UNIVERSITY OF SIENA

Department of Biotechnology, Chemistry and Pharmacy*

PHD SCHOOL IN BIOCHEMISTRY AND MOLECULAR BIOLOGY, XXXIV CYCLE

PhD coordinator: Prof.ssa Lorenza Trabalzini,

(Prof.ssa Annalisa Santucci -2020)

MD*: A novel Molecular Dynamics approach to reveal the Target/small-molecule interaction secretes

S.S.D: BIO/10

Tutor:

Prof.ssa Ottavia Spiga,

Dott. Alfonso Trezza

PhD student:

Adam Gabor Laux

Academic year: 2021/2022

* Department of Excellence 2018-2022

Declaration

I hereby declare that, the contents and organization of this dissertation constitute my own original work and does not compromise in any way the rights of third parties, including those relating to the security of personal data.

Adam Gabor Laux

2022

Dedication:

May every being ailing with disease
Be freed at once from every malady.
May all the sickness that afflicts the living
Be instantly and permanently
healed.

0

Abstract

In 1957, John Kendrew determined successfully the atomic structure of the myoglobin. Since, the scientific community increased the interest towards the knowledge of how the macromolecular structures are able to fold, move and interact inside the biological environment. To know the molecular basis of how a biological system works is a key step to reveal the secrets of the life. Molecular dynamics (MD) simulation, first developed in the late 60s, has advanced from simulating gases as elastic collisions between hard spheres to complex biological systems formed by thousands of atoms. However, several limits occur, such as: computational time, resource usage, non-feasibility etc. Automation, algorithm research and standardizations are crucial in order to curb cost of resources and achievement of results. In particular, accelerated search methods in the phase-space are increasingly studied to overcome the present barriers of IT capabilities.

In this thesis, we expose a novel and innovative approach of MD, named, Molecular dynamics-star (MD*), MD* is an accelerated binding/unbinding path finding MD algorithm based on the semantics from Artificial Intelligence (AI) Astar (A*) informed-search algorithm. MD* is implemented in GROMACS with control and evaluation cycles written in python for the accelerated simulation. The viability of MD* was evaluated simulating the binding/unbinding process of the LUSH protein/Ethanol co-crystal. The MD* simulation showed an accurate overlapping of the ethanol binding pose compared with the crystal, revealing its reliability. Our work, could be open novel frontiers in computational biochemistry field, providing the molecular basis of biological system interactions.

1 Contents

List of Figures

1 Introduction

1.1 Molecular dynamics in context

In the course of scientific development the advent of high performance computing power has opened a spectrum of new possibilities. From physics of materials to astrophysics, via nuclear physics, metrology, economics, ecology, architecture and engineering - simulation and forecast of a wide range of concrete system behavior and phenomena has become feasible. Simulation of such diverse phenomena, its predictive power in the particular cases, highlight the genius of the precomputer era scientists that discovered the basic laws of physics, chemistry that form the bedrock of these simulations – discoveries that are one of the greatest achievements of human kind's history.

Microbiology is not an exception in this regard, in the 1950s via statistical physics, first, the Maxwell-Boltzmann ideal gas' intensive and extensive physical quantities were obtained as averages of the particular solution of the many particle differential equation system of atomic coordinates, velocities and potentials modelled as hard-spheres. The Newton equations of motion of the systems' particles were discretized and integrated on a main-frame computer, starting the era of molecular dynamics (MD). The pioneering results were in accordance with the theoretical laws. Evolving from homogenous compounds via-via to heterogonous systems computer simulation of microbiological systems evolved. Other cornerstones of MD were the field of crystallography x-ray diffraction, in which generation of scientist meticulously mapped and recorded the configuration of biological matter, invaluable data for the initial conditions to be put into the equations of motion; similarly the field of physical-chemistry: the development and parametrization of force fields to be put into the equations as potentials for various molecules.

Nowadays, at pharmaceutical companies, university departments, in vitro laboratory experiments are increasingly substituted by molecular dynamics simulations. For example, in the field of computational drug design, protein unfolding modelling etc. There is a two way interaction in the literature of the virtual laboratory and microbiology experiments, verifying hypothesis suggested by one another.

1.2 Gaps

In the last four decades, on the one hand, computational power increased approximated by the empirical relationship of Moore's law ("the number of transistors in integrated circuits double every two years") on the other hand, drug discovery, is becoming slower and more expensive over time ("inflation-adjusted cost of developing new drugs roughly doubles every nine years") - called Eroom's (anadrome word of Moore) law. This apparent contradiction advises us that to rely solely on computer power will not deliver desired solutions fast enough in the field of microbiology for the future.

In particular, MD - as any technical tool - by its nature is used by scientist to its out most potential to solve ever more and more complex and subtle problems. Computational time, resource usage and non-feasibility of MD simulations, the evaluation cost of the BIG Data it generates, increase as it is used to simulate microbiological systems of increasing number, complexity and phenomena of more subtle nature that occur with less probability.

Automation, algorithm research and standardization is an important factor in the advancement of combinatorial chemistry and computational drug design in the pharmacological industry. These factors curb the cost of resources and can achieve optimal results in more a timely manner. One line of development, MD simulation of the type, that use accelerated search methods in the phase space flow of the modelled system, are being continuously studied to overcome the ever present barriers of IT capabilities, expand the number of candidate molecules in screening for drugs, open up ways to scan and exclude potential undesired interactions with innumerable other parts of the living organism of the candidates.

In the future, accurate and fast MD modelling algorithms - in principal – will have even less environmental impact then performing the given series of real experiments, reduce chemical compound usage, waist product management and not least: animal experimentation. While MD will always need experimental verification, its role is inevitable to increase.

1.3 Objecives

In this thesis a novel approach, Molecular dynamics-star (MD*) simulation is exposed. MD* is an accelerated binding/unbinding path finding molecular dynamics algorithm based on the semantics from Artificial Intelligence (AI) A-star (A*) informed-search algorithm, its precursor is supervised MD (suMD) in chapter 2.2. Alternative paths are evaluated by ranking based on suitable heuristic function for the given problem (Chapters 2.6 and 2.8), and are extended on-the-fly by standard md simulations until a terminal condition. MD* is naturally suitable for parallel computing.

The various possible sources of having alternative paths from (quasi) identical initial conditions in MD simulations is extensively studied among ensembles/dynamic models used in the literature and in the MD parameters space (chapter 2.4) , their feasibility in GROMACS implementation (velocitygeneration, Langevin, Andersen, Berendsen, Velocity-Scaling, Nosé-Hoover (NH), Parinello in chapter 2.5. A „scientific acceptance" index is calculated over a corpus of md publication to ranks the possible ways (Chapter 2.3). On the top of the list – currently – is the Nosé-Hoover (NH), Parinello thermo-, barostat, based on which the Variable Partition (VP) NH MD* algorithm was codified (Chapters 2.11 and 2.12).

MD* algorithms is implemented in GROMACS md with control and evaluation cycles written in python for the accelerated simulation of the 3b7a protein-small molecule binding/unbinding process for six different model choices. (Chapter 3.1 and 3.2)

The viability of MD* was evaluated simulating the binding/unbinding process of the LUSH protein/Ethanol co-crystal. The MD* simulation showed an accurate overlapping of the ethanol binding

pose compared with the crystal, revealing its reliability. Our work, could be open novel frontiers in computational biochemistry field, providing the molecular basis of biological system interactions.

A specific angular orientation heuristic function for this problem based on the statistical properties of the unbinding path vicinity is constructed algorithmically via reference atom choices form the protein. (Chapter 2.7.2 and 3.1.1) The protein's stability was studied during the simulation (Chapter 3.16). MD simulation from the final VPNH MD* configuration small molecule pose was analyzed (Chapter 3.1.7).

The pyGro blended script language was codified and documented in the Appendix for the implementation of the increasingly more complex MD, MD* algorithms. PyGro combines GROMACS md commands, python flow control and evaluation math/stats library code with UNIX shell commands using their original syntax to keep clean and easy to comprehend code. (Appendix I and II)

2 Material and Methods

In the subchapters below I present a concise description of the various methods by which the thesis results were obtained. As materials the concrete software implementation of these.

2.1 Definition of reference points and directions LUSH protein/Ethanol co-crystal

In crystal structures of the Drosophila melanogaster protein LUSH in complexes with short-chain alcohols binding are generally occurs in water-filled pockets and for stable complexes.

Figure 1. PyMol rendering of LUSH protein/Ethanol co-crystal

Because the protein is not a rigid body, for the porous of the characterization of the small molecules movement inside the protein, and the various relative movements of the parts of the protein, its own movements during simulation, the raw coordinates calculated in MD are insufficient. With the method described here it is expressed as distances to some geometrical defined reference points and directions which are fitted to definite parts of the protein and thus move with it. It is a minimization method that finds the direction that gives the minimum moment of inertia around the barycenter of a residue section of the protein, a reference group of atoms.

By visual inspection in the VMS software[1] 4 helical regions of the protein was selected to define reference points and directions. Basically we fix co-moving cylindrical coordinate systems [origin $\underline{O'}$ and coordination (r',θ',h')] to the tubular parts of the protein, each in a way to have the $\underline{e_{z'}}$ axis as close as possible to the symmetry axis of the given part, pointing to the residues' numbers increasing direction and $\underline{O'}$ origin to be on the symmetry axis at the midpoint.

Anticipating, with the below introduced notation,

$\underline{O'} \rightarrow \underline{O}^{ref}$,

$\underline{e_{z'}} \rightarrow \underline{S}ref$,

[1] VMD is a software for molecular dynamics visualization. It provides a variety of GUI tools for trajectory analysis. To analyze the Gromacs trajectory in VMD, load the .gro (coordinate) file and then select "load data into molecule" and load the .xtc or .trr (Gromacs trajectory files) into the .gro structure; from the command line, issue "vmd GRO_FILE.gro XTC_FILE.xtc". This will load the xtc file into the gro structure then. pdb file can be saved which contains for all frames an atom positions in sequence. On the in the chapter's figures graphical representation of water is turned off, the protein is rendered with point style, size 2 and also with tube radius zero, EOH lead atom (a=1959) rendered with point style size 22.

$d^{ax} = h'$,

$d^{rad} = r'$.

Thus the coordinate systems via their $\underline{O'}$ origins and $\underline{e_{z'}}$ directions are floating with the protein during the simulation, fixed to and defined by the residue sequence of the helix, the time dependent transformation ***T*** from (x,y,z) to *(r,θ,h) are defined as*:

1) form residue 106 to 115 containing $n_1^{ref} = 156$ atoms

$\underline{O}_1^{ref}$: position, the geometric barycenter (calculated with unit masses)

$$\underline{O}_1^{ref} = 1/ n_1^{ref} \sum_i 1 \square \underline{x}_i$$

$\underline{s}_1^{ref}$: vector, the direction that minimize the moment of inertia respect to the $\underline{x}_1^{ref}$ point, its direction is pointing towards the 115 residue and it is approximately parallel to the axis of the helix[2].

$$\underline{s}_1^{ref} = \text{minarg}_{\underline{s}} (1/ n_1^{ref} \sum_i 1 \square ((\underline{x}_i - \underline{O}_1^{ref}) \times \underline{s})^2$$

where $\times$ is the sign of vector product multiplication

The moment of inertia's formula with unit mass is:

$$\underline{I}_{1ref} = 1/ n_{1ref} \sum_i 1 \square (\underline{x}_i - \underline{O}_{1ref}) \times \underline{s}_{1ref})_2$$

For illustration their values calculated from the initial configuration (MD_0_2 first frame) are:

$\underline{O}_{1ref}$	[27.03442308 27.19910256 35.77858974]	Å
$\underline{s}_{1ref}$	[0.71044002 -0.05909554 0.7012722]	Å
$\underline{I}_{1ref}$	17.2199	Å$_2$

from which the characteristic radius of the helix is 2.09 - 4.15 Å.

[2] A solid cylinder of diameter *2R*, length *L* has *½ m R²* moment of inertia respect to its symmetry axis, *¼ m R² + 1/12 m L²* respect to its central diameter, for an off barycenter parallel axis a *m ℓ²* component adds, where *ℓ* is the axis's distance from the barycenter. For $L > \square 3\,R$ -which is clearly our case - the axis with the minimum moment of inertia is the symmetry axis. For tubular shapes the condition is $L > \square 6\,R$. *The first* limit is approximately 3 Å for 2 Å characteristic radius, the second is 9 Å for 4 Å, to be compared with the characteristic length of 15-20 Å of the chosen residue sections.

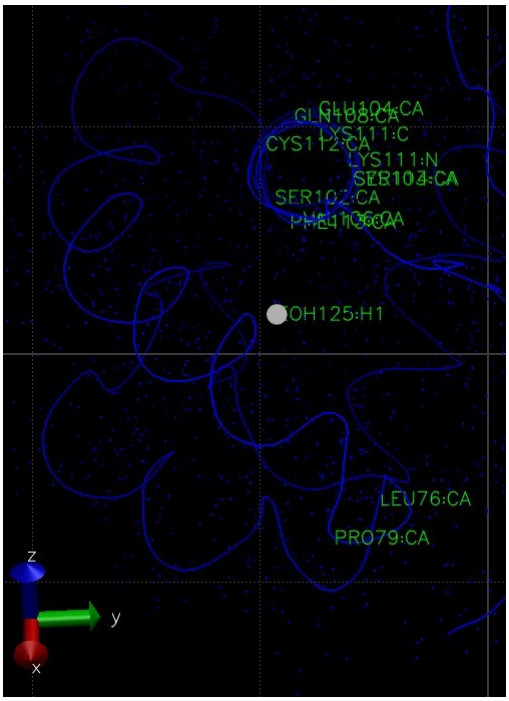


Figure 2 VMS graphical output of the initial configuration, some atoms of residues 106 to 115 are labeled. The direction of sight is approximately $\underline{s}_1^{ref}$, grid lines are 10 Å apart.

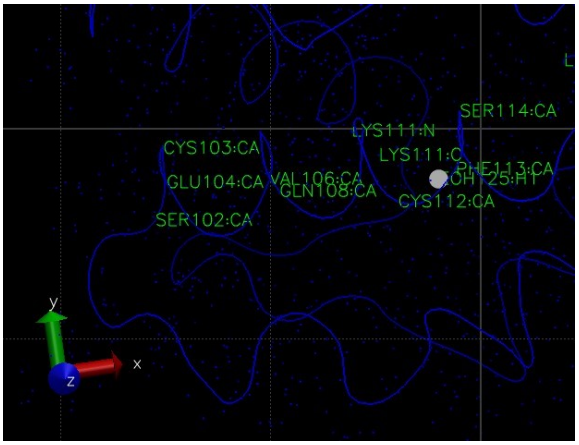


Figure 3 VMS graphical output of the initial configuration, some atoms of residues 106 to 115 are labeled. The direction of sight is the z axis, grid lines are 10 Å apart.

Method:

1) The .pdb coordinate files are generated with a series of command for all *s* and *b-s*. [3] Implementation code: *traj_generator.gup*.

2) These are concatenated to form the escape path and re-indexed by time. Filtered for the residues in the definition to obtain the relevant set of coordinates and calculate the geometric barycenter.

For the $\underline{s}$ vector, an initial normalized direction is taken form the difference of the last atoms' coordinates minus the firsts' in the definition. Starting from this guess a minimization algorithm finds the desired direction that is approximately parallel to the axis of the helix[45]. Implementation code *RG_calc.py*.

The geometric barycenter for index groups is calculated by Gromacs also, while the $\underline{s}$ direction and its moment of inertia isn't.

2) to 4) in similar way for residues 44-54, 82-97, 64-76 respectively. See next table.

Table of the reference points and directions and their values calculated at the initial configuration of the simulation.

ref		residues	quantity	value	unit
1	RG1	106-115	n_{1ref}	156	#

[3] the trjconv output's first line is: CRYST1 60.584 60.584 60.584 90.00 90.00 90.00 P 1 1, meaning position coordinates are already Cartesian, the 90's indicate that the axes are pair-wise perpendicular.

[4] A solid cylinder of diameter *2R*, length *L* has *½ m R²* moment of inertia respect to its symmetry axis, *¼ m R² + 1/12 m L²* respect to its central diameter, for an off barycenter parallel axis a *m ℓ²* component adds, where *ℓ* is the axis's distance from the barycenter. For *L* > □*3 R* -which is clearly our case - the axis with the minimum moment of inertia is the symmetry axis. For tubular shapes the condition is L> □*6 R*. *The first* limit is approximately 3 Å for 2 Å characteristic radius, the second is 9 Å for 4 Å, to be compared with the characteristic length of 15-20 Å of the chosen residue sections.

