Simulating ligand-induced conformational changes in proteins using a mechanical disassembly method

Juan Cortés, Duc Thanh Le, Romain Iehl, Thierry Simeon

Simulating ligand-induced conformational changes in proteins using a mechanical disassembly method

Juan Cortés[*,†] Duc Thanh Le[*] Romain Iehl[*] Thierry Siméon[*,‡]

Abstract

Simulating protein conformational changes induced or required by the internal diffusion of a ligand is important for the understanding of their interaction mechanisms. Such simulations are challenging for currently available computational methods. In this paper, the problem is formulated as a mechanical disassembly problem where the protein and the ligand are modeled like articulated mechanisms, and an efficient method for computing molecular disassembly paths is described. The method extends recent techniques developed in the framework of robot motion planning. Results illustrating the capacities of the approach are presented on two biologically interesting systems involving ligand-induced conformational changes: lactose permease (LacY), and the β_2-adrenergic receptor.

Introduction

Proteins are flexible macromolecules that fluctuate between nearly isoenergetic folded states[1]. In many cases, conformational changes are associated with their function, and they occur through the interaction with other molecules. For instance, conformational changes are of major importance for protein-ligand and protein-protein recognition[2,3].

This paper addresses protein conformational changes induced (or required) by the diffusion of a ligand (or substrate/product) molecule inside the protein. An illustrative example is the permeation of lactose through a membrane transport protein (LacY)[4]. LacY fluctuates between a conformation where lactose is accessible from the cytoplasm, but the channel toward the periplasmic side is closed (Figure 1.a), and the opposite conformation where the channel is open toward the periplasm and closed in the cytoplasmic side (Figure 1.b). The transition between these two conformational states occurs during lactose diffusion inside the protein.

Despite impressive recent advances on the structural determination of protein motions[5,6], currently available experimental methods are unable to provide an atomic-resolution structural description of protein conformational changes associated with ligand diffusion. Computational methods are therefore necessary to better understand such processes. However, the time-scale of the ligand diffusion process from a deep active site to the protein surface is out of range for standard molecular dynamics (MD) simulations. Variants of MD methods such as steered molecular dynamics (SMD)[7] and random acceleration molecular dynamics (RAMD)[8] have been proposed for accelerating the simulation of the ligand exit. Both methods introduce an artificial force in the molecular force field to enhance the ligand motion in a given direction. In SMD simulations, this direction is usually defined by the user through an haptic device. In RAMD simulations, the direction is randomly chosen and iteratively modified after a given number of simulation steps if the ligand gets stuck. Although these methods have been shown to provide biologically relevant information, they remain computationally expensive. Besides, the artificial force introduced for accelerating the simulation may yield biased results about the induced conformational changes, so that the interest of simulating with an accurate molecular force field is partially lost.

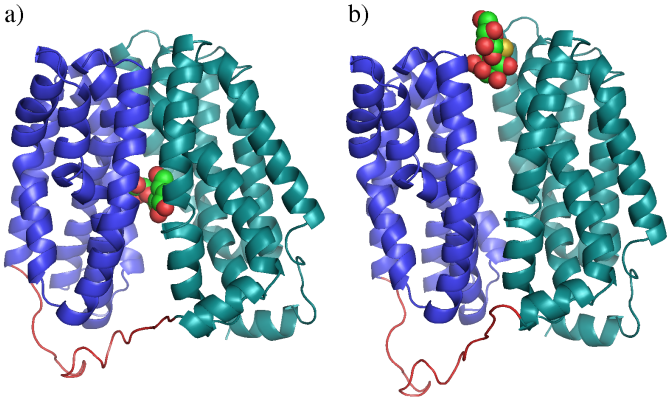

Figure 1: Lactose permease (LacY) conformational transition. a) The crystal structure[9] (PDB ID 1PV7), where the substrate is accessible from the cytoplasm. b) Model of LacY after the conformational change induced by the substrate diffusion toward the periplasm.

This paper presents an alternative method for simulating ligand diffusion motions, together with the possibly induced conformational changes of the protein. Given an initial structure with the ligand docked inside the protein, the proposed method computes a path (i.e. continuous sequence of conformations) simulating the ligand exit. Such a path search problem is formulated as a mechanical disassembly problem, where the protein and the ligand are modeled as articulated mechanisms. The main feature of this method is its computational efficiency, enabling to compute large-amplitude conformation transition paths, such as the one illustrated in Figure 1, in less than one hour of CPU time.

