as well, stating that is possible to have a crossing at time 0 if a trajectory starts from milestone α. Note also that, the conservation of probability imposes that:

$$\sum_\beta \int_0^\infty K_{\alpha\beta}(t) dt = 1 \tag{3.18}$$

which means that a trajectory starting at milestone α at time 0 has to go somewhere.

To solve these equations we employ a few properties of the Laplace transforms that I describe here. Given a function $F(t)$, its Laplace transform is defined as:

$$\tilde{F}(u) = \int_0^\infty e^{-ut} F(t)\, dt \tag{3.19}$$

The Laplace transform of a derivative of a function can be readily obtained integrating by parts:

$$\int_0^\infty e^{-ut} \frac{d}{dt} F(t) = -F(0) + u\tilde{F}(u) \tag{3.20}$$

Given another function $G(t)$, the convolution of this function with $F(t)$ is defined as:

$$F \circ G(t) = \int_0^t F(t')G(t-t')\, dt' \tag{3.21}$$

The Laplace transform of this convolution can be obtained with an appropriate change of variables. Indeed we have:

$$\int_0^\infty e^{-ut} F \circ G(t)\, dt = \int_0^\infty dt\ e^{-ut} \int_0^t F(t')G(t-t')\, dt' \tag{3.22}$$

The change of variables is pictorially represented in Figure 14.

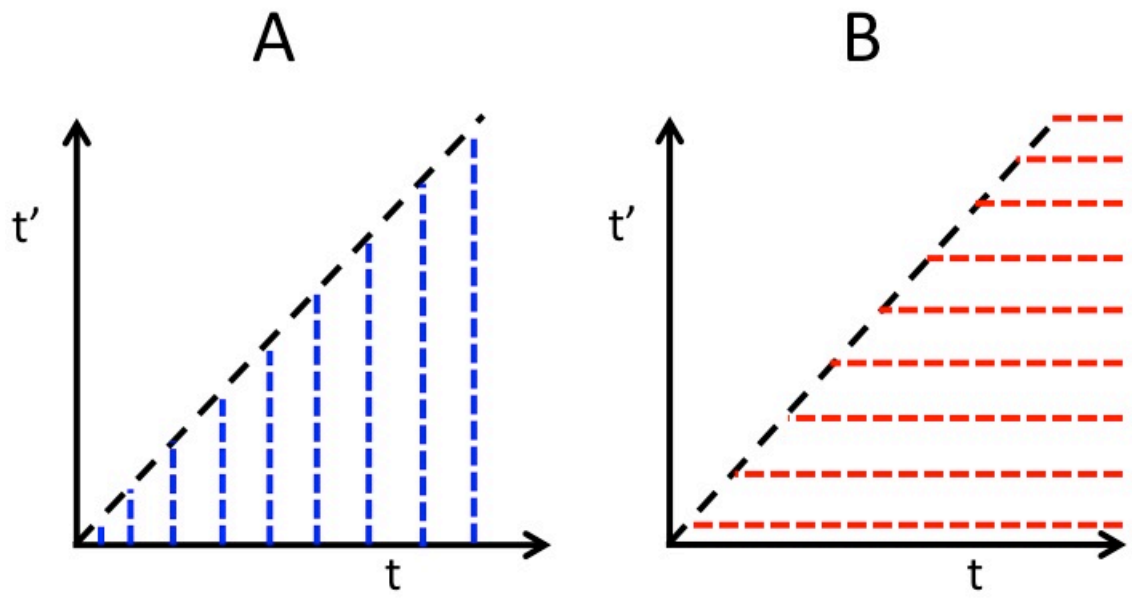


Figure 14 Pictorial illustration of the change of variables used to compute the Laplace transform of a convolution (see Eq.(3.24)**)** and of an integral (see Eq.(3.25)). (A) The integration is performed on the plane marked by the blue dashed lines. The variable t goes from 0 to infinity, while the variable t' goes from 0 to t, as highlighted by the direction of the dashed blued lines. (B) The same integration domain as in (A), but in this case it is spanned in a different way. The variable t' goes from 0 to infinity, while the variable t goes from t' to infinity, as highlighted by the direction of the dashed red lines.

Figure 14A represents the previous integral, in which t' varies between 0 and t, which in turn goes from 0 to infinity. The area covered with blue dashed lines represent the area where the integral is performed. The direction of the lines indicates the "direction" of the integration. The same area can be covered in a different way, as in Fig.

B. In Figure 14B t' spans the range from 0 to infinity, while t goes from t' to infinity. If we apply this transformation of variables to Eq. (3.22) we get:

$$\int_0^\infty dt\ e^{-ut}\int_0^t F(t')G(t-t')dt' = \int_0^\infty dt'\int_{t'}^\infty e^{-ut}F(t')G(t-t')dt \tag{3.23}$$

If we now define $\tau = t - t'$, change the variables in the second integral and substitute in the exponent t with $t'+\tau$, it is readily shown that:

$$\int_0^\infty e^{-ut}F\circ G(t)dt = \tilde{F}(u)\tilde{G}(u) \tag{3.24}$$

The Laplace transform of the integral of a function can be obtained with exactly the same transformation of variables:

$$\int_0^\infty dt\ e^{-ut}\int_0^t dt'F(t') = \int_0^\infty dt'F(t')\int_{t'}^\infty dt\ e^{-ut} = \frac{1}{u}\int_0^\infty dt'F(t')e^{-ut'} = \frac{\tilde{F}(u)}{u} \tag{3.25}$$

Using these properties, we can readily carry out the Laplace transforms of Eq.(3.16) and Eq.(3.17):

$$u\tilde{p}_\alpha(u) = \tilde{q}_\alpha(u)\left[1-\sum_\beta \tilde{K}_{\alpha\beta}(u)\right] \tag{3.26}$$

$$\tilde{q}_\alpha(u) = p_\alpha(0) + \sum_\beta \tilde{q}_\beta(u)\tilde{K}_{\beta\alpha}(u) \tag{3.27}$$

Using Eq.(3.26) and Eq.(3.27) it is possible to derive thermodynamic information, such as the stationary probability of being on each milestone,[53b] and kinetic information, such as the mean first passage time to go from a milestone to another one.[53b] Here, these equations will be used to establish the connection between the Milestoning formalism, the Generalized Master Equation (GME),[53a] the Master Equation (ME),[53b] and to show what formula with should use to derive the rates in the ME from Milestoning.[53b]

To do so, we start by a simple manipulation of the Eq.(3.26):

$$u\tilde{p}_\alpha(u) = \tilde{q}_\alpha(u) - \tilde{q}_\alpha(u)\sum_\beta \tilde{K}_{\alpha\beta}(u) = p_\alpha(0) + \sum_\beta \tilde{q}_\beta(u)\tilde{K}_{\beta\alpha}(u) - \tilde{q}_\alpha(u)\sum_\beta \tilde{K}_{\alpha\beta}(u)$$

where we used Eq.(3.27) to remove the first $\tilde{q}_\alpha(u)$. We can now use Eq.(3.26) to get rid of the flux and have an expression for the probability alone:

$$u\tilde{p}_\alpha(u) - p_\alpha(0) = \sum_\beta \tilde{p}_\beta(u) \frac{u\tilde{K}_{\beta\alpha}(u)}{1 - \sum_\gamma \tilde{K}_{\beta\gamma}(u)} - \tilde{p}_\alpha(u) \sum_\beta \frac{u\tilde{K}_{\alpha\beta}(u)}{1 - \sum_\gamma \tilde{K}_{\alpha\gamma}(u)} \tag{3.28}$$

Now, the GME is:

$$\frac{dp_\alpha(t)}{dt} = \sum_\beta \int_0^t dt' W_{\beta\alpha}(t-t') p_\beta(t') - \sum_\beta \int_0^t dt' W_{\alpha\beta}(t-t') p_\alpha(t') \tag{3.29}$$

The Laplace transform of the GME is:

$$u\tilde{p}_\alpha(u) - p_\alpha(0) = \sum_\beta \tilde{p}_\beta(u) \tilde{W}_{\beta\alpha}(u) - \tilde{p}_\alpha(u) \sum_\beta \tilde{W}_{\alpha\beta}(u) \tag{3.30}$$

The two equations (3.28) and (3.30) describe the same quantity, the Laplace transform of the probability $p_\alpha(t)$. Therefore, comparing the two, we arrive to the following:

$$\tilde{W}_{\alpha\beta}(u) = \frac{u\tilde{K}_{\alpha\beta}(u)}{1 - \sum_\gamma \tilde{K}_{\alpha\gamma}(u)} \tag{3.31}$$

This equation provides the connection between the transition probability per unit time and the Milestoning kernel.

In the case in which the system is Markovian, the GME becomes the ME:

$$\frac{dp_\alpha(t)}{dt} = \sum_\beta W_{\beta\alpha} p_\beta(t) - \sum_\beta W_{\alpha\beta} p_\alpha(t) \tag{3.32}$$

This equation is exactly the same as the one in Eq.(3.2), with the only difference that now the state space is discrete instead of being continuous.

The ME can be derived from the GME if

$$W_{\alpha\beta}(t) = W_{\alpha\beta}\delta(t) \tag{3.33}$$

This means that the system has no memory of where it was before, and the probability of transiting is not affected by the past. This is an approximation. On the other

hand, if we chose properly $W_{\alpha\beta}$ we get two important properties from the Markov model. First of all, it preserves the equilibrium distribution obtained with Milestoning.[53b] Secondly, it preserves the first moment of the first passage time distribution (i.e. the mean first passage time) but not higher moments.[89] The choice that guarantees this is:

$$W_{\alpha\beta} = \int_0^\infty W_{\alpha\beta}(t)\,dt \tag{3.34}$$

From the definition of Laplace transform (Eq.(3.19)), it is readily seen that:

$$W_{\alpha\beta} = \int_0^\infty W_{\alpha\beta}(t)\,dt = \tilde{W}_{\alpha\beta}(0) \tag{3.35}$$

We can now use Eq.(3.31) to get:

$$\tilde{W}_{\alpha\beta}(0) = \lim_{u\to 0} \frac{u\tilde{K}_{\alpha\beta}(u)}{1-\sum_\gamma \tilde{K}_{\alpha\gamma}(u)} \tag{3.36}$$

Recalling the definition of Laplace transform (3.19), we can rewrite the denominator of Eq.(3.36) as:

$$1-\sum_\beta \tilde{K}_{\alpha\beta}(u) = 1-\sum_\beta \int_0^\infty e^{-ut} K_{\alpha\beta}(t)\,dt \tag{3.37}$$

Since we are looking at limit for vanishing u, we can expand the exponent in the integral and, recalling Eq.(3.18) we get:

$$1-\sum_\beta \int_0^\infty \left[1-ut+O\left(u^2\right)\right] K_{\alpha\beta}(t)\,dt = u\sum_\beta \int_0^\infty tK_{\alpha\beta}(t)\,dt + O\left(u^2\right) \tag{3.38}$$

If we plug this result back in the denominator of Eq.(3.36), in the limit for vanishing u we get:

$$W_{\alpha\beta} = \frac{\int_0^\infty K_{\alpha\beta}(t)\,dt}{\sum_\beta \int_0^\infty tK_{\alpha\beta}(t)\,dt} \tag{3.39}$$

We are now left with the interpretation of this result. The numerator is the probability of going from milestone α to milestone β at any time. Let's call it $p(\alpha,\beta)$. The denominator is the probability that the transition from milestone α to any milestone β happens exactly after a time t from the moment that milestone α was crossed, times that time t, integrated on time. This is the average residence time on milestone α, and let's call it $\langle\tau(\alpha)\rangle$. Interestingly, this formula was also derived starting from a Markovian system using a maximum likelihood argument.[61]

How can we practically compute the KM coefficients from a MD simulation? What we need is a way of estimating the rates W from the trajectories obtained in MD. To do so, we use Milestoning.[53] We divide our space using milestones, as illustrated in Figure 15

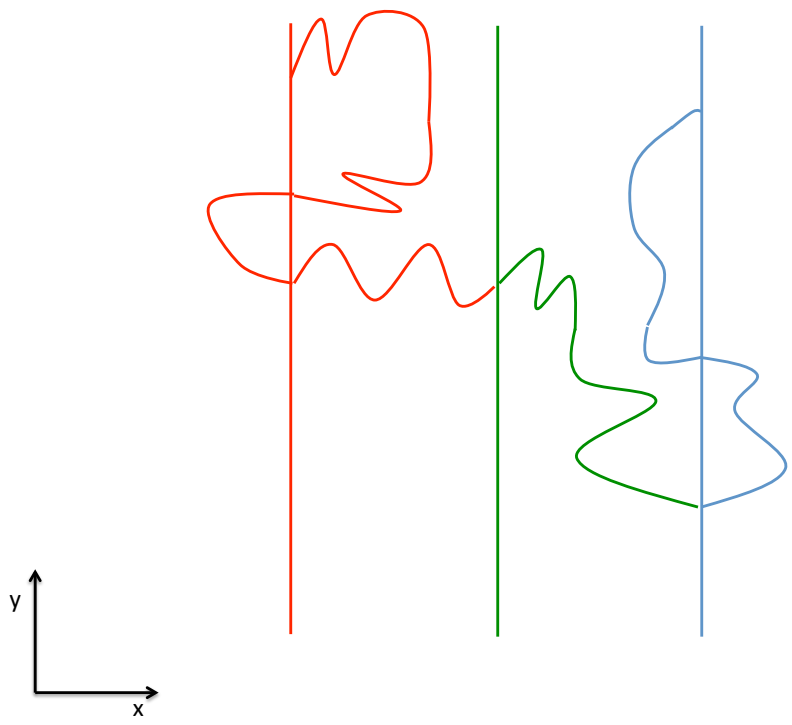


Figure 15: The trajectory crosses milestones in the x direction (red, green and blue vertical lines). Every time that the trajectory crosses a milestone it belongs to that milestone (i.e. it is labeled with the specific color in the figure) until it crosses the next, or the previous milestone.

Here we divided a two dimensional system in N_{MS} milestones defined by lines parallel to the y-axis. The milestones are denoted by Greek letters in sequence, from 1 to N_{MS} and are used to label fragments of a trajectory. Every time a trajectory crosses a milestone it is assigned to that milestone until it crosses another milestone. In this way, the trajectory is coarse grained to hops between a series of milestones. Instead of $\left(x(t),y(t)\right)$ - the full coordinate set as a function of time - we have

$\alpha\left(t_\alpha \le t < t_\beta\right), \beta\left(t_\beta \le t < t_\gamma\right), \gamma\left(t_\gamma \le t\right), \ldots$ where t_ν is the time in which ν was crossed by the trajectory $\left(x(t), y(t)\right)$. Alternatively, we can run multiple trajectories starting from each milestone and terminate them when they cross a neighboring milestone. Both approaches give us the kinetic information that we need, namely the local transition probabilities and local transition time between nearby milestones.

We record the average amount of time that it takes for a trajectory to leave milestone α as $\langle \tau(\alpha) \rangle$ and the probability of going from milestones α to milestone β as $p(\alpha, \beta)$. Therefore, following Eq. (3.39), the rate of going from milestone α to milestone β is: [53b]

$$k(\alpha;\beta) = \frac{p(\alpha,\beta)}{\langle \tau(\alpha) \rangle} \tag{3.40}$$

Once we extract the rate coefficients, we can calculate the force and the space-dependent diffusion using Eq. (3.14) and (3.15). The KM coefficients that we need are:

$$\beta D(x) F(x) + \frac{d}{dx} D(x) = \int dx' (x' - x) W(x' \mid x) \tag{3.41}$$

$$D(x) = \frac{1}{2} \int dx' (x' - x)^2 W(x' \mid x) \tag{3.42}$$

Using the coarse and discrete description provided with Milestoning data, we obtain the rate coefficients between nearby milestones. This means that we should substitute in equations (3.41) and (3.42) $W(x' \mid x)$ with $W(\beta \mid \alpha) = W\left(x' = x_\beta \mid x = x_\alpha\right)$, where x_α and x_β are the position in space of milestone α and β, respectively. Since we only have transitions between neighboring milestones, $W(\beta \mid \alpha)$ is:

$$W(\beta \mid \alpha) = k(\alpha;\alpha+1)\delta\left[x' - \left(x_\alpha + \Delta_x\right)\right] + k(\alpha;\alpha-1)\delta\left[x' - \left(x_\alpha - \Delta_x\right)\right]$$

(3.43)

where the rate coefficients k are given by (3.40), and Δ_x is the distance between two milestones. If we plug this expression in (3.41) and (3.42) we get:

$$\beta D(\alpha)F(\alpha)+\frac{dD(\alpha)}{dx}=\Delta_x\left[k(\alpha;\alpha+1)-k(\alpha;\alpha-1)\right] \tag{3.44}$$

$$D(\alpha)=\frac{\Delta_x^2}{2}\left[k(\alpha;\alpha+1)+k(\alpha;\alpha-1)\right] \tag{3.45}$$

The derivative in space of the diffusion coefficient may be approximated numerically, for instance by

$$\frac{dD(\alpha)}{dx}\approx\frac{D(\alpha+1)-D(\alpha-1)}{2\Delta_x} \tag{3.46}$$

The one-dimensional case is the simplest one. Often though, we need to account for multiple coarse variables. These variables may not be independent, and the analysis of their correlation is of interest. Let's consider a two-dimensional case, where the two coarse variables are (x,y). We need to compute the following KM coefficients:

$$\begin{aligned}&\beta D_{xx}(x,y)F_x(x,y)+\beta D_{xy}(x,y)F_y(x,y)+\frac{\partial}{\partial x}D_{xx}(x,y)+\frac{\partial}{\partial y}D_{yx}(x,y)\\&=\int dx'dy'(x'-x)W(x',y'\mid x,y)\end{aligned} \tag{3.47}$$

$$D_{xx}(x,y)=\frac{1}{2}\int dx'dy'(x'-x)^2W(x',y'\mid x,y) \tag{3.48}$$

$$\begin{aligned}&\beta D_{yx}(x,y)F_x(x,y)+\beta D_{yy}(x,y)F_y(x,y)+\frac{\partial}{\partial x}D_{xy}(x,y)+\frac{\partial}{\partial y}D_{yy}(x,y)\\&=\int dx'dy'(y'-y)W(x',y'\mid x,y)\end{aligned} \tag{3.49}$$

$$D_{yy}(x,y)=\frac{1}{2}\int dx'dy'(y'-y)^2W(x',y'\mid x,y) \tag{3.50}$$

$$D_{xy}(x,y)=\frac{1}{2}\int dx'dy'(x'-x)(y'-y)W(x',y'\mid x,y) \tag{3.51}$$

The KM coefficients in equations (3.47)-(3.51) can be classified in three types. The first two (equations (3.47)-(3.48)) are integrals of the transition rate $W(x',y'\mid x,y)$ multiplied by powers of $(x'-x)$ and are called case (a). The next two (equations (3.49)-(3.50)) involve instead powers of $(y'-y)$ and are called case (b). Finally, the last one (equation (3.51)) is an integral of the product $(x'-x)(y'-y)$ and is called case (c).

To simplify these integrals, we use sets of milestones with different orientations. The first two integrals are simplified if the transitions that we count happen between milestones with fixed distance projected on the x-axis, i.e. $(x'-x)=\pm\Delta_x$. Similar simplified equations (3.49)-(3.50) are obtained if the sets of milestones are with a fixed distance along the y-axis, i.e. $(y'-y)=\pm\Delta_y$. A related reasoning is not possible for the last KM equation (3.51), for which we need a slightly more complicated equation.

We proceed to explain each of these three different cases.

To compute the KM integrals in equations (3.47)-(3.48), we discretize the trajectory shown in Figure 15 as illustrated in Figure 16:

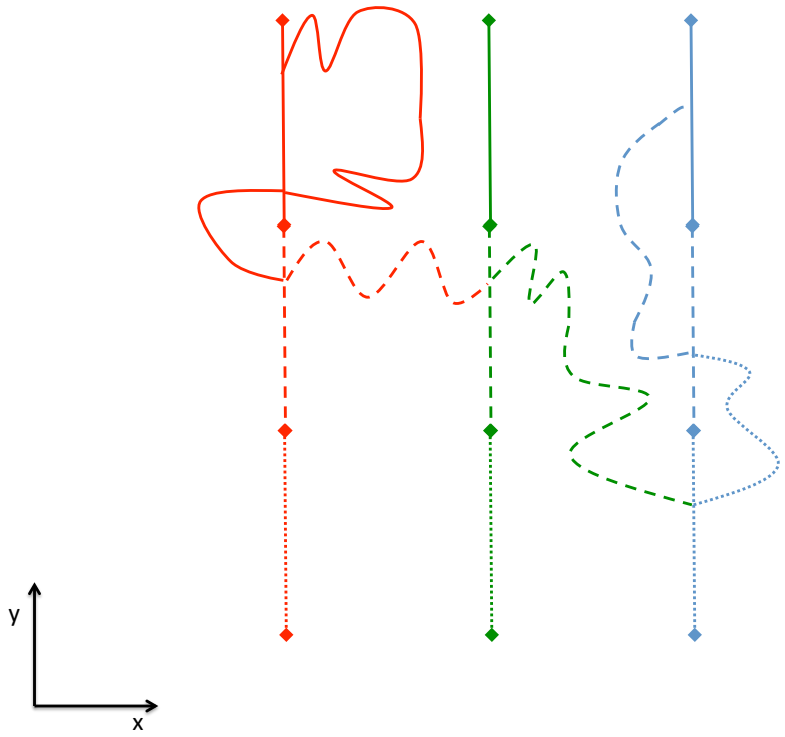


Figure 16: milestones for case (a). All the milestones are parallel to the y-axis, but in contrast to the one-dimensional case (Figure 15), they are not lines. Instead they are segments whose ends are rhomboidal arrows. The transition between different segments is a jump between two milestones.

Each of the N_{MSX} milestones of Figure 15 is now divided into N_{MSY} milestones (represented in Figure 16 by continuous, dashed and dotted lines) of size Δ_y. The idea of partitioning the milestones in this way is of Hawk and Makarov.[90] The N_{MSY} milestones are labeled with increasing integers, from 1 to N_{MSY}, from bottom to top. We need two indices to identify a milestone: one for the position in x (the color in Figure 16, in what follows identified with a Greek letter and the subscript x) and one for the position in y

(the styles of the lines in Figure 16, in what follows identified with a Greek letter and the subscript y). So, a milestone will be identified by the vector $\vec{\alpha}=\left(\alpha_x,\alpha_y\right)$. Note that each milestone has $3N_{MSY}-1$ neighbors. Following Fig. 2, each milestone is defined as the set of points (x,y) that are such that $\left\{x=x_{\alpha_x},\ y_{\alpha_y}-\frac{\Delta_y}{2}\le y<y_{\alpha_y}+\frac{\Delta_y}{2}\right\}$. The Milestoning analysis of the trajectory gives us the rate of going from any point at milestone $\vec{\alpha}$ to any point at milestone $\vec{\beta}$. Since the milestones that are accessible from milestone $\vec{\alpha}$ are those belonging to the set $\left\{\left(\beta_x,\beta_y\right)\,\middle|\,\alpha_x-1\le\beta_x\le\alpha_x+1 \text{ and } \left(\beta_x,\beta_y\right)\ne\left(\alpha_x,\alpha_y\right)\right\}$, the transition probability per unit time obtained with Milestoning $W^{(1)}\left[\left(x',y'\right)\middle|\left(x,y\right)\in\vec{\alpha}\right]$ is

$$\begin{aligned}&W^{(1)}\left[\left(x',y'\right)\middle|\left(x,y\right)\in\vec{\alpha}\right]=\\&\sum_{\beta_y=1}^{N_{MSY}}\Big\{k\left(\alpha_x,\alpha_y;\alpha_x+1,\beta_y\right)\delta\left[x'-\left(x_{\alpha_x}+\Delta_x\right)\right]\Upsilon_{\beta_y}\left(y'\right)\\&+k\left(\alpha_x,\alpha_y;\alpha_x-1,\beta_y\right)\delta\left[x'-\left(x_{\alpha_x}-\Delta_x\right)\right]\Upsilon_{\beta_y}\left(y'\right)\\&+k\left(\alpha_x,\alpha_y;\alpha_x,\beta_y\right)\delta\left(x'-x_{\alpha_x}\right)\Upsilon_{\beta_y}\left(y'\right)\left[1-\delta_{\alpha_y,\beta_y}\right]\Big\}\end{aligned}\tag{3.52}$$

where $\Upsilon_{\beta_y}\left(y'\right)$ is the following function

$$\Upsilon_{\beta_y}\left(y\right)=\begin{cases}\dfrac{1}{\Delta_y} & \text{if } \ y_{\beta_y}-\dfrac{1}{2}\Delta_y\le y<y_{\beta_y}+\dfrac{1}{2}\Delta_y\\ 0 & \text{otherwise}\end{cases}\tag{3.53}$$

Note that we labeled the transition probability per unit time $W^{(1)}\left(\vec{\beta}\,|\,\vec{\alpha}\right)$ with the suffix (1) to highlight that this rate was obtained with the milestones in Figure 16. If we plug equation (3.52) in equations (3.47)-(3.48) we get:

$$\begin{aligned}&\beta D_{xx}^{(1)}\left(\alpha_x,\alpha_y\right)F_x^{(1}\left(\alpha_x,\alpha_y\right)+\beta D_{xy}^{(1)}\left(\alpha_x,\alpha_y\right)F_y^{(1}\left(\alpha_x,\alpha_y\right)+\frac{\partial D_{xx}^{(1)}\left(\alpha_x,\alpha_y\right)}{\partial x}+\frac{\partial D_{xy}^{(1)}\left(\alpha_x,\alpha_y\right)}{\partial y}\\&=\sum_{\beta_y=1}^{N_{MSY}}\Delta_x\left[k\left(\alpha_x,\alpha_y;\alpha_x+1,\beta_y\right)-k\left(\alpha_x,\alpha_y;\alpha_x-1,\beta_y\right)\right]\end{aligned}\tag{3.54}$$

$$D_{xx}^{(1)}\left(\alpha_x,\alpha_y\right)=\frac{1}{2}\sum_{\beta_y=1}^{N_{MSY}}\Delta_x^2\left[k\left(\alpha_x,\alpha_y;\alpha_x+1,\beta_y\right)+k\left(\alpha_x,\alpha_y;\alpha_x-1,\beta_y\right)\right]\tag{3.55}$$

A similar strategy can be devised to compute the KM coefficients in equations (3.49)-(3.50), and the same trajectory is re-analyzed with a new set of milestones parallel to the x axis as in Figure 17

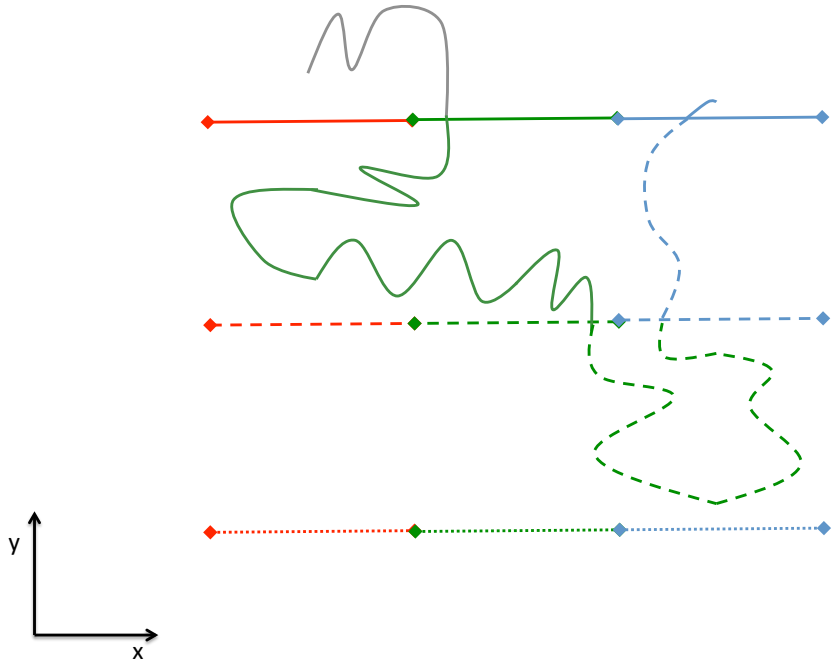


Figure 17 Same trajectory as Figure 15 and Figure 16, but this time discretized to show transitions between milestones parallel to the x-axis. As in Figure 16, the milestones are segments whose ends are represented as rhomboidal arrows.