[5] The lead atom H is on the 57th residue THR thus part of reference group 4

			$\underline{O}_{1ref}$	[27.03442308 27.19910256 35.77858974]	Å
			$\underline{s}_{1ref}$	[0.71044002 -0.05909554 0.7012722]	Å
			$\underline{I}_{1ref}$	17.2199	$Å_2$
2	RG2	44-54	n_{2ref}	175	#
			$\underline{O}_{2ref}$	[24.18840571 32.33626857 26.34574857]	Å
			$\underline{s}_{2ref}$	[0.51366105 -0.3633315 -0.77726607]	Å
			$\underline{I}_{2ref}$	16.2880	$Å_2$
3	RG3	82-97	m_{3ref}	257	#
			$\underline{O}_{3ref}$	[34.36233463 19.27968872 34.21533074]	Å
			$\underline{s}_{3ref}$	[-0.94508985 -0.31535957 -0.08575264]	Å
			$\underline{I}_{3ref}$	16.2802	$Å_2$
4	RG4	64-76[5]	n_{4ref}	257	#
			$\underline{O}_{4ref}$	[37.49658385 22.07 24.3242236]	Å
			$\underline{s}_{4ref}$	[0.4159748 0.90591408 -0.07927579]	Å
			$\underline{I}_{4ref}$	14.099	$Å_2$

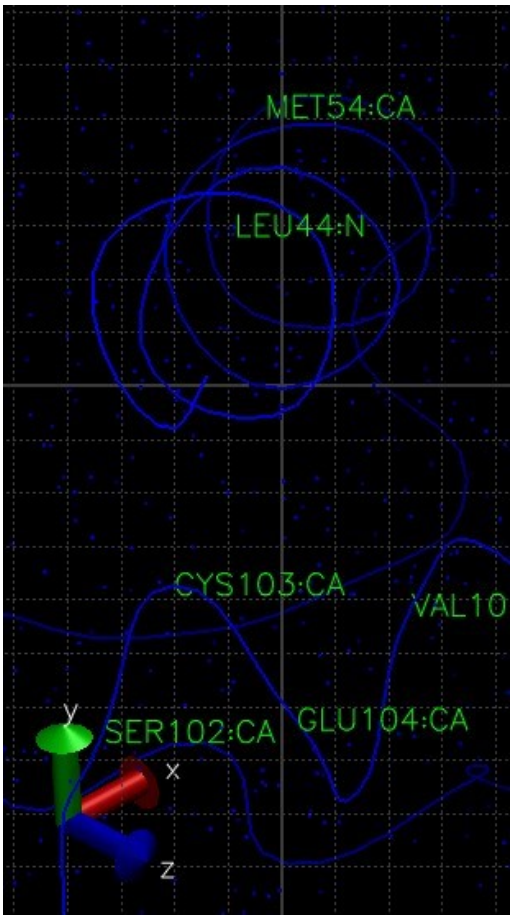


Figure 4. RG2: residues 44-54

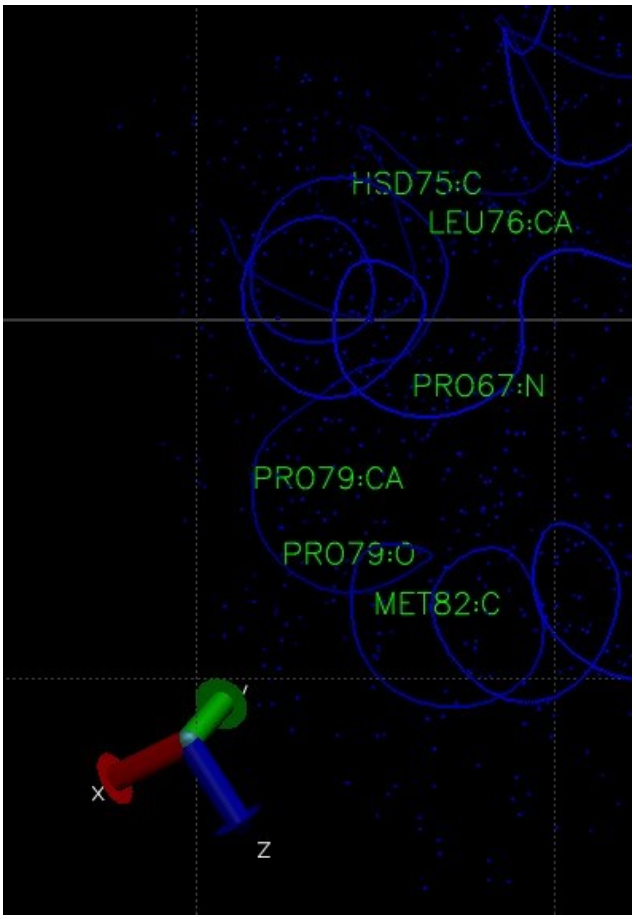


Figure 5. RG4: residues 64-76

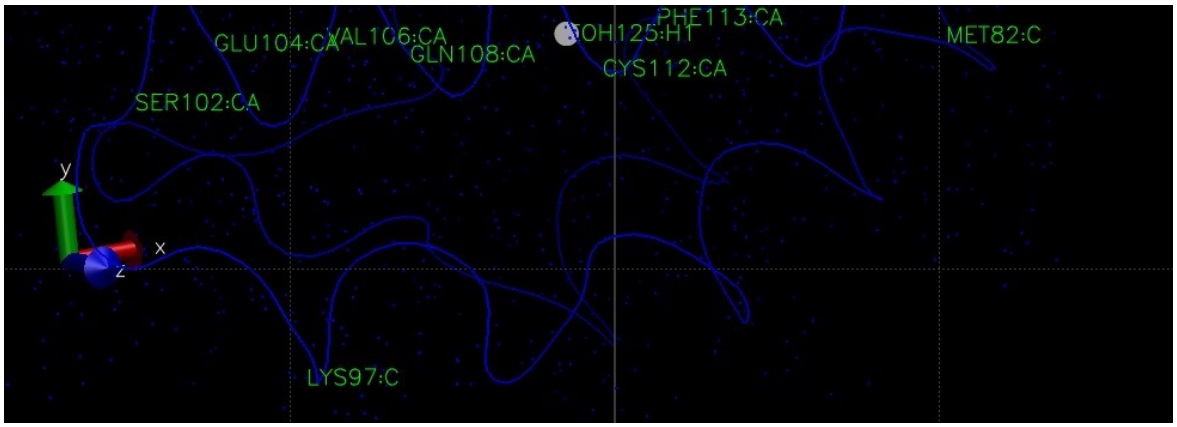


Figure 6. RG3: residues 82-97

2.2.1 Characteristic of the 4 four reference groups: position

They are not in one plain. That is in formula:

$$(\underline{O}_{2ref} - \underline{O}_{1ref}) / |\underline{O}_{2ref} - \underline{O}_{1ref}| \times (\underline{O}_{3ref} - \underline{O}_{1ref}) / |\underline{x}_{3ref} - \underline{O}_{1ref}| \square \underline{O}_{4ref} / |\underline{O}_{4ref}| \neq 0$$

where × is a vector multiplication and □ is a scalar multiplication. The left hand side is -0.34707 for the initial configuration and remains negative during the unbinding. (Implementation code *ref_points_directions.py)*

The axial distance of atom *i* from the *k*-th reference point measured with the positive direction taken along the residues ($\underline{s}_k^{ref}$ orientation) is:

$$d_{ax\ i,k} = (\underline{x}_i - \underline{O}_{kref}) \square \underline{s}_{kref}$$

The radial distance of atom *i* from the *k*-th reference point is:

$$d_{ri,k} = |(\underline{x}_i - \underline{O}_{kref}) \times \underline{s}_{kref}|$$

2.2.2 The small molecule's position respect to the reference points

As for any atom, also for the EOH molecule the *k=1…4* radial and axial reference distances can be calculated. Inversely given the 4,4 quantities o*f* $d^{ax}_{EOH,k}$, $d^{r}_{EOH,k}$, *the small molecules* $\underline{x}$*'* 3 position coordinate components can be calculated by the over specified equation system $d_{ax\ EOH,k} = (\underline{x}' - \underline{O}_{kref}) \square \underline{s}_{kref}$, $k=1…4$ $d^{r}_{EOH,k} = |(\underline{x}' - \underline{O}_k^{ref}) \times \underline{s}_k^{ref}|$, $k=1…4$

The EOH molecule's direction respect to the reference directions

Let $\underline{c}$ be the normalized vector pointing from C1 to the C2 atom of the EOH molecule, then the 4 angels respect to the reference directions are in degrees:

$$\theta_{EOH,k} = 180/\pi \arccos(\underline{c} \cdot \underline{s}_k^{ref})$$

The EOH molecule's velocity respect to the reference directions

$$\dot{d}_{ax\ EOH,k} = (\dot{\underline{x}}' - \underline{O}_{kref}) \cdot \underline{s}_{kref}$$

$$\dot{d}_{rEOH,k} = |(\dot{\underline{x}}' - \underline{O}_{kref}) \times \underline{s}_{kref}|$$

The EOH molecule's angular velocity respect to the reference directions

$$\dot{\theta}_{EOH,k} = 180/\pi \arccos(\dot{\underline{c}} \cdot \underline{s}_k^{ref})$$

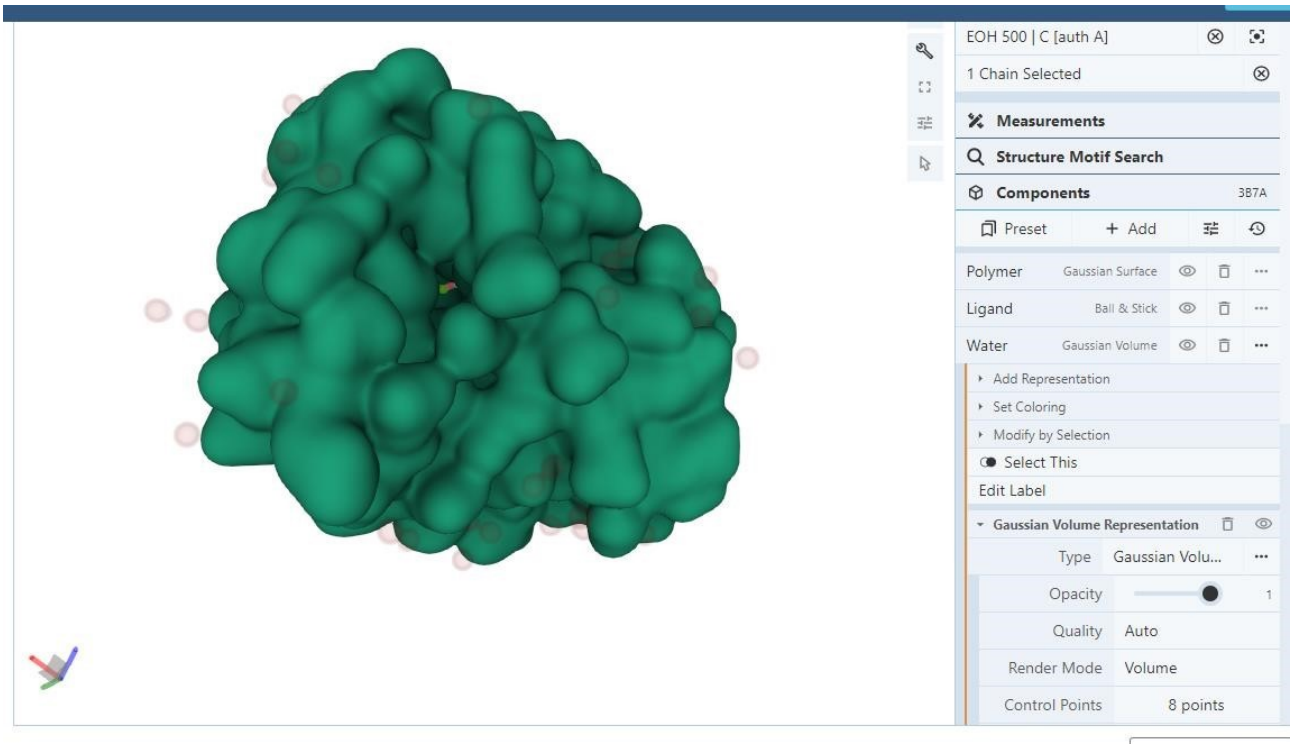


Figure 7. A rare view of the small molecule inside the binding pocket, rendering woth Gaussian Volume Rapresentation by protein explorer app on the PBS website.

2.3 Origin of MD* algorithm

A-star is an informed search (Hart, Nilsson, & Raphael, 1968), optimal pathfinding algorithm.

Typical problems are: 1) optimal path finding in connected closed regions of the Euclidian space (obstacle avoidance), with optimality conditions expressed as minimal distance/time/cost traveled. 2) Minimal path problems in networks, represented as a graph traversal algorithm.

From a starting *vertex* (n_0), *cost* (c) of traversing an *arc* (e_{ji} connection between n_j and n_i) is distance is travel time or economic cost. n_t is the terminal vertex and the solution is a path, a sequence of vertexes (arc traversal sequence) : $n_0, n_1, \ldots, n_t$.

In practical applications a node is associated with the coordinates or labels of a place and the optimal point to point path problem can be formalized as

Point-to-point shortest path problem (P2P):

– Given:

1) directed graph with nonnegative arc lengths $l(n_j, n_i) = c_{ij}$;

2) source vertex n_0; 3) target vertex n_t.

– Goal: find shortest path from n_0 to n_t.

The algorithm starting from the zero node explores a set of increasingly longer paths in iterations, at a given iteration the horizon of the search is the set of the last nodes in each sequence. Estimated total cost at a horizon node is $f(n_h)$ is $c(n_0 n_h) + h(n_h n_t)$, where h is the heuristic function. The ranking of total cost of the sequences determine which one to extend from the horizontal node until the terminal node is reached.

A* terminology	MD* terminology
coordinates, labels	$(\underline{q},\underline{p})$ generic point of the phase space
graph	structure of paths originating from the initial condition
starting *vertex* (n_0)	$(\underline{q}_o,\underline{p}_o)$
arc (e_{ji})	simulated path from $(\underline{q}_i,\underline{p}_i)$ to $(\underline{q}_j,\underline{p}_j)$
n_i, vertex	$(\underline{q}_i,\underline{p}_i)$ simulated phase space configuration of the system
arc lengths $l(n_j, n_i)$	effective simulated time
horizon	set of final $(\underline{q}_i,\underline{p}_i)$ of alternative paths at a given iteration
$f(n_h)$ is the cost of the path from the start vertex to n_h	sum of simulation time from the initial condition to $(\underline{q}_h,\underline{p}_h)$
$h(n_h)$ heuristic function	$h(\underline{q}_i,\underline{p}_i,\underline{q}_j,\underline{p}_j)$ heuristic functional
terminal vertex n_t	$t(\underline{q},\underline{p})$ terminal condition indicator function

Due to the high computational cost of the MD simulations, in the MD* implementation we simulate a limited number of edges, and currently keep track only a substructure of the paths originating from the initial condition. Similarly the high memory allocation for a given system configuration advised us to define the cost function form the start as the effective simulation time – avoiding the need to step to configurations further back in time. For future developments the cost of the path from the start could be defined as the sum of the heuristic's values realized, or other desired cartelistic of the path.

Another difference is that the possible set of moves over the edges in A* are predetermined by the topology of the problem (for example a grid) while in MD* it is the simulated behavior of the whole physical system. In A* the final result is a set of nodes which the agent should traverse to arrive to the terminal node with minimum cost, in MD* the final result is a set of choices of path sections by the algorithm which lead to the termination condition that was achieved minimizing cost of simulation time taking in consideration the constrains over the sunset of possibilities simulated.

How the path sections generated are continued is a crucial problem for the evaluation of the acceptability of MD* results and depends on the model choice of the MD alternative path generation cause, its limitation should be studied. On it depends weather MD* results can be interpreted as a tool of scientific inquiry or a representation of a physical phenomenon under an accepted model.

In particular taken for given that the MD simulation methodology is accepted as a good model of a real phenomenon in nature, the method of generation of the alternative paths should be evaluated separately for its likelihood in reality and embedded in a specific model.

2.4 Methodologies used in MD simulation and estimation their status of being "scientifically accepted" form publications

I classify a corpus of publications by reference to the various methodologies in MD simulation at present by vintage time of publication, and attempt to construct a quantitative accepted index (I_A) for the rough indication of their being scientifically mainstream in the context of Molecular Dynamics.

ranking	methodology	I_A	term
1	Nosé-Hoover	63,99	"Nosé-Hoover"
2	velocity scaling	57,67	"velocity scaling"
3	Andersen	11,45	"Andersen thermostat"
4	Berendsen	9,57	"Berendsen thermostat"

5	velocity generation	<2 see[6]	"velocity generation"

The corpus consist of the free web database of Clarivate PLC public analytics company available at https://www.webofscience.com/ , Biological Abstracts on Web of Science.

Index is calculated as a weighted sum of the publication hits by the reciprocal of the time of publication until 2022.01.01 in years.

Coincidentally there is a strong negative correlation between simplicity of the implementation of a methodology and its I_A .

In the process of research of this thesis we moved upward on the methodology ranking list, adapting methodologies into MD* to arrive to the first ranked from the last: we moved from the simpler to more complicated in terms of implementation as natural is it.

2.5 Sources of different MD path generation

MD simulation integrates the differential equation system based on the choice of the modelled ensemble, via the specific thermo- and barostats. Isothermal and isobaric simulations (NPT) are most relevant to confront with experimental data.

A stochastic thermostat is, for example, the Andersen collision type. During simulation, this model assigns in a stochastic quasi-periodically manner to some part or to all of the systems' atoms kinetic energy (velocities) drawn from the theoretical thermally determined velocity distribution (Maxwell-Boltzmann) of target temperature; overwriting those determined by the prevailing forces trough the Newton equations at the moment [Andersen1980] . In this approximation a non-localized interaction to the heat bath is introduced. The random redistribution erases the old velocities it eventually overwrites, as such, for strong coupling expressed in short characteristic time of the interaction it can make lose the protein motions coherence of different parts. Different realization of the stochastic assignments (the random sequence) leads to different time evolution of the system.

Velocity generation is an instantaneous assignment of atomic velocities at the beginning of a simulation to the atoms from the theoretical thermally determined velocity distribution (Maxwell-Boltzmann) of target temperature

Deterministic thermostats and barostats couple a series of damped harmonic oscillator to all atoms of the system [Martyna1992]. This new system is evolving in an extended phase space (thermostat's position and impulse is added) with a different Hamiltonian. The evolution of the common variables are different respect to the non-extended system's. Further the coupling is switched on during a characteristic time smoothly from an initial level to its full strength.

[6] There were no hits in the public part of the database in relation to MD simulation, limit was estimated from the articles in Appendix III, supposing the abstract contained the search fraise (which was not the case)

Berendsen modeled via weak coupling to an external ‘heat bath’ [Berendsen1984] in which deviation of system from a target temperature is corrected by scaling the velocities, resulting in an exponential decay of temperature deviation.

Berendsen pressure coupling is a weak coupling yields exponential relaxation. Equations of motion are modified with a first order relaxation of pressure towards the target pressure, by rescaling the box and the coordinates with a factor which is proportional to the isothermal compressibility and the coupling time constant causing the volume to change.