Computing disassembly paths for mechanical parts is an important problem in the fields of robotics and manufacturing en-

[*]LAAS-CNRS, Université de Toulouse; 7 avenue du colonel Roche, F-31077 Toulouse, France
[†]E-mail: juan.cortes@laas.fr
[‡]E-mail: thierry.simeon@laas.fr

gineering. In the last years, randomized search algorithms [10] have been demonstrated to be effective computational tools for disassembly path planning [11,12]. Thanks to their generality, this type of algorithms have also been applied to solve problems in computational structural biology [13,14,15]. In this framework, the ML-RRT algorithm [16] was introduced as a general method for computing disassembly paths of objects with articulated parts. ML-RRT has been successfully applied in enzyme enantioselectivity studies for computing ligand exit paths considering the flexibility of the protein side-chains [17,18].

The methodological contribution of this paper is an extension of ML-RRT that enables further introduction of protein flexibility, so that challenging problems involving protein models with flexible backbone segments can be tackled. The improved algorithm is able to consider not only side-chain local flexibility, but also loop or domain motions induced by the ligand along the diffusion pathway. As a proof of concept, the method is applied to two biologically interesting systems involving ligand-induced conformational changes: lactose permease (LacY), and the β_2-adrenergic receptor.

Methods

Outline

Path search problem: The problem of computing the exit path of a ligand from a protein active site is formulated as a mechanical disassembly problem in which molecules are represented as articulated mechanisms. The degrees of freedom of the molecular models correspond to bond torsion (backbone or side-chains) and to rigid-body motion of atoms groups (rigid secondary structure elements). Starting from a given "assembled" (docked) position of the ligand inside the protein, the disassembly problem consists in finding the path leading to a "disassembled" state, where the ligand is located outside the protein. The disassembly path has to be searched in a composite conformational space involving the degrees of freedom of the protein and the ligand. The difficulty for solving such path search problem is due to the very high dimension of this search-space.

Random diffusion trees: The conformational exploration algorithm described in this work is derived from the Rapidly-exploring Random Tree (RRT) algorithm [19], developed in robotics, and which has been demonstrated to perform well for solving complex disassembly problems in constrained spaces. The basic principle of RRT is to iteratively construct a random tree, rooted at a given initial state, and tending to cover the accessible regions of the search-space. The nodes of the tree correspond to states generated by the diffusion process, and the edges correspond to feasible local paths. The RRT construction process is illustrated by Figure 2 on a simple two-dimensional problem. At each iteration of the algorithm, a state $\mathbf{q}_{\text{rand}}$ is randomly sampled following a uniform distribution in the search-space. The nearest node in the tree $\mathbf{q}_{\text{near}}$ is selected, and an attempt is made to expand it in the direction of $\mathbf{q}_{\text{rand}}$. A new node $\mathbf{q}_{\text{new}}$ is generated at the endpoint of the feasible straight-line path (i.e. sub-path satisfying motion constraints) from $\mathbf{q}_{\text{near}}$ to $\mathbf{q}_{\text{rand}}$. The process is iterated until the final state can be connected to the tree. This tree construction strategy favors an efficient exploration biased toward unexplored regions, while converging to a uniform coverage of the space [19]. This technique performs well for solving moderately high-dimensional problems. However, its performance degrades when applied to very-high-dimensional search-spaces.

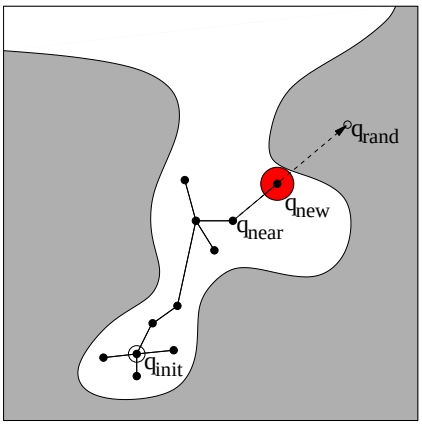

Figure 2: Illustration of the RRT expansion process.

Manhattan-like RRT: The Manhattan-like RRT (ML-RRT) variant [16] was developed to circumvent this limitation of the basic RRT algorithm for dealing with disassembly problems involving complex articulated objects. The main idea is to facilitate the tree expansion by considering separately two types of conformational parameters, called *active* and *passive*. Active parameters are essential for the disassembly problem, and they are directly treated at each iteration of the algorithm. Passive parameters, however, only need to be treated when they hinder the expansion of active parameters. The advantage of this decoupled treatment, that favors the expansion of the active parameters, is to maintain the exploratory strength of the RRT algorithm while dealing with high-dimensional problems. The ML-RRT algorithm was successfully applied in previous work [17,18] for computing ligand exit paths considering the flexibility of the protein side-chains. For this particular application, the partition of the conformational parameters makes the exploration be focused on the ligand diffusion (active parameters), while the protein side-chain motions (passive parameters) are induced by the ligand motion.