In this case, there are N_{MSY} milestones parallel to the x-axis, separated by Δ_y. Each of these milestones is then divided into N_{MSX} milestones of size Δ_x. A milestone $\vec{\alpha} = (\alpha_x, \alpha_y)$ is identified by the set of points (x, y) such that

$\left\{x_{\alpha_x}-\frac{\Delta_x}{2}\le x<x_{\alpha_x}+\frac{\Delta_x}{2},\ y=y_{\alpha_y}\right\}$. As before, the $3N_{MSX}-1$ milestones that are neighbor to milestone $\vec{\alpha}$ are those belonging to the set $\left\{\left(\beta_x,\beta_y\right)\middle|\alpha_y-1\le\beta_y\le\alpha_y+1 \text{ and } \left(\beta_x,\beta_y\right)\ne\left(\alpha_x,\alpha_y\right)\right\}$. We can now define the transition probability per unit time, obtained with a Milestoning analysis performed according to Figure 17, $W^{(2)}\left[\left(x',y'\right)\middle|\left(x,y\right)\in\vec{\alpha}\right]$, which is:

$$\begin{aligned}W^{(2)}\left[\left(x',y'\right)\middle|\left(x,y\right)\in\vec{\alpha}\right]=\sum_{\beta_x=1}^{N_{MSX}}\Big\{&k\left(\alpha_x,\alpha_y;\beta_x,\alpha_y+1\right)\delta\left[y'-\left(y_{\alpha_y}+\Delta_y\right)\right]\Upsilon_{\beta_x}\left(x'\right)\\&+k\left(\alpha_x,\alpha_y;\beta_x,\alpha_y-1\right)\delta\left[y'-\left(y_{\alpha_y}-\Delta_y\right)\right]\Upsilon_{\beta_x}\left(x'\right)\\&+k\left(\alpha_x,\alpha_y;\beta_x,\alpha_y\right)\delta\left(y'-y_{\alpha_y}\right)\Upsilon_{\beta_x}\left(x'\right)\left[1-\delta_{\alpha_x,\beta_x}\right]\Big\}\end{aligned} \tag{3.56}$$

where $\Upsilon_{\beta_x}\left(x'\right)$ is defined

$$\Upsilon_{\beta_x}\left(x\right)=\begin{cases}\dfrac{1}{\Delta_x} & \text{if } \ x_{\beta_x}-\dfrac{1}{2}\Delta_x\le x<x_{\beta_x}+\dfrac{1}{2}\Delta_x\\ 0 & \text{otherwise}\end{cases} \tag{3.57}$$

If we plug equation (3.56) in equations (3.49)-(3.50) we get:

$$\begin{aligned}&\beta D_{yx}^{(2)}\left(\alpha_x,\alpha_y\right)F_x^{(2)}\left(\alpha_x,\alpha_y\right)+\beta D_{yy}^{(2)}\left(\alpha_x,\alpha_y\right)F_y^{(2)}\left(\alpha_x,\alpha_y\right)+\frac{\partial D_{yx}^{(2)}\left(\alpha_x,\alpha_y\right)}{\partial x}+\frac{\partial D_{yy}^{(2)}\left(\alpha_x,\alpha_y\right)}{\partial y}\\&=\sum_{\beta_x=1}^{N_{MSX}}\Delta_y\left[k\left(\alpha_x,\alpha_y;\beta_x,\alpha_y+1\right)-k\left(\alpha_x,\alpha_y;\beta_x,\alpha_y-1\right)\right]\end{aligned} \tag{3.58}$$

$$D_{yy}^{(2)}\left(\alpha_x,\alpha_y\right)=\frac{1}{2}\sum_{\beta_x=1}^{N_{MSX}}\Delta_y^2\left[k\left(\alpha_x,\alpha_y;\beta_x,\alpha_y+1\right)+k\left(\alpha_x,\alpha_y;\beta_x,\alpha_y-1\right)\right] \tag{3.59}$$

We are left with the last case: the calculation of the KM coefficient in equation (3.51). To do so, we plug $W^{(1)}\left(\vec{\beta}\,|\,\vec{\alpha}\right)$ in (3.51). With more algebra than before, and remembering that in the y direction the milestones are labeled with rising integers from bottom to top, we obtain:

$$D_{xy}^{(1)}(\alpha_x,\alpha_y)=\frac{1}{2}\sum_{\beta_y=1}^{N_{MSY}}(\beta_y-\alpha_y)\Delta_y\Delta_x\left[k(\alpha_x,\alpha_y;\alpha_x+1,\beta_y)-k(\alpha_x,\alpha_y;\alpha_x-1,\beta_y)\right] \quad (3.60)$$

Note that, the integrand of equation (3.51), $(x'-x)(y'-y)$, is substituted with the spacing between two milestones in the x direction Δ_x, and the spacing between the center of milestone (β_x,β_y) and milestone (α_x,α_y), which is $(\beta_y-\alpha_y)\Delta_y$ (see Figure 18).

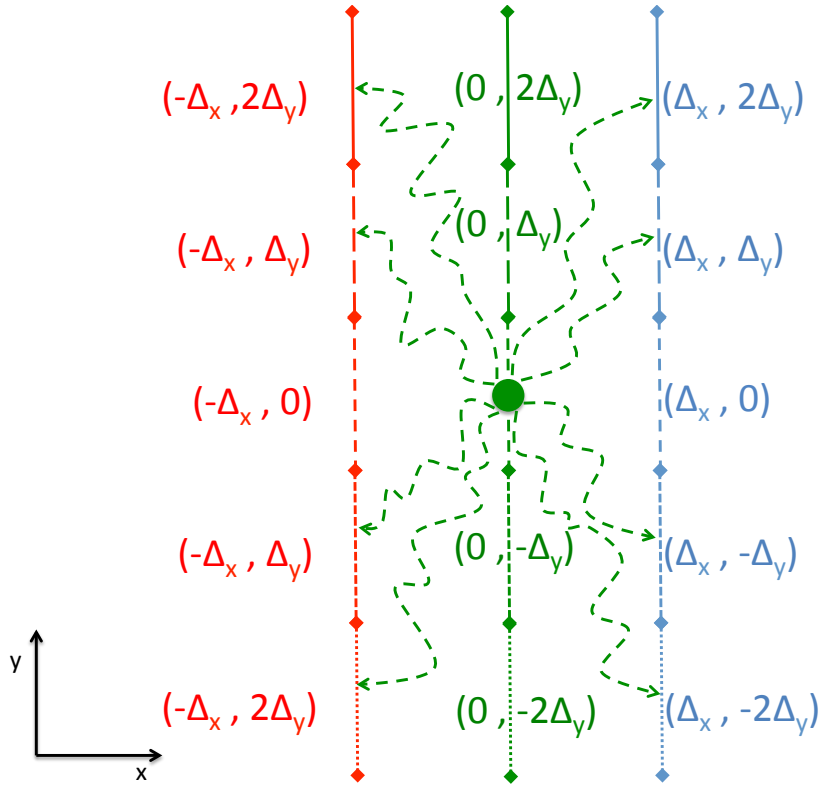


Figure 18 The same milestone scheme as in Figure 16. The starting milestone is labeled with a green dot. The other milestones are reported together with their distance on x and y from the initial milestone.

The derivation assumes that the discretization of spatial coordinates provide an accurate representation of the continuum limit.

If the size of the milestones and the distance between two milestones is "small enough", the discretization of the transition probability per unit time should introduce only a small error. Intuitively, the "smallness" of the discretization step should depend on the curvature of the underlying force and space-dependent diffusion. Since those are not known, it may not be easy to establish a-priori a good spacing for the milestones. However, some experimentation and exploratory simulations of the system are likely to solve this problem. Perhaps more significant is the scale frustration. The particle simulations require significant separation between the milestones to obtain reliable transition dynamics. The estimate of the derivative for the Smoluchowski Eq. is optimal when the distance between the milestones is as small as possible.

A numerical illustration to test the reliability of the method is provided in the next section.

NUMERICAL ILLUSTRATION

As an illustration of the algorithm presented in the previous section, we run a test case for which we know the exact answer. We follow the steps below:

1. Choose a functional form for the potential $U(\vec{x})$ and the matrix $\hat{b}(\vec{x})$ or $\hat{D}(\vec{x})$;

2. Run a Brownian dynamics simulation using Ermak and McCammon formula[91] that follows the Euler-Maruyama algorithm:[92]

$$\vec{x}_{n+1} = \vec{x}_n + \beta\hat{D}(\vec{x}_n)\cdot\vec{F}(\vec{x}_n)dt + \vec{\nabla}\cdot\hat{D}(\vec{x}_n)dt + \sqrt{2dt}\hat{b}(\vec{x}_n)\cdot\vec{N}(0,1) \qquad (3.61)$$

where $\vec{N}(0,1)$ is a vector of independent random numbers sampled from a Gaussian distribution of average 0 and variance 1, and $\vec{F}(\vec{x}) = -\vec{\nabla}U(\vec{x})$ is the force;

3. Perform a Milestoning analysis to compute the rate coefficients from the time traces;
4. Use the KM expansion to extract $\vec{F}_{KM}(\vec{x})$ and $\hat{D}_{KM}(\vec{x})$.
5. Compare the input force vector and diffusion tensor with the results generated by our analysis.

We first consider a one-dimensional test case. We chose the example used in [60] where the force is $\beta F(x) = -\cos(2x)$ and the diffusion constant is $D(x) = [0.2 + 0.1\sin(x)]\text{rad}^2/\text{ps}$. We divide the system into 24 milestones. We run $N_\alpha = 30000$ trajectories starting from each milestone α and terminate when the following or previous milestone is reached. The results for the potential and the diffusion are reported in Figure 19 and Figure 20.

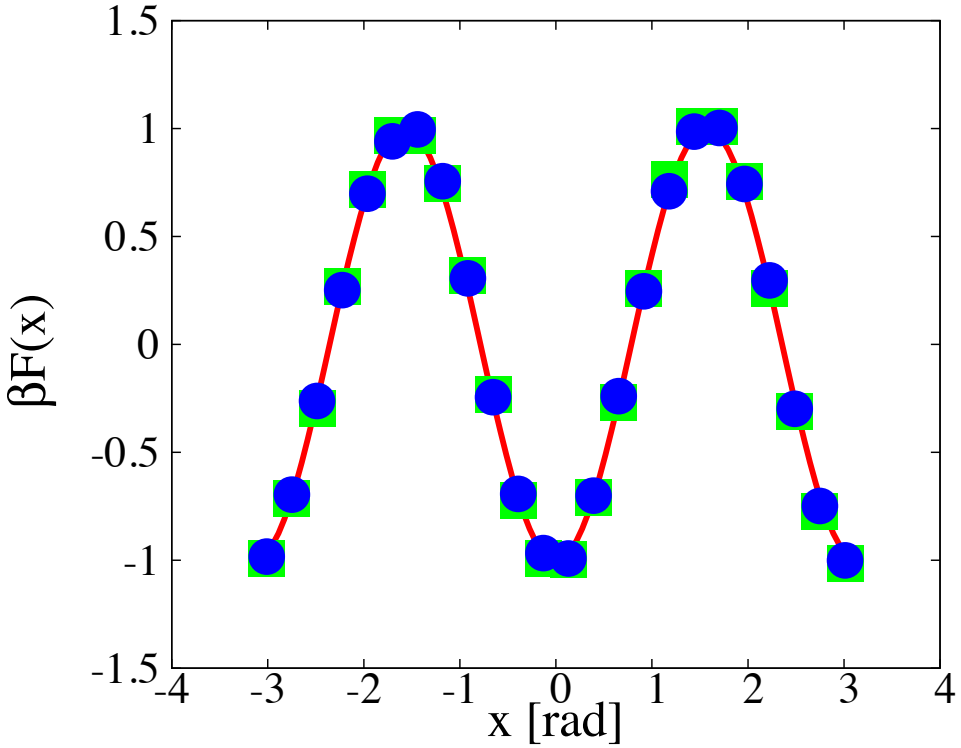


Figure 19 Force used for the 1D test case. The red line shows the input force, the blue dots and the green squares are the results of simulations carried out with time step 0.0001ps and 0.00001ps, respectively. The time scale and units were adopted from [60]. The error bars were computed propagating the statistical error on the rates (3.62) to the expression for the force. The errors are smaller than the size of the symbols.

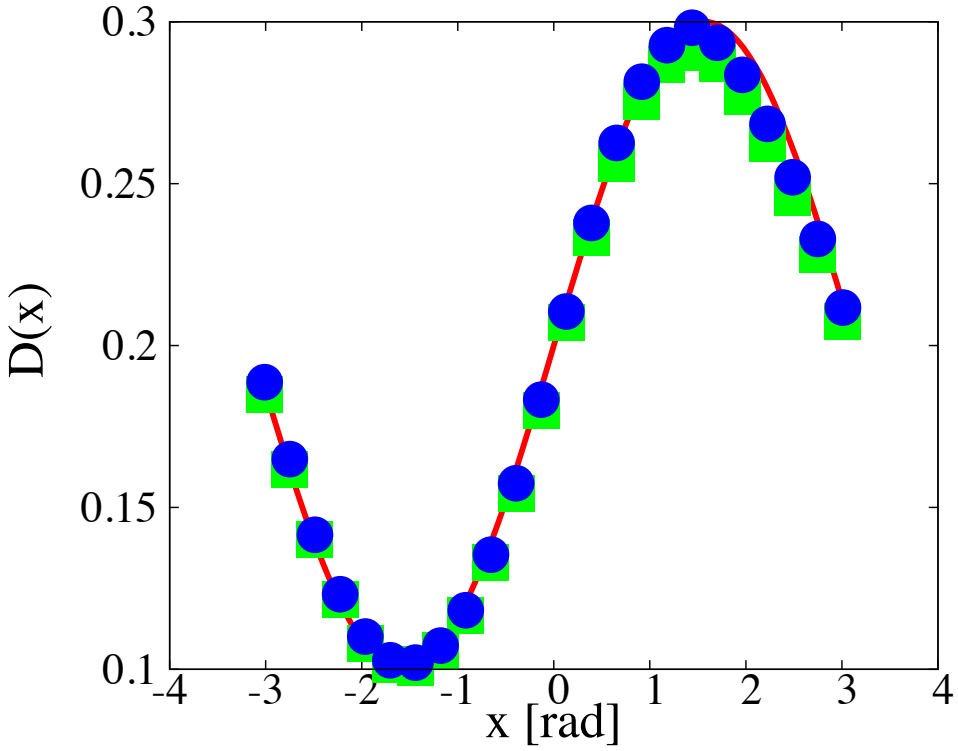


Figure 20 Diffusion coefficient used in the 1D test case. The red line shows the input space-dependent diffusion coefficient, the blue dots and the green squares are the results of simulations carried out with time step 0.0001ps and 0.00001ps, respectively. The time scale and units were adopted from [60]. The error bars were computed propagating the statistical error on the rates (3.62) to the expression for the diffusion. The errors are smaller than the size of the symbols.

The statistical error bars are smaller than the size of the dots in the plot. The statistical error for the evaluation of the rate constant is computed as:

$$\sigma^2\left[k(\alpha,\beta)\right] = \frac{k^2(\alpha,\beta)}{N_{\alpha\beta}}\left[\frac{p(\alpha,\beta)\left[1-p(\alpha,\beta)\right]}{p(\alpha,\beta)^2} + \frac{\left\langle\tau_\alpha^2\right\rangle - \left\langle\tau_\alpha\right\rangle^2}{\left\langle\tau_\alpha\right\rangle^2}\right] \tag{3.62}$$

where $N_{\alpha\beta}$ is the number of trajectories that start from milestone α and reach milestone β before any other milestone.

In equation (3.62) we assumed that $p(\alpha,\beta)$ is binomial. The errors are then propagated to the diffusion and the force. A normal distribution of the errors was assumed in an earlier study[93] which is an approximation to the expression derived here.

The force is recovered with high accuracy for both of the time steps used (Figure 19), while the space-dependent diffusion shows some dependence on the time step (Figure 20). Nevertheless, the algorithm reproduces the expected space-dependent diffusion equation well.

We also simulated two different 2D models. In these cases, we ran one long trajectory of 10^{11} steps with a time step of $10^{-5}\tau$ (τ is an arbitrary unit of time), saving the configurations every 10 steps. The analysis was then performed using 11 milestones per side that were used to truncate the long trajectory to the desired fragments and to estimate the rate coefficients. In both cases, the length of the periodic box was of 2.2λ (λ is an arbitrary unit of length). In the first case, the following 2D potential and 2D diffusion tensor were used

$$\begin{aligned} \beta U(x,y) &= U_0 \cos\left[\frac{2\pi}{L}\left(x-\frac{L}{2}\right)\right]\sin\left[\frac{2\pi}{L}\left(y-\frac{L}{2}\right)\right] \\ \hat{D}(x,y) &= \left\{D_0 + D_1 \cos\left[\frac{2\pi}{L}\left(x-\frac{L}{2}\right)\right]\right\}\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \end{aligned} \tag{3.63}$$

Here, the off-diagonal terms of the diffusion tensor are equal to zero, and the diagonal ones are equal. The parameters U_0, D_0 and D_1 were set equal to 0.5, $0.03\lambda^2/\tau$ and $0.01\lambda^2/\tau$. The results are reported in Figure 21-Figure 25. The statistical errors are derived from the rate coefficients (Eq. (3.62)). An error was also added in the

estimate of the off-diagonal term of the diffusion tensor, where the term $(x'-x)(y'-y)$ in Eq. (3.51) is approximated as $(\beta_y - \alpha_y)\Delta_y\Delta_x$, as in Eq. (3.60). While the contribution coming from the displacement in x is exact, the displacement in y lies in the interval $\left[(\beta_y - \alpha_y - 1)\Delta_y, (\beta_y - \alpha_y + 1)\Delta_y\right]$. The resulting error was computed as if the distribution of distances is uniform, and it found to be $(\Delta_y / 2)^2 / 3N_{\alpha\beta}$.

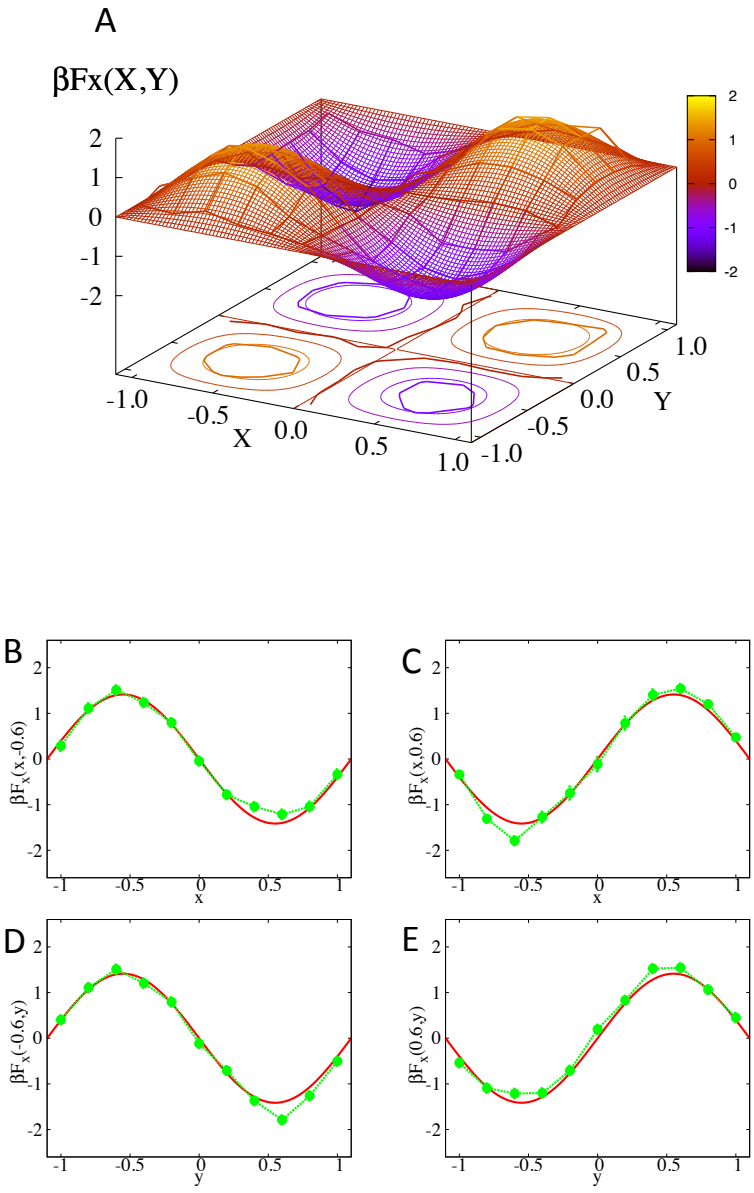

A
βFx(X,Y)
2
1
0
-1
-2
-1.0
-0.5
X
0.0
0.5
1.0
Y
B
C
D
E
βFx(x,-0.6)
βFx(x,0.6)
βFx(-0.6,y)
βFx(0.6,y)
x
y

Figure 21

Figure 21: Force in the x direction. (A) The thin meshed surface represents the force given in input, the thicker lines interpolate the values computed at the 11x11 milestones. (B-E) Different cross-sections of the 3D plot. The red line is the expected profile, the green points are the result of the simulation. (B) The force as a function of x with $y = -0.6$. (C) The force as a function of x with $y = 0.6$. (D) The force as a function of y with $x = -0.6$. (E) The force as a function of y with $x = 0.6$.

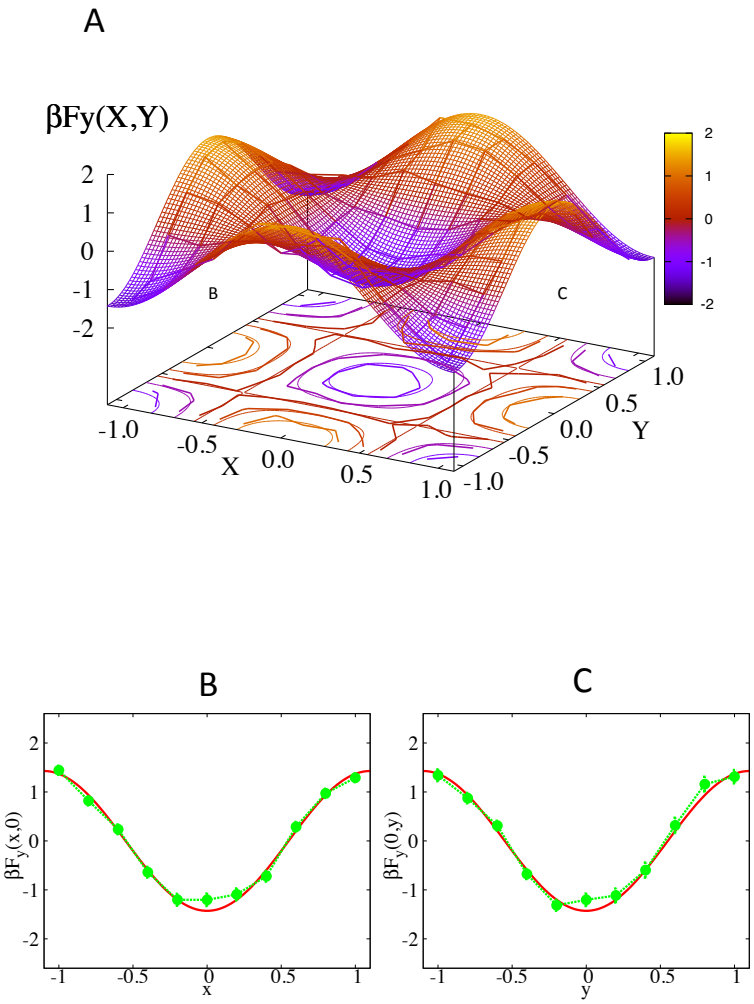


Figure 22 Force in the y direction. (A) The thin meshed surface represents the force given in input, the thicker lines interpolate the values computed at the 11x11 milestones (B-C) Different cross-sections of the 3D plot. The red line is the expected profile, the green points are the result of the simulation. (B) The force as a function of x with $y = 0.0$. (C) The force as a function of y with $x = 0.0$.

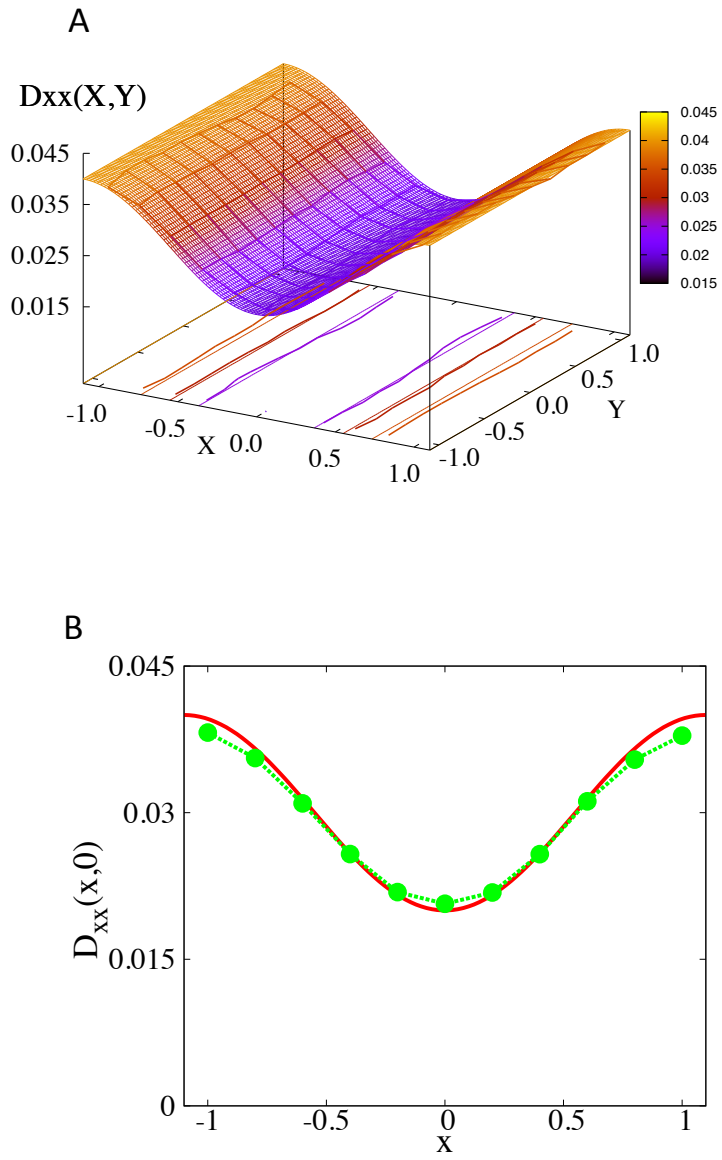


Figure 23 Diffusion tensor, xx component. (A) The thin meshed surface represents the function given in input, the thicker lines interpolate the values computed at the 11x11 milestones. (B) A cross-section corresponding to $y = 0.0$ is shown. The red line is the exact result, the green dots are the result for the simulation.

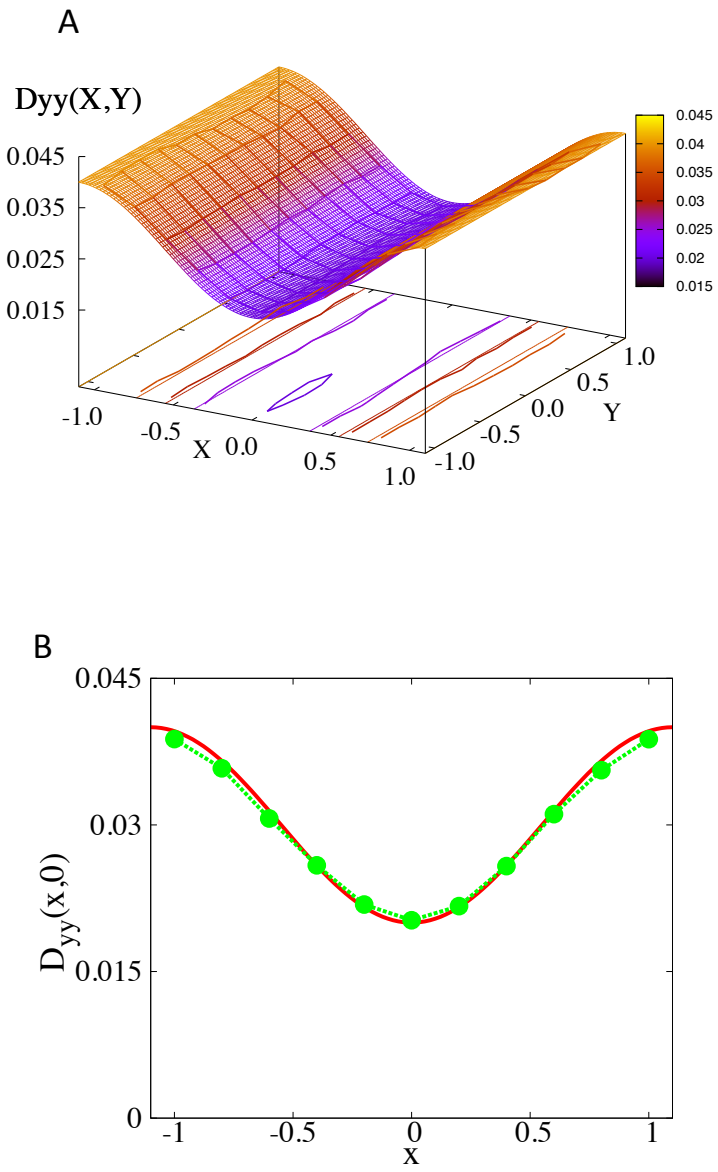


Figure 24 Diffusion tensor, *yy* component. (A) The thin meshed surface represents the function given in input, the thicker lines interpolate the values computed at the 11x11 milestones. (B) A cross section corresponding to $y = 0.0$ is shown. The red line is the exact result, the green dots are the result for the simulation.

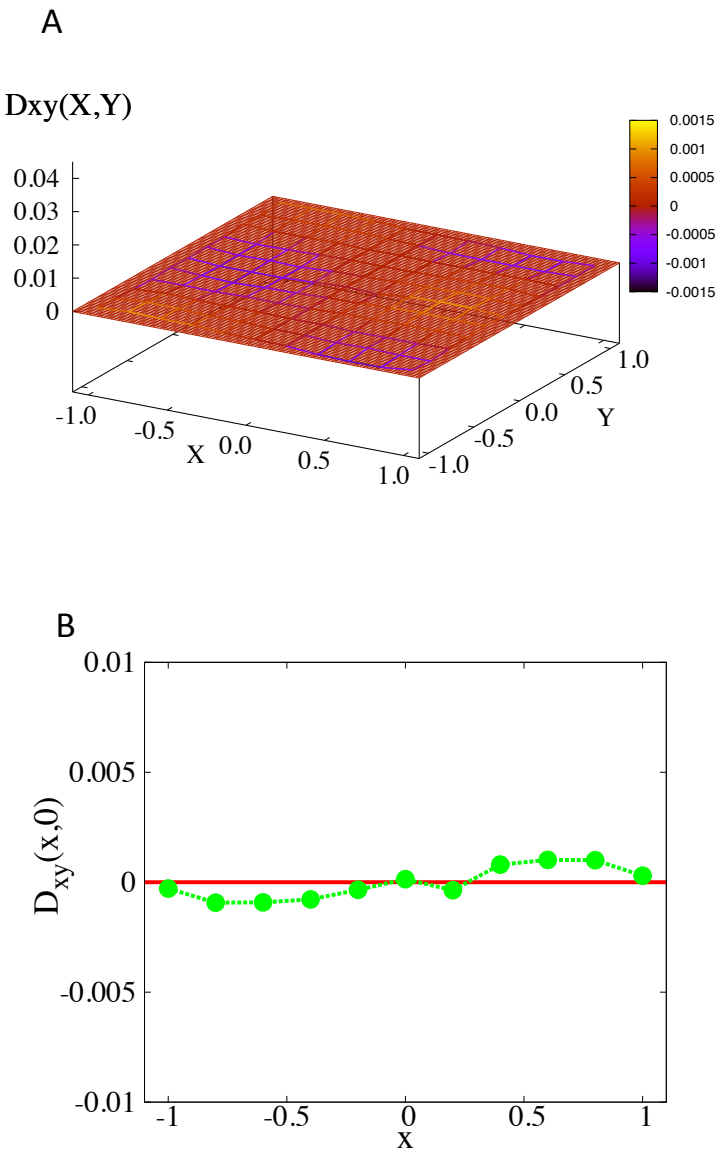


Figure 25 Diffusion tensor, xy component. (A) The lines interpolate the values computed at the 11x11 milestones. The expected result is 0. (B) A cross-section corresponding to $y = 0.0$ is shown. The red line is the exact result, the green dots are the result for the simulation. Note that the deviations from zero are an order of magnitude smaller than sizes of other elements of the diffusion tensor.