Parrinello-Rahman pressure coupling uses an extended Hamiltonian with extra degree of freedom where volume and shape of the system allowed to fluctuate [Parrinello1981]. Most cases the Parrinello-Rahman barostat is combined with the Nosé-Hoover thermostat. Velocity scaling is a combination of Berendsen weak coupling with an additional stochastic component.

A standard MD practice is to assign different thermostats to different group of atoms in parallel with identical target temperatures to prevent prolonged temperature differences of separate components called ‘hot-solvent, cold-solute’ phenomenon.

In general different coupling characteristic times yield different evolution of the system.

		time reversible dynamics [7]	*stochastic nature* [8]	*case for*	*Different*[9] *MD paths are generated by*	*Typical value*	

[7] Theoretically, currently GROMACS do not allow negative time steps and reversal of velocities as initial conditions. In practice only on identical hardware/software configuration, numerical representation and compiled version *reproducibility* can be archived in MD. *Reproducibility* is necessary condition for time reversibility.

[8] Fixing the random seed variables in the md parameter files for the simulation of random extractions reproducibility can be archived to a certain degree on the same hardware configuration and numerical representation.

[9] With identical initial conditions of the (non-extended) system

1)	*Weak-coupling scheme of Berendsen*	*yes*	*no*	*a)*	*different initial scaling factors:* λ_0	*formula, implemented with max 1.25, min 0.80*	
				b)	different n_{TC} time steps	*1,10*	
				c)	*different time constants* τ	*0.05 (equilibration 0.01)*	
2)	*Velocityrescaling weakcoupling scheme*	*no*	*yes*	*a)*	*different realization of the dW a Wiener process*		

				b)	*different time constants* τ_T	*0.05 (equilibration 0.01)*	
3)	*Andersen thermostat*	*no*	*yes*	*a)*	*different time constants:* τ_T	*>10 ps*	
				b)	*simultaneously (massive collision) vs. probabilistic partial*		
				c)	*different realization of random extraction form M. dist*		
4)	*Extendedensemble approach by Nosé - Hoover*	*yes*	*no*	*a)*	*different mass parameter of the reservoir: Q*	$Q = \frac{\tau_T^2 T_0}{4\pi^2}$	
				b)	*different initial heat bath parameters:* $\xi_i(t=0)$		
				c)	*different number of chains of thermostats: M*	*10*	
5)	*Group temperature coupling*			*a)*	*different group and permutation of previous parameters described*		
				b)	*not thermostat some groups (protein)*	*time constant* $\tau_T = -1$	
6)	*Velocity generation*	*no*	*yes (only initial condition)*	*a)*	*keep only coordinates re-generate initial velocities*		

2.6 GROMACS implementation for sources of different MD path generation

The GROMACS implementation of the various MD models make it possible to interact with the parameters mainly in two ways, the parameter file and the index file for definition of groups [GROMACS2019]. In the following tabular form the technical name and way to interact with the parameters are listed for the previous point's model choices:

		case	*variable, parameter*	*GROMACS implementation in md parameter file*

1)	*Weak-coupling scheme of Berendsen*[7]	*a)*	λ_{0i}	tcoupl = berendsen Not directly set from *md parameter file* *By using* tc-grps and *defining different partitions as groups in index files* for the different md runs. *min/max hardcoded in:* src/gromacs/mdlib/coupling.cpp *real lll = std::sqrt(1.0 + (dt / opts->tau_t[i]) * (reft / T - 1.0)); ekind->tcstat[i].lambda = std::max<real>(std::min<real>(lll, 1.25), 0.8);*
		b)	n_{TC}	**tcoupl = berendsen** **nsttcouple = …**
		c)	τ	**tcoupl = berendsen tau-t = …**
2)	*Velocityrescaling weak-coupling scheme*	*a)*	*dW*	**tcoupl = v-rescale** **ld-seed = … ; (integer 0 … 2^{63}) to have only stochastic component: tau-t = 0;**
		b)	τ_T	**tcoupl = v-rescale** **tau-t = … ; (workts also for 0.0)**
3)	*Andersen thermostat*	*a)*	τ_T	**tcoupl = andersen tau-t = …** **andersentemperaturecoupling.cpp**
		b)		**tcoupl = andersen-massive vs. tcoupl = andersen**
		c)		**andersen_seed = … ; (0 … 2^{63})** **(default = 0)**
4)	*Extendedensemble approach by Nosé - Hoover*	*a)*	*Q*	**tcoupl = nose-hoover tau-t = …** $Q = \frac{\tau_T^2 T_0}{4\pi^2}$
		b)	$\xi_i(t=0)$, $p_{\xi i}(t=0)$	Not directly set from *md parameter file*
				By using tc-grps and *defining different partitions as groups in index files* for the different md runs.
		c)	*different number of chains of thermostats: M*	**nh-chain-length = …**
5)	*Group temperature coupling*	*a)*	*different group and*	**tc-grps**

[7] https://gitlab.com/gromacs/gromacs/-/blob/master/src/gromacs/mdlib/coupling.cpp , berendsen_tcoupl

		b)	*not thermostat some groups*	**tau-t = -1; thermostat off**
6)		*a)*	*initial velocity generation*	**vel_gen = yes** **continuation = no gen_seed** **= … ; (0 … 2^{63}) gen_temp** **= … ; in K**

2.7 Description of h_d, the heuristic functional:

Description of h_d, the heuristic functional:

The heuristic functional is a real valued expression of the $n+1$ snapshots of lead atom distance measurements taken during the simulation taken at $\Delta t = t_{sym}/n$ intervals. In particular it is a weighted average of the mean velocity and trend of the distance measurements of the two lead atoms. The weight factor (w) is a function of the final distance of lead atoms and two characteristic lengths:

$$w = min(max(d_{EOH\_T} / (d_{c2}-d_{c1}), 0), 1)$$

The mean velocity (v) is final minus initial distance divided by the time length of the simulation (250ps) expressed in nm/ps units, while trend (u) is the slope of the linear fit over the $n+1$ lead atom distance measurements versus time in nm/ps units. The heuristic h is equal

$$h_d = w\, v + (1 - w)\, u \quad .$$

In general a positive heuristic means an increasing distance of, or trend in the motion of, the ligand's lead atom respect to the binding site; in the vicinity (at the first characteristic length) it is dominated by the displacement, afar (at the second characteristic length) by motions trending further away, in between an average depending on the evolution of the unbinding process.

	symbol	value	unit
section simulation length	t_{sym}	250	ps
number of snapshots	n	10	
snapshot intervals	Δt	25	ps
weight factor	w	0 – 1 interval	
lead atom distance	d_{EOH_T}		nm
characteristic length 1	d_{c1}		nm
characteristic length 2	d_{c2}		nm
mean velocity	v		nm/ps
trend	u		nm/ps
heuristic functional	h		nm/ps

2.8 Description of angular orientation of the path:

2.8.1 The idealized case:

Let's examine the idealized case of a regular octagonal exit region from the protein. We imagine the protein as an almost entirely close globular shape with a tubular channel filled partially with water molecules that connects the region of the target lead atom to the outside, the small molecule travels in this tunnel from the target during unbinding. We model the opening of the tunnel as a regular octagon on the surface of the globular shape. We take a reference point (apex) inside the protein in the normal direction of the octagon (base). This direction is also parallel to the entry zone of the tunnel. The tunnel could deviate versus the target lead atom further inside form its entry zone. The base octagon is a window or entry area, its center is far enough to the tunnel walls to make the lead atom not interact with the perimeter.

Thus the apex and the octagonal base forms a regular octagonal cone. For this geometry, the eight $\square_{EOH,i} = x_{EOH} - apex - h_i$ angles have zero standard deviation in case the EOH lead atom is on the symmetry axis of the octagonal cone. For any off axis position, x_{EOH} , the standard deviation is positive - increasing as the distance from the symmetry axis increases. Ergo the minimum value of the standard deviation of the $\square_{EOH,i}$ angles is a one-to one indication that the lead atom is on the symmetry axis.

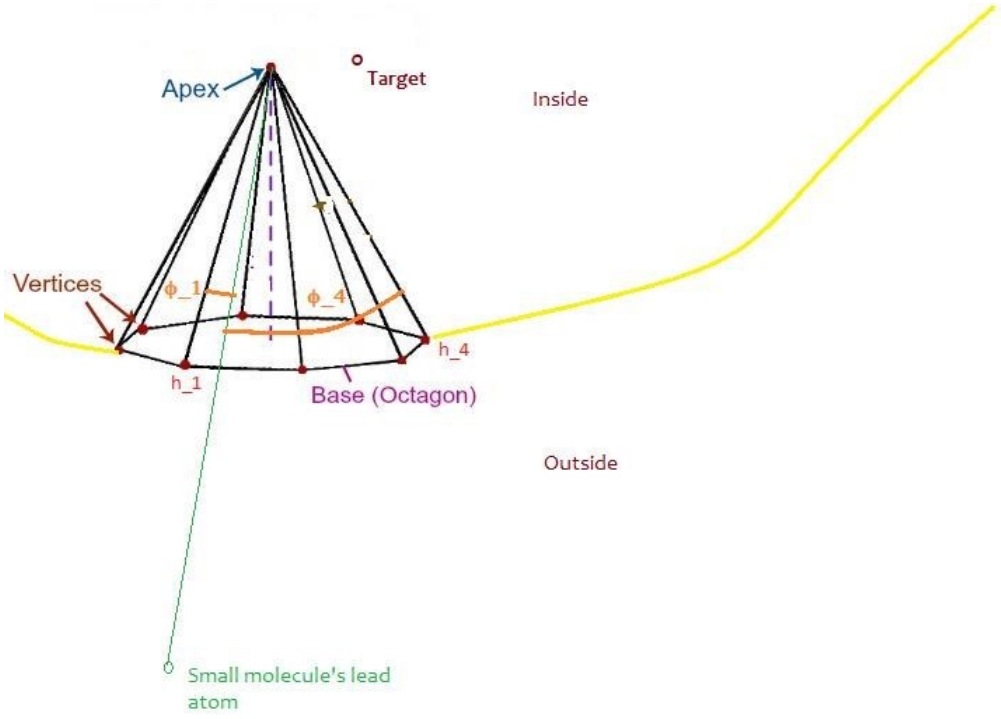


Figure 8. Schematic drawing of the angular orientation

Schematic drawing of the angular orientation

We extend the *minimal condition* to an irregular octagon cone shape: *the minimum standard deviation of vertex-apex-lead atom angles* is an indication that lead atom is in the near *vicinity of the approximate symmetry axis of the cone*.

In other words the path of the lead atom that travels through the central region of the base octagon in its normal direction can be characterized by low value of the *vertex-apex-lead atom* angles' standard deviation.

2.8.2 In particular in case of the 3b7a protein

In particular in case of the 3b7a protein the examination of the unbinding path provided evidence on the contour of the surface area where the small molecule left the protein. Namely the ranking of the nearest residues during the exit phase indicated the semi – perimeter as composed of some atoms on the 76,9,1,12,13 residues.

RESIDUE Number	#First nearest	#Second nearest	#Third nearest
76	14	7	9
9	10	11	10
1	8	9	7
12	6	6	5
13	6	4	2
5	5	4	9
6	4	7	2
8	4	3	5
54	3	4	1
75	3	1	3
15	2	4	0
10	2	3	0
11	2	0	0
55	1	5	3
51	1	0	2
19	1	0	2
77	0	2	2
73	0	2	1

An irregular octagonal regions' vertices were individuated choosing from each of the best ranking residues by cross referencing the nearest atom's list and by visual inspection in the VMS software: those few that "stick out" and not "shadowed" by others. One of the vertices on residue 5 was added to close the base perimeter of the hexagon.

Vertices			
h_1	HB1	SER	9
h_2	HA	PHE	6

h_3	HE1	MET	1
h_4	O	SER	9
h_5	HD1	ILE	13
h_6	HA	ALA	55
h_7	3HD2	LEU	76
h_8	HA	LEU	76

The chosen surface of the irregular octagon region is the window where the small molecule passes during unbinding to exit the protein. While the eight atoms do not lie perfectly on a plain (base) nevertheless in orthogonal projection it was possible to pin point a reference atom (*apex)* in the direction of the approximate symmetry axis (normal of the base surface), further on the other side of the protein. [8]

apex	C	GLY	34	

Definition:

$\square_{\square}$:= std.dev ($\square_{EOH,i}$) where $\square_{EOH,i}$ are the $x_{EOH} - a - h_i$ angles in *radians*

is the "angular orientation" factor versus the direction of entry into the protein. From which we construct the angular component of the heuristic functional using a functional form similar to the directional heuristic employed for the unbinding simulations, as follows:

2.9 Description of $h_{\square}$, the angular orientation heuristic functional:

The heuristic functional is a real valued expression of the *n+1* snapshots of lead atom angular orientation measurements taken during the simulation taken at $\square t = t_{sym} / n$ intervals. In particular it is a weighted average of the mean velocity and trend of the angular orientation measurements of the ligand's lead atom respect to the hexagon cone. In its formula, the weight factor ($w_{\square}$) is a function of the final angular orientation of the lead atom and two characteristic angular deviations :

$$w_{\square} = min(max(\square_{\square\ EOH\_T} / (\square_{c2} - \square_{c1}), 0), 1)$$

The mean velocity of angular orientation ($\square_v$) is final minus initial $\square_{\square}$ divided by the time length of the simulation expressed in rad/ps units. Final lead atom distance (d_{EOH_T}) times $\square_v$ is traversal directional velocity of centering the desired orientation in nm/ps units. Trend ($\square_u$) is the slope of the linear fit over the *n+1* lead atom angular orientation versus time in rad /ps units. The core of the heuristic $h_{\square}$ is equal

[8] This step could be automatized by minimizing the unit mass moment of inertia of the eight points, and measuring along the direction 2-3 times the diameter of the octagon.

$[\ w_\theta\ \omega_v + (1 - w_\theta)\ \omega_u\ ]$.

In general a positive heuristic means a deviation in angular orientation from the path of possible entry/exit via the window, or trend; in the vicinity (at the first characteristic length) it is dominated by the angular displacement, afar (at the second characteristic length) by angular motions trending further away, in between an average depending on the evolution of the binding process.

Definition:

$$h_\theta = \min(d_{EOH\_T} - d_\theta , 0)\ \ [\ w_\theta\ \omega_v + (1 - w_\theta)\ \omega_u\ ]$$

the angular part of the heuristic functional.

Definition:

$$h = h_d + \Sigma\ h_\theta$$

the binding heuristic functional.

	symbol	tipical value	unit
section simulation length	t_{sym}	250	ps
number of snapshots	n	10	
snapshot intervals	Δt	25	ps
angular orientation factor	σ_θ	std.dev of x_{EOH} – apex – h_i angles	rad
weight factor	w_θ	0 – 1 interval	
lead atom distance	d_{EOH_T}	d(EOH-O(t_f), THR-57-H(t_f),)	nm
characteristic length 1	ω_{c1}	0.8	nm
characteristic length 2	ω_{c2}	0.2	nm
mean velocity of angular orientation	ω_v	$\Delta\theta_\theta / \Delta t$	nm/ps
trend of angular orientation	ω_u	slope of $\theta_\theta(t_k)$ *vs. time*	nm/ps
angular part of heuristic	h_θ	*min(* $d_{EOH,\text{THR-57-H}}$ - d_θ *, 0)* . *[* $w_\theta\ \omega_v + (1 - w_\theta)\ \omega_u$ *]*	nm/ps

distance part of heuristic	h_d	see table x	nm/ps
heuristic functional	h	$h = h_d +$ □ h□	nm/ps
angular orientation cut off	d□	0.5	nm
angular importance factor	□	2	

2.10 Variable partition thermostat scheme

Let's define solvent, ligand and protein groups. Let N_{sol} be the number of solvent molecules. Further divide the solvent into k_v groups, each with approximately N_{sol}/ k_v molecules, name them SOL_1, SOL_2,… SOL_k_v . These group of molecules will be part of separate thermostats, for example Nosé-Hoover ones, with identical target temperature as specified for the npt ensemble.

2.10.1 Definition of partition into groups

Let's define solvent, ligand and protein groups. Let N_{sol} be the number of solvent molecules. Further divide the solvent into k_v groups, each with approximately N_{sol}/ k_v molecules, name them *SOL_1, SOL_2,… SOL_k_v* . These group of molecules will be part of separate thermostats, for example Nosé-Hoover ones, with identical target temperature as specified for the npt ensemble.

2.10.1.1 Definition of partition into groups

Let $a_1, a_2,..., a_n$ be n objects. Let $g_1, g_2, ..., g_k$ be k (with $k < n$) groups to which assign the n objects. n_1 objects can be assigned to group g_1 ,n_2 objects can be assigned to group g_2 and so on. $n_1, n_2..., n_k$ are such that:

$$n = n_1 + n_2 +...+ n_k$$

A **partition** of $a_1, a_2,..., a_n$ into the k groups $g_1, g_2, ..., g_k$ is one of the possible ways to assign the n objects to the k groups.

Let denote $P_{n1,n2,...,nk}$ the **number of possible partitions** into the k groups (where group i contains n_i objects). Then $P_{n1, n2,...,nk} = n! / n_1!n_2! \dots n_k!$, the multinomial coefficient.

For the dynamics, the groups' index sequence is irrelevant respect to its permutations in the differential equation system: the number of physically different configurations, *c,* are*:* $P_{n1, n2,...,nk} /k!$.

In our case - for $N_{sol} = n = 6000$, $k_v = k = 5$, $n_i = 1200$, i.e. forming five equal cardinality subgroups from the 6000 solvent molecules - the number of different configurations are $c = 2.683\ 10^{20065} / (5 * 3.316\ 10^{5735} * 120) = 1.348 * 10^{14327}$.

Estimate using the Stirling-formula [Pearson1924] for the logarithm of a factorial:

$$\ln n! \approx n \ln n - n$$

$\ln c = n \ln n - n - n/5 \ln (n/5) - n/5 - 4.787 = 46197 - 7308 - 6.396 = 38882 = \log(16887)$

$$c \approx 10^{16867}$$

An astronomically high number, for each of which there is - theoretically - a different deterministic solution of the dynamics starting from same set of initial conditions (considering the non-extended phase space variables). The initial values of the extended variables, those that describe the different thermostats' internal states are determined by each groups own energy E_i (temperature T_i) at t_0. A group molecules evolves by being coupled only to its specific thermostat by the modified newton equations of the relevant model, equally interacting with all other parts of the system by the usual intermolecular forces.