Building on this prior work, we describe below an extension of ML-RRT that enables the simulation of loop/domain motions induced by the ligand diffusion. The proposed generalization of the ML-RRT principle relies on a classification and hierarchization of the different elements in the mechanistic molecular model, receiving each a specific treatment during the exploration.

Model and parameters

Mechanistic molecular model: The proposed method deals with all-atom models of molecules, which are represented as articulated mechanisms. Groups of atoms form the bodies, and the articulations between bodies correspond to bond torsions. The size of the atom groups depends on the level of flexibility allowed to different parts of the molecule. Flexible and rigid regions can be assigned based on structural knowledge. In the present work, flexibility is defined by the user. Note however that the identification of rigid and flexible regions may be automated using computational methods such as FIRST [20].

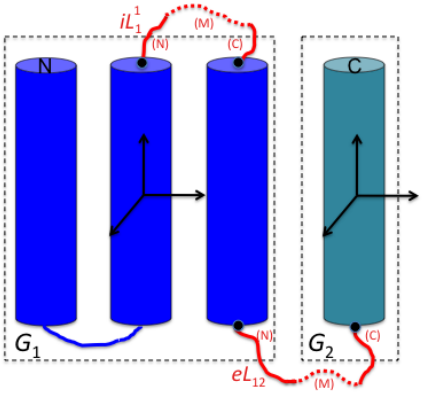

Figure 3: Schematic representation of a flexible protein model. The three secondary structure elements grouped in G_1 are modeled as a rigid solid. The group G_2 involves only one secondary structure element. The loop/linker $eL_{1,2}$, between G_1 and G_2, is flexible. Loops connecting elements in a group can be flexible or not. Only the intra-domain loop iL_1^1 is flexible in this example.

Figure 3 illustrates the mechanistic model of a protein. The following notation is used:

- G_i group: set of rigid secondary structure elements (with flexible side-chains), possibly connected by flexible loops.
- iL_i^k intra-group loop: k^{th} flexible segment between two secondary structure elements of group G_i.
- $eL_{i,i+1}$ inter-group loop/linker: flexible segment between secondary structure elements in consecutive groups G_i and G_{i+1}.

Each group G_i holds free rigid body mobility, independently from the other groups. Therefore, loop-closure constraints have to be imposed on flexible segments $eL_{i,i+1}$ and $eL_{i-1,i}$ connecting G_i to its neighboring groups, in order to maintain the molecular chain integrity. As indicated in Figure 3, several parts are differentiated inside inter- or intra-group loops: the N-terminal and C-terminal segments, and the middle part (M), which is composed by a tripeptide. Such a decomposition is required for the treatment of loop motions that will be explained below. Additionally, geometric (distance and orientation) constraints can be introduced between any pair of elements (rigid groups or loops) in order to model interactions such as hydrogen bonds or disulfide bonds. All these constraints will be satisfied during the conformational exploration.

Side-chains (not represented in the figure) are generally modeled as flexible elements with freely rotatable bond torsions. By default, the ligand is also fully flexible. Nevertheless, the user can arbitrarily define the flexibility of the ligand and the side-chains.

Conformational parameters: The protein conformation is defined by the parameters determining the pose (position and orientation) of all the groups G_i, the values of the bond torsions in intra- and inter-group loops, and the bond torsions of the side-chains. The conformational parameters of the ligand are the six parameters defining the pose of its reference frame (associated with its center of mass), and the values of the allowed bond torsions.

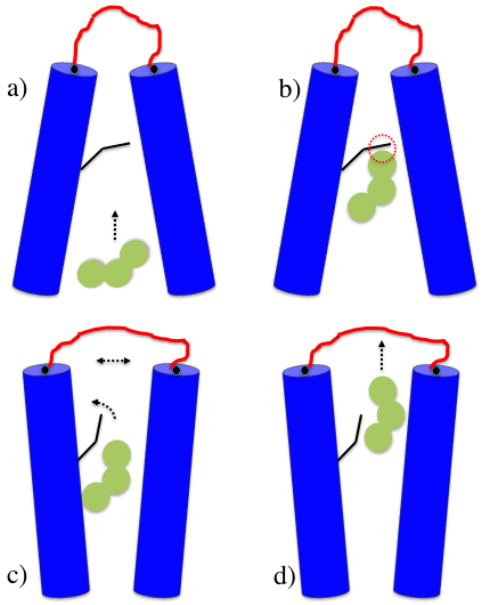

Figure 4: Illustration of the decoupled exploration of active and passive parameters within ML-RRT. a) Expansion of active parameters corresponding to the motion of the ligand. b) Identification of the passive parts hindering the ligand motion. c) The expansion of passive parameters yielding the opening motion of the protein. d) New iteration of the active parameters expansion.