Figure 21 and Figure 22 show a comparison between the forces inserted in input and the output of the analysis carried out according to the previous section. The statistical errors are roughly of the size of the dots. The overall χ^2 is equal to 1.64 for $F_x(\alpha,\beta)$ and 0.99 for $F_y(\alpha,\beta)$. Figure 23-Figure 25 show the different components of the diffusion tensor. The results are less noisy than those in Figure 21. However the simulated space-dependent diffusion coefficient is slightly under-estimated, resulting in a χ^2 of around 10 for the diagonal terms and 3 for the off diagonal term. The reasons for this possible discrepancy are discussed in the end of this section. The shapes are recovered quite well, and the off-diagonal term, which was expected to be zero, is found to be more than an order of magnitude smaller than the diagonal terms.

In the second 2D test case, we used the same potential as in the first case (eq. (3.63)) but a different diffusion tensor. In particular, we decided to test a case in which the off-diagonal terms of the diffusion tensor were of the same order of magnitude as the diagonal terms. We used a $\hat{b}(x,y)$ tensor with the following components:

$$\begin{aligned} b_{xx}(x,y) &= \sqrt{D_0 + D_1 \sin\left[\frac{2\pi}{L}\left(x - \frac{L}{2}\right)\right]} \\ b_{yy}(x,y) &= \sqrt{D_0 + D_1 \sin\left[\frac{2\pi}{L}\left(y - \frac{L}{2}\right)\right]} \\ b_{xy}(x,y) = b_{yx}(x,y) &= \sqrt{\alpha_0 D_0 + \alpha_1 D_1 \sin\left[\frac{2\pi}{L}\left(x - \frac{L}{2}\right)\right]} \end{aligned} \tag{3.64}$$

with the parameters α_0, α_1, D_0 and D_1 set equal to 0.25, 0.125, $0.03\,\lambda^2/\tau$ and $0.01\,\lambda^2/\tau$. The diffusion tensor $\hat{D}(x,y)$ is obtained from $\hat{b}(x,y)$ as in eq. (3.11). The results are reported in the following figures.

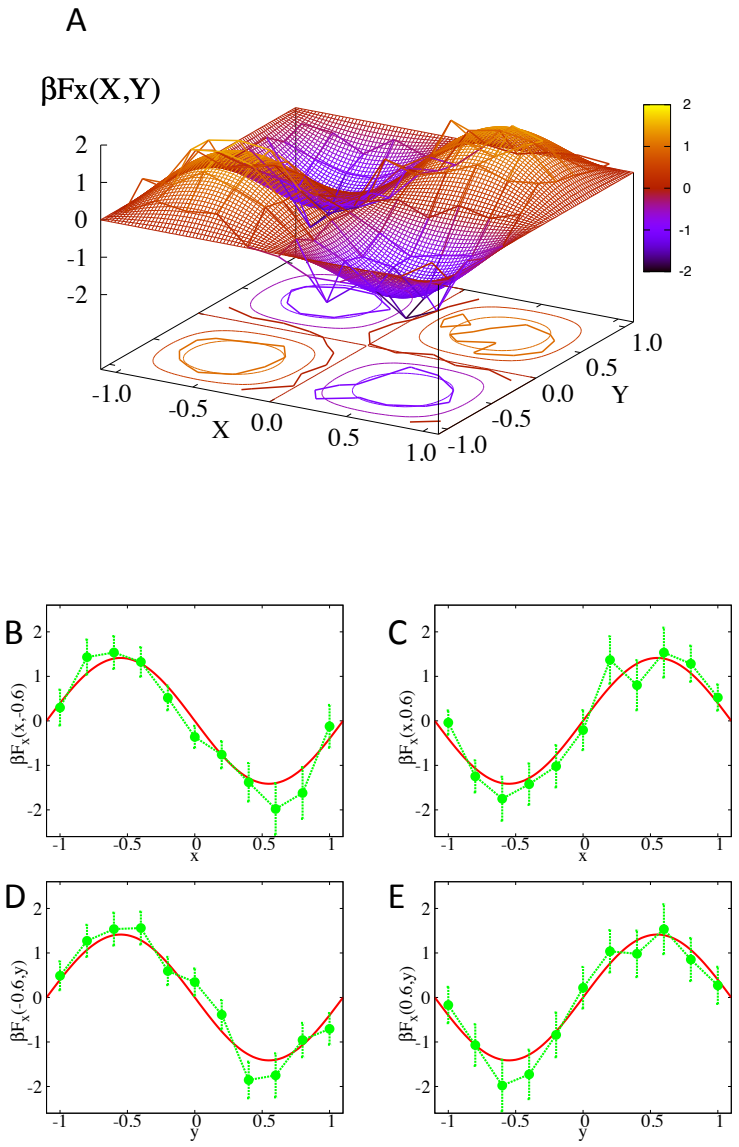


Figure 26 Force in the x direction for the second 2D test case. (A) The thin meshed surface represents the force given in input, the thicker lines interpolate the values computed at the 11x11 milestones. (B-E) Different cross-sections of the 3D plot. The red line is the expected profile, the green points are the result of the simulation. (B) The force as a function of x with $y = -0.6$. (C) The force as a function of x with $y = 0.6$ (D) The force as a function of y with $x = -0.6$. (E) The force as a function of y with $x = 0.6$.

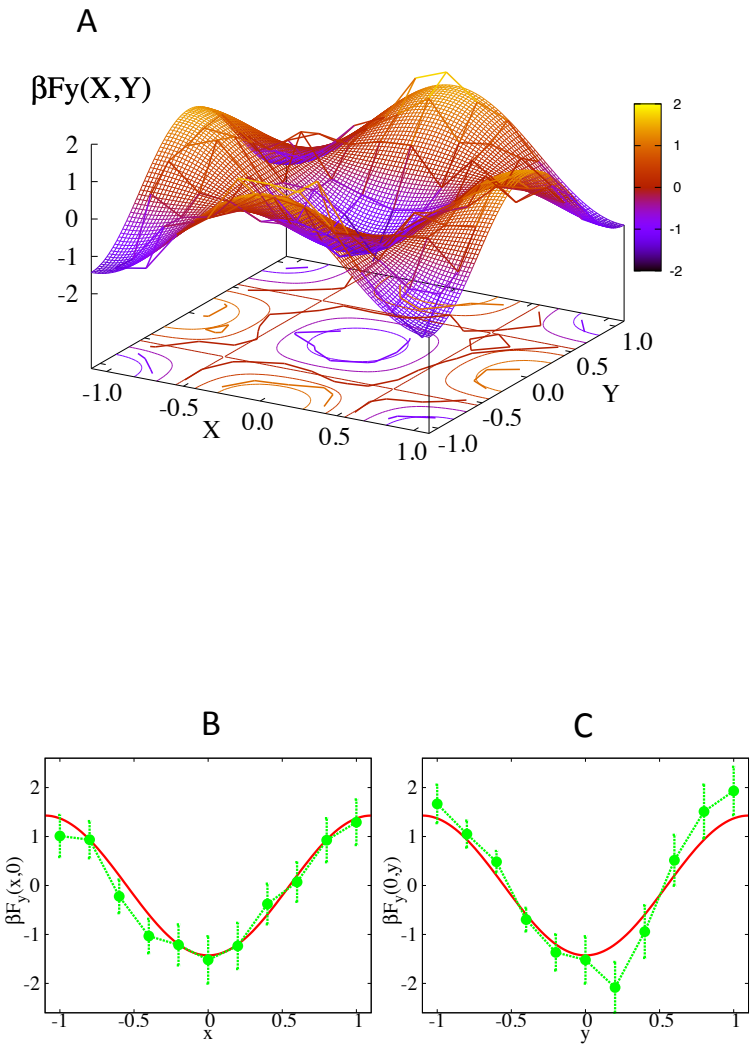


Figure 27 Force in the y direction for the second 2D test case. (A) The thin meshed surface represents the force given in input, the thicker lines interpolate the values computed at the center of the 11x11 milestones (B-C) Different cross-sections of the 3D plot. The red line is the expected profile, the green points are the result of the simulation. (B) The force as a function of x with $y = 0.0$. (C) The force as a function of y with $x = -0.6$.

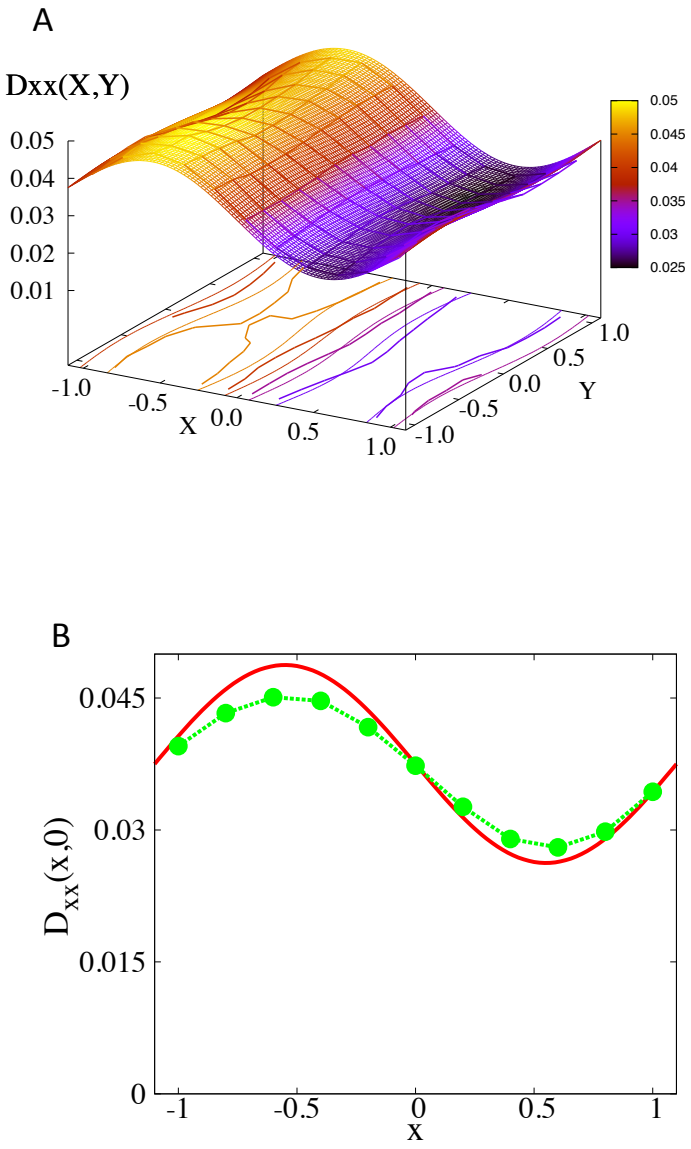


Figure 28 Diffusion tensor for the second 2D test case, xx component. (A) The thin meshed surface represents the function given in input, the thicker lines interpolate the values computed at the 11x11 milestones. (B) A cross-section corresponding to $y = 0.0$ is shown. The red line is the exact result, the green dots are the result for the simulation.

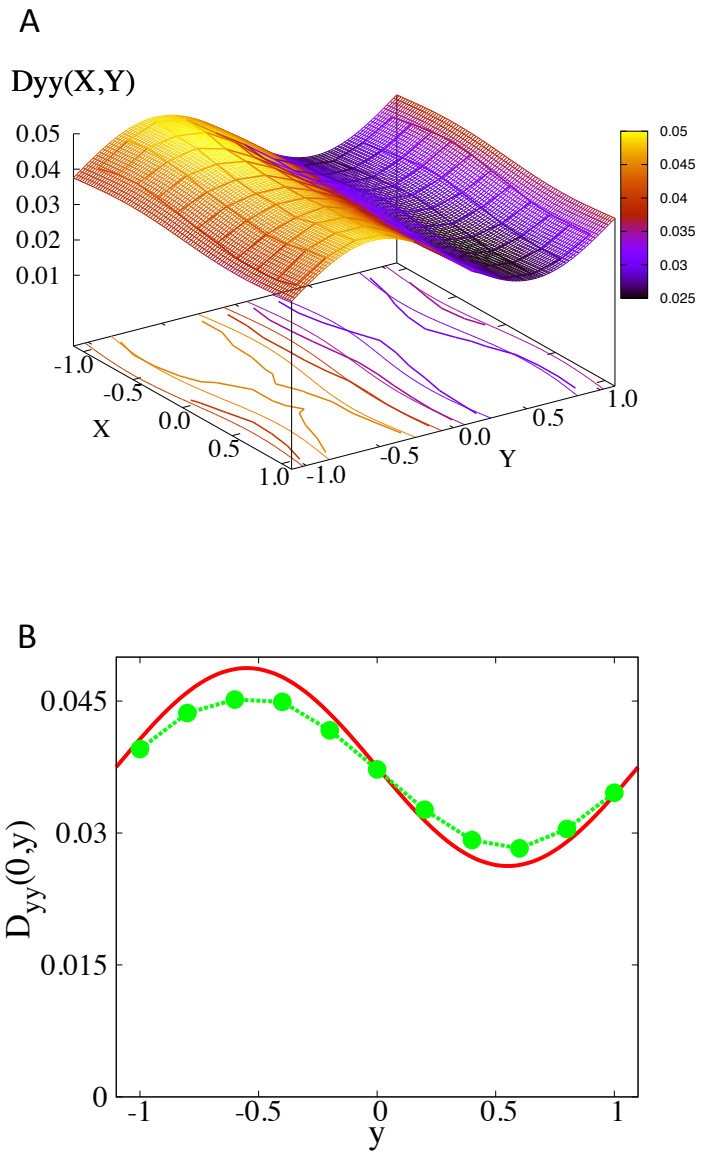


Figure 29 Diffusion tensor for the second 2D test case, *yy* component. (A) The thin meshed surface represents the function given in input, the thicker lines interpolate the values computed at the 11x11 milestones. (B) A cross-section corresponding to $x = 0.0$ is shown. The red line is the exact result, the green dots are the result for the simulation.

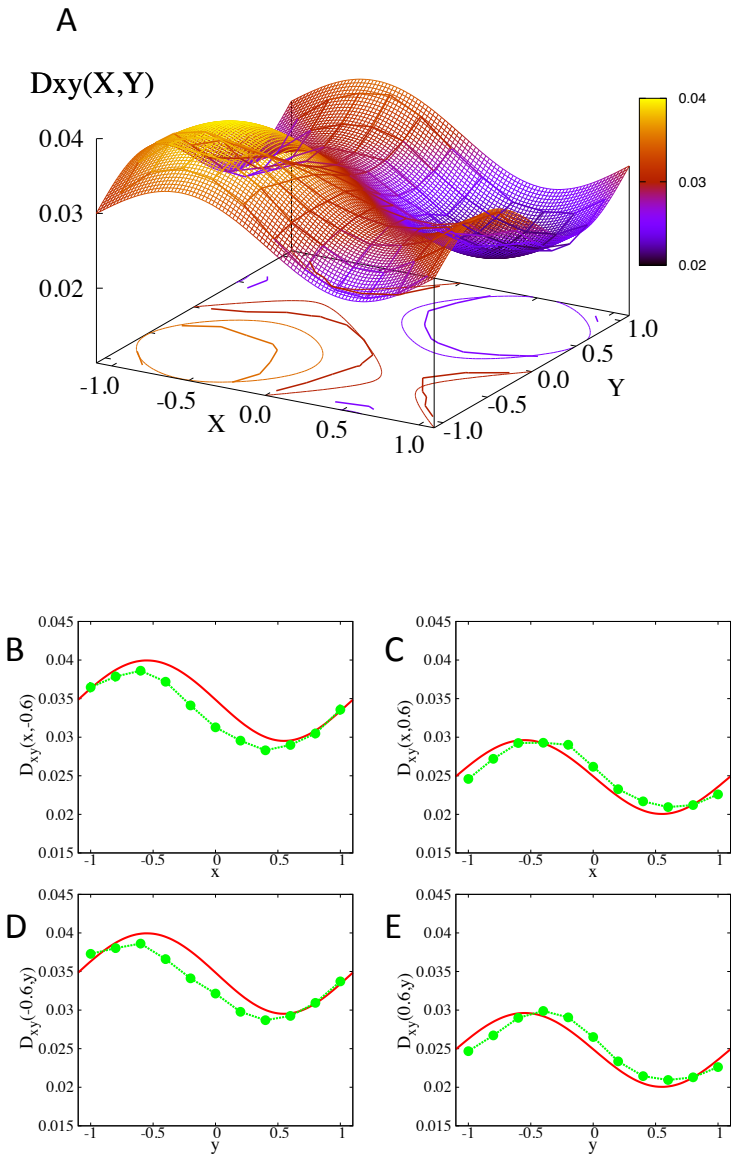


Figure 30 Diffusion tensor for the second test case, xy component. (A) The thin meshed surface represents the function given in input, the thicker lines interpolate the values computed at the 11x11 milestones. (B-E) Four cross-sections of the 2D surface: (B) cross-section corresponding to $y = -0.6$; (C) cross-section corresponding to $y = 0.6$; (D) cross-section corresponding to $x = -0.6$; (E) cross-section corresponding to $x = 0.6$. The red line is the exact result, the green dots are the result for the simulation.

Figure 26 and Figure 27 show the two components of the force vector. The expected value is the same as in Figure 21 and Figure 22, the recovery of the two-dimensional surface is noisier, but the results are again within the statistical error (χ^2 of 0.34 and 0.36 for the two components of the force). Figure 28-Figure 30 show the different components of the diffusion tensor. The shape of each of the elements of the diffusion tensor is recovered even though, as highlighted from the one-dimensional sections of the two-dimensional surfaces (Figure 28B, Figure 29B, and Figure 30B-E), there is a slight underestimate of the maxima and overestimate of the minima, which exceeds the statistical error (χ^2 of around 10-20). As shown in the one-dimensional test case, the accuracy of the integrator used, and the choice of the time-step, might account, at least partially, to this small discrepancy. To test this hypothesis, we evaluated and reported in Figure 31 the dependence of the error in the evaluation of $D_{xx}(x,y)$ at $x=-0.6$ and $y=0.0$ from

1. The size of the milestone in the y direction, Δ_y (Figure 31A);
2. The distance between two milestones in the x direction, Δ_x (Figure 31B);
3. The time step (Figure 31B).

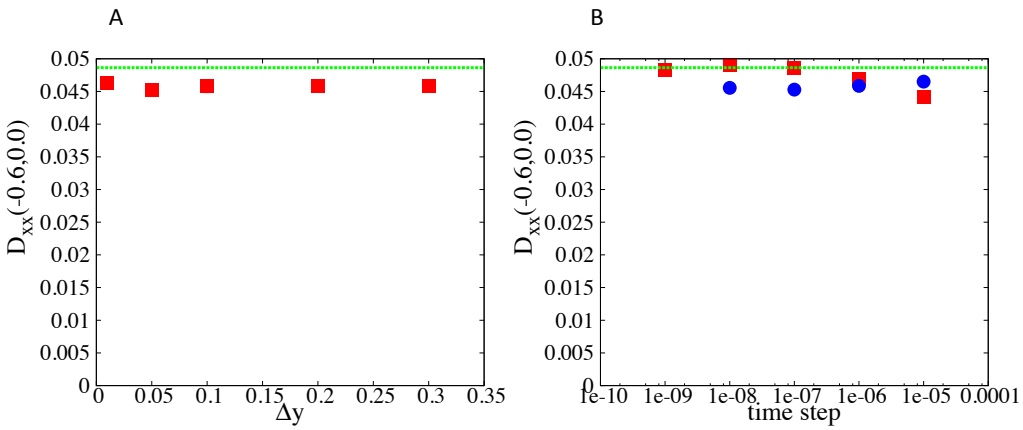


Figure 31 Analysis of the numerical errors in the calculations of the diffusion tensor. The green lines are the exact results. (A) The red squares are computed for distances between two milestones $\Delta_x = 0.2$ and variable size Δ_y of a milestone. (B) The blue dots are simulations carried out with $\Delta_x = 0.2$ and $\Delta_y = 0.2$ using the time steps reported on the x-axis, while the red squares were obtained from simulations performed after changing the spacing between milestones to $\Delta_x = 0.01$.

The results in Figure 31 were obtained by starting the trajectories at the milestone $\left\{(x,y) \mid x = 0.6 \text{ and } -\frac{\Delta_y}{2} \le y < \frac{\Delta_y}{2}\right\}$, by drawing randomly from a uniform distribution the starting point in y. Each point was extracted running 10^4 trajectories, with the

exception of the first and second point in Figure 31A, that were obtained from $3 \cdot 10^4$ and 10^5 runs, respectively.

The dependence on the size of the milestone Δ_y is weak, as we can see from Fig. 17A. On the other hand, Δ_y is expected to affect more significantly the estimation of the off-diagonal terms of the diffusion tensor. If we use the spacing between milestones and the size of the milestones used in Figure 21-Figure 30, reducing the time step does not give a significant improvement (Figure 31B, blue dots). To improve the results we reduced of the spacing between the milestones Δ_x and a reduced of the time step (Figure 31B, red squares). Hence, it is possible to obtain highly accurate results if significantly larger computational resources are used.

Chapter 4: The Stereospecificity of the Enzyme Ketoreductase[6]

The study of the enzymatic reactions is a prime goal of biochemistry. Many enzymes perform their functions by finding the correct substrate and modifying its chemical structure. Typically, this occurs in a number of steps.[68] First of all the substrate is weakly bound to the enzyme. Then the enzyme undergoes a conformation transition that strengthens the binding and prepares the chemistry step (induced fit). Then the chemical processing takes place, in which bonds may be broken and new ones may be formed, and the products are formed. Finally, the enzyme releases the product.

A key question here is: how does the enzyme find the proper substrate? In the cellular milieu there are many candidate substrates, but to perform its function properly the enzyme has to be specific for one of them. How does this specificity occur? In principle every step of the mechanism for enzyme reaction is involved. The size of the binding site in the open configuration of the enzyme forbids the binding of large substrates. The position of the reactive amino acids selects a particular geometry of the ligand. Interestingly, also the dynamics of the enzyme once bound to the ligand might affect the specificity. Experimental findings,[68] and results from MD simulations[70] have shown that DNA polymerase changes its structure from the open to the close conformation only if the matching nucleic acid is bound.

Here, I performed a number of MD simulations of the enzyme ketoreductase (KR), which is a domain of the polyketidase (PK), a large molecular factory that produces the secondary metabolites polyketides.[72] The enzyme reduces the β-keto oxygen of the substrate, with the help of NADPH.[72] The ligand might presents itself to the protein

---

[6] The work presented in this chapter has been done under the supervision of my advisor Prof. Elber and in collaboration with Prof. Keatinge-Clay. I gratefully acknowledge the contribution of Dr. Jianting Zheng, who provided the PDB files of the ternary A1-dkD complex, and Dr. Yue Shi, who carried out the quantum mechanical calculations that were necessary to develop the force field for the ligands.

in two different optical isomers: the one in which the α-methyl substituent has a D orientation, and the one with the L orientation. In a racemic mixture, the A1 KR is capable of selecting and reducing the D optical isomer, while no product corresponding to the L optical isomer is formed.[72, 74] How does the enzyme distinguish these two ligands? Mutations in the binding pocket of the enzyme have shown that a glutamine is necessary for the specificity, because if mutated in histidine the enzyme looses its stereospecificity.[74] Upon one further mutation in the binding pocket, from glycine to threonine, the stereospecificity is inverted.[75]

The MD simulations that I carried out show the atomistic reason of the role in stereospecificity of this glutamine. Furthermore, other relevant interactions for the correct alignment of the ligand in the binding pocket are highlighted.

MODELING THE SUBSTRATES INTO THE BINDING SITE

The crystal structure for the A1 ketoreductase[74] (molecule B from PDB 3MJS) was used as the starting structure in the preparation of the simulations. The structure was determined without the substrate, while the coordinates of the co-factor NADPH were resolved. The substrate used for the simulations is 2-methyl-3-oxopentanoate-S-N acetyl cysteamine, which from now on will be called diketide (dk). The enzyme is stereospecifc for the D isomer, dkD, while it does not form products with the stereoisomer, dkL. The dkD ligand was modeled into the binding pocket. In the first step of the modeling it was docked into the protein binding site. In Figure 32 we sketch the positions of the critical protein groups (tyrosine 371) that participate in the reaction, or that are considered important for the alignment of the substrate in the reactive configurations (tryptophan 363). We also show the spatial position of the NADPH. Note that the numbering of the

amino acids is presented here as the numbering that is produced by the MD package MOIL,[23] which was used to generate the initial structure and geometry.

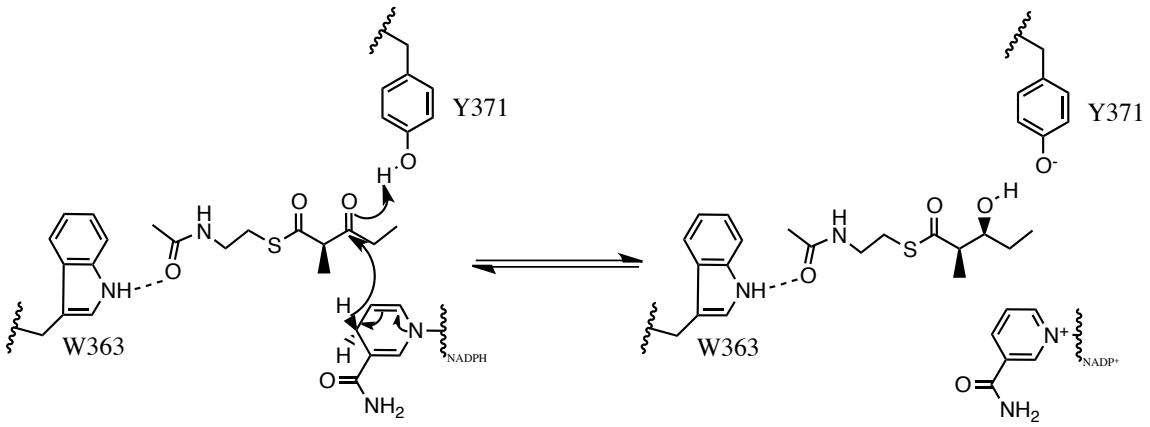


Figure 32 Mechanism of the enzyme.[72] The ligands are shown together with two amino acids involved in the reactions. The Y371 donates a hydrogen to the ligand. The W363 is considered important for the correct alignment of the ligand in the binding pocket.[75]

The amino acid Y371 is directly involved in the chemical reaction. It donates the hydrogen of the hydroxyl group of the tyrosine side chain to the β-carbonyl oxygen of dk. The NADPH is part of the reaction as well. It is oxidized, and donates an hydrogen of the nicotinamide ring to the β-carbon of the dkD. The hydrogen of the nicotinamide ring is bound to a prochiral carbon, and the hydrogen that is donated to the ligand is the pro-S hydrogen. Another interaction that is believed to be important for the correct alignment of the ligand in the binding pocket is the hydrogen bond of the dkD amide carbonyl group and the side chain nitrogen of W363. This interaction, proposed by early biochemical insight,[75] helped in the initial modeling of the correct dkD ligand into the binding pocket. The incorrect enantiomer, dkL, was obtained by inverting the methyl group of the α substituent with the hydrogen

at the α carbon. The structures obtained so far were the starting point for all the simulations that followed. Below we use the name E0 (experimental 0) to denote these initial structures.[7] The E0 structure is shown in Figure 33A.

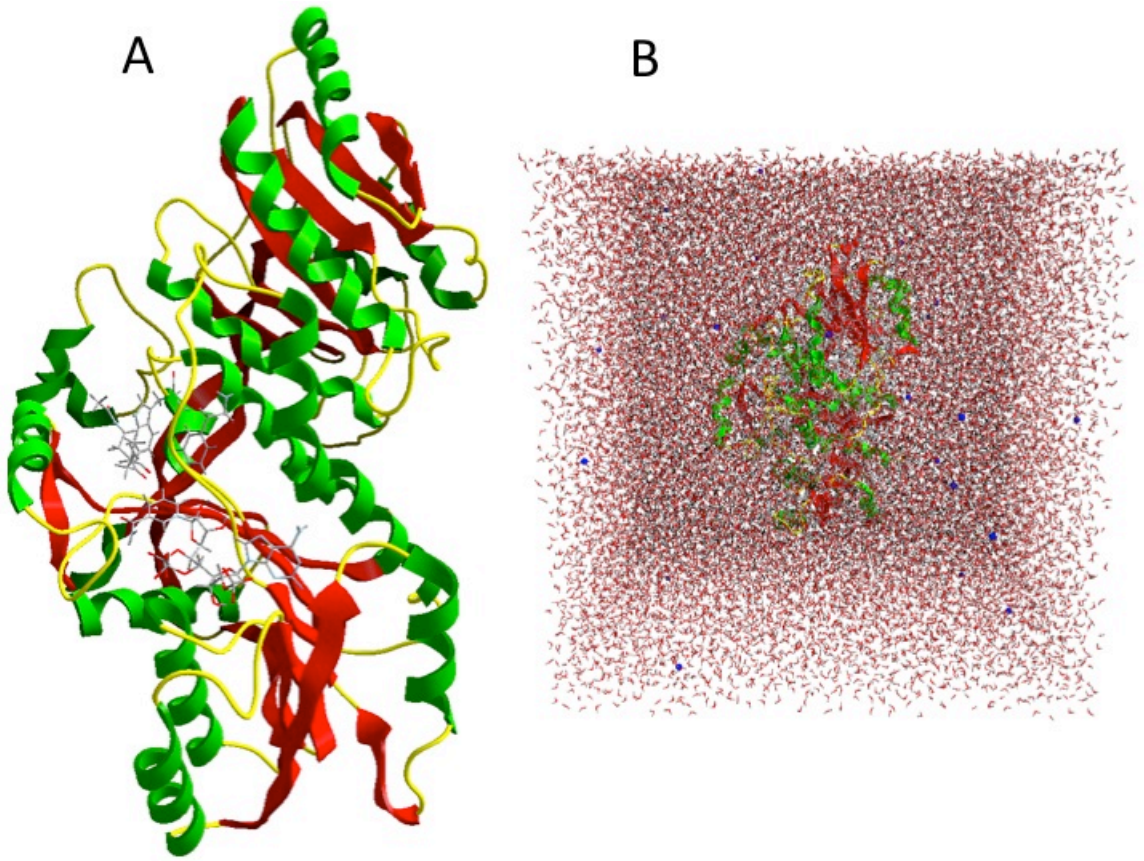


Figure 33 (A) The E0 ternary structure. The protein is shown in cartoon mode, the ligands, W363, and Y371 are drawn in stick mode. The atoms of the protein and the NADPH are determined by X-ray crystallography, and are taken from molecule B of PDB 3MJS. The dk was modeled in the binding pocket. (B) The solvated A1 enzyme. The sodium ions are shown in blue.