A consequence of the astronomically high number of configurations is that for large enough n we can considering a continuous spectrum of the energy levels of the thermostats around the mean energy of the system.

At the initial moment, for a given $\varepsilon > 0$ number, fix one arbitrary molecule, then there exists $n_{epsilon}$ for which there will be always a molecule among those in one of the other thermostats with absolute difference of its kinetic energy less then epsilon respect to the fixed one. By swapping the attribution of the two molecules between the two thermostats involved, their (the thermostats) respective initial energy levels will change by less the epsilon. In consequences the thermostat's state variables will differ only by a bounded function of epsilon, while the other state variables initial values will be identical. So there going to be a t_ε time for which the maximum difference of the solutions will be less then epsilon according to the theorem of the differential equation system solution's continuous and differentiable dependence on initial conditions and parameters. [Arnold1973 Chapter 9.4]. Specifically because of the different set of differential equation for the two systems partitioned differently the solutions will be different for non-zero energy difference between partitions.

The different initial distribution of E_i among thermostats according to the partition – will determine the difference in the dynamics in a smooth (continuous and differentiable) way.

2.11 Implementation of the Variable partition thermostat scheme for

MD in GROMACS

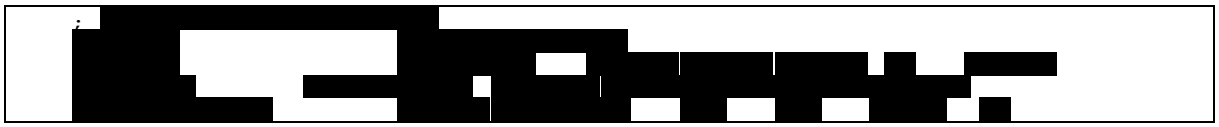
Technically, in GROMACS, generate for a given MD run an index.ndx file by insert at the end k_v section definitions containing the atom numbers of the molecules from a randomly chosen partition.

Example of a group definition section of ndx file (here the ion is added to the first group):

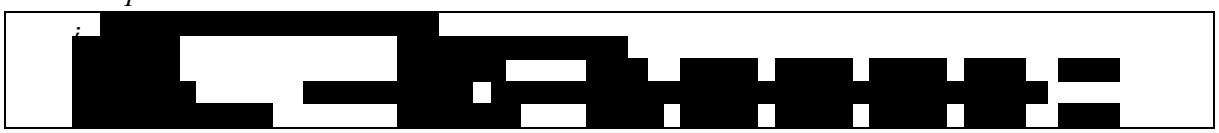
[SOL_1]
1966 1979 1980 1981 1991 1992 1993 2045 2046 2047 2084 2085 2086 2090 2091
2092 2096 2097 2098 2099 2100 2101 2111 2112 2113 2123 2124 2125 2168 2169
...
22432 22466 22467 22468 22469 22470 22471 22478 22479 22480 22490 22491 22492 22505 22506
22507 22520 22521 22522
...
[SOL_5]
1967 1968 1969 1973 1974 1975 1976 1977 1978 2006 2007 2008 2024 2025 2026
2042 2043 2044 2054 2055 2056 2060 2061 2062 2069 2070 2071 2072 2073 2074 ...
22427 22428 22429 22439 22440 22441 22451 22452 22453 22457 22458 22459 22472 22473 22474

22508 22509 22510

Specify in the md parameters file the temperature coupling *section as:*



Example:



Example of molecular dynamics parameter file used

2.12 Variable partition thermostat scheme MD*

To generate the different alternative paths needed in MD* we employ the Variable partition thermostat scheme for Nosé-Hoover or Berstened during the simulation of binding and unbinding processes. In particular at a given non initial section of the MD* simulation new partitions for the thermostat groups are generated for the alternative branches, notwithstanding keeping for two branches the partitions, for the best and the second best ranked ones.

In the MD* "5+2" scheme:

b, branch number of section s	*branch initial condition is final configuration of*	*branch re-partition*
1	*best ranked branch among s-1*	*no: keep best ranked of s-1*
2	*best ranked branch among s-1*	*yes*
3	*best ranked branch among s-1*	*yes*
4	*best ranked branch among s-1*	*yes*
5	*best ranked branch among s-1*	*yes*
6	*2^{nd} best ranked branch among s-1*	*no: keep 2^{nd} best ranked of s-1*

7	*2nd best ranked branch among s-1*	*yes*

For each branch of the initial section, solvent molecules are re-partitioned between the thermostats because the npt MD had one common thermostat.

2.13 Implementation of the Variable partition thermostat scheme for MD* in GROMACS

For MD*, in GROMACS, for each sections' branches MD run generate a new index.ndx file - if needed according the description above - by inserting at its end k_v section definitions containing the atom numbers of the molecules from a randomly chosen partition. The ion is added to the last group. This was done by the solvent permutation function in the pygro_utils module, and the aforementioned index file was saved with a name identifying the run in question in common /ndx subdirectory. In cases in which re-partitioning was not commanded the respective old index file was copied with the proper new name.

The proper index file is used for the gromacs precompile command argument:

After – in a parallel or serial manner – all branch MD's of a given section were computed, the evaluation phase ranks the alternatives according the heuristic function of the problem for the continuation.

3 Results and Discussion

.

3.1 MD* simulation binding/unbinding process of the LUSH protein/Ethanol co-crystal

The unbinding process of the Ethanol small molecule form the LUSH protein was simulated with GROMACS using the MD* algorithm with the following models: Initial velocity generation, Variable thermostat partition Nosé-Hoover, Andersen thermostat, Variable thermostat partition, Berendsen scheme.

MD* underlying model type	*way of implementation*	*ref in chapter 2.4 by*	*success*	*date*	*simulation id*
Variable thermostat partition Nosé-Hoover	*ligand unbinding*	*4.b), 5.a)*	*yes*	*2022-01-08*	*3b7a_3_unbind_VP_NH*[9]
Andersen thermostat	*ligand unbinding*	*3.c)*	*yes*	*2021-12-04*	3b7a_3_unbind_andersen[13]
Variable thermostat partition Berendsen scheme	*ligand unbinding*	*1.a), 5.a)*	*yes*	*2022-01-09*	*3b7a_3_unbind_VP_BR*

[9] simulation directory: /home/trezzaa/trezzaa/tesi/adam/3b7a/3b7a_3/3b7a_3_unbind_VP_NH [13] simulation directory: /home/trezzaa/trezzaa/tesi/adam/3b7a/3b7a_3/3b7a_3_rep1/1

Variable thermostat partition Velocity-rescaling weakcoupling scheme	*ligand unbinding*	*2.a), 5.a)*	*yes*	*2022-01-10*	*3b7a_3_unbind_VP_VS*[10]
Initial velocity generation	*ligand unbinding*	*6.a)*	*yes*	*2021-11-27*	*3b7a_3_unbind_VG*[11]

Summary of MD* results for unbinding Table

3.1.1 Brief description of the MD* algorithm:

The MD* the simulations were divided in successive sections of t_{sym} = 250 ps duration each with the relevant temperature and pressure coupling in the table 3.1. The starting point of all simulations were a common initial configuration: the energy minimized, temperature and pressure equilibrated *(T=303.5 +- .5 K, p = 0.8 +-3.8 K)* molecular configuration (npt_0) obtained with iterative MD simulation cycles using Chemistry at HARvard Molecular Mechanics [CHARMM] force field, on atomic coordinate files form the Protein Data Bank [PDB] solvated in 60x60x60 A rectangular box. Each successive section of the simulation consists of $n_b = 7$ parallel run MD simulations, branches of the given section. The simulated LUSH-EOH molecular system at the *b*-th branch of the *s*-th segment travels from α_{sb} initial configuration via $\Pi_{\alpha\beta}$ path to the ω_{sb} final configuration in the phase space. The n_b branches were ranked by a heuristic functional: $h(\Pi_{\alpha\beta})$, the ranking determined which one's to be continued. The initial condition of the MD simulations of the next section's branches are determined as follows $\alpha_{s+1,b}$: for *5* of the new branches take the maximal ranked final configuration ω_s of the previous section. For the remaining two the second best's ω_s, . (Chapter 2.2)

3.1.2 Discussion of the utility of the heuristic functional parameter choices from the results and the type of underlying MD* model:

The heuristic functional h_d (Chapter 2.5) with parameters t_{sym} *= 250 ps , n =10 ,* Δt *= 25ps* d_{c1} *= 0.3 nm* d_{c2} *= 0.02 nm* was evaluated over the *11* snapshots of lead atom lead atom (EOH O, THR 57 H) distance measurements taken during the simulation at *25 ps* intervals. In particular the weighted average of the mean velocity and trend of the distance measurements was calculated. The weight factor (*w*) as a function of the final distance of lead atoms and the two characteristic lengths. The mean velocity (*v*) is final minus initial distance divided by the time length of the simulation (250ps) expressed in *nm/ps* units, while trend (*u*) is the slope of the linear fit over the 11 lead atom distance measurements versus time in *nm/ps* units. The heuristic *h* was equal weighted average of the mean velocity and trend over the characteristic distances. In visual analysis of the unbinding path with pyMOL software, a positive heuristic indicated an increasing distance of, or trend in the motion of, the ligand's lead atom respect to the binding site; in the vicinity (d<*0.07 nm*) it was dominated by displacement, afar (d>*0.13 nm*) by motions trending further away from the protein, in between a motion that can be characterized by a diffusion in channel connecting the biding site to the

[10] simulation directory: /home/trezzaa/trezzaa/tesi/adam/3b7a/3b7a_3/3b7a_3_unbind_VP_VS

[11] simulation directory: /home/trezzaa/trezzaa/tesi/adam/3b7a/3b7a_3/3b7a_3

surrounding solvent. This confirmed the parameter choice of d_{c1}, d_{c2} characteristic lengths based on the protein and the solvated box's volume.

The choice of parameters captured the intended direction and trend motion of the ligand in the resultant unbinding path.

The same h_d parameters were used with 5 different MD* underlying model type for the thermo- and barostats, the simulated velocity of the unbinding process was confronted - albite with a poor statistics: The slowest by far was the Anderson collision thermostat (by factor of 3), using already the lowest thermal coupling characteristic time suggested in the GROMACS documentation. Among the other 4 underlying models the *Initial velocity generation* result indicated the highest velocity of the unbinding event. This confirms our suspicion that this model by introducing concentrated velocity shocks yields unnatural paths, as it distorts, interferes with the proteins natural coordinated movements that helps the small molecule stay in the water filled pocket.

3.1.3 Characterization of the resultant path of the simulations:

3.1.3.1 Description of lead atom path during unbinding

I expose the characteristic results obtained in all unbinding simulated in terms of lead atomic distance and vicinity during the unbinding process. The first one is a fast confirmation of the event of unbinding, the second one is important, yielding information to construct the angular heuristic functional for the binding simulation, by the path vicinity reference atom assignment preclude described in Chapter 2.7.2. The structural integrity of the protein during simulation was studied and documented in Appendix IV.

a) Lead atom distances:

Fix the *O'* origin of a co-moving spherical coordinate system (r,θ,φ) to one of the lead atoms (EOH O) and plot the *r* distance of the other lead atom (THR 57 H) during the unbinding process.

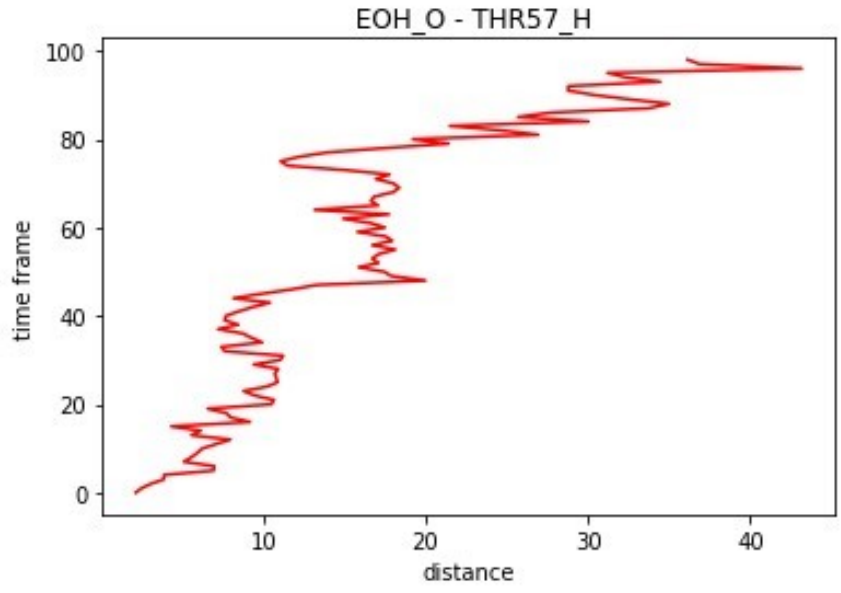


Figure 9. Lead atom distance in Å vs simulation time frame number: t = 0.250 * n_f in ns

From the initial 2.14 Å, distance between them gradually increases to cc 10 Å, at around 12 ns to 20 Å after which a clear departing phase brings it to a distance of 40 Å from 20 to 25 ns outside the protein. Ignoring any consideration on the direction in which the small molecule went, as a rough estimate we consider that during the simulation the distance from THR 57 H of any protein atom was more than 29.78 Å (average 27.32 Å , std.dev. 0.90 Å), complete unbinding event occurred.

The heuristic functional values during the simulation paths' segments showed, at the beginning of unbinding, a strong correlation with the mean velocity of departure, near the terminal condition with periodic trends of increasing distance.

In particular, from the initial configuration of npt_0 the path was discovered by the MD* as the follows sequence of MD simulations (with simulation end point lead atom distance and the heuristic's value):

	d [nm] , h [nm/ps]
md_0_6 starting from npt_0	0.592, 0.0015
md_1_0 starting from md_0_6	0.770, 0.0008
md_2_5 starting from md_1_0	1.069, 0.0011
md_3_6 starting from md_2_5	0.875, -0.0008
md_4_5 starting from md_3_6	0.988, 0.0007
md_5_3 starting from md_4_5	1.712, 0.0022
md_6_6 starting from md_5_3	1.772, 0.0
md_7_5 starting from md_6_6	1.776, 0.0002
md_8_6 starting from md_7_5	2.693, 0.0044
md_9_2 starting from md_8_6	3.037, 0.0036
md_10_5 starting from md_9_2	4.267, 0.006

b) Characterization of the lead atom's vicinity

Examine which atoms of the protein has the lowest distance *(r)* respect to the lead atom via the unbinding path as follows:

1) The nearest atom of the protein respect to the EOH O lead atom during the unbinding

for each frame

$$a' = \mathrm{argmin}_a \; d_{EOH\_O,a}$$

where a is member of the protein

2) The list of the 3 nearest atoms of the protein to the EOH O lead atom during the unbinding

for each frame

$$a',a'',a''' = \mathrm{argminsort}_a \; d_{EOH\_O,a}$$

where a is member of the protein

3) The list of the nearest 3 different residues to the EOH O lead atom during the unbinding

for each frame

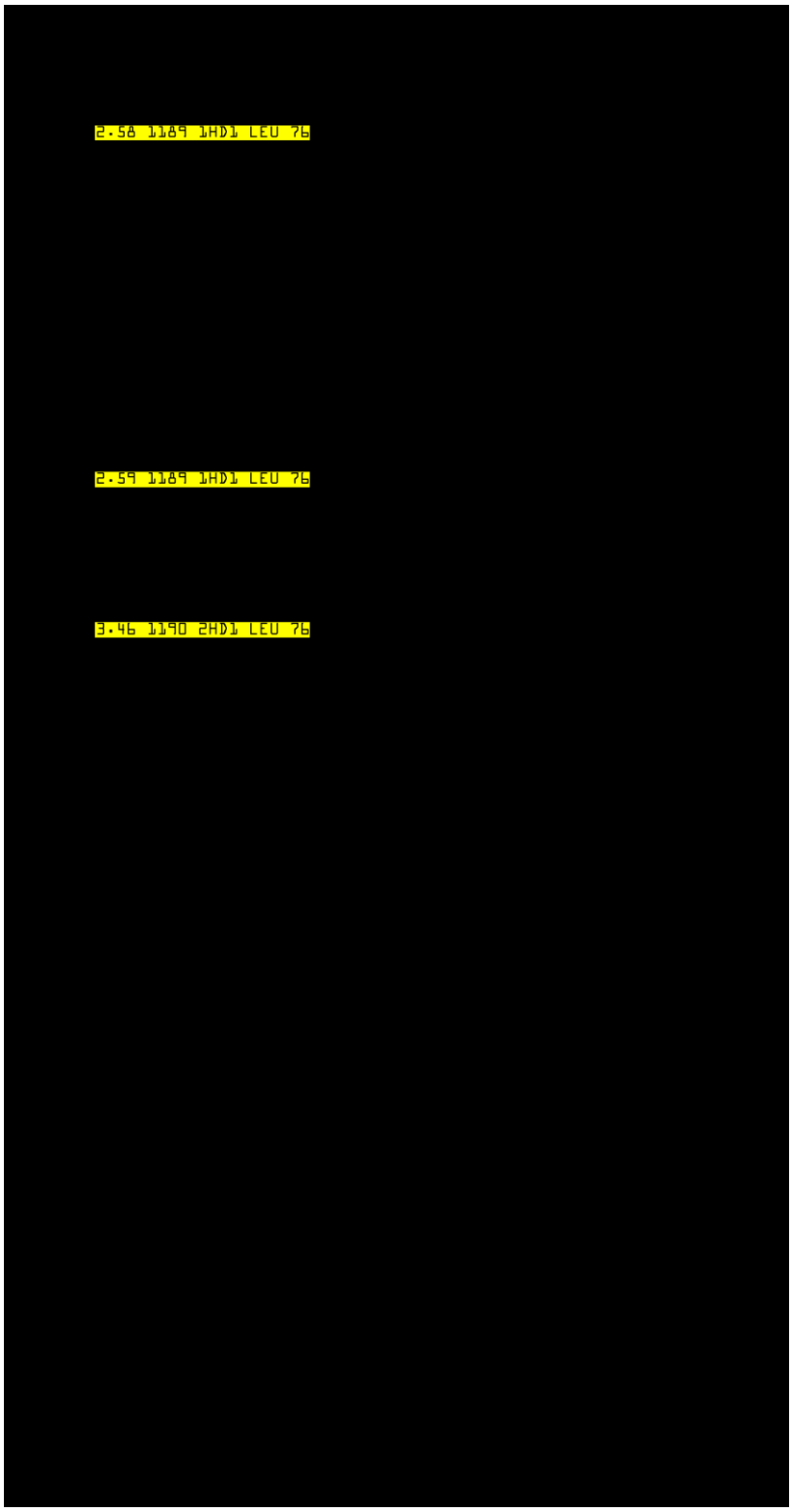
$$a',a'',a''' = \mathrm{argminsort}_a \; d_{EOH\_O,a}$$

where a is member of the protein and a',a'',a''' are part of different residues

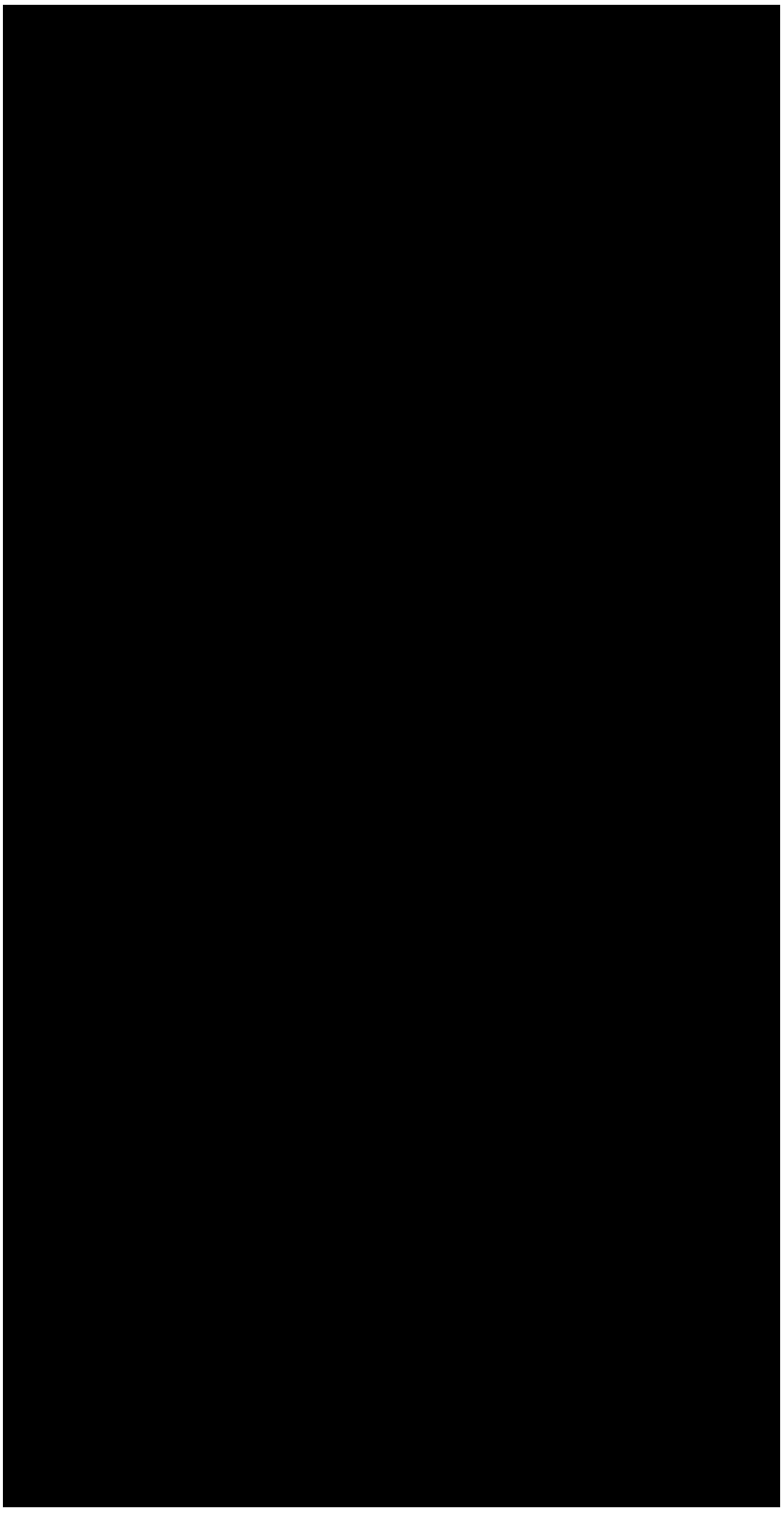
Implementation code path_near_list.py

In the following tables the first column contains the frame number n_f from which simulation time is obtained as $t = 250\ n_f\ ps$; second column, the distance of the nearest protein atom; column 3-6 , the nearest atom's description. The following columns analogously describe the second and third nearest to the protein:

1) and 2) List of the nearest atoms along the path of the s. molecule



3). List of the 3 nearest residues along the path of the s. molecule:



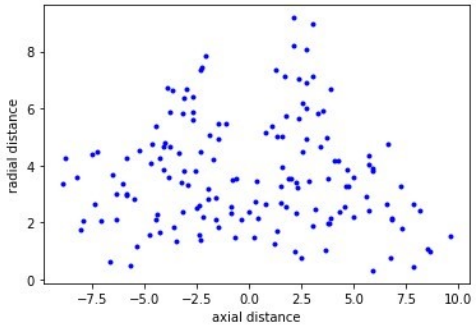
All MD* simulations performed with different underlying model type fund similar unbinding paths. That is, the top ranking of the nearest residues during unbinding were among the 76,9,1,12,13 residues in all. Due to the limited number of simulations this do not exclude categorically that there are other pathways, and/or could be an artifact of the radial heuristic function which could oversample relatively straight paths in the time scale studied. It is a strong indication that the feasibility of binding via the same regions of the protein connecting the internal binding site to the outside solvent discovered in unbinding is much more promising than elsewhere. The method utilized to translate the

information obtained from the unbinding path into mathematical language for the MD* algorithm is the construction of angular orientation heuristic functional based on the reference atoms forming an (octagonal) cone around entry zone (Chapter 2.7). The paths will be ranked and continued respect to their affinity to the symmetry axis of the cone defined.

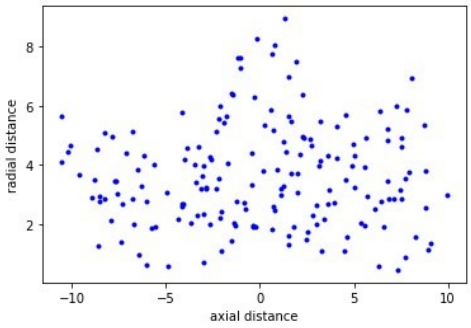
3.1.4 Characterization of the protein's movement respect to its parts:

Using the method of reference point and directions defined on the protein (Chapter 2.1) the radial and axial distribution of the atoms constituting the four groups were calculated. The results are as follows:

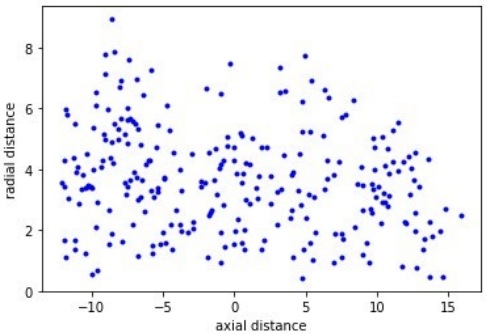
$\underline{x}_{1,i}^{ref}$ in RG1: residues 106-115



$\underline{x}_{2,i}^{ref}$ *in RG2: residues 44-54*



$\underline{x}_{3,i}^{ref}$ *in RG3: residues 82-97*



$\underline{x}_{4,i}^{ref}$ *in RG4: residues 64-76*

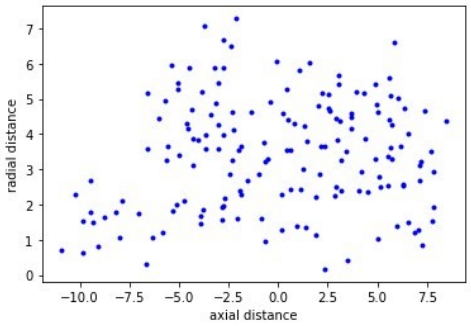


Figure 10. Radial and axial distribution of the atoms constituting the four reference groups

Residues 64-76 (the 4rd ref group) minimal distance respect to the 3th ref group (82-97) is 4A due to the „U" shape with a turn at residues 78-80.

$\underline{x}_{4,iref}$ *in RG3*

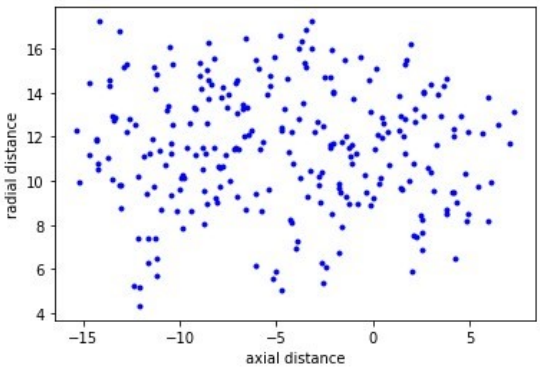


Figure 11. Radial and axial distribution of the atoms constituting the 4th reference group respect to RG3

The average of radial distribution of cross distances was calculated and gives a picture of the protein's change in shape during the simulation. It describes the movement of its helices respect to one another. In this case it is a *4x4* matrix, *its j,k* element is the mean radial distance of the atoms in the *j-th* ref groups respect to the *k*-th ref point/direction :

2.1477	14.0868	12.8671	14.7839
16.7412	3.4620	18.5998	10.2632
11.2345	19.7923	2.2441	12.4127
19.4417	11.8785	12.8018	3.5820

3.1.5 The lead atom's movements respect to the protein

During unbinding the 3b7a EOH molecular system from its initial configuration arrives to the final configuration via the path. I transform the $\underline{x}_{EOH}(t)$, $\underline{c}_{EOH}(t)$ path into the $d^{ax}{}_{EOH,k}(t)$, $d^{r}{}_{EOH,k}(t)$, $\theta_{EOH,k}(t)$ coordinates to characterize its movement respect to the stable structural parts of the protein.

The binding site lead atom H is on the 57th residue THR part of reference group 4, its movement is on graph 31.5 coloring from white to dark red is first to last frame. It remains in a cc 2A diameter circular area during unbinding.

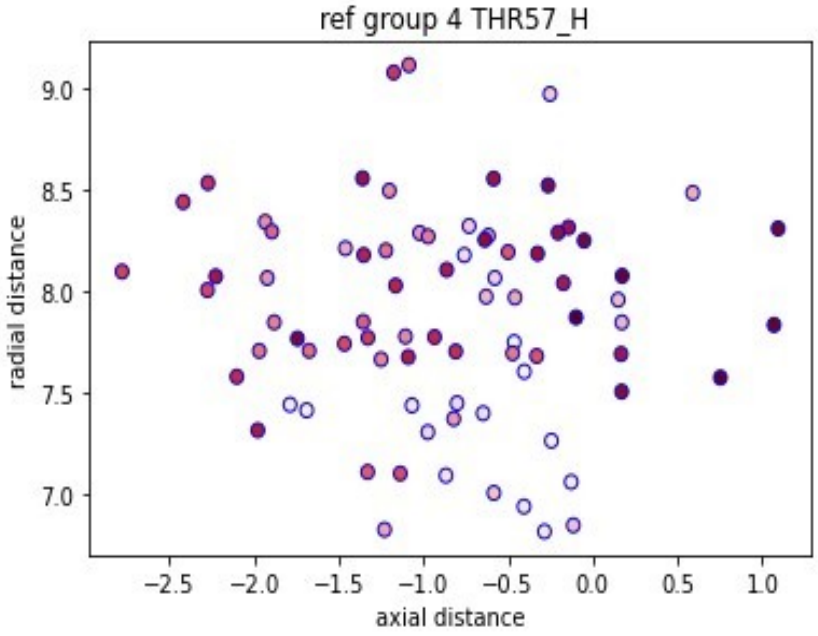


Figure 12. Movement of the binding site lead H atom on the 57th residue THR during unbinding

The EOH O lead atom moves versus the direction of the lower residue number of the protein along the RG3, it distances itself in two cc 5 A steps.

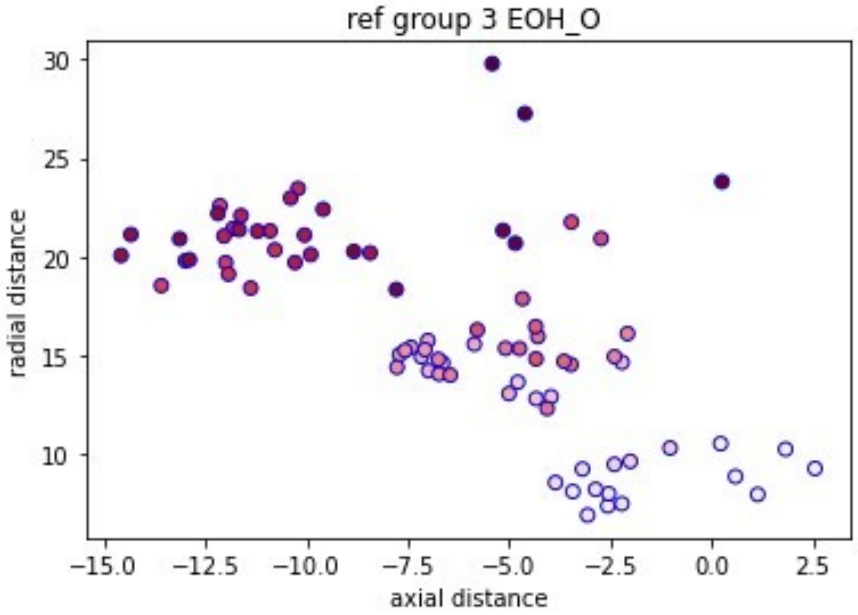


Figure 13. Movement of the lead ligand atom during unbinding

3.1.6 Verification of the structural stability of the protein during unbinding

I studied the cross movement of the various reference groups during unbinding, results is illustrated on the matrix plot of

$\underline{O_j}^{ref}(t)$ points $d^{ax}{}_k(t)$, $d^{r}{}_k(t)$ coordinates versus time – where

j column: the *j*-th reference point, *k* row: RG*k*, the coordinate systems fixed to *k*-th reference group;

plots: axial vs radial distances in Å; time form light to dark color, table: standard deviations in Å.

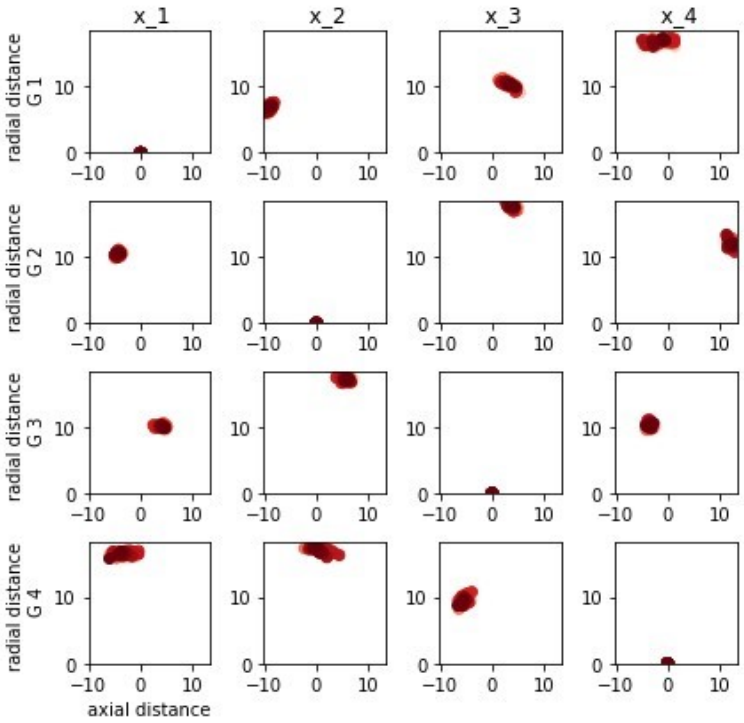


Figure 14. Structural stability of the protein during unbinding

RG3 and RG4 displayed highly correlated motion as they form a U shape' two limbs which are structurally connected. RG1 and RG2 movement was also coordinated. RG4 moved against RG1 and RG2 cc. 5 Å in the axial direction.

[redacted]

Table: standard deviation of $\underline{O_j}^{ref}(t)$ points

in $d^{ax}{}_k(t)$, $d^{rad}{}_k(t)$ coordinates during unbinding
Implementation code: path_graphs.py

3.1 MD* simulation binding process of the LUSH protein/Ethanol co-crystal

The binding process of the Ethanol small molecule form the LUSH protein was simulated with GROMACS using the MD* algorithm with the following models: Initial velocity generation, Variable thermostat partition Nosé-Hoover, Variable thermostat partition, Berendsen scheme. The heuristic functional h_d with parameters $t_{sym} = 250\ ps$, $n = 10$, $\Delta t = 25ps$ $d_{c1} = 0.3\ nm$ $d_{c2} = 0.02\ nm$, the angular heuristic $h_\Box$ with parameters $\Box_{c1}$ =0.8 nm, $\Box_{c2}$ =0.2 nm, $d_\Box$ = 0.5 and $\Box = 2$ (Chapters 2.6 ,and 2.8).