SETUP AND LIST OF SIMULATIONS THAT WERE CONDUCTED

We ran 4 different sets of simulations:

[7] The pdb file of the E0 structure was provided to me by Dr. Jianting Zheng, whose contribution is gratefully acknowledged.

1. One 20ns simulation for each of the complexes A1-dkD and A1-dkL, which we refer to as L1.
2. A set of 20 shorter simulations of the complexes A1-dkD and A1-dkL, each of 1ns; we refer to those as S1;
3. A longer simulation of both complexes of 50ns each, which we will refer to as L2;
4. A set of 20 short simulations of both complexes, each of 800ps; we will refer to those as S2.

Besides the lengths, the differences between L1, S1, and L2, S2 are in the choice of the initial condition for the simulations.

We started the L1 and S1 simulation from the E0 structures. These structures include experimentally determined positions of the protein atoms and of the modeled coordinates of the substrate. We optimize the coordinates of the substrate as follows. We restrained the positions of all the atoms resolved by X-ray crystallography (i.e. protein and NADPH) by attaching a stiff spring to their starting position. We added the penalty function $U_{\text{tether}} = \sum_i (1/2) k \left(x_i - x_{0i} \right)^t \left(x_i - x_{0i} \right)$ where x_i is the current 3D position vector of the atom and x_{0i} is the three-dimensional coordinate vector of the "tether" atom in the E0 structure. The force constant was chosen to be highly stiff and of 100 kcal/mol/Å^2. The dk ligand, on the other hand, was left unrestrained, so that the algorithm could modify its position in space and its internal structure following the minimization of the energy of the interactions with the X-ray resolved atoms and within the substrate. Hence we adjust the modeled structure but the experimentally determined coordinates are remained roughly unchanged. We call these new structures M1 (minimized 1).

Simulations L2 and S2 also started from the structures E0 as well, but we ran a different energy minimization procedure. As before, we restrained the atoms resolved by the crystal structures. At variance with the construction of M1, we added two more distance restraints on the dk ligand to better mimic the reactive configuration (Figure 32). One constrained distance is between the β-carbonyl oxygen and the reactive hydrogen of Y371, the other distance is between the β-carbonyl carbon and the pro-S hydrogen of NADPH. These restraints were modeled as springs with equilibrium distance of 2Å and spring stiffness of 10 kcal/mol $Å^2$. After the minimization of the energy of interaction, we equilibrate the system for 10ps at 10K allowing only the ligand to move, but keeping the restraints on the reactive distances. The structures obtained in this way are called M2 (minimized 2).

In all the simulation sets, the system was solvated in a 97.5 $Å^3$ box water molecules (see Figure 33B). The size of the box was chosen to ensure that in the 8 corners of the box the density of water was close to the expected one for the A1-dkD enzyme-ligand complex (0.988 g/cm^3 for A1-dkD complex in the L1 setup computed over 1ns simulation; the number slightly larger then the expected one for the water model used, TIP3, which has density 0.982 g/cm^3 [81] at 298K and 1atm). The box was filled with roughly 28540 water molecules, with a variation between the 8 different initial structures (four simulation sets, two complexes each) of less then 10 water molecules. Twenty-seven sodium ions were added to ensure the neutrality of the simulation box. The solvated M1 and M2 structures are addressed as SolvM1 and SolvM2.

The configurations from which the production runs were started were generated following slightly different heating and equilibration protocols.

For L1, we first relaxed the structure SolvM1 by heating it from 10K to 300K in 20ps, we then run 1ns of equilibration before sampling the initial structure for this run.

For S1, we decided to look at the immediate response of the protein to the ligand starting from a configuration close to the X-ray structure. We were concerned that some of our conclusions may be affected by inaccuracies in the force field that brings us to an equilibrium conformation which is not sufficiently close to the coordinates from crystallography. Therefore, as for L1, we relaxed the structure SolvM1 by increasing the temperature from 10K to 300K in 20ps, and that was the initial structure for the 20 short runs in S1.

For L2, we relaxed the structures SolvM2 by heating them from 10K to 300K in 100ps while keeping the restraints on the reactive distances. The structure obtained in this way was our initial structure.

For S2, we took the initial structure of L2 and we ran other 2ns at 300K keeping the restraints. Out of these 2ns of simulation, we selected 20 configurations per complex to start the short runs of S2.

All the simulations were carried out using the MD software MOIL-OPT,[24] adopting similar setups. The Particle Mesh Ewald (PME)[80] algorithm was used to account for the long-range electrostatic interactions. The cutoff for van der Waals interactions and for the real space part of PME was set to 9.5Å. The PME tolerance was set to 10^{-9}, and the PME grid to 100^3. A spring was attached to the geometric center of the enzyme to restrain its center of mass translation. The SHAKE algorithm[94] in its MATRIX form[95] was used to constrain bonds and angles in the water molecules. All the bonds in the enzyme and ligands were also constrained using matrix shake[96] in L1/S1, while in L2/S2 only the light atoms were constrained with shake.

A "temperature rescaling" thermostat was used to run the simulations in the NVT ensemble.[83] The time step was 1fs. We used the RESPA algorithm,[86] which allowed us to

compute the long-range part of PME (i.e. the reciprocal space part of the calculation) once every four steps.

The force field that we used for the protein was the all atom version of OPLS.[4] For the water molecules the TIP3 force field was adopted.[81] We could not find a force field for the ligands, therefore we generated one using OPLS energy terms of similar chemical species for bonded energy terms and van der Waals interactions. We computed the partial charge distribution, and the missing terms for the torsion potential performing some quantum mechanical calculations.[8]

The L1 and L2 simulations were carried out using the optimized version of the MOIL code,[24] parallelized to run on GPUs and multiple CPUs (three or four in our simulations) at the same time. The many short simulations (S1 and S2) were run on CPUs, and parallelized to work on three threads.

The different initial structures affect our analysis significantly. The results highlight different aspects of the process of specificity, but at the same time show the common feature that the correct ligand is a preferred initial binder compared to the incorrect one.

EVIDENCES FROM THE LONG SIMULATIONS L1/L2

The L1/L2 simulations for both of the complexes show that the force field that we used is accurate enough to keep the correct ligand (dkD) closer to the reactive configuration.

In Figure 34 and Figure 35 we show the "reactive" distances Y371-dk and NADPH-dk as a function of time.

---

[8] The quantum mechanical calculations were performed by Dr. Yue Shi, whose contribution is gratefully acknowledged.

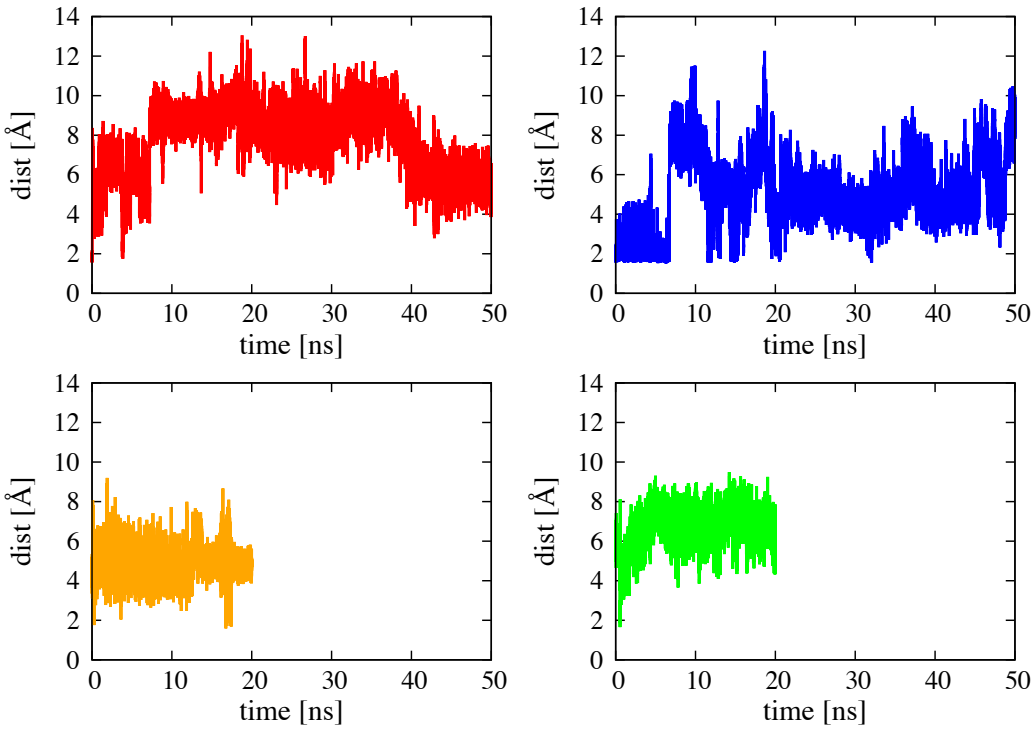


Figure 34 Distance between Y371 side-chain hydroxyl hydrogen and dk β-carbonyl oxygen (Y371-dk). The red line shows A1-dkD for L2 simulations, the blue line A1-dkL for L2, the orange line A1-dkD for L1, and the green line A1-dkL for L1.

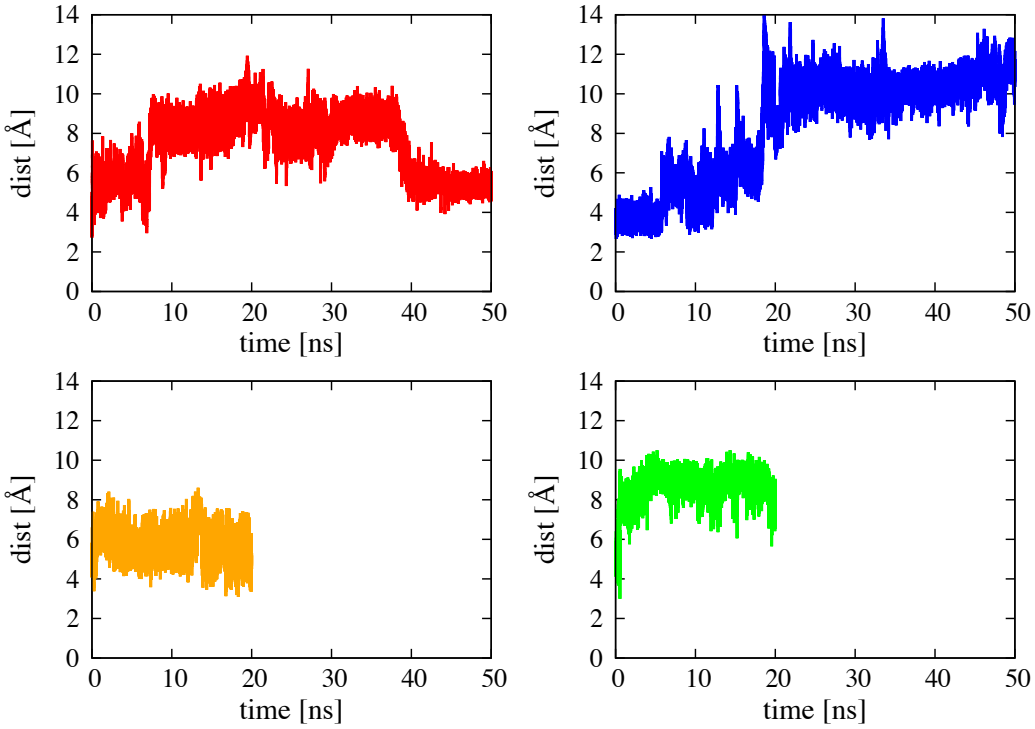


Figure 35 Reactive distance between pro-S hydrogen of NADPH and the dk β-carbonyl carbon (NADPH-dk). The red line shows A1-dkD for L2 simulations, the blue line A1-dkL for L2, the orange line A1-dkD for L1, and the green line A1-dkL for L1.

The L1 simulations (orange and green) reach quickly (within few ns) a steady state that is kept throughout the rest of the simulation. The L2 simulations (red and blue) show instead a significant drift. This might suggest that initial structure of the L1 simulations is closer to a stable bound state than the one of the L2 simulations.

In the first 7.5ns of the L2 simulations the A1-dkL complex (blue) shows more reactive configurations than the A1-dkD complex (red), i.e. the Y-dk (Figure 34) and NADPH-dk (Figure 35) distances are less than 4 Å. On the other hand, after this initial

stage, the NADPH-dk distance for the A1-dkL complex (blue, Figure 35) starts to drift away from the reactive configuration, while the NADPH-dk distance for the A1-dkD complex (red, Figure 35), after an initial drift, goes back to roughly 6 Å. In the last 10ns of the L2 simulations, the Y-dk distance (Figure 34) for A1-dkD (red) and A1-dkL (blue) tends to stay at around 6 Å, with A1-dkD showing smaller fluctuations than A1-dkL.

Since the Y-dk and NADPH-dk distances in the L1 trajectories are more stationary, properties averaged over the simulations are more likely to be converged. We consider first the average displacement of the correct and incorrect ligand from their initial structure. To do so, we align all the structures sampled in the L1 simulations to the initial structure (as initial structures we used the structures obtained after energy minimization - SolvM1 and the SolvM2 configurations - for the L1 and L2 simulation sets, respectively). To do so, the Kabsch algorithm[97] is used to align the C_α carbons of all the protein residues.[9] We then measure the average distance of the dk ligand from the initial one using the following formula for the average RMSD:

$$\left\langle RMS_{\mathrm{dk}} \right\rangle = \frac{1}{T}\sum_{t=1}^{T}\frac{1}{N_{dk}}\sum_{i\in dk}^{N_{dk}}\sqrt{\left(x_t - x_0\right)^2 + \left(y_t - y_0\right)^2 + \left(z_t - z_0\right)^2} \qquad (4.1)$$

where the subscript of x, y, and z refers to the number of the sampled configuration. The starting configuration is labeled "0", the total number of configurations is T, N_{dk} is the number of atoms belonging to dk. The index t runs from 1 to T. The result is that the average displacement for the dkD ligand in the binding pocket from the initial structure is 2.09Å, while the RMSD of dkL is 4.19Å. This

[9] In what follows, all the structural alignments are performed by aligning the C_α carbons of all the protein residues, unless otherwise stated.

suggests that the correct enantiomer stays closer to the initial configuration compared to the incorrect substrate, as it should.

We also examined the number of configurations that are reactive, i.e. structures in which the Y-dk and NADPH-dk distances are simultaneously below 4 Å. There are 46 such configurations for the A1-dkD complex, spanning the whole length of the simulation (the first reactive structure is observed after around 0.25ns, the last after 19.97ns). There are only 4 reactive configurations found for A1-dkL, all in the first 0.554ns.

In conclusion, the L1 and L2 simulations show that the A1 enzyme is able to recognize dkD as a preferential ligand at the level of weak physical interactions. They also show that the system setup of L1 generates trajectories that tend to drift from the initial structure less than those generated with the L2 setup, suggesting initial structures for the L2 simulations carry significant internal strain. This might be due to the restrain on the reactive hydrogen bonds that were added in the generation of the initial configurations for L2/S2. Instead of facilitating the generation of an initial structure close to the reactive configuration, these restrains may have introduced some strain that is reflected in a more mobile ligand. On the other hand, it is still remarkable that the correct ligand dkD finds its way back to a configuration much closer to the reactive structure compared to dkL.

In both L1 and L2, when the ligand abandons the reactive configuration, it moves away within the first few nanoseconds (L2 for A1-dkD, L1 and L2 for A1-dkL). In the L2 simulation of the A1-dkD complex the substrate returns to a configuration close to the reactive one. In the other cases, during the rest of the simulation dk does not get close again to a reactive configuration. A different simulation strategy may be in place if we

want to understand the behavior of the complex in the neighborhood of the reactive configuration. Such a strategy is described in the next section.

EVIDENCES FROM THE SHORT SIMULATIONS SETS: S1

Twenty S1 trajectories are generated starting from SolvM1 configuration with different initial velocities. The initial configuration for the S1 simulation set is not equilibrated, so the configurations tend to drift rapidly from the initial structure. Nevertheless, it is interesting to look at the average displacement of the ligand (see eq. (4.1)) and at the number of reactive configurations found. The data is reported in Table 10.

	dkD		dkL	
	<RMS>	# React Conf	<RMS>	# React Conf
A1	2.07Å	3957	2.51Å	1163

Table 10 Average displacement from the initial structure (see definition in Eq.(4.1)) and number of reactive configurations (i.e. instances in which NADPH-dk distance and Y371-dk distance are simultaneously below 4Å) for A1-dkD and A1-dkL complexes in the S1 simulation set.

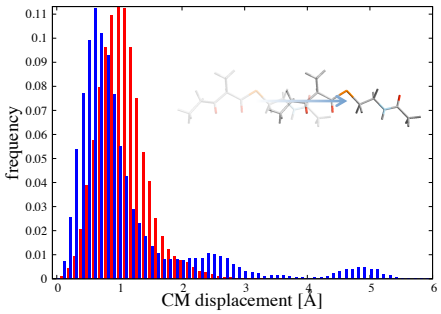
The A1-dkD has more reactive configuration and the ligand moves less from the initial condition. The short simulations set S1 reproduces this qualitative feature of the L1 simulation set. Why is the dkL ligand moving more than the dkD ligand? The incorrect ligand may abandon the reactive configuration because of three different types of movement:

1. A rigid body displacement of the ligand out of the binding pocket;
2. A rigid body rotation of the ligand in the binding pocket;
3. A distortion of the internal degrees of freedom.

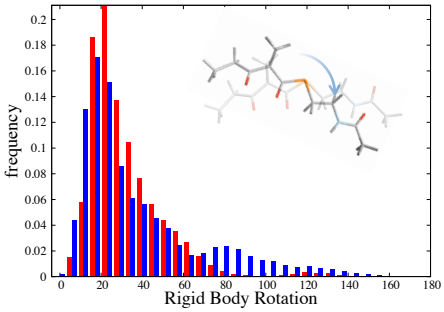
Figure 36 shows:

A. the distribution of the center of mass displacement in the binding pocket for dkD (red) and dkL (blue) with respect to the initial position;
B. the distribution of the angles of rigid body rotation for dkD (red) and dkL (blue) respect to the initial position;
C. the “end-to-end” distance of the ligand, i.e. the distance between the initial and final methyl group for dkD in the A1 enzyme (red), for dkL in the A1 enzyme (blue), and for dkD and dkL in solution (green and gold, respectively).

(A)



(B)



(C)

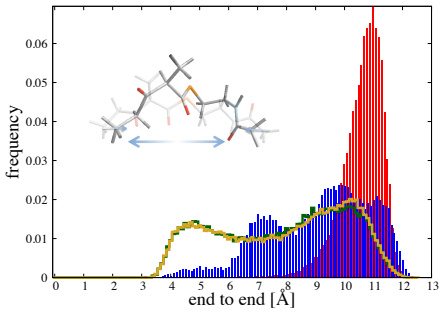


Figure 36

Figure 36: Different types of movement of the ligand in the binding pocket. (A) Displacement of the center of mass of the ligand; (B) rigid body rotation; (C) change of the internal degrees of freedom, measured as the end to end distance, i.e. the distance between the methyl carbon C1 and the methyl carbon C10 (see Figure 37 for the naming of the atoms). In red we show the distribution for dkD, in blue for dkL. In (C) the green and gold histograms report the distribution of end-to-end distances in solution for dkD (green) and dkL (gold). The rigid body displacements were computed aligning all the trajectory configurations to the initial structure. The atoms used for the alignment were those belonging to the residues around the binding pocket (residues A313-A317, W363-Y371, W397-M405, P409-Q418, and E451-A459). The translation is defined as the distance from the center of mass of the ligand in the initial configuration. The overall rotation was computed by aligning the ligand (already rotated to account for the changes in the position of the binding pocket) with the ligand in the initial configuration. From the rotation matrix $\hat{\Omega}$ we extracted the rotation angle as $\arccos\left\{\frac{1}{2}\left[\mathrm{Trace}\left(\hat{\Omega}\right)-1\right]\right\}$. There was no control on the rotation axis, which may change in different configurations.

From Figure 36A,B, it is clear that the overall translations and rotations are similar for the correct and incorrect substrates; indeed it even seems that the correct ligand undergoes a larger motion than the incorrect one.

The sharply peaked distribution of the end-to-end distance of the correct substrate in Figure 36C illustrates that dkD is held stretched by the enzyme. In contrast the dkL ligand has a much broader distribution of distances that resembles the distribution of end-to-end distance of both the enantiomers in solution. This clearly suggests that the incorrect ligand is unable to "hook" to the protein with specific interactions.

The Reactive Configuration

The significant sample of short trajectories allows us to examine how the enzyme in the crystal structure responds to the two ligands (dkD and dkL) placed in a reactive configuration. Recall that we define a reactive configuration by only two distance constraints between the ligand, the protein and NADH (one with the β-carbon C3 and one with the β-carbonyl oxygen O3, see Figure 37 for the name of the atoms). It is not clear if these restraints are sufficient to uniquely define the structure of the bound complex. We therefore examine below additional degrees of freedom.

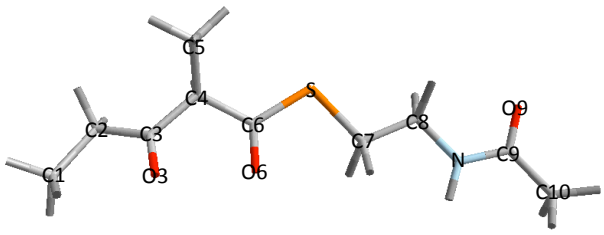


Figure 37 A schematic drawing of the diketide substrate with explicit atom names.

We consider the dihedral angles of the ligand that include only carbon or nitrogen atoms (the main chain). A histogram of the distribution of these dihedral angles in the trajectories (with bin size 30 degrees, so 12 bins from -180 to 180) was computed. Let's call $p_{i\alpha}$ the frequency of times the dihedral α is found in the i-th bin. Then, to measure the spread of the distribution, we use the "entropy"[10] S_α

$$S_\alpha = -\sum_{i=1}^{12} p_{i\alpha} \ln p_{i\alpha} \tag{4.2}$$

[10] Since it is not an equilibrium calculation, we do not mean to give to this "entropy" a thermodynamic meaning but rather information context. To remind this to the reader, we will call this quantity S "entropy", with quotation marks.

This function is positive definite, and maximal when the distribution is uniform (so, when the information is minimal and the spread maximal). The larger is the value of the function, the least informative, or the more spread, is the distribution.

In Table 11 we report the values of the "entropies" measured on the whole set of configurations, only on the reactive configurations, and in a 50ns simulation of the two enantiomers in water. The calculations were conducted for both ligands A1-dkD and A1-dkL.

	A1-dkD		A1-dkL		dkD	dkL
	all	reactive	all	reactive		
10 C1-C2-C3-C4	1.94	1.51	2.16	2.18	2.11	2.11
13 C2-C3-C4-C5	1.82	1.26	2.10	2.10	1.86	1.87
14 C2-C3-C4-C6	1.81	1.26	2.10	2.08	1.88	1.88
19 C3-C4-C6-S	1.84	0.97	1.54	1.36	2.12	2.13
23 C5-C4-C6-S	1.85	0.99	1.55	1.34	2.11	2.12
28 C4-C6-S-C7	0.71	0.71	0.72	0.72	0.71	0.71
30 C6-S-C7-C8	1.59	0.60	1.63	1.59	1.89	1.91
33 S-C7-C8-N	1.05	0.84	1.23	1.19	1.45	1.48
42 C7-C8-N-C9	1.23	0.58	1.27	1.31	0.92	1.02
45 C8-N-C9-C10	0.70	0.67	0.68	0.68	0.70	0.70
TOTAL	14.54	9.39	14.98	14.55	15.75	15.93

Table 11 "Entropies" of the different torsions in the diketide. The torsions are reported in the first column. The second and third column show the "entropy" of the torsions for the A1-dkD complex in all the configurations (second) and in the reactive configurations (third). The fourth and fifth columns show the "entropies" of the torsions for the A1-dkL complex. The fourth column shows the result for all the configurations, the fifth for the reactive configurations. The sixth column has the "entropies" of the torsions for the dkD ligand in water, the seventh for the dkL ligand in water.

In the A1-dkD complex, the reactive configurations are always characterized by a reduction of the "entropy" S. This happens also with torsions 33 and 42, which are far away from the atoms whose positions is used to define the reactive configuration. This is not the case for A1-dkL: the "entropy" S in the reactive configurations is smaller for some dihedrals, though larger for others. The selection of a specific reactive configuration and locking into a particular configuration seem to be properties of only the correct complex, A1-dkD. Also, the sum of the "entropies"[11] of the dkL ligand is larger than for the dkD, in agreement with what we stated before: the internal degrees of freedom of the incorrect ligand are looser. The last two columns show the "entropy" of all the torsions for the two stereoisomers of the ligand in water solution. Note that, in the case of the dkD molecules, with the exception of torsion 42, all the dihedrals in water have larger "entropy". For the dkL molecule instead, the "entropy" in the binding site is larger than in water for dihedral 10, 13, 14, 28 (almost the same), and 42. The flexibility of the α and β carbon region of the dkL ligand in the binding pocket suggests highly unfavorable interactions with protein groups around the reactive site.

A pictorial representation of this difference is given in Figure 38. First, we show ten dkD ligands randomly selected from reactive configurations, and then aligned (Figure 38A). Then we show ten structures taken randomly from the whole sample of structures (Figure 38B). Clearly, imposing the two conditions on the β-carbonyl carbon and oxygen reduces the configurations available throughout the ligand. The same conditions do not reduce significantly the flexibility of the dkL ligand in reactive configurations (Figure 38C): the superposition of the randomly chosen structures shows poor structure selection

[11] The sum of the "entropies" corresponds to the total "entropy" under the assumption that all the torsions are independent. This is, of course, false. Nevertheless, it is a simple representation of the overall flexibility of the ligand.

for the incorrect enantiomer, making Figure 38C similar to the sample of ten structure from the whole simulations of A1-dkL shown in Figure 38D.

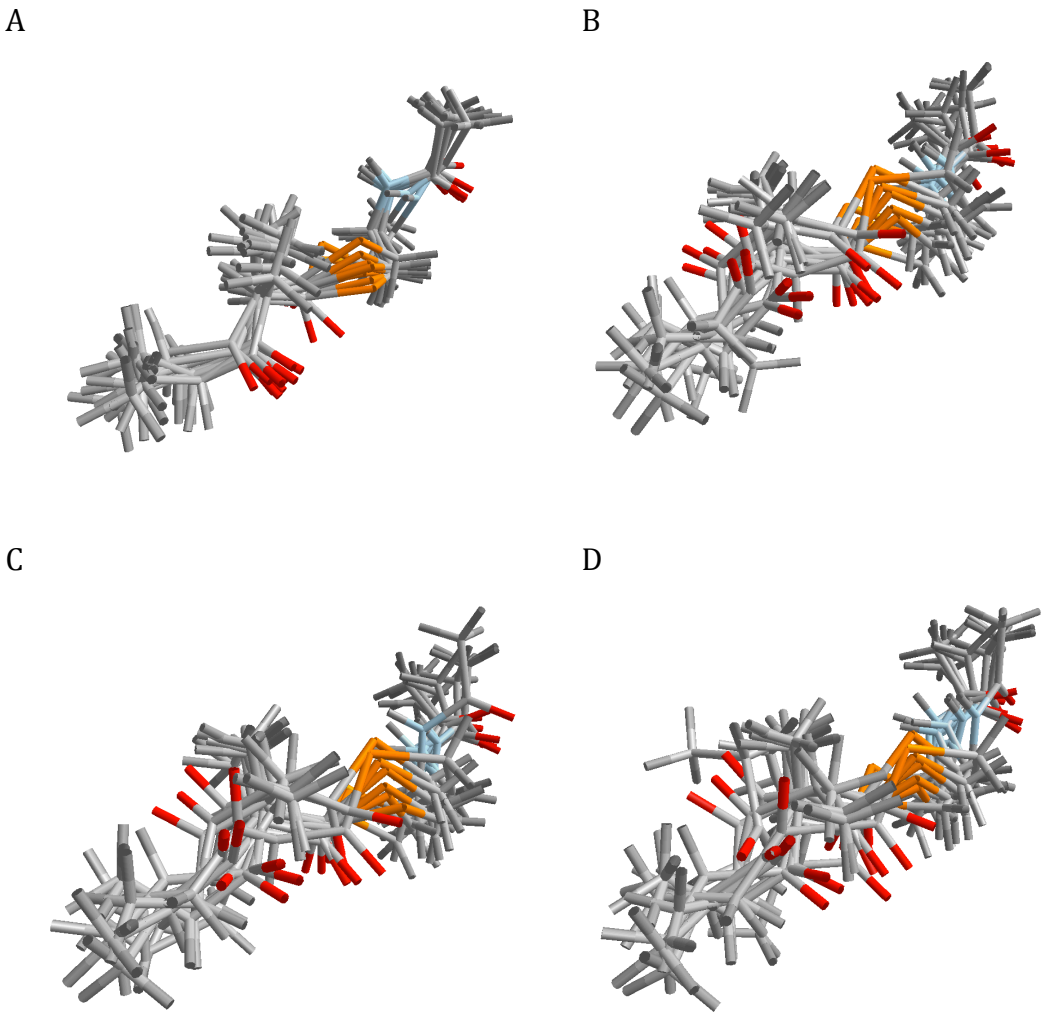


Figure 38 Superposition of ten structures of the dk randomly selected among the following sets of structures: (A) dkD in reactive state, (B) dkD in any enzyme-ligand configuration, (C) dkL in reactive state, (D) dkL in any enzyme-ligand configuration.