MD* underlying model type	*way*	*ref in 2.5) implementation - chapter 2.4 by*	*success*	*date*	*simulation id*
Variable thermostat partition NoséHoover	*ligand binding*	*4.b), 5.a) and 5.b)*	*no*[12]	*2022-01-05*	*3b7a_3_bind_VP_NH*[13]
Variable thermostat partition NoséHoover	*ligand binding*	*4.b), 5.a)*	*yes*	*2022-01-06*	3b7a_3_bind_VP_NH_2[14]
Andersen thermostat	*ligand binding*	*3.c)*	*no*[15]	*2021-12-28*	*3b7a_3_bind_andersen*[16]
Variable thermostat partition Berendsen scheme	*ligand binding*	*1.a), 5.a)*	*yes*	*2022-01-07*	*3b7a_3_bind_VP_BR*[17]
Initial velocity generation	*ligand binding*	*6.a) without specifying* **`gen_temp`**	*yes*	*2021-12-23*	*3b7a_3_bind_VG*[18]
Initial velocity generation	*ligand binding*	*6.a)*	*yes*	*2021-12-31*	*3b7a_3_bind_VG_2*[19]

3.1.1 Brief description of the MD* binding algorithm:

The MD* algorithm for the binding simulations differs in 1) initial condition 2) ranking order 3) heuristic functional used form the one described for the unbinding (Chapter 3.1.2)

[12] small molecule stopped its movement inside the protein as the protein group was without thermostat, t_tau = -1

[13] simulation directory: /home/trezzaa/trezzaa/tesi/adam/3b7a/3b7a_3/3b7a_3_bind_VP_NH

[14] simulation directory: /home/trezzaa/trezzaa/tesi/adam/3b7a/3b7a_3/3b7a_3_bind_VP_NH_2

[15] alternative paths generated by different seeds for the Andersen thermostat were very close to each other almost indistinguishable, only rarely differed from one another

[16] simulation directory: /home/trezzaa/trezzaa/tesi/adam/3b7a/3b7a_3/3b7a_3_back_andersen

[17] simulation directory: /home/trezzaa/trezzaa/tesi/adam/3b7a/3b7a_3/3b7a_3_bind_VP_BR

[18] simulation directory: /home/trezzaa/trezzaa/tesi/adam/3b7a/3b7a_3/3b7a_3_back

[19] simulation directory: /home/trezzaa/trezzaa/tesi/adam/3b7a/3b7a_3/3b7a_3_back2

1) The common initial condition for all underlying MD* models was a generic configuration obtained by unbinding MD* simulations with the ligand outside the protein. Temperature and pressure averages deviations were compatible with fluctuations observed in standard iterative MD *(T=303.2 +- .5 K, p = 0.9 +-2.1 K)* .
2) The ranking order was inverted, for the evaluation of the candidate paths to be evaluated reflecting the intended direction of the motion.
3) As the h_d heuristic functional by construction is sensitive only to radial directional movements an angular component was added that privileged paths in the vicinity of the previously discovered channel's entrance connecting the binding site with the outside solvent.

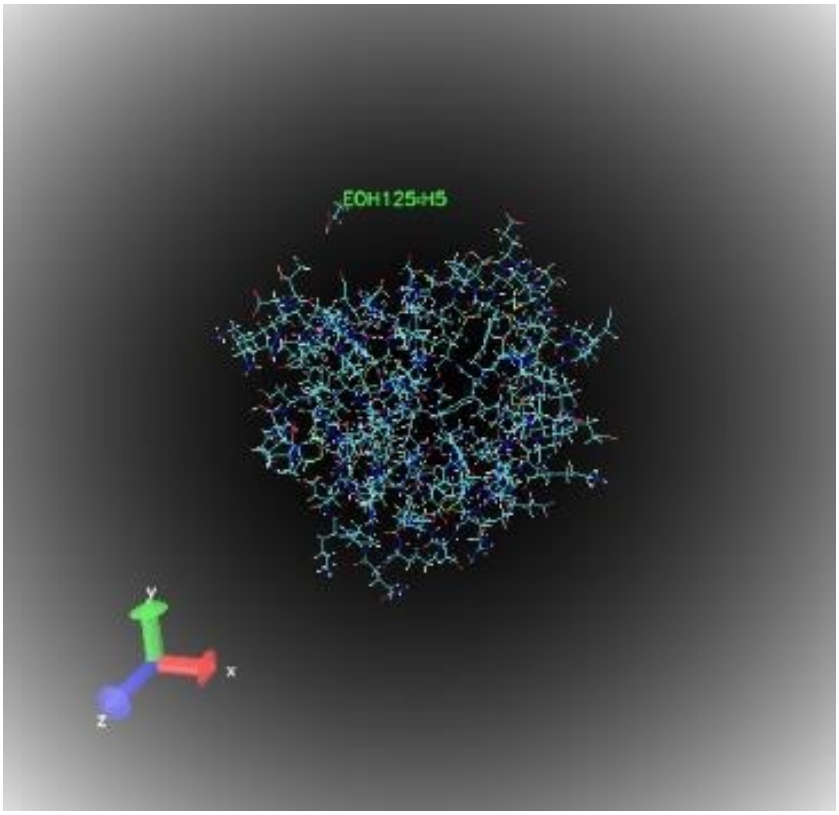


Figure 15. PyMol rendering of the initial condition of the binding simulation

3.1.2 Discussion of the utility of the heuristic functional parameter choices from the results and the type of underlying MD* model for binding:

The same parameters were used for the radial part of the heuristic functional h_d (Chapter 2.5) with (t_{sym} *= 250 ps , n =10 ,* Δt *= 25ps* d_{c1} *= 0.3 nm* d_{c2} *= 0.02 nm).* For the angular orientation the path vicinity reference atom assignment preclude described in Chapter 2.7.2. was performed yielding

to the octagonal cones bases as atoms : *HB1 SER9,HA PHE6,HE1 MET1,O SER9,HD1 ILE13,HA ALA55,3HD2 LEU76, HA LEU76, with apex: C GLY34.* The angular heuristic $h_{\square}$ with parameters were $\square_{c1}$ =0.8 nm, $\square_{c2}$ =0.2 nm, $d_{\square}$ = 0.5 $\square$ = *2* (Chapters 2.6 and 2.8). The resultant functional form *h* was evaluated over the *11* snapshots of lead atom lead atom (EOH O, THR 57 H) distance measurements taken during the simulation at *25 ps* intervals. In visual analysis of the binding path with pyMOL software, a negative heuristic indicated a directional movement or trend versus the approximate symmetry axis of the octagonal cone of reference atoms for the beginning of the simulation outside of the protein (d>*0.13 nm*) and a motion of the ligand's that can be characterized by successive diffusive movement versus the channel connecting the biding site to the surrounding solvent until its entry. In the vicinity of the binding site (d<*0.07 nm*) the path had similar characteristic than the unbinding path in this region with time symmetry.

This confirmed the parameter choice of d_{c1}, d_{c2} , $\square_{c1}$, $\square_{c2}$, $d_{\square}$ characteristic lengths based on the protein and the solvated box's volume and on the geometrical configuration of the channel connecting the binding site to the outside solvent.

The choice of parameters captured the intended directional motion versus the entry of zone and a successive approach to the lead atoms in the internal of the protein of the ligand in the resultant binding path.

The same h_d parameters were used with 5 different MD* underlying model type for the thermo- and barostats, the simulated velocity of the unbinding process was confronted - albite with a poor statistics: The Anderson collision thermostat did not provide a binding path, using already the lowest thermal coupling characteristic time suggested in the GROMACS documentation.

The Variable thermostat partition Nosé-Hoover model was performed in two version, one where only the solvents partition were coupled to the thermostat, but the protein and ligand was not, this simulation did not provide un unbinding path, it was observed that the ligand entered the protein, successively its movement slowed down and its movement stalled at cc 0.8 nm from the target, and the simulation were stopped after 3o segments. This indicates that the transmission of the thermal movement of the solvent via the protein structure via the ligand was not sufficient to provide the diffusion enough strength, a 'hot-solvent, cold-solute' situation occurred in the time scale simulated.

In the other version of Variable thermostat partition Nosé-Hoover model where all components of the system were coupled to the thermostats with the same target temperature the binding path was discovered by the MD* algorithm. Similarly to the Variable thermostat partition Berendsen scheme and the Initial velocity generation underlying models.

Among the other 3 underlying models the Variable thermostat partition Nosé-Hoover model result indicated the highest velocity of the binding event.

3.1 Overlapping of the ethanol binding pose compared with the crystal

MD simulation from the final VPNH MD* configuration small molecule pose was analyzed by trajectory analysis of the EOH molecule in 2.5 ns MD simulation from the VPNH MD* final configuration. After 750ps the maximal interatomic difference of distance of EOH atoms and the ten nearest protein atom respect to the PDB atomic coordinates decreased until 0.35 A.

4 Conclusion

The viability of MD* simulation was demonstrated by simulating the binding/unbinding process of the LUSH protein/Ethanol co-crystal. The MD* simulation revealed the route of the unbinding path at in an order of magnitude less simulation computer time than standard MD. The state of the art thermos- and barostat models were used to accomplish this result, by the Variable Partition Scheme applied for the solvent. This, respect to previous alternative path generations in suMD greatly enhance the comparison the MD generated data to be cross referenced to published results in the field and experimental data.

The viability of the simulation of the binding path during binding showed the importance the choice of the underlying model and the parametrization of the heuristic functional respect to geometry of the problem. An accurate overlapping of the ethanol binding pose compared with the crystal was found.

Further investigation respect to the general application of the MD* algorithms proposed are:

1) Study the properties of the simulation respect to the Variable Partitions Schemes, in particular the dependence on the number of thermostats on the statistical properties of the proteins movement and the small molecules path.
2) Study the cost/benefit of keeping track in the memory of more data of discarded path segments/branches combined with heuristics functional forms that can command the algorithm to continue from further back in time in case of slow convergence.
3) Estimate the statistical probabilities of the path (diffusion, angular diffusion) in run time by sampling over the alternatives and compare it similar estimates on the movement of solvent molecules to calibrate the relationship between simulated time in MD* and conventional MD.
4) Study the algorithms bi-directional version, in which the small molecule's path is continued forward and backward in time form a generic position. This can be done as soon as in Gromacs negative time step simulation will be available.

Possible further development in the specific case of the LUSH protein/Small molecule co-crystal system are proposed as:

1) Re-run the analysis on different residue 52, 54 substituted LUSH proteins
2) Extend the analysis to butanol e propanol small molecules
3) Study binding properties by appropriate heuristic functional with lead atom on residue 52, and on the geometric mean position of 57, 52 hydrogen leads, and analyze the accuracy of the binding pose.

Our work, could be open novel frontiers in computational biochemistry field, providing the molecular basis of biological system interactions.

5 References

Arnold1973 Arnold, V.I. - Ordinary differential equations, (MIT Press) Translation of e differentsial'nye Uravneniya. 1973)

Berendsen1984 Berendsen, H.J.C., Postma, J.P.M., DiNola, A., Haak, J.R. Molecular dynamics with coupling to an external bath. J. Chem. Phys. 81:3684-3690, 1984

Berendsen1991 Berendsen, H.J.C. Transport properties computed by linear response through weak coupling to a bath. In: Computer Simulations in Material Science. Meyer, M., Pontikis, V. eds. Kluwer 1991, 139-155

Bissaro2021 Maicol Bissaro and Giovanni Bolcato and Matteo Pavan and Davide Bassani and Mattia Sturlese and Stefano Moro (2021) Inspecting the Mechanism of Fragment Hits Binding on SARS-CoV-2 Mpro by Using Supervised Molecular Dynamics (SuMD) Simulations ChemMedChem 16, doi: 10.1002/cmdc.202100156

CHARMM2021 Chemistry at HARvard Molecular Mechanics, https://charmmgui.org/doc.html

Ciancetta2016 Antonella Ciancetta and Alberto Cuzzolin and Giuseppe Deganutti and Mattia Sturlese and Veronica Salmaso and Andrea Cristiani and Davide Sabbadin and Stefano Moro (2016), New Trends in Inspecting GPCR-ligand Recognition Process: the Contribution of the Molecular Modeling Section (MMS) at the University of Padova, Molecular Informatics, doi :10.1002/minf.201501011

Deganutti2020 "Deganutti G. Stefano Moro S, Reynolds C A. A Supervised Molecular Dynamics Approach to

Unbiased Ligand–Protein Unbinding J. Chem. Inf. Model. 2020, 60, 3, 1804–1817"

Deganutti2021 Giuseppe Deganutti and Filippo Prischi and Christopher A. Reynolds (2021) Supervised molecular dynamics for exploring the druggability of the SARS-CoV-2 spike protein, Journal of Computer-Aided Molecular Design 35, doi:10.1007/s10822-020-00356-4

GROMACS2019 GROningen MAchine for Chemical Simulations, https://www.gromacs.org/documentation2019

Hart1972 Hart, Peter E.; Nilsson, Nils J.; Raphael, Bertram (1972). Correction to 'A Formal Basis for the Heuristic Determination of Minimum Cost Paths. ACM SIGART Bulletin (37): 28–29. doi:10.1145/1056777.1056779. ISSN 0163-5719. S2CID 6386648.

Hoover1985 Hoover, W.G. Canonical dynamics: equilibrium phase-space distributions. Phys. Rev. A 31:1695-1697, 1985

Jorgensen1983 [1] Jorgensen WL, Chandrasekhar J, Madura JD, Impey RW, Klein ML (1983) Comparison of simple potential functions for simulating liquid water. J Chem Phys 79:926. doi: 10.1063/1.445869

Maier215 Maier JA, Martinez C, Kasavajhala K, Wickstrom L, Hauser KE, Simmerling C (2015) ff14SB: Improving the Accuracy of Protein Side Chain and Backbone Parameters from ff99SB. J Chem Theory Comput 11:3696–3713. doi: 10.1021/acs.jctc.5b00255

Nosé1983 Nosé, S., Klein, M.L. Constant pressure molecular dynamics for molecular systems. Mol. Phys. 50: 1055-1076, 1983

Nosé1984 Nosé, S. A molecular dynamics method for simulations in the canonical ensemble. Mol. Phys. 52:255-268, 1984

Parrinello1981 Parrinello, M., Rahman, A. Polymorphic transitions in single crystals: A new molecular dynamics method. J. Appl. Phys. 52:7182-7190, 1981 PDB Protein Data Bank

pyMOL2022 open-source model visualization software tools in structural biology, https://pymol.org/documentation2022

Russell2009 Russell, Stuart; Norvig, Peter (2009) [1995]. Artificial Intelligence: A Modern Approach (3rd ed.). Prentice Hall. p. 103. ISBN 978-0-13-604259-4.

Sabbadin2014 Davide Sabbadin and Stefano Moro, (2014) Supervised molecular dynamics (SuMD) as a helpful tool to depict GPCR-ligand recognition pathway in a nanosecond time scale, Journal of Chemical Information and Modeling 54, doi:10.1021/ci400766b

VMD2016 https://www.ks.uiuc.edu/Research/vmd/current/docs.html

Wang2004 Wang J, Wolf RM, Caldwell JW, Kollman PA, Case DA (2004) Development and testing of a general amber force □eld. J Comput Chem 25:1157–1174. doi: 10.1002/jcc.20035

Wang2006 Wang J, Wang W, Kollman PA, Case DA (2006) Automatic atom type and bond type perception in molecular mechanical calculations. J Mol Graph Model 25:247–260. doi: 10.1016/j.jmgm.2005.12.005

1 Appendix I.

1.1 pyGro a blended script language format for Gromacs (v1.0)

pyGro takes advantege of phyton's flxibilty combined with unix shell comands for variable definitions and substitutions for scripting of the gromax workflow of complex, multistep MD simulations in a unique, easy to read and use, blended script language. It has an easy variable definition and replacement fot the unix comands and costum replacemeten in file content manipulation.

pyGro adopts from Gromacs the basic principle that is to keep things as simple as possible.

In first approximation the script ca be a unix comand list with comments, then one line python executiables can be added. Script execution flow commands are with capital letters *usage from shell*:

```
python script.gup [echo]
```

where `script.gup` is pyGro, gromacs/unix/python script with echo optional parameter that runs the script in test only mode where the unix shell comands are displayed only, not executed. The

script is a text file in which lines starting with character

1) # are comments
2) **$** are executed as unix commands in the shell *3)* other characters: python executables.
4) > labels for execution flow controll of the script in 2) the unix commands, pyGro substitutes variabels defined in as `subs` dictionary keys to their values.

Example usage:

1) `script.gup is`

```
# pyGro test file #
definitions
subs["!BOXSIZE"]="20 20 20"
subs["!MUSTER"]="/home/trezzaa/tesi/adam/water_only/step0/"
$ cd -r !MUSTER
# solvate
$ gmx solvate -cs spc216.gro -o conf.gro -box !BOXSIZE -p topol.top #
end
```

2) shell command:

```
python script.gup
```

3) expectaed behavior: will change the directory to `/home/trezzaa/tesi/adam/water_only/step0/` then execute the gromacs gmx solvate -cs spc216.gro -o conf.gro -box 20 20 20 -p topol.top command.