A typical reactive configuration obtained from our simulation for the A1-dkD complex is shown in Figure 39.

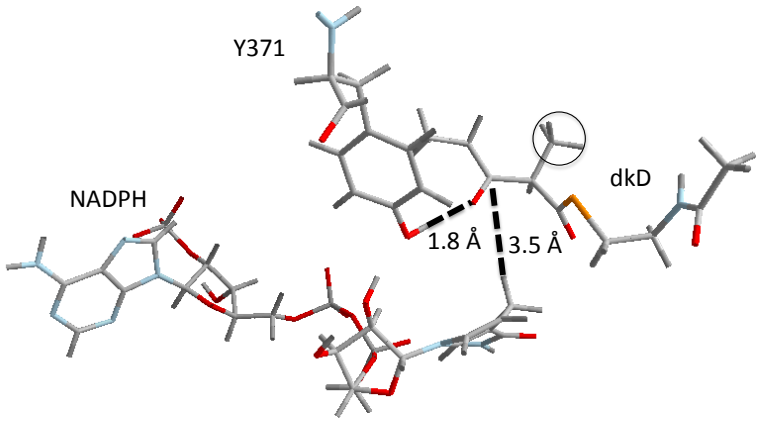


Figure 39 A reactive configuration for the A1-dkD complex. Only the ligand (dkD), the reactive tyrosine (Y371), and the NADPH are shown. In dashed lines the NADPH-dk and Y371-dk distances are shown.

In Figure 39 we show a snapshot from the trajectory of a reactive configuration of Y371 together with the NADPH and with dkD. The typical reactive configuration is characterized by a rotation of the non-reactive β-carbonyl away from Y371. At the same

time, the sulfur (in orange) is moved towards the reactive oxygen. As shown in Figure 38A, this feature seems to be common to many reactive configurations, while this is not true in general (see Figure 38B).

The vicinity of the sulfur to the β-carbonyl oxygen in the reactive configuration might play a role in facilitating the reaction catalyzed by the enzyme. Indeed, the vicinity of two strongly electronegative atoms may assist the transfer of a proton from Y371 to the β-carbonyl oxygen, and reduce the electrostatic repulsion. The repulsive energy between the sulfur and the reactive oxygen is shown in Figure 40, where the red histogram shows the distribution of interaction energies (electrostatic and van der Waals) in all the configurations found, the green for the non-reactive configurations and the blue for the reactive. Clearly, in the reactive configurations there is a strong repulsion between the sulfur and the β-carbonyl oxygen.

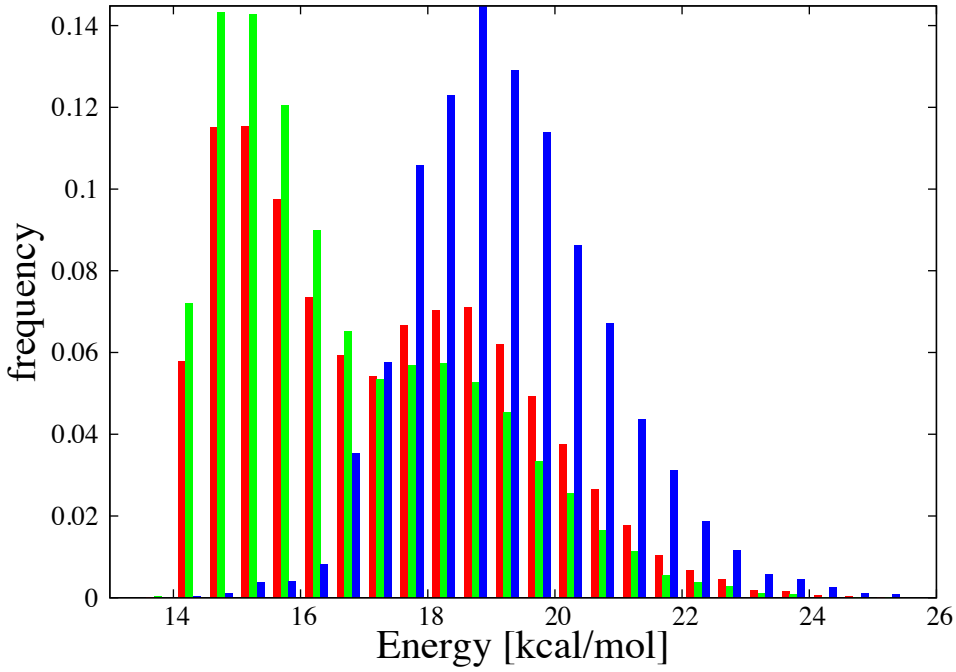


Figure 40 Distribution of energies of interaction (van der Waals and electrostatic) between the sulfur of the ligand S and the β-carbonyl oxygen O3 (for nomenclature see Figure 37) for the A1-dkD complex. In red we report the distribution for all the configurations, in green the distribution for the non-reactive ones, and in blue the distribution for the reactive configurations.

At this stage, the role played by the repulsion between the sulfur and the β-carbonyl oxygen in facilitating the reaction is suggestive. Quantum mechanical calculations could further assist in determining whether the interactions include higher order terms such as polarization and charge transfer. Moreover, extensive sampling will be required to learn about thermodynamics and kinetics of the process. It has been

pointed out in many circumstances how important is the role of electrostatic interactions in the catalytic power of enzyme.[98]

Microscopic Reasons For Specificity

So far, we illustrate by analyses of trajectory data that formation of weakly bound protein-ligand complex is favorable for the correct ligand. In particular we showed that more reactive configurations are found, and that they are characterized by a specific geometry (Figure 38) that fits the desired protein function (Figure 39). We did not discuss the microscopic interactions responsible for the selection of the proper enantiomer; a topic which is addressed in the present section.

We start by looking at NADPH and Y371, which are both involved in the reaction. We ask "Is their behavior significantly different for the correct and incorrect binding cases?" It seems that it is not. NADPH is strongly bound for both the correct and the incorrect complex (average RMSD of 1.03Å and 1.02Å, respectively). The same is true for the reactive tyrosine Y371 (average RMSD of 0.91Å and 0.92Å, respectively). There is a strong hydrogen bond between the 2' hydroxyl group of the ribose ring of NADPH and Y371 side-chain oxygen (see Figure 41), which holds in position the tyrosine ring. This is regardless of the chirality of the substrate.

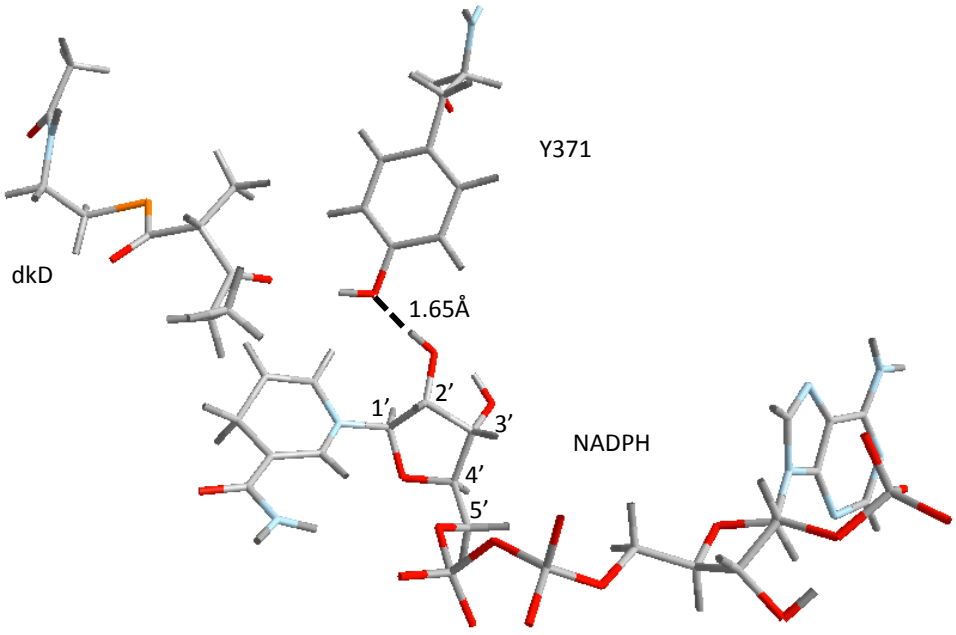


Figure 41 dkD ligand, NADPH and reactive tyrosine Y371 shown together as in Figure 39, but from a different angle. The hydrogen bond between the 2' hydroxyl group and Y371 side-chain oxygen is shown in dashed lines. The carbons in the ribose ring of NADPH are numbered.

The distributions of the hydrogen bond distance for the correct (red) and incorrect (blue) enzyme-ligand pairs are shown in Figure 42.

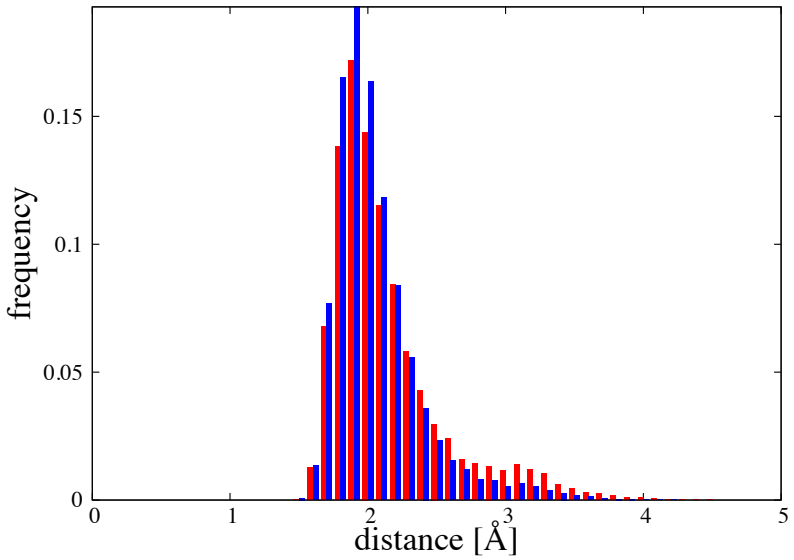


Figure 42 Distribution of the hydrogen bond distance between the NADPH 2' hydroxyl hydrogen and Y371 side-chain oxygen (atoms connected with a dashed line in Figure 41) for A1-dkD (red) and A1-dkL (blue).

The hydrogen bond between the amide carbonyl group of the ligand and the side-chain of W363 is considered important for properly aligning the ligand in the binding pocket (see Figure 43).

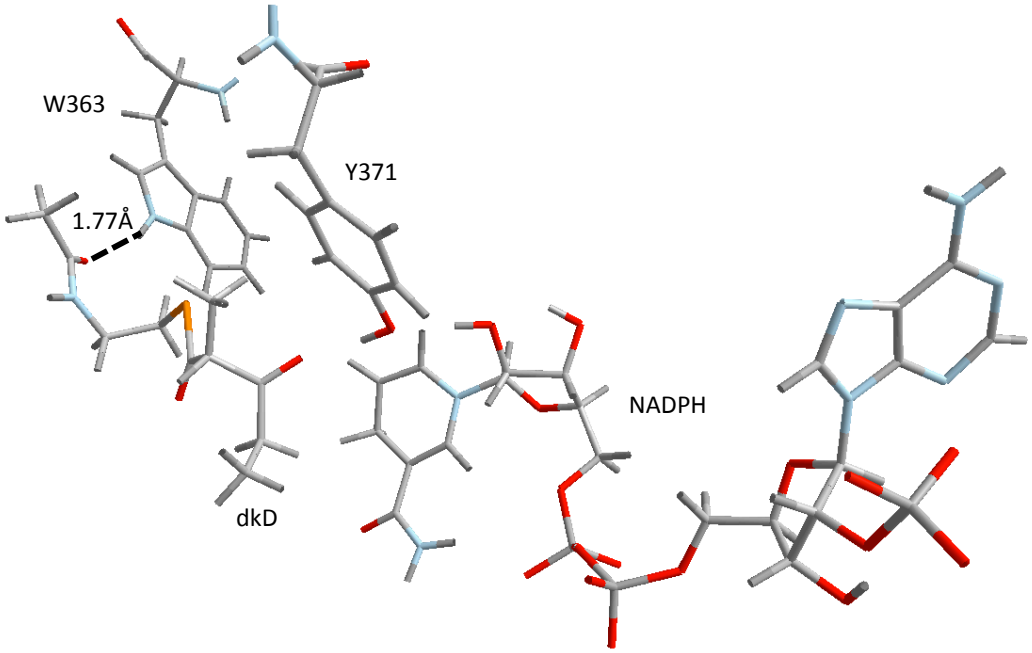


Figure 43 dkD ligand, tyrosine Y371, NADPH, and tryptophan W363 are displayed in a reactive configuration. The hydrogen bond between the dk oxygen O9 (see Figure 37 for the nomenclature) and the side-chain of W363 is shown as a black dashed line.

Figure 44 shows that this hydrogen bond is tighter in the case of the correct complex (A1-dkD, red histogram Figure 44A) than with the wrong one (A1-dkL, red histogram Figure 44B).

(A) (B)

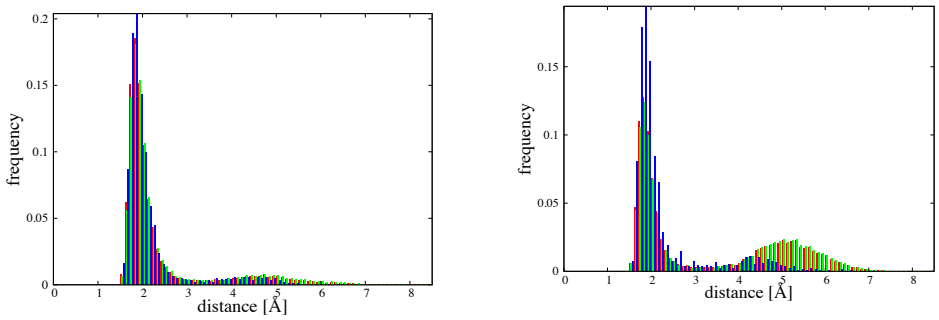


Figure 44 Distribution of distances between W363 side-chain and the O9 oxygen of the dk ligand (see Figure 37 for nomenclature and the dashed black line in Figure 43) for A1-dkD complex (A) and A1-dkL complex (B). The red histograms show the distribution of distances in all configurations. The green histograms show the distribution of distances for non-reactive configurations, and the blue ones the distribution for reactive configurations.

To explore the role of the α-methyl substituent, we looked at what is the amino acid whose center of mass is closer to the α-methyl substituent at every instant of time. The data for the A1-dkD complex for reactive (green) and non-reactive (red) configurations is shown in Figure 45, together with the two distributions for A1-dkL complex: in orange the one for reactive configurations, and in blue the one for non-reactive configurations.

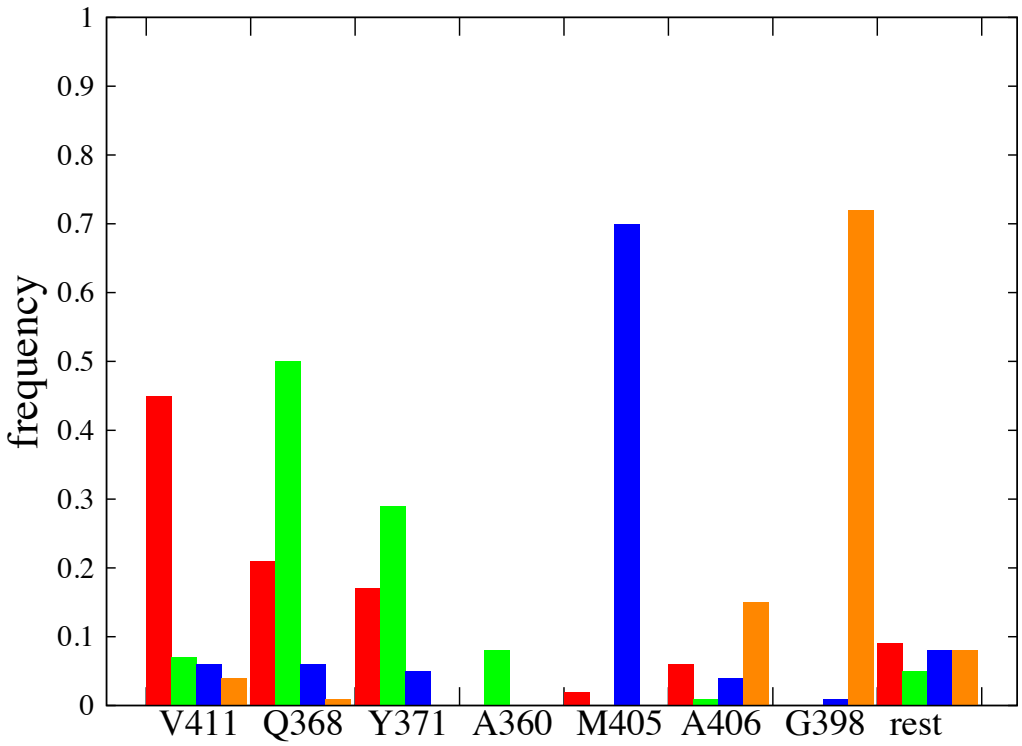


Figure 45 Histogram of the number of times the center of mass of the amino acid listed on the x-axis is the closest to the α-methyl substituent. Red: A1-dkD complex, non-reactive configurations. Green: A1-dkD complex, reactive configurations. Blue: A1-dkL complex, non-reactive configurations. Orange: A1-dkL complex, reactive configurations.

In Figure 46 and Figure 47, the amino acids discussed in Figure 45 are colored as the highest bar in Figure 45. So, V411 is reported in red, A360, Q368, and Y371 are green, M405 is blue and G398 and A406 are orange.

For A1-dkD, the non-reactive configurations are peaked on the V411 amino acid (red, see Figure 46B), which is on the lid helix (see Figure 33A, the lid helix is the helix on the opposite side of the binding pocket with respect to the Y371). The reactive

configurations are instead peaked on Q368 and Y371 (green, see Figure 46A), which are on the opposite side of the binding pocket. We observe from Figure 45 that the non-reactive configurations (red) have a large population also for the Q368 and Y371 amino acids, indicating that the sets of configurations, reactive and non-reactive, are not dramatically different. Note also that in non-reactive configurations the oxygen of the β-carbonyl group of dkD is pointing towards the glutamine 368, as shown by the dashed black line in Figure 46B.

A B

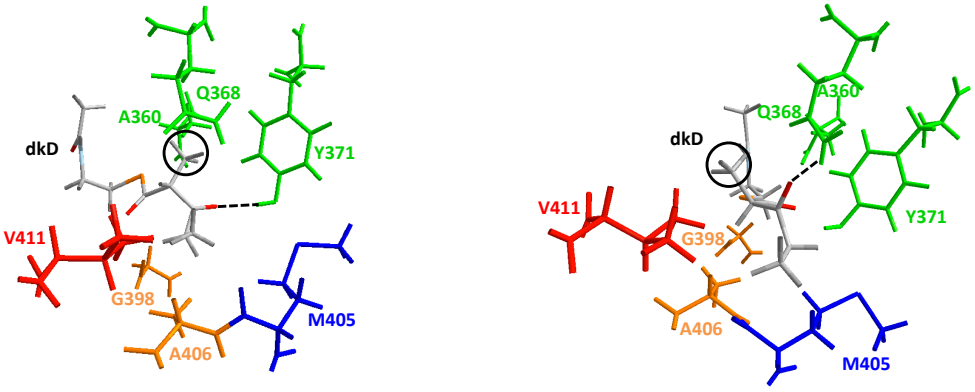


Figure 46 (A) A1-dkD complex, reactive configuration. All the amino acids discussed in Figure 45 are shown. Note that the D-α-methyl substituent (highlighted in a black circle) is close to the green amino acids, particularly very close to Q368. The dashed line shows that the reactive oxygen of dkD is aligned with the reactive hydrogen of Y371. (B) A1-dkD complex, non-reactive configuration. In this case the D-α-methyl substituent (highlighted in a black circle) is close to the red amino acid, V411. The dashed line shows that the reactive oxygen of the β-carbonyl of dk points towards the glutamine 368.

For A1-dkL instead (see Figure 47), the non-reactive configurations (blue, see Figure 45) are peaked close to the M405 amino acid (reported in blue in Figure 47B), which is buried in the binding pocket, away from the cleft, close to the NADPH (non

displayed in Figure 47 for clarity, but it would be under Y371, on the right side of M405). The reactive configurations (orange, see Figure 45) are peaked near G398 (reported in orange in Figure 47A), which is still away from the cleft, but on the opposite side with respect to the NADPH, closer to W363 (not reported on Figure 47 for clarity, but it would be close to A360, on its left). Note that, as in Figure 46B, also in this case for non-reactive configurations the oxygen of the β-carbonyl group of dkL is pointing towards the glutamine 368, shown with the dashed lines.

A B

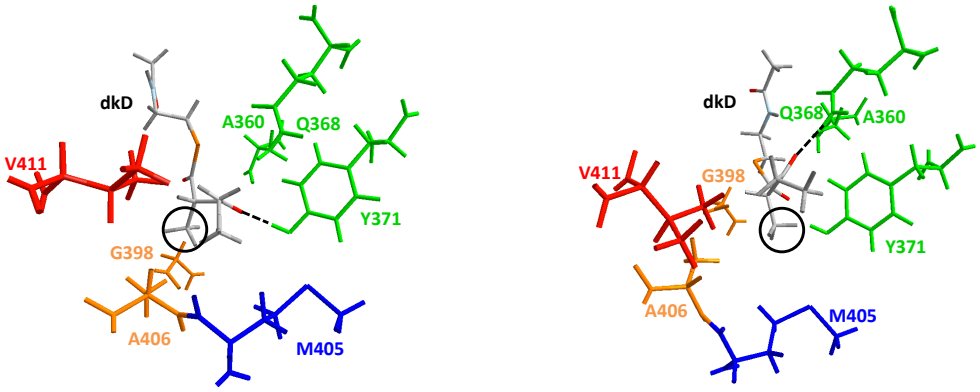


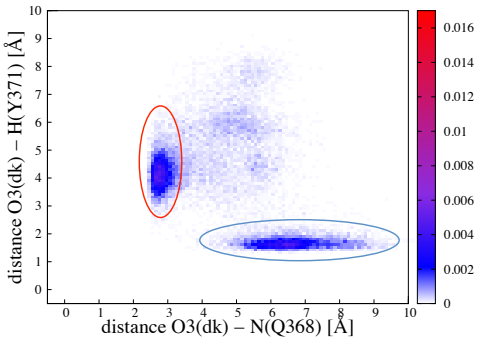
Figure 47 (A) A1-dkL complex, reactive configuration. All the amino acids discussed in Figure 45 are here shown. Note that the L-α-methyl substituent (highlighted in a black circle) is close to the orange amino acids, G398 and A406. (B) A1-dkL complex, non-reactive configuration. In this case the L-α-methyl substituent (highlighted in a black circle) is close to the blue amino acid, M405. Note that the dashed line shows that the reactive oxygen of the β-carbonyl of dk points towards the glutamine 368.

This analysis suggests the presence of two different modes in the binding pocket for the β-carbonyl of the ligand: a reactive one, where it binds with the hydroxyl group of Y371 (Figure 46A and Figure 47A, dashed lines), and a non-reactive one, where it moves

closer to the side chain of Q368 (Figure 46B and Figure 47B, dashed lines). To further assess this conjecture, we plot a two dimensional histogram. On the x-axis there is the distance between the ligand O3 β-carbonyl oxygen and the Q368 side-chain nitrogen (the distance highlighted in dashed lines in Figure 46B and Figure 47B, let's refer to it as Q368-dk). On the y-axis there is the distance between the same dk oxygen O3 and the hydrogen of the hydroxyl group of Y371 (the distance highlighted in dashed lines in Figure 46A and Figure 47A, which we will refer to as Y371-dk, as we did in the previous paragraphs).

In Figure 48A the result is shown for the A1-dkD complex, in Figure 48B for the A1-dkL complex.

A



B

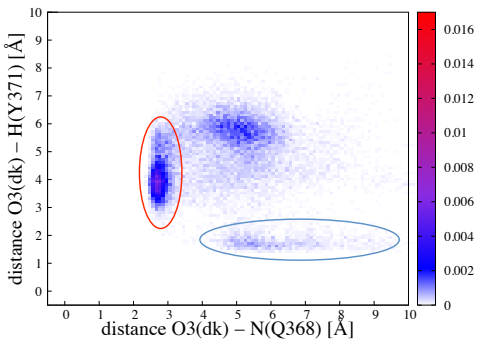


Figure 48 Two-dimensional probability density of critical distances. (A) A1-dkD complex, two-dimensional histogram. The horizontal axis is the distance between the nitrogen on the side-chain of Q368 and the dk reactive oxygen O3 (Q368-dk). The vertical axis is the Y371-dk distance. The probability density is color-coded. The circles highlight the two possible contacts. The red circle corresponds to the region where the oxygen is in contact with Q368 (see Figure 46B, dashed lines), the blue corresponds to the reactive hydrogen bond between Y371 and the β-carbonyl oxygen (see Figure 46A, dashed lines) (B) A1-dkL, same figure. The red region corresponds to the bond highlighted with a dashed line in Figure 47B; the blue region to the one highlighted with a dash line in Figure 47A.

Figure 48A has two clear peaks, one corresponding to a hydrogen bond between the reactive oxygen of dkD and the reactive hydrogen of Y371 (in a blue circle), and another in which instead the reactive carbonyl is closer to Q368 side-chain (in a red circle). In Figure 48B we notice that the first of the two peaks is significantly weaker (blue circle), while the second is more pronounced (red circle). This suggests that glutamine Q368 is able to "fish out" the reactive carbonyl from its reactive configuration more efficiently in the A1-dkL enzyme than in the A1-dkD.

What is the role of the α-methyl substituent in this different behavior? Figure 46A shows that the D-α-methyl substituent in a reactive configuration places itself between the reactive carbonyl and Q368, while in a non-reactive configuration (Figure 46B) it rotates away from Q368, facilitating the interaction between the reactive carbonyl of the ligand and the amide group of Q368 side-chain. In the A1-dkL complex (Figure 47) the L-α-methyl substituent is oriented towards the inside of the cleft and is unable to perturb the interaction between the reactive carbonyl and Q368. These features are common to a significant number of reactive configurations, as we can illustrate looking at the histogram of the following order parameter:

$$T = d(\text{dkO3, dkC5}) + d(\text{dkC5, Q368}) - d(\text{Q368, dkO3}) \qquad (4.3)$$

where $d(a,b)$ is the distance between a and b, dkO3 is the reactive oxygen of the ligand, dkC5 is the α-methyl substituent carbon and Q368 is the position of the nitrogen of the glutamine residue. There are three possibilities, as illustrated in Figure 49:

A. The triangle with vertexes dkO3, dkC5, and the nitrogen of Q368 becomes close to a line segment with dkC5 lying between dkO3 and the nitrogen of Q368. In this case $T \approx 0$;

B. The three atoms form a triangle; the further away dkC5 is from the segment connecting dkO3 and Q368, the larger is T ,

C. dkC5 is close to the line identified by dkC5 and Q368, but outside of the line segment connecting dkO3 and Q368; in this case T is positive and roughly twice as large as the distance between dkC5 and the closest other atom.

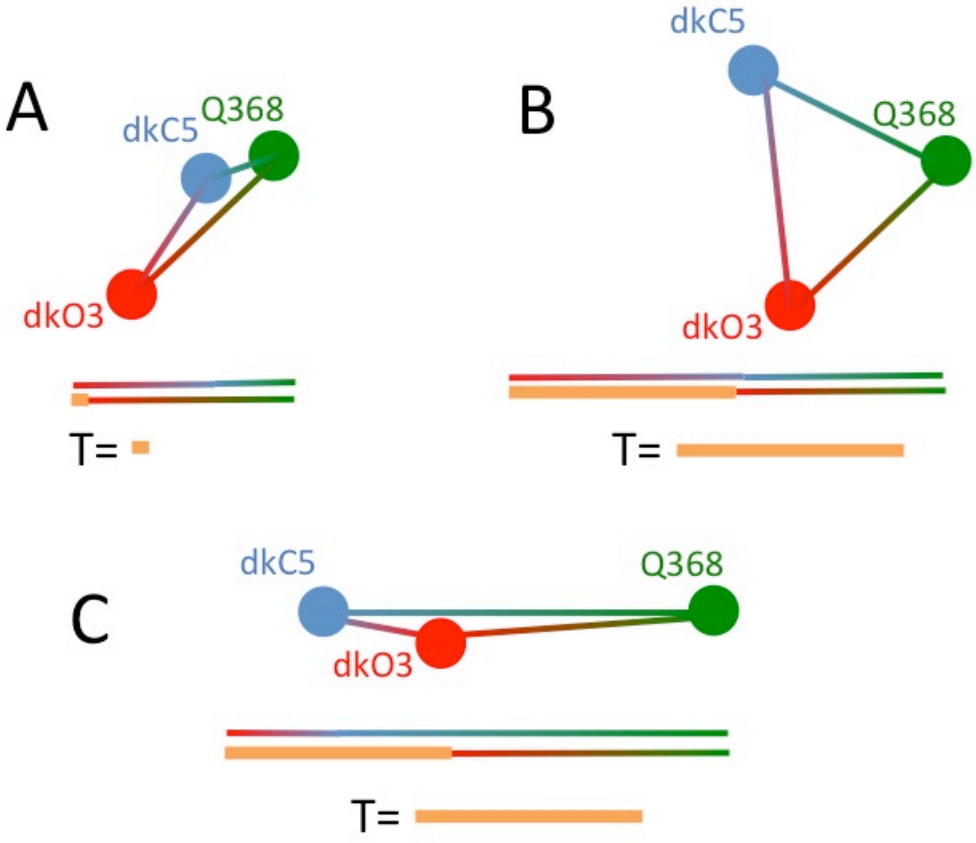


Figure 49 Three pictorial illustrations of the size of the parameter T in three different cases. In red the dkO3 is reported, in blue the dkC5, and in green Q368. The orange thick line represents the parameter T , as introduced in Eq.(4.3). (A) The case in which the three atoms almost lie on the same line, with dkC5 between dkO3 and Q368. The parameter T is close to zero. (B) When the triangle is formed, T is large. (C) If the three atoms almost lie on the same line, but with dkC5 outside of the segment connecting dkO3 and Q368, then the parameter T is large.

In Figure 50A T is shown for the A1-dkD complex for all the configurations (red), for the non-reactive configurations (green), and for the reactive configurations (blue). The same color code holds for Figure 50B, where T is shown for the complex A1-dkL.

A B

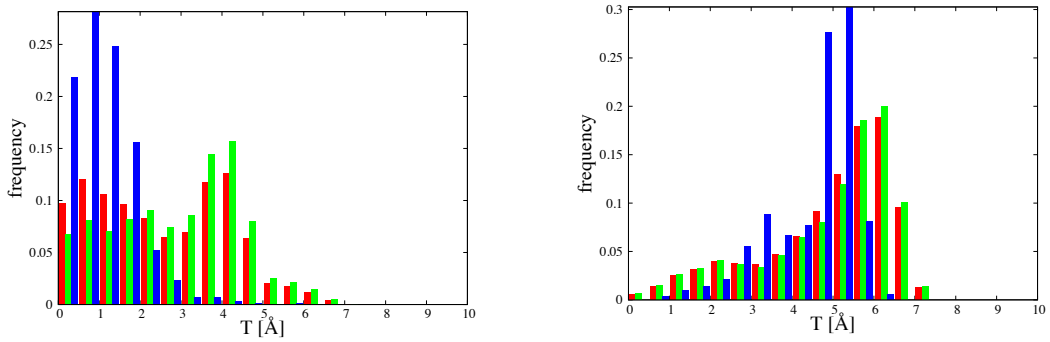


Figure 50 (A) Parameter T (see equation (4.3) for the definition) for A1-dkD complex. In red the distribution for all the configurations is reported, in green the distribution for the non-reactive and in blue for the reactive ones. (B) Same as (A) but for the complex A1-dkL.

Figure 50 shows that the distribution of T has a peak for small values with the A1-dkD complex in the reactive state (Figure 50A, blue), suggesting that indeed the dkC5 places itself between dkO3 and Q368, keeping them away and reducing the probability that Q368 will interact with the reactive carbonyl. This is not true for the A1-dkL complex (Figure 50B).

Experiments have shown a drastic reduction of specificity upon mutation of Q368 with histidine,[74] an amino acid that cannot form a strong bond with the β-carbonyl group.

The MD simulations suggest the reason of the importance of this amino acid for the stereospecificity of the A1 enzyme.

EVIDENCES FROM THE SHORT SIMULATIONS SETS: S2

We showed that in the long simulations L1 and L2, the initial structure used for simulations L2 is further away from the bound state. Indeed, in L2 (Figure 34 and Figure 35, red and blue lines) we observe significant movement of the ligand, which is not seen in the L1 simulation (Figure 34 and Figure 35, green and orange). The short simulation set S2 starts from a configuration (solvM2) similar to one of the L2 simulations. Hence, it is in a configuration that is further away from the bound state. Nevertheless, it is instructive to see what is happening to the ligand in these simulations.

First we examine the perturbation of Q368 interaction to the reactive configurations of the ligand. Figure 51 is the same as Figure 48 but for the S2 simulation set.

A B

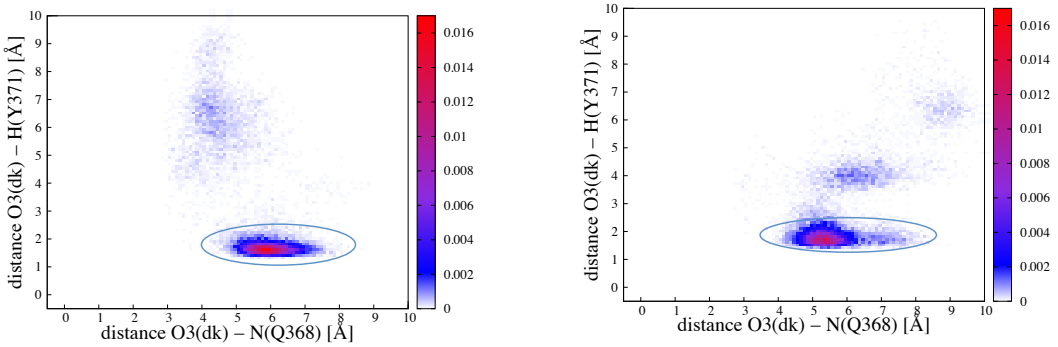


Figure 51 (A) A1-dkD complex, S2 simulation set, two-dimensional histogram. On the x-axis the Q368-dk distance is shown. On the y-axis the Y371-dk distance is shown. The probability is color-coded. The blue circle corresponds to the reactive hydrogen bond between Y371 and the β-carbonyl oxygen (see dashed line in Figure 46A for the S1 simulations set) (B) A1-dkL, same figure. In this case the blue region corresponds to the interaction highlighted with a dash line in Figure 47A for the S1 simulation set.

As expected, the peak corresponding to the Y371-dk hydrogen bond formed (circled in blue in Figure 51) is larger than for the S1 simulation set (Figure 48). This is because in the generation of the starting configuration we imposed the restraint on the distance Y371-dk.

Clearly there is no sign of strong interaction between the ligand and the Q368, as highlighted by the fact that the region circled in red in Figure 48 is missing in Figure 51. To understand why this is the case, we looked at the changes in the structure of the enzyme around Q368, and compared them with sets L1, L2, and S1. In Figure 52 we show the average RMSD (see eq.(4.1)) of some amino acids near Y371, and the diketide for both A1-dkD and A1-dkL complex (Figure 52A and Figure 52B, respectively). In red

we show the results for the L1 simulation set, in green for the S1, in blue for the L2 and in orange for the S2.

A B

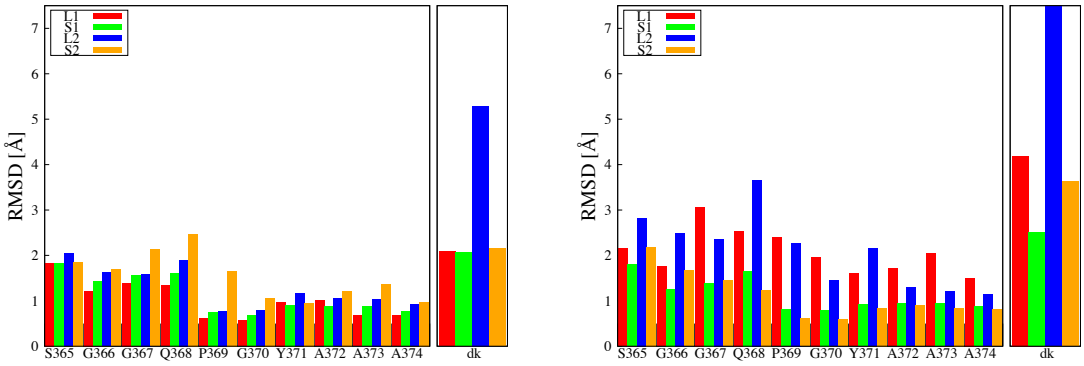


Figure 52 (A) RMSD (see eq(4.1)) of some amino acids around Y371 for the complex A1-dkD. As highlighted in the key, in red it is shown the result of the L1 simulation, in green for the S1, in blue for L2, and in orange for S2. As a reference initial structure we used the M1 structure for L1 and S1 simulation sets, and the M2 structure for the L2 and S2 simulation sets. (B) Same as (A) but for the A1-dkL complex.

For the A1-dkD complex (Figure 52A), the S2 simulation set (orange) shows the largest displacements from the initial structure. The L2 simulation set (blue) is consistently second. This observation suggests that the restraint between the ligand and Y371 in the generation of the initial structure M2 added strain that is released by distorting the structure. In the case of the A1-dkL complex (Figure 52B), the picture is different. The long simulations (L1 in red, L2 in blue) deviate the most from the initial structure, as expected. On the other hand, if we look at the average displacement of the ligand, we note that in the S2 simulation set (orange), even though it is the shortest set of simulation, the dkL ligand moves away from the initial position much more than in the

S1 simulation set (green), almost as much as in the L1 (red), where rarely reactive configurations are found.

In both cases it seems that the M2 initial structure has generated some strain in the protein. This is released in the dkD case with a displacement of the amino acids from the initial structure that is larger than the RMSD of the same amino acids in the L1/S1 simulation sets. In the A1-dkL, the strain is released by displacing the ligand away from the reactive configuration.

What is happening in the A1-dkL enzyme then? What interactions with the ligand are looser than in the A1-dkD complex, so to justify the excess average RMSD of the ligand?

There is a network of interactions that position properly the Y371 side-chain to form the hydrogen bond with the β-carbonyl group of the ligand. In Figure 53 we highlight in black and red two of them: in black we show the hydrogen bond between the 2' hydroxyl group of the NADPH ribose ring (see Figure 41 for nomenclature) and the hydroxyl group of the side-chain of Y371; in red we show the interaction between the side-chains of S358 and Y371. The configuration reported here is one of the 3957 reactive configurations for the A1-dkD complex of the S1 set of simulations.

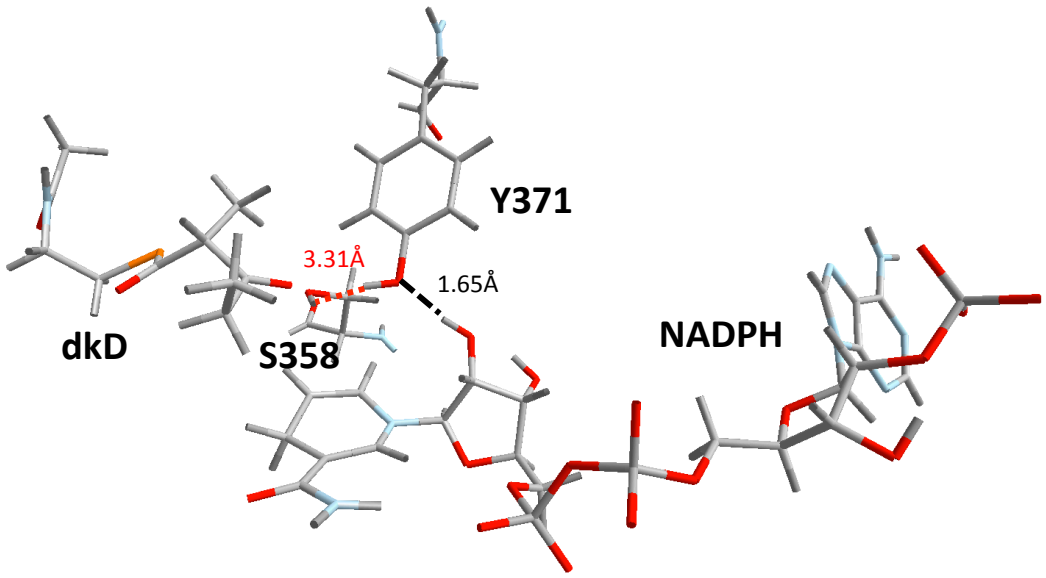


Figure 53 A reactive configuration for the A1-dkD complex during the S1 set of simulations. A black dashed line denotes a hydrogen bond between the 2' hydroxyl group of the ribose ring of NADPH (see Figure 41 for nomenclature) and the side-chain of Y371. A red dotted line illustrates the interaction between S358 side-chain hydroxyl group and Y371. The distance is between the serine's hydrogen and the tyrosine's oxygen.

This pattern can be replaced by others in which the hydroxyl group of Y371 rotates to point towards either the serine (Figure 54A) or to the 2' oxygen of NADPH (Figure 54B).

A B

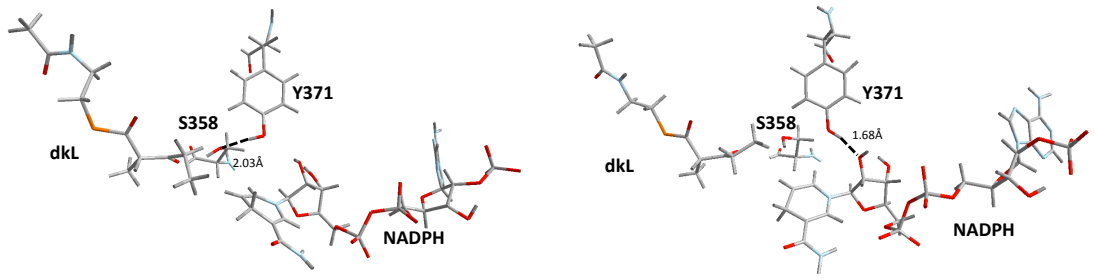


Figure 54 (A) A configuration of the complex A1-dkL extracted from the S2 simulations in which the hydroxyl group of Y371 side-chain points towards the oxygen of S358 serine instead of towards the reactive β-carbonyl group. The dashed black line highlights this interaction. (B) A configuration of the complex A1-dkL extracted from the S2 simulations in which the hydroxyl group of Y371 side-chain, instead of pointing towards the reactive β-carbonyl group, points towards the 2' oxygen of NADPH ribose ring (see Figure 41 for nomenclature). The dashed black line highlights this interaction.

Of course, the occurrence of these configurations reduces the probability of being in a reactive conformation, and weakens an attractive interaction between the ligand and the enzyme. Figure 55 shows the distribution of configurations as a function of the distances between the oxygen of the side-chain of S358 and the hydrogen of Y371 side-chain (x-axis), and the distance between the same hydrogen of Y371 and the 2' hydrogen of NADPH ribose ring (y-axis). In the S1 simulation (Figure 55A-B), the hydroxyl group of Y371 side-chain rarely, and almost exclusively for the A1-dkL complex (Figure 55B), forms a hydrogen bond with S358 (highlighted by a red circle). In the S2 simulation set (Figure 55C-D), there is a very strong peak for S358-Y371 hydrogen bond for A1-dkL (Figure 55D, red circle).

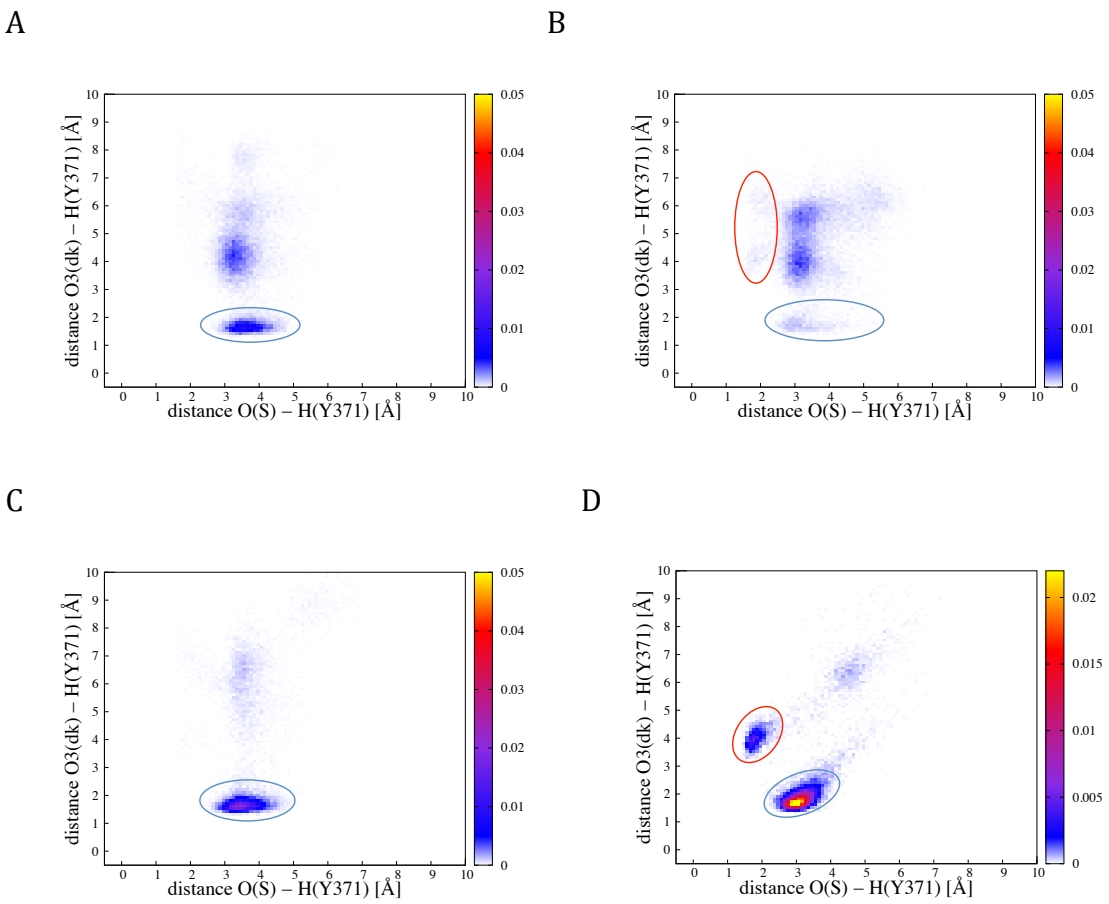


Figure 55 Two-dimensional distribution of distances between the reactive hydrogen of Y371 and the oxygen of S358 side-chain (horizontal) (see dashed line in Figure 54A), The vertical direction is the reactive β-carbonyl oxygen of the ligand. (A) A1-dkD complex, S1 simulation set. (B) A1-dkL complex, S1 simulation set. (C) A1-dkD complex, S2 simulation set. (D) A1-dkL complex, S2 simulation set. In a blue circle all the regions showing abundance of configurations with the reactive hydrogen bond formed are highlighted. In a red circle, we highlight the regions with abundance of configurations showing a strong interaction between S358 and Y371.

In Figure 56, we show the probability that Y371 forms an hydrogen bond with the β-carbonyl group of the ligand (regions highlighted in a blue circle), and the probability of having the same hydroxyl of Y371 rotating towards the NADPH and forming an hydrogen bond with the 2' oxygen of the ribose ring as in Figure 54B (regions highlighted in a red circle). The non-reactive hydrogen bond with NADPH is rarely formed during the S1 simulation set (Figure 56A-B), much more commonly during the S2 simulation set, and both for A1-dkD (Figure 56C) and A1-dkL (Figure 56D).

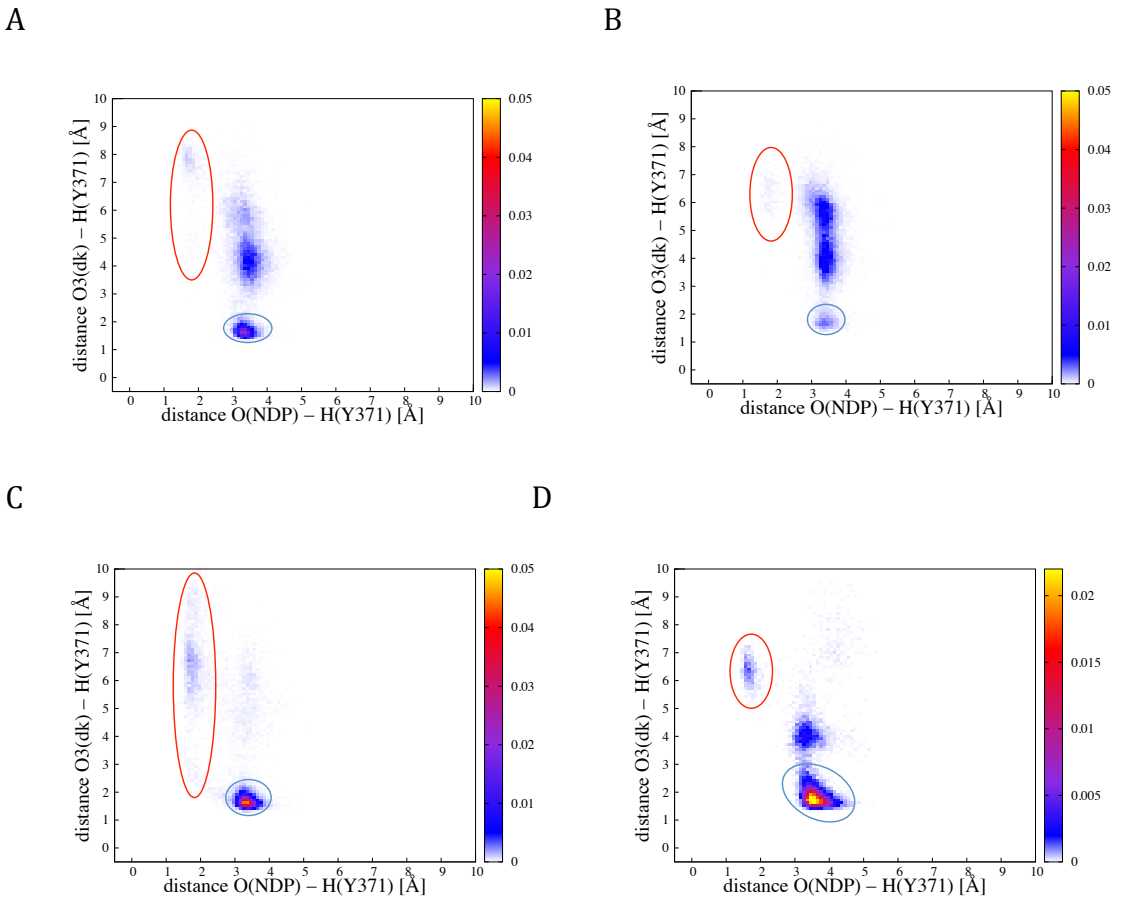


Figure 56 Two-dimensional distribution of distances between the reactive hydrogen of Y371 and, on the x-axis the 2' oxygen of NADPH ribose ring (see dashed line in Figure 54B), on the y-axis the reactive β-carbonyl oxygen of the ligand. (A) A1-dkD complex, S1 simulation set. (B) A1-dkL complex, S1 simulation set. (C) A1-dkD complex, S2 simulation set. (D) A1-dkL complex, S2 simulation set. In a blue circle all the regions showing abundance of configurations with the reactive hydrogen bond formed are highlighted. In a red circle, we highlight the regions with abundance of configurations showing the hydrogen bond between Y371 and NADPH.

The last two figures showed that in the S2 simulation set the ligand often abandons the reactive configuration because the hydroxyl group of Y371 rotates towards

a neighbor polar oxygen and forms a hydrogen bond with it. The same mechanism was possible, but is less frequent within the S1 simulation set.

ANALYSIS OF THE DYNAMICS OF THE D AND L DIKETIDES IN VACUUM, AQUEOUS SOLUTION, AND IN THE BINDING POCKET

The diketide undergoes different fluctuation in different environments. We compare 100ns simulation carried out in vacuum using a Langevin dynamics algorithm[99] at 300K with 50ns carried out in solution. We also compare the fluctuations of the ligand with the S1 simulation set, to estimate the effect of the binding pocket on the dynamics of the diketide. In Figure 57A, we report the distributions of end-to-end distances of the dkD ligand in water (red), vacuum (green) and in the binding pocket (blue). In Figure 57B the same color code is adopted for dkL.

A B

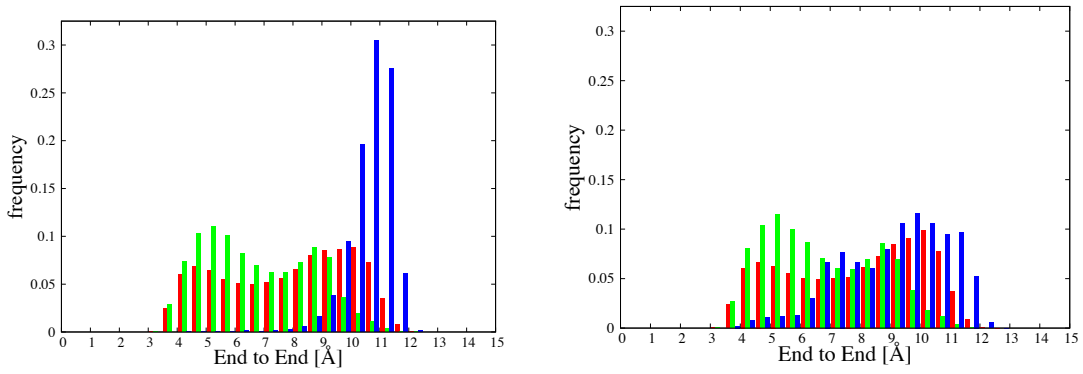


Figure 57 End-to-end distance of (A) dkD and (B) dkL in the binding pocket of A1 (blue, data taken from the S1 simulation set), in water (red) and in vacuum (green).

As we noted in Figure 36C, in the binding pocket (blue) the ligand tends to stay more stretched, particularly the dkD ligand. In aqueous solution (red) the distribution is wider, and in vacuum (green) the diketide tends to have a more collapsed structure.

Next we want to look at the following torsion: O3-C3-C6-O6 (see Figure 37 for more details on the structure). C3 and O3 form the reactive β-carbonyl group, C6-O6 is the nearby carbonyl group. In Figure 58A the distribution is shown for dkD in water (red), vacuum (green), and in A1 (blue). Same color code holds for Figure 58B, where the dkL is reported.

A B

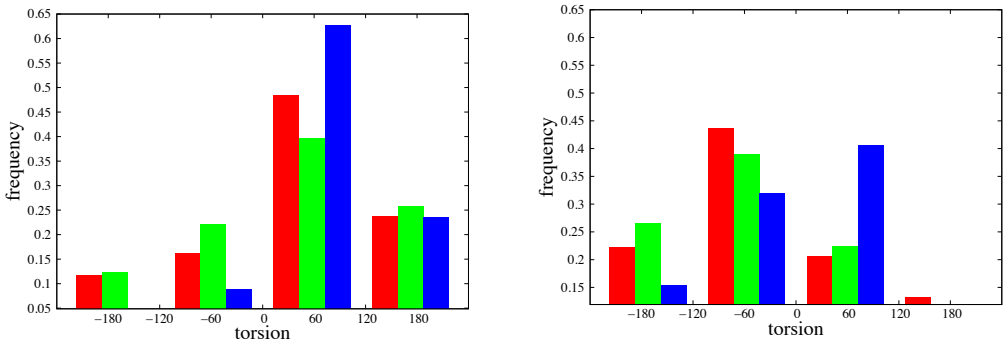


Figure 58 Distribution of value of the torsion O3-C3-C6-O6 (see Figure 37 for the nomenclature) of the D enantiomer (A) and the L enantiomer (B) of the diketide. In red we report the results in water, in green those obtained from vacuum simulations, and in blue those obtained from the binding pocket.

The distributions in vacuum and in water are almost specular images of each other with respect to 0, as shown in Figure 59 which reports the distribution for dkD in water and vacuum (red and green, respectively) and the specular image respect to 0 of the dkL in water and in vacuum (blue and orange, respectively).

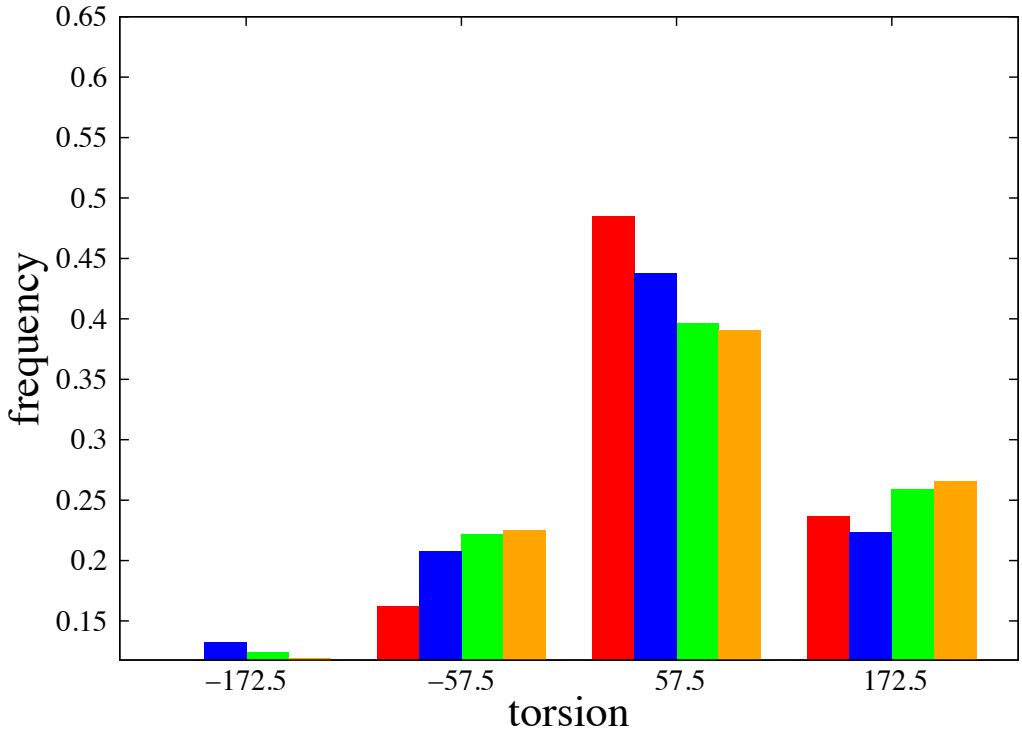


Figure 59 Distribution of value of the torsion O3-C3-C6-O6 (see Figure 37 for the nomenclature) of the D enantiomer in water (red), and in vacuum (green), and the specular image with respect to zero of the L enantiomer in water (blue), and in vacuum (orange).

This seems reasonable, as the α-mehtyl substituent lies between the two carbonyl groups C3-O3 and C6-O6. Interestingly, the maximum of the distribution for the ligand in the binding pocket is the same for both dkD and dkL (Figure 58, blue histogram). For A1-dkD that peak of the distribution corresponds to the peak in vacuum and water (Figure 58A), for dkL this is not the case (Figure 58B). This suggests that in the A1 enzyme less strain on the reactive region is applied on the dkD ligand, which has a distribution of this torsion analogous to the one in water and in vacuum. The dkL ligand

instead seems to be more distorted from its fluctuations outside of the binding pocket. This might also explain why the A1-dkD complex has a larger number of reactive configurations compared to the A1-dkL complex.