4) output: line number , tab, and 1) comments, 2) the commands with substitutions, 3) looktrough of the python code from the `script.gup`:

```
1  # pyGro test file
2  # definitions
3  subs["!BOXSIZE"]="20 20 20"
4  subs["!MUSTER"]="/home/trezzaa/tesi/adam/water_only/step0/
5  $ cd -r  /home/trezzaa/trezzaa/tesi/adam/water_only/step0/
6  # solvate
7  $ gmx solvate -cs spc216.gro -o conf.gro -box 20 20 20 -p topol.top 8 # end
```

1.1.1 The predefined commands and variables in pyGro v1.0 are:

1.1.1.1 ***subs_in_file(input_file , output_file ,subs_file)****: which substitutes the content of the input file according to the subs_file dictionary keys to their values and save it to the output file.*

subs: dictionary for substitution in unix commands (key-> value)

example script:

```
iteration = 0; subs["!ITERATION!"] = str(iteration ).strip();subs["!NEXT_ITERATION!"] = str(iteration+1).strip()
$ gmx grompp -f ./mdp/nvt.mdp -c nvt_!ITERATION!.gro -r nvt_!ITERATION!.gro -n index.ndx -t nvt_!ITERATION!.cpt -p topol.top -o nvt_!NEXT_ITERATION!.tpr -maxwarn 1000
$ gmx mdrun -deffnm nvt_!NEXT_ITERATION! -ntomp 4 -ntmpi 1
```

expected behavior: script runs the gromacs preprocessor and an MD simulation:

gmx grompp -f ./mdp/nvt.mdp -c nvt_0. gro -r nvt_0.gro -n index.ndx -t nvt_0.cpt -p topol.top -o nvt_1.tpr -maxwarn 1000 gmx mdrun

-deffnm nvt_1 -ntomp 4 -ntmpi 1

1.1.2 Standard script elements and their usage:

1.1.2.1 ***!LIVE****: variable the current directory* **subs_pdb**: dictionary for substitution in .pdb and .top files (key-> value) **subs_mdp**: dictionary for substitution in .mdp files (key-> value)

1.1.2.2 ***GOTO("label"):*** *Continue script execution from a named label , label is in quotes*

example script:

```
kk = -1
GOTO("bookmark")
```

```
>loopstart kk
= kk + 1
>bookmark
subs["!K!"] = str(kk).strip()
$ cp em.mdp em_!K!.mdp if
kk<3: GOTO("loopstart")
```

expected behavior produce the shell copy comands:

cp em.mdp em_-1.mdp

cp em.mdp em_0.mdp

cp em.mdp em_1.mdp

cp em.mdp em_2.mdp

label: Defined by the string in a line starting with „>" charcter, without quotes, if there are more then one in the script the last line where it apears is considered

*1.1.2.3 **EXIT()**: Terminate script execution*

*1.1.2.4 **LN = n**: Executes code form line n-th onward of the script (first line is zero, c convention)*

example script:

```
import random
# line 1
if random.random() >0.95: EXIT()
LN = 1
```

expected behavior: an indefinit length loop with 5% chance of exit in each iteration

*1.1.2.5 **SCRIPTNAME:** Contains the name of the scrip running.*

example script file named script_foo.gup:

```
print(SCRIPTNAME)
```

expected behavior:

prints out „script_foo.gup”

1.1.2.6 ECHO_OFF(key) and ECHO_ON(key) with key = "COMMENTS" , "SHELL", "PYTHON","EXIT","GOTO".

Turns off or on the standard output visualization of the respective type of lines in the script during the run

*1.1.2.7 **WAIT(n):** Pause execuiton for n seconds*

*1.1.2.8 **WAITFILE(source_dir = "./",filename, p=2,f=10):***

Waits till the file is created or modified in the following for *f* seconds,

but if the file is already present and was modified in the last *p* seconds it does not pause.

1.1.3 pygro_util.py module contains:

1.1.3.1 get_energy(file_name, quantity):

quantity : "Pressure" | "Temperature"

returns the average pressure (bar) , temperature (K) values and their estimeted error, RMSD, drift form the gmx energy files

1.1.3.2 approach(filename,group,tp)

group: index file group name

tp: *displacement, meanvelocity, linreg_disp, smooth*

returns a quantity decribing the approch of on group to another form a dist.xvg file generateg with

displacement: final minus inital distance (in nm units)

meanvelacity: velocity of approach (in nm/ps iunits)

linreg_disp: y = mx + c , regession coefficients: gradient (in nm/ps units) and constant (nm units) fited to the distance of group.

2 Appendix II.

2.1 pyGro script for standard em minization and nvt, npt equilibration

2.1.1 Directory structure:

The MD* run's main directory contains the followind directories are:

ndx - index files mdp -

parameter files pdb -

trajectory files eval -

merged trajectory

toppar- force fields

script_backup - backup of the script that generated the simulazione

2.1.2 File name convention

Files names fromualted as "type_S_B", where *type* is md for molecular dynamics .gro. trj , S integer is the section number, b integer the branch number. Tempalte file of "type_Template" serve as the templates for index and paramtere files.

2.1.3 Log files:

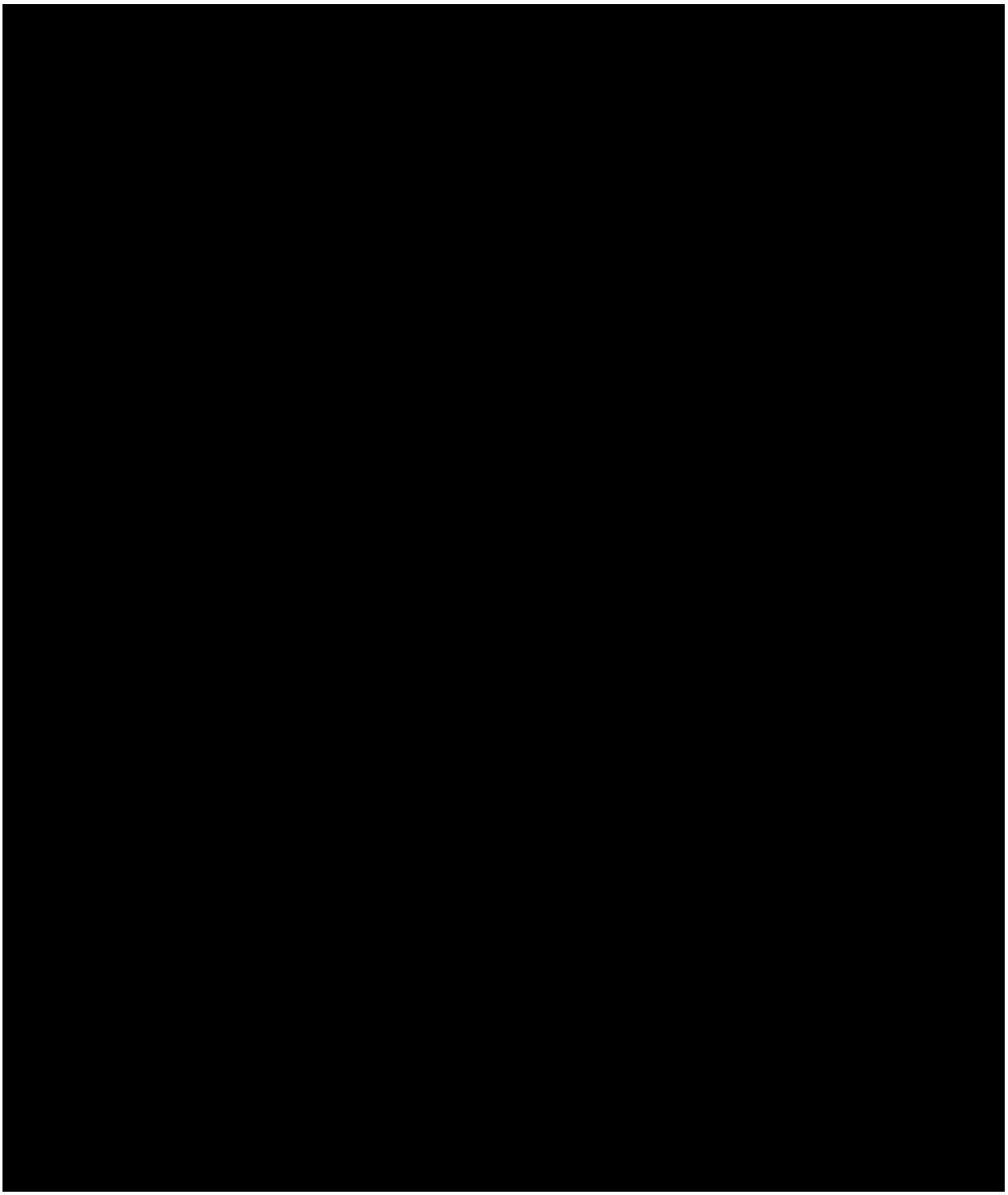
mds_run.log - contains step-by step information on MD segments, their chaining, decision variables (potential, temperature, pressure, distances, heuristics and ranking). Example output:

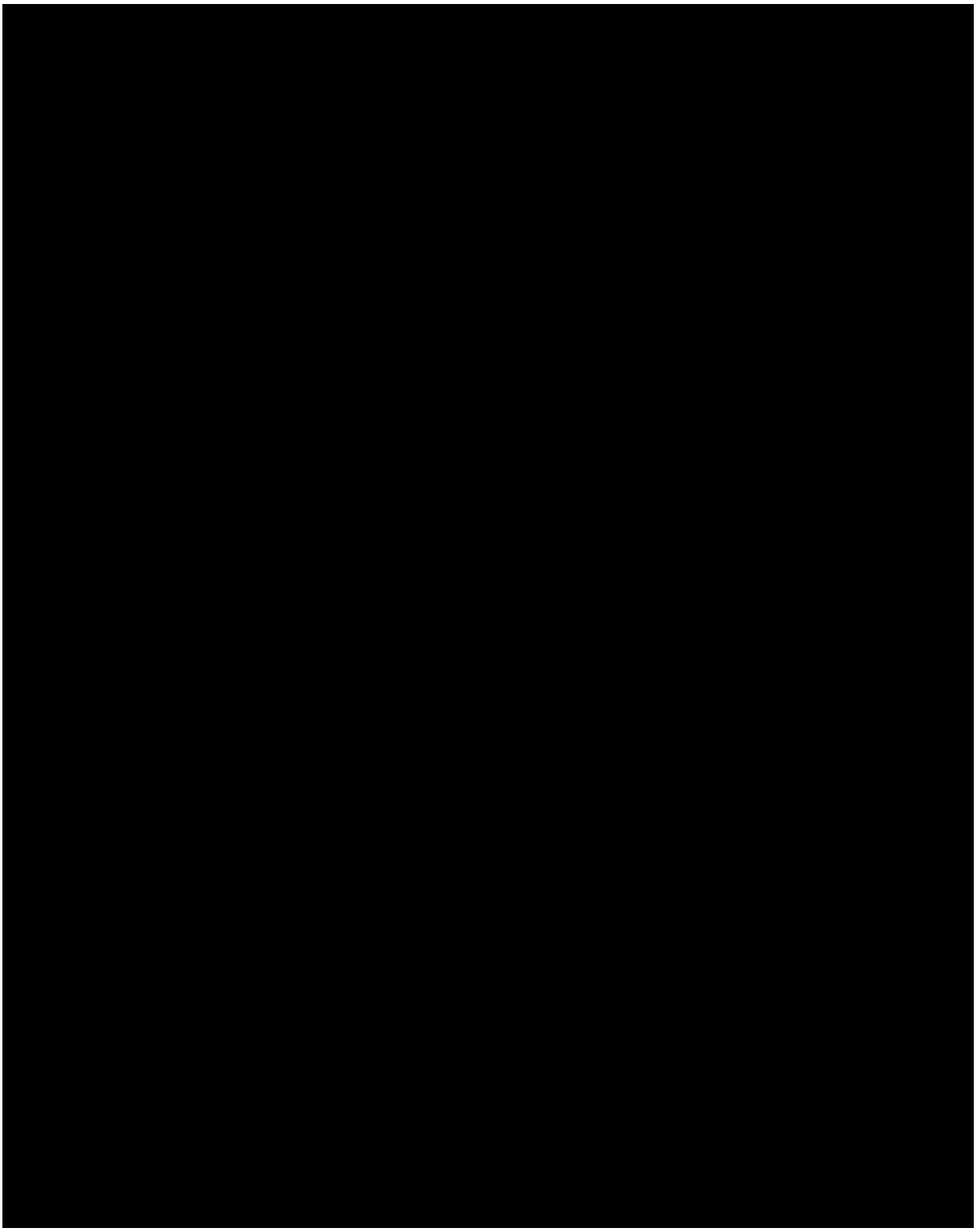
[redacted]

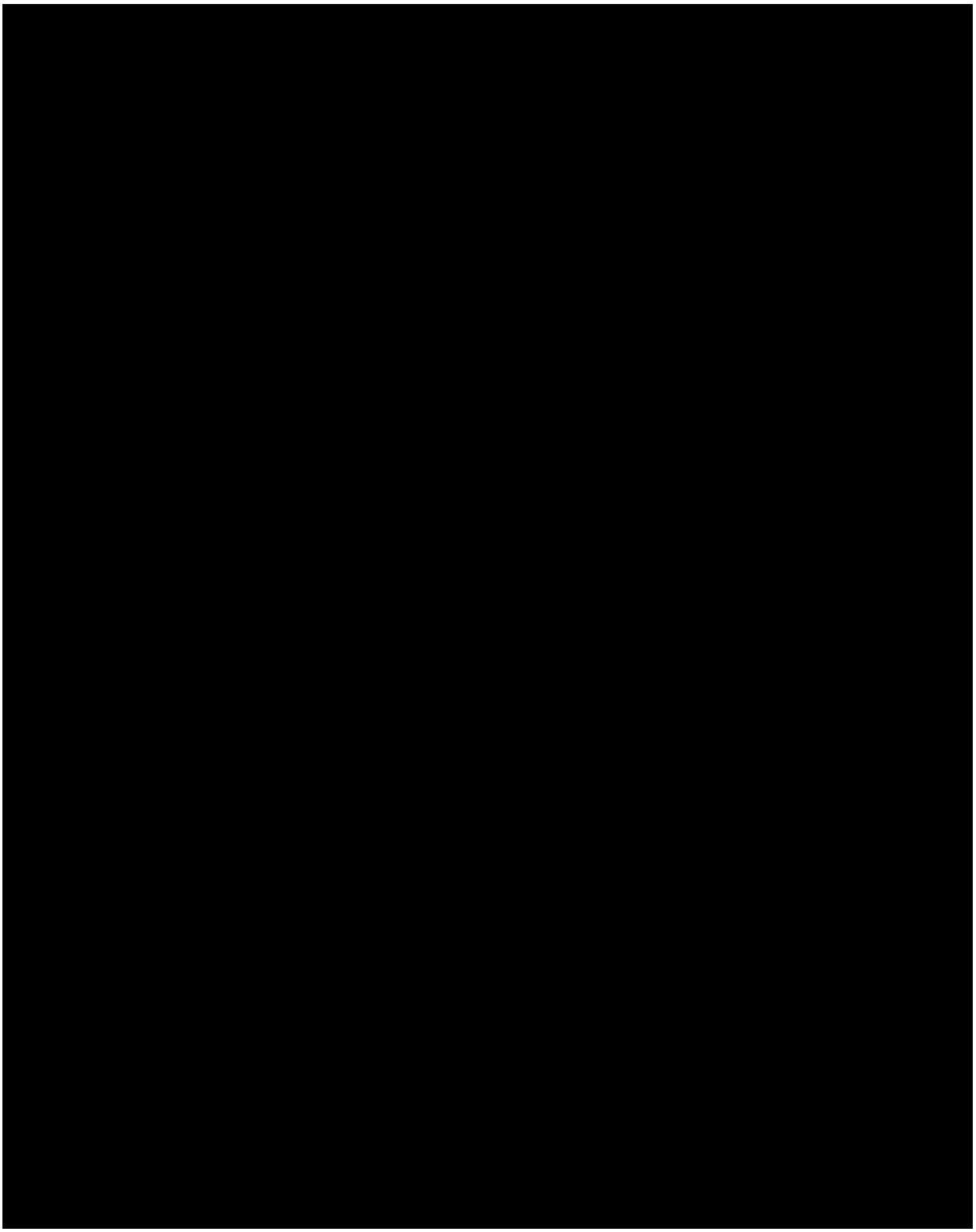
2.1.4 **The common part of the mds.gup script:**

[redacted]

2.1.5 **EM – Energy minimization**







2.1.8 Standard MD

3 Appendix III.

3.1 III. Literature review on SuMD

In this appendix I track the technical descriptions of Supervised MD found in the literature, from its first appearance till the modified versions reported recently. We can observe that there are increasing amount of technical detail and computational performance measures published.

3.1.1 III.1 Description of the algorithm used in Sabbadin [2014]:

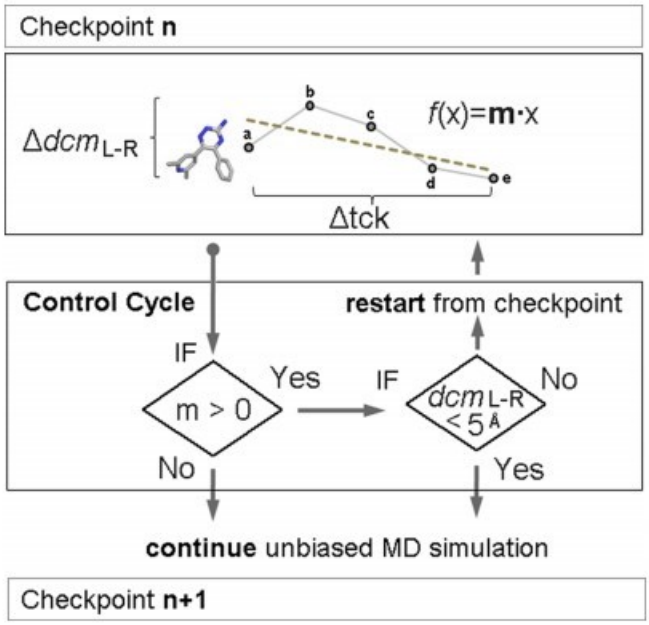
In Sabbadin [2014] the supervised MD was introduced and tested on human A2A

adenosine receptor hA2AAR complex with three binders: ZM 241385, T4G ,T4E and a weaker binder: caffeine

The SuMD consists of a series of short MD simulation segments of 600 *ps* length in time.

The ligand - target binding site distance is recorded at $n=5$ timestamps: (t_a, t_b, t_c, t_d, t_e,) as (d_a,d_b,d_c,d_d,d_e). linear function f(t) = m × t is fitted on the [(d_a,d_b,d_c,d_d,d_e) , (t_a, t_b, t_c, t_d, t_e,)] dataset.