Appendix A: Free Energy Contribution of Urey-Bradley Potential

In the thermodynamic cycle that we compute, we remove some ILE sidechain analog angular interactions (between P and N particles) and create some GLN sidechain analog angular interactions (between P and M particles) in water. Then we do the opposite in vacuum. The list of the angles that are substituted (i.e. removed or created) is reported in the following Table.

Created/annihilated angles in ILE side chain analog				Created/annihilated angles in GLN side chain analog			
P		N		P		M	
HS1	CS	CI1		HS1	CS	CQ1	
HS2	CS	CI1		HS2	CS	CQ1	
HS3	CS	CI1		HS3	CS	CQ1	
CS		CI1	HI11	CS		CQ1	HQ11
CS		CI1	HI12	CS		CQ1	HQ12
CS		CI1	CI2	CS		CQ1	CQ2

Table 12 This table reports the list of angles that are removed/created in the mutations in Figure 7 (horizontal arrows). These are all the angles that involve P and N particles or P and M particles. The name of the atoms involved in these angles are reported according to Figure 8.

As already stated, it is convenient to remove/add Urey-Bradley bonds instead of regular angular interactions. Clearly, the free energy difference of removing/creating Urey-Bradley bonds is different from the free energy difference of removing/creating bond angle interactions since the functions are different. According to our protocol as in Table 1 and Table 2, we performed the mutation simulations (horizontal arrows in Figure 7) using a Urey-Bradley bond. In the solvation simulations (vertical arrows in Figure 7) the sampling of configurations was performed according to the regular bond-angle potential. Therefore, we need to correct for using different functional forms computing

explicitly the free energy contribution of the potential change. Potential switches occur at the four corners of Figure 7. In the top left corner we change from the regular angular potential to the Urey-Bradley potential for all the substituted angles in ILE sidechain analog (Table 12). This free energy difference is defined $\Delta F_{I,solv}^{A\to UB}$. We then reach the top right corner with a Urey-Bradley potential for the substituted angles in GLN sidechain analog (Table 12), so we need to compute the free energy difference of removing them and creating the regular angular potential. We refer to this term as $-\Delta F_{Q,solv}^{A\to UB}$. In the bottom right corner of Figure 7 in the text we have regular bond-angle potential, but the mutation is performed according to Urey-Bradley terms. Therefore we need to compute the free energy difference of substituting the regular bond angle term to Urey-Bradley terms for all the substituted angles in GLN sidechain analog in vacuum (Table 12). We refer to this term as $\Delta F_{Q}^{A\to UB}$. Finally, in the bottom left corner of Figure 7 all the substituted angles in ILE sidechain analog (Table 12) are described according to a Urey-Bradley potential, but in the solvation we use the regular bond-angle term. The free energy difference of performing this substitution in vacuum is $-\Delta F_{I}^{A\to UB}$. Overall, the total correction to the cycle due to our inconsistent use of different angle potentials is:

$$\oint dF^{A\to UB} = \Delta F_{I,\,\mathrm{solv}}^{A\to UB} - \Delta F_{I}^{A\to UB} - \Delta F_{Q,\,\mathrm{solv}}^{A\to UB} + \Delta F_{Q}^{A\to UB} \qquad \text{(A1)}$$

Each free energy difference in Eq.(A1) was computed using the Bennett Acceptance Ratio (BAR) method[17, 27]. According to this method, the free energy difference associated with our change in the force field is computed using the following formula:

$$\Delta F^{A\to UB} = -k_B T \ln \frac{\left\langle \dfrac{1}{1+\exp\left[\beta\left(U_{UB}-U_{A}-C\right)\right]} \right\rangle_{A}}{\left\langle \dfrac{1}{1+\exp\left[-\beta\left(U_{UB}-U_{A}-C\right)\right]} \right\rangle_{UB}} + C \qquad \text{(A2)}$$

Here, U_A represents the angular potential for all those angles that are alchemically removed (see Table 12), U_B is instead the potential for the Urey-Bradley bond that substitutes the regular angular potential. The symbol $\langle ... \rangle_A$ refers to an average performed over an ensemble of structures sampled when the angular potential is used. The symbol $\langle ... \rangle_{UB}$ refers to an average performed over an ensemble of structures sampled when the Uery-Bradley potential is used. The value of C is determined according to the following formula:

$$C = \Delta F^{A \to UB} + k_B T \ln \frac{n_{UB}}{n_A} \tag{A3}$$

where n_{UB} is the number of structures in the sample performed with the Urey-Bradley bonds, and n_A is the number of structures in the sample performed with the regular angular potential. The two equations (A2) and (A3) can be used iteratively to obtain the value of the free energy difference.

The computation of the variance of the free energy associated with BAR method was performed according to the following formula[41]:

$$\sigma^2_{BAR} = \frac{1}{n_A \beta^2} \left[\frac{\left\langle \left(\frac{1}{1+\exp\left[\beta\left(U_{UB} - U_A - C\right)\right]} \right)^2 \right\rangle_A}{\left\langle \frac{1}{1+\exp\left[\beta\left(U_{UB} - U_A - C\right)\right]} \right\rangle_A^2} \right] + \frac{1}{n_{UB} \beta^2} \left[\frac{\left\langle \left(\frac{1}{1+\exp\left[-\beta\left(U_{UB} - U_A - C\right)\right]} \right)^2 \right\rangle_{UB}}{\left\langle \frac{1}{1+\exp\left[-\beta\left(U_{UB} - U_A - C\right)\right]} \right\rangle_{UB}^2} \right] \tag{A4}$$

The simulations are carried out using the same systems as those of the alchemical substitutions. The solvated simulations were 2ns long, while the simulations of the system in vacuum were 10ns long. The results are in the following table:

$\Delta F_{I,solv}^{A->UB}$	(0.1421±0.0088)kcal/mol
$\Delta F_{I}^{A->UB}$	(0.2007±0.0047)kcal/mol
$\Delta F_{Q,solv}^{A->UB}$	(0.1505±0.0010)kcal/mol
$\Delta F_{Q}^{A->UAB}$	(0.2038±0.0048)kcal/Mol

Table 13: The results of the free energy difference upon substitution of the regular angular interactions with the Urey-Bradley interactions are reported with their errors.

Hence the free energy of changing the potential of the angles between the fragment and the molecule from the regular bond angle potential to the Urey-Bradley potential is small for each individual term. In the context of a comparison with experiment it is one order of magnitude smaller than the expected systematic errors (~1-2kcal/mol in the case of ligand binding[22]). In the present context of testing numerical accuracy we exploit our knowledge that the entire cycle must be zero. Therefore, we consider only the contribution to the entire cycle. According to Eq.(A1) this number is:

$$\oint dF^{A\to UB} = \Delta F_{I,\,\mathrm{solv}}^{A\to UB} - \Delta F_{I}^{A\to UB} - \Delta F_{Q,\,\mathrm{solv}}^{A\to UB} + \Delta F_{Q}^{A\to UB} = (-0.005 \pm 0.011)\,\mathrm{kcal/mol} \quad \text{(A5)}$$

It turns out that this correction due to the changes of the angular potential is negligible.

Appendix B: Derivation of the Fokker-Planck Equation from the Master Equation

A derivation very similar to this one, and two possible other variants, can be found in [100].

We start from the Master equation for a time homogeneous system as in the main text:

$$\frac{\partial p(\vec{x},t)}{\partial t} = \int d\vec{y}\left[W(\vec{x}\,|\,\vec{y})\,p(\vec{y},t) - W(\vec{y}\,|\,\vec{x})\,p(\vec{x},t)\right] \tag{B1}$$

Following the idea of the Kramers-Moyal expansion, we can expand $W(\vec{y}\,|\,\vec{x})$ in a series. To do so, we can use the Fourier transform of $W(\vec{x}\,|\,\vec{y})$:

$$C(\vec{u},\vec{y}) = \int d\vec{x} W(\vec{x}\,|\,\vec{y}) e^{i\vec{u}\cdot(\vec{x}-\vec{y})} \tag{B2}$$

If we expand in Taylor series the exponential we obtain:

$$C(\vec{u},\vec{y}) = \int d\vec{x} \sum_{n=0}^{\infty} \frac{\left[i\vec{u}\cdot(\vec{x}-\vec{y})\right]^n}{n!} W(\vec{x}\,|\,\vec{y}) = \int d\vec{x} W(\vec{x}\,|\,\vec{y}) + \int d\vec{x} \sum_{n=1}^{\infty} \frac{\left[i\vec{u}\cdot(\vec{x}-\vec{y})\right]^n}{n!} W(\vec{x}\,|\,\vec{y})$$

The sum of all the exiting rates from $\vec{y}$ is

$$\gamma(\vec{y}) = \int d\vec{x} W(\vec{x}\,|\,\vec{y}) \tag{B3}$$

therefore, we can write for the Fourier transform:

$$C(\vec{u},\vec{y}) = \gamma(\vec{y}) + \int d\vec{x} \sum_{n=1}^{\infty} \frac{\left[i\vec{u}\cdot(\vec{x}-\vec{y})\right]^n}{n!} W(\vec{x}\,|\,\vec{y}) \tag{B4}$$

We can now transform back $C(\vec{u},\vec{y})$ into $W(\vec{y}\,|\,\vec{x})$, remembering that the N-dimensional delta function is

$$\delta(\vec{x}-\vec{y}) = \frac{1}{(2\pi)^N} \int d\vec{u} e^{-i\vec{u}\cdot(\vec{x}-\vec{y})}$$

we get

$$W(\vec{x},\vec{y}) = \frac{1}{(2\pi)^N}\int d\vec{u} e^{-i\vec{u}\cdot(\vec{x}-\vec{y})} C(\vec{u},\vec{y}) = \gamma(\vec{y})\delta(\vec{x}-\vec{y}) +$$

$$+\frac{1}{(2\pi)^N}\sum_{n=1}^{\infty}\int d\vec{u} e^{-i\vec{u}\cdot(\vec{x}-\vec{y})}\int d\vec{x}' \frac{\left[i\vec{u}\cdot(\vec{x}'-\vec{y})\right]^n}{n!} W(\vec{x}'\,|\,\vec{y})$$

In the second term of the right hand side of this equation, the scalar product should be rewritten in a more convenient way:

$$\frac{1}{(2\pi)^N}\sum_{n=1}^{\infty}\int d\vec{u}\ e^{-i\vec{u}\cdot(\vec{x}-\vec{y})}\int d\vec{x}' \frac{\left[i\vec{u}\cdot(\vec{x}'-\vec{y})\right]^n}{n!} W(\vec{x}'\,|\,\vec{y}) =$$

$$= \frac{1}{(2\pi)^N}\sum_{n=1}^{\infty}\frac{1}{n!}\int d\vec{u}\ e^{-i\vec{u}\cdot(\vec{x}-\vec{y})}\int d\vec{x}'\left[\sum_{j=1}^{N} iu_j\left(x'_j - y_j\right)\right]^n W(\vec{x}'\,|\,\vec{y}) =$$

$$= \frac{1}{(2\pi)^N}\sum_{n=1}^{\infty}\frac{1}{n!}\int d\vec{u}\ e^{-i\vec{u}\cdot(\vec{x}-\vec{y})}\int d\vec{x}'\left[\sum_{j_1=1}^{N} iu_{j_1}\left(x'_{j_1} - y_{j_1}\right)\right]\cdots\left[\sum_{j_n=1}^{N} iu_{j_n}\left(x'_{j_n} - y_{j_n}\right)\right] W(\vec{x}'\,|\,\vec{y}) =$$

$$= \frac{1}{(2\pi)^N}\sum_{n=1}^{\infty}\sum_{j_1=1}^{N}\cdots\sum_{j_n=1}^{N}\int d\vec{u}\ e^{-i\sum_{j=1}^{N} u_j\left(x_j - y_j\right)} iu_{j_1}\cdots iu_{j_n}\frac{1}{n!}\int d\vec{x}'\left(x'_{j_1} - y_{j_1}\right)\cdots\left(x'_{j_n} - y_{j_n}\right) W(\vec{x}'\,|\,\vec{y}) =$$

$$= \frac{1}{(2\pi)^N}\sum_{n=1}^{\infty}\sum_{j_1=1}^{N}\cdots\sum_{j_n=1}^{N}\int d\vec{u}\ e^{-i\sum_{j=1}^{N} u_j\left(x_j - y_j\right)} iu_{j_1}\cdots iu_{j_n} D^{(n)}_{j_1\cdots j_n}(\vec{y})$$

Here, we defined the KM coefficient as

$$D^{(n)}_{j_1\cdots j_n}(\vec{y}) = \frac{1}{n!}\int d\vec{x}'\left(x'_{j_1} - y_{j_1}\right)\cdots\left(x'_{j_n} - y_{j_n}\right) W(\vec{x}'\,|\,\vec{y}) \qquad \text{(B5)}$$

Now we can carry out the integration in $\vec{u}$ by making use of the following:

$$\frac{1}{(2\pi)^N}\int du(iu)^n e^{-iu(x-y)} = \frac{1}{(2\pi)^N}\frac{\partial^n}{\partial y^n}\int du e^{-iu(x-y)} = \frac{\partial^n}{\partial y^n}\delta(x-y)$$

Therefore:

$$\frac{1}{(2\pi)^N}\sum_{n=1}^{\infty}\sum_{k_1=1}^{N}\cdots\sum_{k_n=1}^{N}\int d\vec{u} e^{-i\vec{u}\cdot(\vec{x}-\vec{y})} D^{(n)}_{k_1\ldots k_n}(\vec{y}) iu_{k_1}\cdots iu_{k_n} =$$

$$= \sum_{n=1}^{\infty}\sum_{k_1=1}^{N}\cdots\sum_{k_n=1}^{N} D^{(n)}_{k_1\ldots k_n}(\vec{y})\frac{\partial}{\partial y_{k_1}}\cdots\frac{\partial}{\partial y_{k_n}}\cdot\delta(\vec{x}-\vec{y})$$

This formula can be used to derive the following equation for $W(\vec{y}\,|\,\vec{x})$

$$W(\vec{x}\,|\,\vec{y}) = \gamma(\vec{y})\delta(\vec{x}-\vec{y}) + \sum_{n=1}^{\infty}\sum_{k_1=1}^{N}\cdots\sum_{k_n=1}^{N}\left\{D^{(n)}_{k_1\cdots k_n}(\vec{y})\frac{\partial^n}{\partial y_{k_1}\cdots\partial y_{k_n}}\delta(\vec{x}-\vec{y})\right\} \tag{B6}$$

Now, if we substitute (B6) in the Master equation (B1), we get:

$$\frac{\partial p(\vec{x},t)}{\partial t} = \int d\vec{y}\delta(\vec{x}-\vec{y})\gamma(\vec{y})p(\vec{y},t) +$$

$$+\sum_{n=1}^{\infty}\sum_{k_1=1}^{N}\cdots\sum_{k_n=1}^{N}\int d\vec{y}D^{(n)}_{k_1\cdots k_n}(\vec{y})p(\vec{y},t)\frac{\partial^n}{\partial y_{k_1}\cdots\partial y_{k_n}}\delta(\vec{x}-\vec{y}) - \int d\vec{y}W(\vec{y}\,|\,\vec{x})p(\vec{x},t)$$

Recalling (B3), we get that

$$\int d\vec{y}\delta(\vec{x}-\vec{y})\gamma(\vec{y})p(\vec{y},t) = \gamma(\vec{x})p(\vec{x},t) = \int d\vec{y}W(\vec{y}\,|\,\vec{x})p(\vec{x},t)$$

This term cancels out with the last one, and we are left with

$$\frac{\partial p(\vec{x},t)}{\partial t} = \sum_{n=1}^{\infty}\sum_{k_1=1}^{N}\cdots\sum_{k_n=1}^{N}\int d\vec{y}D^{(n)}_{k_1\cdots k_n}(\vec{y})p(\vec{y},t)\frac{\partial^n}{\partial y_{k_1}\cdots\partial y_{k_n}}\delta(\vec{x}-\vec{y}) \tag{B7}$$

Integrating by parts the delta, and finally integrating the delta away, we obtain our final result.

$$\frac{\partial p(\vec{x},t)}{\partial t} = \sum_{n=1}^{\infty}\sum_{k_1=1}^{N}\cdots\sum_{k_n=1}^{N}(-1)^n\frac{\partial^n}{\partial x_{k_1}\cdots\partial x_{k_n}}D^{(n)}_{k_1\cdots k_n}(\vec{x})p(\vec{x},t) \tag{B8}$$

Appendix C: Derivation of the Kramers-Moyal Coefficients For Overdamped Dynamics

This derivation is based upon the derivation of the 1-dimensional KM coefficients for an overdamped system, which can be found in [56].

The dynamics in a N dimensional space of coarse variables is modeled with a set of overdamped equations as in (3.1):

$$d\vec{x} = \vec{a}(\vec{x})dt + \sqrt{2}\hat{b}(\vec{x})d\vec{W}(t) \tag{C.1}$$

Given this dynamics, we want to compute the first and the second KM coefficients:

$$\begin{aligned} D_i^{(1)}\left(\vec{\xi}\right) &= \int d\vec{x}\left(x_i - \xi_i\right) W\left(\vec{x}, t+\tau \mid \vec{\xi}, t\right) \\ D_{ij}^{(2)}\left(\vec{\xi}\right) &= \frac{1}{2}\int d\vec{x}\left(x_i - \xi_i\right)\left(x_j - \xi_j\right) W\left(\vec{x}, t+\tau \mid \vec{\xi}, t\right) \end{aligned} \tag{C.2}$$

If we plug the definition of the transition probability per unit time given in Eq. (3.3) in the KM coefficients, since the term with the Dirac's delta cancels out we get:

$$D_i^{(1)}\left(\vec{\xi}\right) = \lim_{\tau\to 0}\frac{1}{\tau}\int d\vec{x}\left(x_i - \xi_i\right) p\left(\vec{x}, t+\tau \mid \vec{\xi}, t\right) = \lim_{\tau\to 0}\frac{1}{\tau}\left\langle x_i(t+\tau) - x_i(t)\right\rangle \tag{C.3}$$

$$\begin{aligned} D_{ij}^{(2)}\left(\vec{\xi}\right) &= \lim_{\tau\to 0}\frac{1}{2\tau}\int d\vec{x}\left(x_i - \xi_i\right)\left(x_j - \xi_j\right) p\left(\vec{x}, t+\tau \mid \vec{\xi}, t\right) \\ &= \lim_{\tau\to 0}\frac{1}{2\tau}\left\langle\left[x_i(t+\tau) - x_i(t)\right]\left[x_j(t+\tau) - x_j(t)\right]\right\rangle \end{aligned} \tag{C.4}$$

Note that with the symbol $\langle ... \rangle$ we refer to the averaged over the conditional probability of being in $\vec{\xi}$ at time t.

To compute these averages, we need to integrate equation (C.1) for a short time τ. What makes this calculation tricky is the presence of a stochastic term in equation (C.1). To integrate the stochastic process we need to follow Ito's rules of calculus. We recall here the few properties[87] that we will need to solve equations (C.3)-(C.4):

1. Property (a): the stochastic process $\vec{x}(t)$ it is continuous but not differentiable; we also assume that $a_i\left[\vec{x}(t)\right]$ and $b_{ij}\left[\vec{x}(t)\right]$ are continuous.
2. Property (b): given any function $G\left[\vec{x}(t)\right]$ which is bounded and non-anticipating, i.e. independent on the Wiener process for $s > t$, we have:

$$\left\langle \int_{t_0}^{t} dW(t')G\left[\vec{x}(t')\right] \right\rangle = 0 \tag{C.5}$$

   The functions $a_i\left[\vec{x}(t)\right]$ and $b_{ij}\left[\vec{x}(t)\right]$ are non-anticipating.
3. Property (c): given any arbitrary function of a stochastic process as the one in (3.1), that function obeys the following differential equation:

$$\begin{aligned} df\left[\vec{x}(t)\right] = & \left\{ \sum_{i=1}^{N} a_i\left[\vec{x}(t)\right] \frac{\partial f\left[\vec{x}(t)\right]}{\partial x_i} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N} b_{ik}\left[\vec{x}(t)\right] b_{jk}\left[\vec{x}(t)\right] \frac{\partial^2 f\left[\vec{x}(t)\right]}{\partial x_i \partial x_j} \right\} dt \\ & + \sum_{i=1}^{N}\sum_{j=1}^{N} \sqrt{2} b_{ij}\left[\vec{x}(t)\right] \frac{\partial f\left[\vec{x}(t)\right]}{\partial x_i} dW_j(t) \end{aligned} \tag{C.6}$$

This last equation is known as Ito's formula. We will use it in its integrated form:

$$\begin{aligned} f\left[\vec{x}(t)\right] = & f\left[\vec{x}(t_0)\right] \\ & + \int_{t_0}^{t} \left\{ \sum_{i=1}^{N} a_i\left[\vec{x}(t')\right] \frac{\partial f\left[\vec{x}(t')\right]}{\partial x_i} + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N} b_{ik}\left[\vec{x}(t')\right] b_{jk}\left[\vec{x}(t')\right] \frac{\partial^2 f\left[\vec{x}(t')\right]}{\partial x_i \partial x_j} \right\} dt' \\ & + \int_{t_0}^{t} \sum_{i=1}^{N}\sum_{j=1}^{N} \sqrt{2} b_{ij}\left[\vec{x}(t')\right] \frac{\partial f\left[\vec{x}(t')\right]}{\partial x_i} dW_j(t') \end{aligned} \tag{C.7}$$

The average in the first KM coefficient can be rewritten by integrating equation (C.1) over a short time τ, i.e.

$$\left\langle x_i(t+\tau) - x_i(t) \right\rangle = \left\langle \int_{t}^{t+\tau} dt' a_i\left[\vec{x}(t')\right] + \sqrt{2} \sum_{j=1}^{N} \int_{t}^{t+\tau} dW_j(t') b_{ij}\left[\vec{x}(t')\right] \right\rangle$$

The average of the second integral is zero because of property (b). Therefore, we are left with:

$$\lim_{\tau\to 0}\frac{\langle x_i(t+\tau)-x_i(t)\rangle}{\tau}=\lim_{\tau\to 0}\frac{\left\langle\int_t^{t+\tau}dt'\,a_i\left[\vec{x}(t')\right]\right\rangle}{\tau}$$
$$=\lim_{\tau\to 0}\left\langle a_i\left[\vec{x}(t+\tau)\right]\right\rangle=\left\langle a_i\left[\vec{x}(t)\right]\right\rangle=a_i\left(\vec{\xi}\right)$$

where in the second step of the derivation we used l'Hôpital's rule and in the last step the continuity property (a). This proves that:

$$D_i^{(1)}\left(\vec{\xi}\right)=\lim_{\tau\to 0}\frac{1}{\tau}\langle x_i(t+\tau)-x_i(t)\rangle=a_i\left(\vec{\xi}\right) \tag{C.8}$$

For the second KM coefficient we need to compute:

$$\left\langle\left[x_i(t+\tau)-x_i(t)\right]\left[x_j(t+\tau)-x_j(t)\right]\right\rangle=\left\langle x_i(t+\tau)x_j(t+\tau)\right\rangle-\left\langle x_i(t+\tau)x_j(t)\right\rangle$$
$$-\left\langle x_i(t)x_j(t+\tau)\right\rangle+\left\langle x_i(t)x_j(t)\right\rangle$$
$$=\left\langle x_i(t+\tau)x_j(t+\tau)\right\rangle-\left\langle x_i(t+\tau)\right\rangle\xi_j-\left\langle x_j(t+\tau)\right\rangle\xi_i+\xi_i\xi_j$$

Note that we explicitly used the fact that the average is conducted over a conditional probability that $\vec{x}(t)=\vec{\xi}$.

Now, from equation (C.8) we have that:

$$\left\langle x_i(t+\tau)\right\rangle=\xi_i+a_i\left(\vec{\xi}\right)\tau+O\left(\tau^2\right)$$

so

$$\left\langle\left[x_i(t+\tau)-x_i(t)\right]\left[x_j(t+\tau)-x_j(t)\right]\right\rangle=\left\langle x_i(t+\tau)x_j(t+\tau)\right\rangle-\xi_i\xi_j-a_i\left(\vec{\xi}\right)\xi_j\tau-\xi_i a_j\left(\vec{\xi}\right)\tau+O\left(\tau^2\right)$$

To evaluate the first average, we use Ito's formula (C.7) for the function $x_i(t+\tau)x_j(t+\tau)$:

$$\left\langle x_i(t+\tau)x_j(t+\tau)\right\rangle = \left\langle x_i(t)x_j(t)\right\rangle + \left\langle \int_{t_0}^{t}\left\{\sum_{k=1}^{N} a_k\left[\vec{x}(t')\right]\frac{\partial x_i(t')x_j(t')}{\partial x_k} + \sum_{k=1}^{N}\sum_{l=1}^{N}\sum_{m=1}^{N} b_{km}\left[\vec{x}(t')\right]b_{lm}\left[\vec{x}(t')\right]\frac{\partial^2 x_i(t')x_j(t')}{\partial x_k\,\partial x_l}\right\}dt' + \sqrt{2}\int_{t_0}^{t}\sum_{k=1}^{N}\sum_{l=1}^{N} b_{kl}\left[\vec{x}(t')\right]\frac{\partial x_i(t')x_j(t')}{\partial x_k}dW_l(t')\right\rangle$$

As before, the last integral is zero because of property (b). Now, carrying out the derivatives we get:

$$\left\langle x_i(t+\tau)x_j(t+\tau)\right\rangle = \xi_i\xi_j + \left\langle \int_{t}^{t+\tau} a_i\left[\vec{x}(t')\right]x_j(t')dt'\right\rangle + \left\langle \int_{t}^{t+\tau} a_j\left[\vec{x}(t')\right]x_i(t')dt'\right\rangle + 2\left\langle \int_{t}^{t+\tau}\sum_{m=1}^{N} b_{im}\left[\vec{x}(t')\right]b_{jm}\left[\vec{x}(t')\right]dt'\right\rangle$$

Therefore, we have:

$$\left\langle\left[x_i(t+\tau)-x_i(t)\right]\left[x_j(t+\tau)-x_j(t)\right]\right\rangle = \left\langle \int_{t}^{t+\tau} a_i\left[\vec{x}(t')\right]x_j(t')dt'\right\rangle + \left\langle \int_{t}^{t+\tau} a_j\left[\vec{x}(t')\right]x_i(t')dt'\right\rangle + 2\left\langle \int_{t}^{t+\tau}\sum_{m=1}^{N} b_{im}\left[\vec{x}(t')\right]b_{jm}\left[\vec{x}(t')\right]dt'\right\rangle - a_i\left(\vec{\xi}\right)\xi_j\tau - \xi_i a_j\left(\vec{\xi}\right)\tau + O\left(\tau^2\right)$$

If we divide by τ and take the limit for $\tau \to 0$, using l'Hopital's rule and property (a) like before, we get:

$$\lim_{\tau\to 0}\frac{\left\langle\left[x_i(t+\tau)-x_i(t)\right]\left[x_j(t+\tau)-x_j(t)\right]\right\rangle}{\tau} = a_i\left(\vec{\xi}\right)\xi_j + \xi_i a_j\left(\vec{\xi}\right) + 2\sum_{m=1}^{N} b_{im}\left(\vec{\xi}\right)b_{jm}\left(\vec{\xi}\right) - a_i\left(\vec{\xi}\right)\xi_j - \xi_i a_j\left(\vec{\xi}\right)$$

So that, finally,

$$D_{ij}^{(2)}\left(\vec{\xi}\right) = \lim_{\tau\to 0}\frac{1}{2\tau}\left\langle\left[x_i(t+\tau)-x_i(t)\right]\left[x_j(t+\tau)-x_j(t)\right]\right\rangle = \sum_{m=1}^{N} b_{im}\left(\vec{\xi}\right)b_{jm}\left(\vec{\xi}\right) \quad \text{(C.9)}$$

Finally, we need to show that the third KM coefficient is zero, i.e.

$$D_{ijk}^{(3)}\left(\vec{\xi}\right) = \lim_{\tau\to 0}\frac{1}{6\tau}\left\langle\left[x_i(t+\tau)-x_i(t)\right]\left[x_j(t+\tau)-x_j(t)\right]\left[x_k(t+\tau)-x_k(t)\right]\right\rangle = 0 \quad \text{(C.10)}$$

Using Pawula's theorem,[54a, 54c] this proves that all the other coefficients are zero. Let's rewrite the average in equation (C.10):

$$\left\langle \left[x_i(t+\tau)-x_i(t)\right]\left[x_j(t+\tau)-x_j(t)\right]\left[x_k(t+\tau)-x_k(t)\right]\right\rangle = \left\langle x_i(t+\tau)x_j(t+\tau)x_k(t+\tau)\right\rangle$$
$$-\xi_i\left\langle x_j(t+\tau)x_k(t+\tau)\right\rangle - \xi_j\left\langle x_i(t+\tau)x_k(t+\tau)\right\rangle - \xi_k\left\langle x_i(t+\tau)x_j(t+\tau)\right\rangle +$$
$$+\xi_i\xi_j\left\langle x_k(t+\tau)\right\rangle + \xi_i\xi_k\left\langle x_j(t+\tau)\right\rangle + \xi_j\xi_k\left\langle x_i(t+\tau)\right\rangle - \xi_i\xi_j\xi_k$$

From equation (C.9) we get that

$$\left\langle x_i(t+\tau)x_j(t+\tau)\right\rangle = \left\langle x_i(t+\tau)\right\rangle\xi_j + \xi_i\left\langle x_j(t+\tau)\right\rangle - \xi_i\xi_j + 2\left[\hat{b}\left(\vec{\xi}\right)\hat{b}^T\left(\vec{\xi}\right)\right]_{ij}\tau + O\left(\tau^2\right)$$