The resulting slope (*m*) of a fitted linear function (distance vs. time) is negative or below a user selected threshold (a heuristic, showing sign of advancing in the binding process) the segment deemed *approaching* and the next simulation segment is started from the last set of coordinates and velocities produced by the previous segment; otherwise, the simulation is restarted from the original set of coordinates of the previous segment by randomly[20] assigning the atomic velocities to the coordinates of the previous segment coherently to the NVT ensemble.



Part of the scheme of the ligand−receptor distance supervision algorithm reported by Sabbadin [2014] in our taxonomy Checkpoint n is the n-th approaching segment simulated.

Termination condition was ligand−receptor distance less than 5 Å. Each SuMD was run 3 times.

[20] a point of interest wheter check the velocity of the center of mass of the small molecule .

The authors conclude that, it was possible for them to easily determine and characterize all possible ligand binding sites that chronologically anticipate the orthosteric one form the SuMD .The SuMD facilitate a better understanding of all GPCR−ligand recognition pathways.

hA2AAR complex biding with ZM 241385 the one of 3 SuMD simulation results in [Sabbadin 2014]

Summary table of the supervised molecular dynamics simulation results in [Sabbadin 2014]:

interaction description	***event***	***a***	***b***	***c***	***d***	***e***	***unit***
adenosine receptor antagonist ZM241385−human A2A adenosine receptor	time	1.2	2.9	5.5	8.1	59	*ns*
	distance	43	16	8	5	4	*Angstrom*
	interaction energy	-1	-42	-55	-48	-59	*kcal/mol*
T4G−human A2A adenosine receptor recognition mechanism	time	0.2	6	14.7	46	62	*ns*
	distance	43	11	8		4	*Angstrom*
	interaction energy	-3	-32	-31	-20	-57	*kcal/mol*
T4E−human A2A adenosine Receptor recognition mechanism	time	0.1	5.7	16	46.7	105	*ns*
	distance	40	15	8	5	4	*Angstrom*
	interaction energy	-1	-17	-32	-15	-47	*kcal/mol*

Summary table of the supervised molecular dynamics simulation results of [Sabbadin 2014]:

In [Cuzzolin 2016] simulate globular and trans-membrane proteins targets with supervised MD algorithm. They report that the SuMD simulation engine was interfaced with the ACEMD engine and supported AMBER and CHARMM force fields.

3.1.2 III.2 Description of the algorithm used in Cuzzolin [2016]:

The SuMD consists of a series of short MD simulation segments of 600 *ps* length in time.

The ligand - target binding site distance is recorded at 0,75,150,225,300,375,450,525,600 *ps*.

The resulting slope of a fitted linear function (distance vs. time) is negative or below a user selected threshold (a heuristic, showing sign of advancing in the binding process) the segment deemed *approaching* and the next simulation segment is started from the last set of coordinates and velocities produced by the previous segment; otherwise, the simulation is restarted by randomly[21] assigning the atomic velocities to the coordinates of the previous segment.

A preliminary run is performed as follows, in the first simulation segment if 31 consecutive nonapproaching sequence was simulated new initial coordinates are randomly selected. (schema of the algorithm from the authors on the right)

Each segment records the distance revelations; the resulting slope value; distance distribution counters for all previous valid segments in the ranges in the 0–2, 2–5, and 5–9 Å; electrostatic and van der Waals potential energy contributions of the ligand–receptor interaction energy (IE).

[21] a point of extreme interest how relaxation is applied here b the authors if they do, whether they check the velocity of the center of mass of the small molecule if compatible with expectations of statistical mechanics.

The last simulation frame data of each sequence (the end point at 600 ps) is separately stored.

For example the authors report the 30th simulation segment that resulted in 7.435 Å final distance which was the 13th attempt after the 29th segment (that is they simulate effectively 13*600 ps 7.8 ns to arrive to the last 600ns approaching segment, slope = -.282 which is average approaching velocity of … Å/*ps*), and the 5–9 Å distance counter (Dist.9) is 19, i.e. the ligand approached for 11.9 ns 9 Å near, but never reached 5 Å distance as the 3–2 Å (Dist.5) is zero. Cuzzolin [2016 supp]

suMD_Step	Slope	Last_Distance	Ele_Int.	VdW_Int.	Lig_Int.	Try	Dist.2	Dist.5	Dist.9
Step_num 2	-2.523	51.417	0.000	0.000	0.000	1	0	0	0
Step_num 3	2.482	63.416	0.000	0.000	0.000	1	0	0	0
Step_num 3	-1.234	36.119	0.000	0.000	0.000	2	0	0	0
...									
Step_num 16	-0.889	26.406	2.101	-10.484	-18.383	1	0	0	0
...									
Step_num 30	-0.282	7.435	-6.289	-16.619	-42.909	13	0	0	19

Example output on which supervision decisions are based (Step corresponds to segment in our taxonomy)

If the simulation segment end point distance drops below 5 Å the simulation proceeds a classical MD simulation with unspecified length (by the authors) in the Introduction while in section 2.5.4 describe a more complex Termination Criteria is complex as set of rules describing the overflow of the counters:

more than 17 consecutive reruns of the segment generation all with non negative slope, that is 10.2 ns effective simulation length spent; any of the distance counter exceeds 17 , 11.4 nonconsecutive ns spent in the 5–9 Å , 3–2 Å or the 2–0 Å range.

Authors report computation time in single to lower double-digit hours for the case studies in the article (list on the left), and that some SuMD trajectories converge in a different way to the structure of the complex as seen by Xray crystallography. They present tree hypothesis to investigate this. Further on-rate binding kinetics property estimations are in approximate agreement with experimental measurements.

Acid Ellagic-CK2
SASP-GSTP1-1
Benzen-1,2-diol–PDRX:
(*S*)-naproxen-HAS
(*S*)-fluoxetine-LeuT
NECA–hA_{2A} AR
Acid Ellagic-CK2

List of interactions simulated by Cuzzolin

[2016]

In Deganutti [2020] simulated unbinding of small druglike molecules from fundamental pharmacological targets – six different G protein-coupled membrane receptors - by a modified version of the supervised MD algorithm.

According to the authors the changes in the algorithm were aimed to optimize the simulation performance specifically for unbinding of small molecules and

3.1.3 III 3. Description of the algorithm used in Deganutti [2020]:

The SuMD consists of a series of short unbiased MD simulation segments.

The length of each simulation segment's run depends on the ligand-protein distance of the previous simulation segment. According to the authors of 3, 5, and 8 Å distance regimes are 'normally' appropriate to distinguish the simulation length in time to use. The multiplicative values are tabulated for the seven interactions in the article, for example

Complex	**Δt_0(ps)**	**D_1 (Å)**	**N_t_1**	**D_2(Å)**	**N_t_2**	**D_3 (Å)**	**N_t_3**
Adenosine A_1 receptor - adenosin	100	3	4	5	10	8	20

The algorithm in the Adenosine A1 receptor – Adenosin unbinding SuMD simulation segment generation, for ligand–protein distance smaller that 3 Å uses a 100 *ps* time window for each run. In the 3-5 Å range a four-fold, in 5-8 Å a 10, for higher than 8 Å a 20-fold increase is applied to the 100

ps base MD time window as the next simulation segment's length. That is 100, 400, 1000, 2000 ps window length runs for the different distance regimes.[22]

The unbinding simulation segment runs thus are iterated until no ligand–protein van der Waals contact is detected[23]. The authors use the ligand and the protein's specific residue center of masses to measure distances.[24] For example in the Adenosine A1 receptor - Adenosine unbinding SuMD simulation the center of mass of th S61.29 to K301 8.56protein residues define the distance form the Adenosin's one.[25]

In each simulation segment the ligand-protein distance is collected at regular time intervals. If the resulting slope of a fitted linear function (distance vs. time) is positive (a heuristic, showing sign of advancing in the unbinding process), the next simulation segment is started from the last set of coordinates and velocities produced by the previous segment; otherwise, the simulation is restarted by randomly[26] assigning the atomic velocities to the coordinates of the previous segment.

Results:

The SuMD was repeated three times, for each seven interaction in the right list. The authors claim that without the input of any energy bias to facilitate the dissociation mechanism of druglike small molecules were successfully simulated by their modified supervised MD (SuMD) algorithm. This approach sheds light on the multistep nature of ligand–receptor dissociation, can rationalize previous experimental data and elaborate for structure–kinetics relationships hypothesizes to be tested.

protein	ligand
A2A receptor, $A_{2A}R$	adenosine
A1 receptor, A_1R	adenosine
A2A receptor, $A_{2A}R$	NECA
A2A receptor, $A_{2A}R$	ZMA
Orexin 2 receptor, OX_2R	EMPA
Muscarinic 2 receptor, M_2R	QNB
soluble epoxide hydrolase, sEH	TPPU

List of unbinding simulated by Deganutti [2020]

[22] Table S2 [Deganutti 2020 Supp]
[23] GetContacts script https://getcontacts.github.io/
[24] PLUMED 2 [Tribello 2014]
[25] Table S3 [Deganutti 2020 Supp]
[26] a point of extreem interest how realxation is applied here b the authors if they do, wheter thay chec the voelocity of the center of mass of the small molecule if compatible with expectations of statistical mechanics.

The authors propose:

- rescaling the simulation time to kinetically rank similar compounds in SuMD
- rescaling the simulation time to a priori valuation of the ratio of the total and the productive SuMD simulation time needed for dissociating structurally related compounds.
- combine SuMD with other adaptive sampling methods to yield in the construction of kinetic Markov state models.
- to reconstruct the energy surface of the transitions over path collective variable resulting from SuMD

In Deganutti [2021 pr.pr] four SuMD simulations of the ACE2:RBD complex and the RBD:cefsulodin:ACE2 ternary complex was performed respectively with 6 FDA-approved drugs prescreen from 2421.

Simulation predicted to binding to the SARS-CoV-2 S glycoprotein RBD for (cefsulodin, cromoglycate, nafamostat, nilotinib, pen□uridol, and radotinib

3.1.4 III 4. Description of the algorithm used in Deganutti [2021]:

25 Å away from initial distance between the centroid of RBD residue Q493 and the centroid of ACE2 residues K31, E35, with 2 ns monitoring frequency until 7 Å, then 200 ns final sequence MD run.

For the initial complex configuration of RBD:cefsulodin was from a post docking MD simulation of 500 ns.

Results:

Simulation predicted to binding to the SARS-CoV-2 S glycoprotein RBD for (cefsulodin, cromoglycate, nafamostat, nilotinib, pen□uridol, and radotinib

In particular the SuMD, the overall binding path between the two proteins by the presence of cefsulodin in the RBD pocket during the approach to ACE2 was different.

In [Bissaro 2021] SuMD simulated the Influenza A promoter,HIV-1 RevREE complex ,SAH riboswitch ,PreQ1 riboswitch,PreQ1 riboswitch, Corn aptamer.

3.1.5 Description of the algorithm used in Bissaro [2021]:

Segments are 600 ps unbiased MD trajectories, approaching segments are defined by the ligand center of mass with respect to the ribonucleic acid binding site distances periodic relevations' fitted slope against time. For negative slope, the segment deemed approaching. For approaching segments, the next segment's simulation is a proper MD continuation, its initial conditions are coincident to the final conditions.

For non approaching segments, the next segment's simulation is restarted from the previous set of coordinates but randomly assigning new atomic velocities. Termination condition is a ligand-ribonucleic binding site distance below 5 Å, a short (cc. 15 ns ?) classical MD simulation was performed, allowing the system to relax. 10 runs of SuMD simulations were collected for each experimental biding setup.

Simulation efficiency of is the ratio of number of approaching segments and all segments simulated.

Also, a technical note on the performance of SuMD was reported, from which we computed the Simulation efficiency, which remarkably low for SAH riboswitch, and predominantly in the 200% are:

Ribonucleic Systems	Simulated atoms	SuMD time (ns)	Simulation efficiency	total MD segment time (ns)	Performance (ns/day)
Influenza A promoter	66944	27	226%	61	102
HIV-1 Rev-REE complex	80339	32	147%	47	92
SAH riboswitch	70801	23	113%	26	103
PreQ1 riboswitch	47286	34	206%	70	141
PreQ1 riboswitch	48514	32	191%	61	139
Corn aptamer	103591	29	203%	59	67

The authors valuate SuMD as a valid computational method to generate binding hypothesis for ribonucleic targets in a nanosecond timescale. considering flexibility of the macromolecule the role of solvent. Help interpretation and investigation of the complex mechanism of recognition characterizing guide, rational discovery and optimization of the compounds in question.

3.1.6 Summary table of Molecular Dynamics Simulation and SuMD parameters used in the literature:

		Cuzzolin 2014			Deganutti [2021	Deganutti [2021	
		Transmembrane Systems	Globular Systems		spike protein RBD	Post Docking molecular dynamics simulations	ACE2:RBD complex
Force Field			AMBER14SB		Amber14SB[27]	Amber14SB/GAFF[28]	Amber14SB
box		VMD28 membrane builder plugin, lipids within 0.6 Å from amino acid atoms were removed	Cubic, 12 Å away from any protein or ligand atom		90 Å x 92 Å x 73 Å	77 Å x 95 Å x 71 Å box	124 Å x 117 Å x 160 Å
solvate		TIP3P, Solvate 1.0	TIP3P water model.	TIP3P, Solvate 1.0	TIP3P[2930]	TIP3P	TIP3P
charge neutrality		Na+/Cl− counterions to a final concentration of 0.154 M.	Na+/Cl− counterions were added to a final salt concentration of 0.150 M	Na+/Cl− counterions to a final concentration of 0.154 M.	two Cl- ions added	Cl- or Na+ counterions added	21 Na+ ions.

[27] Maier JA, Martinez C, Kasavajhala K, Wickstrom L, Hauser KE, Simmerling C (2015) ff14SB: Improving the Accuracy of Protein Side Chain and Backbone Parameters from ff99SB. J Chem Theory Comput 11:3696–3713. doi: 10.1021/acs.jctc.5b00255

[28] Wang J, Wang W, Kollman PA, Case DA (2006) Automatic atom type and bond type perception in molecular mechanical calculations. J Mol Graph Model 25:247–260. doi: 10.1016/j.jmgm.2005.12.005 and Wang J, Wolf RM, Caldwell JW, Kollman PA, Case DA (2004) Development and testing of a general amber force □eld. J Comput Chem 25:1157–1174. doi: 10.1002/jcc.20035

[29] Jorgensen WL, Chandrasekhar J, Madura JD, Impey RW, Klein ML (1983) Comparison of simple potential functions for simulating liquid water. J Chem Phys 79:926. doi: 10.1063/1.445869

equilibration[31]		2000 cycles of a conjugate gradient; 10 ns of MD of NPT ensemble, restraining ligand, protein atoms by a force constant of 1 kcal mol−1 Å−2.	2000 steps with the conjugate gradient method; 50,000 steps of NVE (100 ps) followed by 1 ns of NPT simulation, (2fs), on protein , ligand atoms harmonic positional constrain reduction by scaling factor of 0.1.	2000 cycles of a conjugate gradient 10 ns of MD, NPT ensemble, restraining ligand and protein atoms by a force constant of 1 kcal mol−1 Å−2.	34 ACEMD	ACEMD 3x 100ns	ACEMD 200ns
		T=310 K by Langevin thermostat[32] low damping constant= 1 ps-1 Pressure at 1 atm using a	Langevin thermostat[33] low damping constant= 1 ps-1 Pressure at 1 atm by Berendsen barostat[34]	Langevin thermostat[35] low damping constant = 1 ps-1 Pressure at 1 atm by Berendsen		Restraints applied to protein alpha carbon atoms with gradually released in2 ns	generalized Born and surface area continuum solvation[36] (MMPBSA.py)
		Berendsen barostat		barostat[39]			
Simulation MD					4 fs, canonical ensemble (NVT).		

[31] Harvey MJ, Giupponi G, Fabritiis GD (2009) ACEMD: Accelerating Biomolecular Dynamics in the Microsecond Time Scale. J Chem Theory Comput 5:1632–1639. doi: 10.1021/ct9000685

[32] Loncharich, R. J.; Brooks, B. R.; Pastor, R. W. Langevin Dynamicsof Peptides: The Frictional Dependence of Isomerization Rates of NAcetylalanyl- N'-Methylamide. Biopolymers 1992, 32, 523−535.

[33] Loncharich, R. J.; Brooks, B. R.; Pastor, R. W. Langevin Dynamicsof Peptides: The Frictional Dependence of Isomerization Rates of NAcetylalanyl- N'-Methylamide. Biopolymers 1992, 32, 523−535.

[34] Berendsen, H. J. C.; Postma, J. P. M.; van Gunsteren, W. F.; DiNola, A.; Haak, J. R. Molecular Dynamics with Coupling to an External Bath. J. Chem. Phys. 1984, 81, 3684−3690.

[35] Loncharich, R. J.; Brooks, B. R.; Pastor, R. W. Langevin Dynamicsof Peptides: The Frictional Dependence of Isomerization Rates of NAcetylalanyl- N'-Methylamide. Biopolymers 1992, 32, 523−535.

[36] Miller BR, McGee TD, Swails JM, Homeyer N, Gohlke H, Roitberg AE (2012) MMPBSA.py: An E□cient Program for End-State Free Energy Calculations. J Chem Theory Comput 8:3314–3321. doi: 10.1021/ct300418h

thermostat damping					0.1 ps-1;		
cut-off distance for electrostatic interactions					9 Å, switching function applied beyond 7.5 Å.		
Long range Coulomb interactions					particle mesh Ewald summation method (PME), 41m.spacing 1.0 Å.		

4

39 Berendsen, H. J. C.; Postma, J. P. M.; van Gunsteren, W. F.; DiNola, A.; Haak, J. R. Molecular Dynamics with Coupling to an External Bath. J. Chem. Phys. 1984, 81, 3684−3690.

41 Essmann U, Perera L, Berkowitz ML, Darden T, Lee H, Pedersen LG (1995) A smooth particle mesh Ewald method. J Chem Phys 103:8577. doi: 10.1063/1.470117