We can use this result to get:

$$\left\langle \left[x_i(t+\tau)-x_i(t)\right]\left[x_j(t+\tau)-x_j(t)\right]\left[x_k(t+\tau)-x_k(t)\right]\right\rangle = \left\langle x_i(t+\tau)x_j(t+\tau)x_k(t+\tau)\right\rangle$$
$$-\left\langle x_i(t+\tau)\right\rangle\xi_j\xi_k - \xi_i\left\langle x_j(t+\tau)\right\rangle\xi_k - \xi_i\xi_j\left\langle x_k(t+\tau)\right\rangle + 2\xi_i\xi_j\xi_k$$
$$-2\tau\left\{\left[\hat{b}\left(\vec{\xi}\right)\hat{b}^T\left(\vec{\xi}\right)\right]_{ij}\xi_k + \left[\hat{b}\left(\vec{\xi}\right)\hat{b}^T\left(\vec{\xi}\right)\right]_{ik}\xi_j + \left[\hat{b}\left(\vec{\xi}\right)\hat{b}^T\left(\vec{\xi}\right)\right]_{jk}\xi_i\right\}\tau + O\left(\tau^2\right)$$

Similarly to what was done to derive the second KM coefficient, we use equation (C.8) to get:

$$\left\langle \left[x_i(t+\tau)-x_i(t)\right]\left[x_j(t+\tau)-x_j(t)\right]\left[x_k(t+\tau)-x_k(t)\right]\right\rangle = \left\langle x_i(t+\tau)x_j(t+\tau)x_k(t+\tau)\right\rangle$$
$$-\tau\left\{a_i\left(\vec{\xi}\right)\xi_j\xi_k + \xi_i a_j\left(\vec{\xi}\right)\xi_k + \xi_i\xi_j a_k\left(\vec{\xi}\right)\right\} - \xi_i\xi_j\xi_k$$
$$-2\tau\left\{\left[\hat{b}\left(\vec{\xi}\right)\hat{b}^T\left(\vec{\xi}\right)\right]_{ij}\xi_k + \left[\hat{b}\left(\vec{\xi}\right)\hat{b}^T\left(\vec{\xi}\right)\right]_{ik}\xi_j + \left[\hat{b}\left(\vec{\xi}\right)\hat{b}^T\left(\vec{\xi}\right)\right]_{jk}\xi_i\right\}\tau + O\left(\tau^2\right)$$

Now we can rewrite the first average using Ito's formula (C.7) for the function $x_i(t+\tau)x_j(t+\tau)x_k(t+\tau)$:

$$\left\langle x_i(t+\tau)x_j(t+\tau)x_k(t+\tau)\right\rangle = \left\langle x_i(t)x_j(t)x_k(t)\right\rangle$$
$$+\left\langle \int_t^{t+\tau}\left\{\sum_{l=1}^{N} a_l\left[\vec{x}(t')\right]\frac{\partial x_i(t')x_j(t')x_k(t')}{\partial x_l}+\sum_{l=1}^{N}\sum_{m=1}^{N}\sum_{n=1}^{N} b_{lm}\left[\vec{x}(t')\right]b_{nm}\left[\vec{x}(t')\right]\frac{\partial^2 x_i(t')x_j(t')x_k(t')}{\partial x_l\,\partial x_n}\right\}dt'\right.$$
$$\left.+\sqrt{2}\int_t^{t+\tau}\sum_{l=1}^{N}\sum_{m=1}^{N} b_{lm}\left[\vec{x}(t')\right]\frac{\partial x_i(t')x_j(t')x_k(t')}{\partial x_l}dW_m(t')\right\rangle$$

Again, the last term is zero because of equation (C.5). Once we carry out the derivatives we obtain:

$$\left\langle x_i(t+\tau)x_j(t+\tau)x_k(t+\tau)\right\rangle = \xi_i\xi_j\xi_k$$
$$+\left\langle \int_t^{t+\tau} a_i\left[\vec{x}(t')\right]x_j(t')x_k(t')dt'\right\rangle+\left\langle \int_t^{t+\tau} x_i(t')a_j\left[\vec{x}(t')\right]x_k(t')dt'\right\rangle+\left\langle \int_t^{t+\tau} x_i(t')x_j(t')a_k\left[\vec{x}(t')\right]dt'\right\rangle$$
$$+2\left\langle \int_t^{t+\tau}\left\{\hat{b}\left[\vec{x}(t')\right]\hat{b}^T\left[\vec{x}(t')\right]\right\}_{ij} x_k(t')dt'\right\rangle+2\left\langle \int_t^{t+\tau}\left\{\hat{b}\left[\vec{x}(t')\right]\hat{b}^T\left[\vec{x}(t')\right]\right\}_{ik} x_j(t')dt'\right\rangle$$
$$+2\left\langle \int_t^{t+\tau}\left\{\hat{b}\left[\vec{x}(t')\right]\hat{b}^T\left[\vec{x}(t')\right]\right\}_{jk} x_i(t')dt'\right\rangle$$

Therefore, we get:

$$\left\langle\left[x_i(t+\tau)-x_i(t)\right]\left[x_j(t+\tau)-x_j(t)\right]\left[x_k(t+\tau)-x_k(t)\right]\right\rangle = \left\{\left\langle \int_t^{t+\tau} a_i\left[\vec{x}(t')\right]x_j(t')x_k(t')dt'\right\rangle-\tau a_i\left(\vec{\xi}\right)\xi_j\xi_k\right\}$$
$$+\left\{\left\langle \int_t^{t+\tau} x_i(t')a_j\left[\vec{x}(t')\right]x_k(t')dt'\right\rangle-\tau\xi_i a_j\left(\vec{\xi}\right)\xi_k\right\}+\left\{\left\langle \int_t^{t+\tau} x_i(t')x_j(t')a_k\left[\vec{x}(t')\right]dt'\right\rangle-\tau\xi_i\xi_j a_k\left(\vec{\xi}\right)\right\}$$
$$+2\left\{\left\langle \int_t^{t+\tau}\left\{\hat{b}\left[\vec{x}(t')\right]\hat{b}^T\left[\vec{x}(t')\right]\right\}_{ij} x_k(t')dt'\right\rangle-\tau\left[\hat{b}\left(\vec{\xi}\right)\hat{b}^T\left(\vec{\xi}\right)\right]_{ij}\xi_k\right\}$$
$$+2\left\{\left\langle \int_t^{t+\tau}\left\{\hat{b}\left[\vec{x}(t')\right]\hat{b}^T\left[\vec{x}(t')\right]\right\}_{ik} x_j(t')dt'\right\rangle-\tau\left[\hat{b}\left(\vec{\xi}\right)\hat{b}^T\left(\vec{\xi}\right)\right]_{ik}\xi_j\right\}$$
$$+2\left\{\left\langle \int_t^{t+\tau}\left\{\hat{b}\left[\vec{x}(t')\right]\hat{b}^T\left[\vec{x}(t')\right]\right\}_{jk} x_i(t')dt'\right\rangle-\tau\left[\hat{b}\left(\vec{\xi}\right)\hat{b}^T\left(\vec{\xi}\right)\right]_{jk}\xi_i\right\}+O\left(\tau^2\right)$$

If we divide by τ and take the limit for $\tau\to 0$, using l'Hopital rule and property (a), we can show that each term in curly brackets vanishes, giving as a result that the third KM coefficient is zero.

References

1. Karplus, M.; McCammon, J. A., Molecular dynamics simulations of biomolecules. *Nat Struct Biol* **2002,** *9* (9), 646-652.
2. Frenkel, D.; Smit, B., *Understanding molecular simulation : from algorithms to applications*. 2nd ed.; Academic Press: San Diego, 2002; p xxii, 638 p.
3. McQuarrie, D., *Statistical Mechanics*. University Science Books: 2000.
4. Kaminski, G. A.; Friesner, R. A.; Tirado-Rives, J.; Jorgensen, W. L., Evaluation and reparametrization of the OPLS-AA force field for proteins via comparison with accurate quantum chemical calculations on peptides. *J Phys Chem B* **2001,** *105* (28), 6474-6487.
5. (a) Torrie, G. M.; Valleau, J. P., Non-Physical Sampling Distributions in Monte-Carlo Free-Energy Estimation - Umbrella Sampling. *J Comput Phys* **1977,** *23* (2), 187-199; (b) Sugita, Y.; Okamoto, Y., Replica-exchange molecular dynamics method for protein folding. *Chem Phys Lett* **1999,** *314* (1-2), 141-151.
6. Tembe, B. L.; Mccammon, J. A., Ligand Receptor Interactions. *Comput Chem* **1984,** *8* (4), 281-283.
7. Schames, J. R.; Henchman, R. H.; Siegel, J. S.; Sotriffer, C. A.; Ni, H. H.; McCammon, J. A., Discovery of a novel binding trench in HIV integrase. *J Med Chem* **2004,** *47* (8), 1879-1881.
8. (a) Zwanzig, R., Memory Effects in Irreversible Thermodynamics. *Phys Rev* **1961,** *124* (4), 983-&; (b) Mori, H., Transport Collective Motion and Brownian Motion. *Prog Theor Phys* **1965,** *33* (3), 423-&.
9. Koshland, D. E., The Key-Lock Theory and the Induced Fit Theory. *Angew Chem Int Edit* **1994,** *33* (23-24), 2375-2378.
10. Teague, S. J., Implications of protein flexibility for drug discovery. *Nat Rev Drug Discov* **2003,** *2* (7), 527-541.
11. Freire, E., Do enthalpy and entropy distinguish first in class from best in class? *Drug Discov Today* **2008,** *13* (19-20), 869-874.
12. Zidek, L.; Novotny, M. V.; Stone, M. J., Increased protein backbone conformational entropy upon hydrophobic ligand binding. *Nat Struct Biol* **1999,** *6* (12), 1118-1121.
13. Jorgensen, W. L., The many roles of computation in drug discovery. *Science* **2004,** *303* (5665), 1813-1818.
14. Kua, J.; Zhang, Y. K.; McCammon, J. A., Studying enzyme binding specificity in acetylcholinesterase using a combined molecular dynamics and multiple docking approach. *J Am Chem Soc* **2002,** *124* (28), 8260-8267.
15. Dror, R. O.; Dirks, R. M.; Grossman, J. P.; Xu, H. F.; Shaw, D. E., Biomolecular Simulation: A Computational Microscope for Molecular Biology. *Annu Rev Biophys* **2012,** *41*, 429-452.

16. Shan, Y. B.; Kim, E. T.; Eastwood, M. P.; Dror, R. O.; Seeliger, M. A.; Shaw, D. E., How Does a Drug Molecule Find Its Target Binding Site? *J Am Chem Soc* **2011,** *133* (24), 9181-9183.
17. Chipot, C.; Pohorille, A., *Free Energy Calculations. Theory and Applications in Chemistry and Biology*. Springer-Verlag: Berlin, Heidelberg, 2007.
18. Pearlman, D. A., A Comparison of Alternative Approaches to Free-Energy Calculations. *J Phys Chem-Us* **1994,** *98* (5), 1487-1493.
19. (a) Karplus, M.; Prevost, M.; Tidor, B.; Wodak, S., Simulation Analysis of the Stability Mutants R96h of Bacteriophage-T4 Lysozyme and 196a of Barnase. *Ciba F Symp* **1991,** *161*, 63-74; (b) Prevost, M.; Wodak, S. J.; Tidor, B.; Karplus, M., Contribution of the Hydrophobic Effect to Protein Stability - Analysis Based on Simulations of the Ile-96-]Ala Mutation in Barnase. *P Natl Acad Sci USA* **1991,** *88* (23), 10880-10884; (c) Tidor, B.; Karplus, M., Simulation Analysis of the Stability Mutant R96h of T4 Lysozyme. *Biochemistry-Us* **1991,** *30* (13), 3217-3228; (d) Sun, Y. C.; Veenstra, D. L.; Kollman, P. A., Free energy calculations of the mutation of Ile96->Ala in barnase: Contributions to the difference in stability. *Protein Eng* **1996,** *9* (3), 273-281.
20. (a) Bash, P. A.; Singh, U. C.; Brown, F. K.; Langridge, R.; Kollman, P. A., Calculation of the Relative Change in Binding Free-Energy of a Protein-Inhibitor Complex. *Science* **1987,** *235* (4788), 574-576; (b) Jorgensen, W. L.; Buckner, J. K.; Boudon, S.; Tirado-Rives, J., Efficient Computation of Absolute Free-Energies of Binding by Computer-Simulations - Application to the Methane Dimer in Water. *Journal of Chemical Physics* **1988,** *89* (6), 3742-3746.
21. Miller, J. L.; Kollman, P. A., Solvation free energies of the nucleic acid bases. *J Phys Chem-Us* **1996,** *100* (20), 8587-8594.
22. Chodera, J. D.; Mobley, D. L.; Shirts, M. R.; Dixon, R. W.; Branson, K.; Pande, V. S., Alchemical free energy methods for drug discovery: progress and challenges. *Curr Opin Struc Biol* **2011,** *21* (2), 150-160.
23. Elber, R.; Roitberg, A.; Simmerling, C.; Goldstein, R.; Li, H. Y.; Verkhivker, G.; Keasar, C.; Zhang, J.; Ulitsky, A., Moil - a Program for Simulations of Macromolecules. *Comput Phys Commun* **1995,** *91* (1-3), 159-189.
24. Ruymgaart, A. P.; Cardenas, A. E.; Elber, R., MOIL-opt: Energy-Conserving Molecular Dynamics on a GPU/CPU System. *Journal of Chemical Theory and Computation* **2011,** *7* (10), 3072-3082.
25. Kirkwood, J. G., Statistical Mechanics of Fluid Mixtures. *Journal of Chemical Physics* **1935,** *3* (5), 14.
26. Zwanzig, R. W., High-Temperature Equation of State by a Permutation Method I. Nonpolar Gases. *Journal of Chemical Physics* **1954,** *22* (8), 7.
27. Bennett, C. H., Efficient Estimation of Free-Energy Differences from Monte-Carlo Data. *J Comput Phys* **1976,** *22* (2), 245-268.
28. Jarzynski, C., Nonequilibrium equality for free energy differences. *Phys Rev Lett* **1997,** *78* (14), 2690-2693.
29. Kong, X. J.; Brooks, C. L., lambda-Dynamics: A new approach to free energy calculations. *Journal of Chemical Physics* **1996,** *105* (6), 2414-2423.

30. Mugnai, M. L.; Elber, R., Thermodynamic Cycle without Turning Off Self-Interactions: Formal Discussion and a Numerical Example. *Journal of Chemical Theory and Computation* **2012,** *8* (9), 3022-3033.
31. Shirts, M. R.; Pande, V. S., Comparison of efficiency and bias of free energies computed by exponential averaging, the Bennett acceptance ratio, and thermodynamic integration. *Journal of Chemical Physics* **2005,** *122* (14).
32. Gilson, M. K.; Given, J. A.; Bush, B. L.; McCammon, J. A., The statistical-thermodynamic basis for computation of binding affinities: A critical review. *Biophys J* **1997,** *72* (3), 1047-1069.
33. Shirts, M. R.; Pitera, J. W.; Swope, W. C.; Pande, V. S., Extremely precise free energy calculations of amino acid side chain analogs: Comparison of common molecular mechanics force fields for proteins. *Journal of Chemical Physics* **2003,** *119* (11), 5740-5761.
34. (a) Wong, C. F.; Mccammon, J. A., Dynamics and Design of Enzymes and Inhibitors. *J Am Chem Soc* **1986,** *108* (13), 3830-3832; (b) Chipot, C.; Rozanska, X.; Dixit, S. B., Can free energy calculations be fast and accurate at the same time? Binding of low-affinity, non-peptide inhibitors to the SH2 domain of the src protein. *J Comput Aid Mol Des* **2005,** *19* (11), 765-770.
35. Jas, G. S.; Kuczera, K., Free-energy simulations of the oxidation of C-terminal methionines in calmodulin. *Proteins* **2002,** *48* (2), 257-268.
36. (a) Boresch, S.; Karplus, M., The role of bonded terms in free energy simulations: 1. Theoretical analysis. *J Phys Chem A* **1999,** *103* (1), 103-118; (b) Boresch, S.; Karplus, M., The role of bonded terms in free energy simulations. 2. Calculation of their influence on free energy differences of solvation. *J Phys Chem A* **1999,** *103* (1), 119-136.
37. Shobana, S.; Roux, B.; Andersen, O. S., Free energy simulations: Thermodynamic reversibility and variability. *J Phys Chem B* **2000,** *104* (21), 5179-5190.
38. Boresch, S., The role of bonded energy terms in free energy simulations - Insights from analytical results. *Mol Simulat* **2002,** *28* (1-2), 13-37.
39. Boresch, S.; Tettinger, F.; Leitgeb, M.; Karplus, M., Absolute binding free energies: A quantitative approach for their calculation. *J Phys Chem B* **2003,** *107* (35), 9535-9551.
40. Beutler, T. C.; Mark, A. E.; Vanschaik, R. C.; Gerber, P. R.; Van Gunsteren, W. F., Avoiding Singularities and Numerical Instabilities in Free-Energy Calculations Based on Molecular Simulations. *Chem Phys Lett* **1994,** *222* (6), 529-539.
41. Pohorille, A.; Jarzynski, C.; Chipot, C., Good Practices in Free-Energy Calculations. *J Phys Chem B* **2010,** *114* (32), 10235-10253.
42. Schuler, B.; Eaton, W. A., Protein folding studied by single-molecule FRET. *Curr Opin Struc Biol* **2008,** *18* (1), 16-26.
43. (a) Liphardt, J.; Dumont, S.; Smith, S. B.; Tinoco, I.; Bustamante, C., Equilibrium information from nonequilibrium measurements in an experimental test of Jarzynski's equality. *Science* **2002,** *296* (5574), 1832-1835; (b) Liphardt, J.; Onoa, B.; Smith, S. B.; Tinoco, I.; Bustamante, C., Reversible unfolding of single RNA molecules by mechanical force. *Science* **2001,** *292* (5517), 733-737.

44. Bustamante, C.; Chemla, Y. R.; Forde, N. R.; Izhaky, D., Mechanical processes in biochemistry. *Annu Rev Biochem* **2004,** *73*, 705-748.
45. (a) Soranno, A.; Buchli, B.; Nettels, D.; Cheng, R. R.; Muller-Spath, S.; Pfeil, S. H.; Hoffmann, A.; Lipman, E. A.; Makarov, D. E.; Schuler, B., Quantifying internal friction in unfolded and intrinsically disordered proteins with single-molecule spectroscopy. *P Natl Acad Sci USA* **2012,** *109* (44), 17800-17806; (b) Lannon, H.; Haghpanah, J. S.; Montclare, J. K.; Vanden-Eijnden, E.; Brujic, J., Force-Clamp Experiments Reveal the Free-Energy Profile and Diffusion Coefficient of the Collapse of Protein Molecules. *Phys Rev Lett* **2013,** *110* (12).
46. Yamakawa, H.; Tanaka, G.; Stockmay.Wh, Correlation-Function Formalism for Intrinsic-Viscosity of Polymers. *Journal of Chemical Physics* **1974,** *61* (11), 4535-4539.
47. Zwanzig, R., From Classical Dynamics to Continuous-Time Random-Walks. *J Stat Phys* **1983,** *30* (2), 255-262.
48. Lange, O. F.; Grubmuller, H., Collective Langevin dynamics of conformational motions in proteins. *Journal of Chemical Physics* **2006,** *124* (21).
49. Berne, B. J.; Tuckerman, M. E.; Straub, J. E.; Bug, A. L. R., Dynamic Friction on Rigid and Flexible Bonds. *Journal of Chemical Physics* **1990,** *93* (7), 5084-5095.
50. Straub, J. E.; Borkovec, M.; Berne, B. J., Calculation of Dynamic Friction on Intramolecular Degrees of Freedom. *J Phys Chem-Us* **1987,** *91* (19), 4995-4998.
51. Zwanzig, R., *Nonequilibrium statistical mechanics*. Oxford University Press: Oxford ; New York, 2001; p ix, 222 p.
52. Carof, A.; Vuilleumier, R.; Rotenberg, B., Two algorithms to compute projected correlation functions in molecular dynamics simulations. *Journal of Chemical Physics* **2014,** *140* (12).
53. (a) Faradjian, A. K.; Elber, R., Computing time scales from reaction coordinates by milestoning. *Journal of Chemical Physics* **2004,** *120* (23), 10880-10889; (b) West, A. M. A.; Elber, R.; Shalloway, D., Extending molecular dynamics time scales with milestoning: Example of complex kinetics in a solvated peptide. *Journal of Chemical Physics* **2007,** *126* (14).
54. (a) Risken, H., *The Fokker-Planck equation : methods of solution and applications*. 2nd ed.; Springer-Verlag: New York, 1996; p xiv, 472 p; (b) Moyal, J. E., Stochastic Processes and Statistical Physics. *J Roy Stat Soc B* **1949,** *11* (2), 150-210; (c) Lax, M., Classical Noise .4. Langevin Methods. *Rev Mod Phys* **1966,** *38* (3), 541-&.
55. Murphy, T. J.; Aguirre, J. L., Brownian Motion of N Interacting Particles .1. Extension of Einstein Diffusion Relation to N-Particle Case. *Journal of Chemical Physics* **1972,** *57* (5), 2098-&.
56. Schuss, Z., *Theory and Applications of Stochastic Processes: An Analytical Approach*. Springer: New York, 2009; Vol. 170.
57. (a) Best, R. B.; Hummer, G., Diffusive model of protein folding dynamics with Kramers turnover in rate. *Phys Rev Lett* **2006,** *96* (22); (b) Zhang, Q.; Brujic, J.; Vanden-Eijnden, E., Reconstructing Free Energy Profiles from Nonequilibrium Relaxation Trajectories. *J Stat Phys* **2011,** *144* (2), 344-366.

58. Hinczewski, M.; von Hansen, Y.; Dzubiella, J.; Netz, R. R., How the diffusivity profile reduces the arbitrariness of protein folding free energies. *Journal of Chemical Physics* **2010,** *132* (24).
59. Woolf, T. B.; Roux, B., Conformational Flexibility of O-Phosphorylcholine and O-Phosphorylethanolamine - a Molecular-Dynamics Study of Solvation Effects. *J Am Chem Soc* **1994,** *116* (13), 5916-5926.
60. Hummer, G., Position-dependent diffusion coefficients and free energies from Bayesian analysis of equilibrium and replica molecular dynamics simulations. *New J Phys* **2005,** *7*.
61. Schutte, C.; Noe, F.; Lu, J. F.; Sarich, M.; Vanden-Eijnden, E., Markov state models based on milestoning. *Journal of Chemical Physics* **2011,** *134* (20).
62. Bicout, D. J.; Szabo, A., Electron transfer reaction dynamics in non-Debye solvents. *Journal of Chemical Physics* **1998,** *109* (6), 2325-2338.
63. Comer, J.; Chipot, C.; Gonzalez-Nilo, F. D., Calculating Position-Dependent Diffusivity in Biased Molecular Dynamics Simulations. *Journal of Chemical Theory and Computation* **2013,** *9* (2), 876-882.
64. Turkcan, S.; Alexandrou, A.; Masson, J. B., A Bayesian Inference Scheme to Extract Diffusivity and Potential Fields from Confined Single-Molecule Trajectories. *Biophys J* **2012,** *102* (10), 2288-2298.
65. Comer, J.; Schulten, K.; Chipot, C., Calculation of Lipid-Bilayer Permeabilities Using an Average Force. *Journal of Chemical Theory and Computation* **2014,** *10* (2), 554-564.
66. Comer, J.; Schulten, K.; Chipot, C., Diffusive Models of Membrane Permeation with Explicit
Orientational Freedom. *Journal of Chemical Theory and Computation* **2014,** *10* (7), 2710-2718.
67. Cardenas, A. E.; Elber, R., Computational study of peptide permeation through membrane: searching for hidden slow variables. *Molecular Physics* **2013,** *111* (22-23), 3565-3578.
68. Johnson, K. A., Role of induced fit in enzyme specificity: A molecular forward/reverse switch. *J Biol Chem* **2008,** *283* (39), 26297-26301.
69. vanBeek, J.; Callender, R.; Gunner, M. R., The contribution of electrostatic and van der Waals interactions to the stereospecificity of the reaction catalyzed by lactate dehydrogenase. *Biophys J* **1997,** *72* (2), 619-626.
70. Kirmizialtin, S.; Nguyen, V.; Johnson, K. A.; Elber, R., How Conformational Dynamics of DNA Polymerase Select Correct Substrates: Experiments and Simulations. *Structure* **2012,** *20* (4), 618-627.
71. de Graaf, C.; Oostenbrink, C.; Keizers, P. H. J.; van Vugt-Lussenburg, B. M. A.; Commandeur, J. N. M.; Vermeulen, N. P. E., Free energies of binding of R- and S-propranolol to wild-type and F483A mutant cytochrome P450 2D6 from molecular dynamics simulations. *Eur Biophys J Biophy* **2007,** *36* (6), 589-599.
72. Keatinge-Clay, A. T., The structures of type I polyketide synthases. *Nat Prod Rep* **2012,** *29* (10), 1050-1073.

73. Whicher, J. R.; Dutta, S.; Hansen, D. A.; Hale, W. A.; Chemler, J. A.; Dosey, A. M.; Narayan, A. R. H.; Hakansson, K.; Sherman, D. H.; Smith, J. L.; Skiniotis, G., Structural rearrangements of a polyketide synthase module during its catalytic cycle. *Nature* **2014,** *510* (7506), 560-564.
74. Zheng, J. T.; Taylor, C. A.; Piasecki, S. K.; Keatinge-Clay, A. T., Structural and Functional Analysis of A-Type Ketoreductases from the Amphotericin Modular Polyketide Synthase. *Structure* **2010,** *18* (8), 913-922.
75. Zheng, J. T.; Piasecki, S. K.; Keatinge-Clay, A. T., Structural Studies of an A2-Type Modular Polyketide Synthase Ketoreductase Reveal Features Controlling alpha-Substituent Stereochemistry. *Acs Chem Biol* **2013,** *8* (9), 1964-1971.
76. Straatsma, T. P.; Mccammon, J. A., Multiconfiguration Thermodynamic Integration. *Journal of Chemical Physics* **1991,** *95* (2), 1175-1188.
77. Dellago, C.; Hummer, G., Computing Equilibrium Free Energies Using Non-Equilibrium Molecular Dynamics. *Entropy-Switz* **2014,** *16* (1), 41-61.
78. Herschbach, D. R.; Johnston, H. S.; Rapp, D., Molecular Partition Functions in Terms of Local Properties. *Journal of Chemical Physics* **1959,** *31* (6), 10.
79. *Wolfram Research, Inc., Mathematica, Version 8.0*. Champaign, IL, 2010.
80. Essmann, U.; Perera, L.; Berkowitz, M. L.; Darden, T.; Lee, H.; Pedersen, L. G., A Smooth Particle Mesh Ewald Method. *J Chem Phys* **1995,** *103* (19), 8577-8593.
81. Jorgensen, W. L.; Chandrasekhar, J.; Madura, J. D.; Impey, R. W.; Klein, M. L., Comparison of Simple Potential Functions for Simulating Liquid Water. *Journal of Chemical Physics* **1983,** *79* (2), 926-935.
82. Allen, M. P.; Tildesley, D. J., *Computer Simulation of Liquids*. Oxford University Press: New York, 1987.
83. Heyes, D. M., Molecular-Dynamics at Constant Pressure and Temperature. *Chem Phys* **1983,** *82* (3), 285-301.
84. Papoulis, A., *Probability, Random Variables, and Stochastic Processes*. Third ed.; McGraw-Hill Inc.: New York, 1991.
85. Wolfenden, R.; Andersson, L.; Cullis, P. M.; Southgate, C. C. B., Affinities of Amino-Acid Side-Chains for Solvent Water. *Biochemistry-Us* **1981,** *20* (4), 849-855.
86. Tuckerman, M.; Berne, B. J.; Martyna, G. J., Reversible Multiple Time Scale Molecular-Dynamics. *Journal of Chemical Physics* **1992,** *97* (3), 1990-2001.
87. Gardiner, C. W., *Handbook of stochastic methods for physics, chemistry, and the natural sciences*. 3rd ed.; Springer-Verlag: Berlin ; New York, 2004; p xvii, 415 p.
88. Kampen, N. G. v., *Stochastic processes in physics and chemistry*. Rev. and enl. ed.; North-Holland: Amsterdam ; New York, 1992; p xiv, 465 p.
89. Hanggi, P.; Talkner, P., Memory Index of 1st-Passage Time - a Simple Measure of Non-Markovian Character. *Phys Rev Lett* **1983,** *51* (25), 2242-2245.
90. Hawk, A. T., Milestoning with coarse memory. *Journal of Chemical Physics* **2013,** *138* (15).
91. Ermak, D. L.; Mccammon, J. A., Brownian Dynamics with Hydrodynamic Interactions. *J Chem Phys* **1978,** *69* (4), 1352-1360.

92. Kloeden, P. E.; Platen, E., *Numerical solution of stochastic differential equations*. Springer-Verlag: Berlin ; New York, 1992; p xxxv, 632 p.
93. Kreuzer, S.; Elber, R., Catch bond-like kineics of helix cracking: Network analysis by molecular dynamics and milestoning. *Journal of Chemical Physics* **2013,** *139*.
94. Ryckaert, J. P.; Ciccotti, G.; Berendsen, H. J. C., Numerical-Integration of Cartesian Equations of Motion of a System with Constraints - Molecular-Dynamics of N-Alkanes. *J Comput Phys* **1977,** *23* (3), 327-341.
95. Weinbach, Y.; Elber, R., Revisiting and parallelizing SHAKE. *J Comput Phys* **2005,** *209* (1), 193-206.
96. Ruymgaart, A. P.; Elber, R., Revisiting Molecular Dynamics on a CPU/GPU System: Water Kernel and SHAKE Parallelization. *Journal of Chemical Theory and Computation* **2012,** *8* (11), 4624-4636.
97. Kabsch, W., Solution for Best Rotation to Relate 2 Sets of Vectors. *Acta Crystallogr A* **1976,** *32* (Sep1), 922-923.
98. Warshel, A., Electrostatic origin of the catalytic power of enzymes and the role of preorganized active sites. *J Biol Chem* **1998,** *273* (42), 27035-27038.
99. Allen, M. P.; Tildesley, D. J., *Computer simulation of liquids*. Clarendon Press ; Oxford University Press: Oxford England
New York, 1987; p xix, 385 p.
100. Risken, H., *The Fokker-Planck Equation* Springer: 1996.