Let $\mathbf{q}$ denote the array containing the values of all the conformational parameters of the protein and the ligand. The ML-RRT algorithm explores the composite conformational space $\boldsymbol{C}$, which is the set of all conformations $\mathbf{q}$. As mentioned above, the conformational parameters are partitioned into *active* and *passive* on the basis of their role in the disassembly problem. Active parameters are essential for carrying out the disassembly task, while passive parameters only need to move if they hinder the progress of the process. Thus, the mobile parts of the molecular model are separated into two lists P_{act} and P_{pas} containing the active and the passive parts respectively. For a given partition, the conformational parameters are separated into two sets: $\mathbf{q} = \{\mathbf{q}^{\text{act}}, \mathbf{q}^{\text{pas}}\}$, where $\mathbf{q}^{\text{act}}$ is the set of conformational parameters associated with the parts in P_{act} and $\mathbf{q}^{\text{pas}}$ is the set associated with P_{pas}. For the protein-ligand disassembly problems addressed in this paper, $\mathbf{q}^{\text{act}}$ involves the ligand parameters, while $\mathbf{q}^{\text{pas}}$ concerns the protein flexibility.

Additionally, a mobility coefficient $\delta \in (0,1]$ is assigned to each passive parameter. This coefficient is used to differentiate passive parts that are allowed to move easily from those that should be moved only if the solution path cannot be found otherwise. By default, the mobility coefficient of all side-chains is set to 1, meaning that they will systematically move if they are identified during the exploration. Lower mobility is allowed to loops and secondary structure groups, with $\delta = 0.5$ and $\delta = 0.2$ respectively in the current implementation.

Conformational exploration algorithm

ML-RRT computes the motion of parts associated with active and passive parameters in a decoupled manner. Figure 4 provides a simple illustration of the process, which alternates expansion attempts of these parameter subsets.

The ML-RRT algorithm is sketched in Algorithm 1. At each

Algorithm 1: Construct_ML-RRT

input : the conformational space $\boldsymbol{C}$;
the initial conformation $\mathbf{q}_{\text{init}}$;
the partition $\{P_{\text{act}}, P_{\text{pas}}\}$;
output : the tree τ;
begin
  $\tau \leftarrow$ `InitTree`($\mathbf{q}_{\text{init}}$);
  while not `StopCondition`(τ) **do**
    $\mathbf{q}^{\text{act}}_{\text{rand}} \leftarrow$ `SampleConf`($\boldsymbol{C}$, P_{act});
    $\mathbf{q}_{\text{near}} \leftarrow$ `NearestNeighbor`(τ, $\mathbf{q}^{\text{act}}_{\text{rand}}$, P_{act});
    ($\mathbf{q}_{\text{new}}$, $P^{\text{col}}_{\text{pas}}$) $\leftarrow$ `Expand`($\mathbf{q}_{\text{near}}$, $\mathbf{q}^{\text{act}}_{\text{rand}}$);
    while $P^{col}_{pas} \neq \emptyset$ **do**
      $P^{\text{mov}}_{\text{pas}} \leftarrow$ `PartsToMove`($P^{\text{col}}_{\text{pas}}$);
      $\mathbf{q}^{\text{pas}}_{\text{rand}} \leftarrow$ `PerturbConf`($\boldsymbol{C}$, $\mathbf{q}_{\text{new}}$, $P^{\text{mov}}_{\text{pas}}$, $\mathbf{q}_{\text{near}}.n_{\text{fail}}$);
      ($\mathbf{q}$'$_{\text{new}}$, P'$^{\text{col}}_{\text{pas}}$) $\leftarrow$ `Expand`($\mathbf{q}_{\text{new}}$, $\mathbf{q}^{\text{pas}}_{\text{rand}}$);
      $P^{\text{col}}_{\text{pas}} \leftarrow P$'$^{\text{col}}_{\text{pas}} \setminus P^{\text{col}}_{\text{pas}}$;
      $\mathbf{q}_{\text{new}} \leftarrow \mathbf{q}$'$_{\text{new}}$;
    if not `TooSimilar`($\mathbf{q}_{\text{near}}$, $\mathbf{q}_{\text{new}}$) **then**
      `AddNewNode`(τ, $\mathbf{q}_{\text{new}}$);
      `AddNewEdge`(τ, $\mathbf{q}_{\text{near}}$, $\mathbf{q}_{\text{new}}$);
      $\mathbf{q}_{\text{near}}.n_{\text{fail}} \leftarrow 0$;
    else $\mathbf{q}_{\text{near}}.n_{\text{fail}} \leftarrow \mathbf{q}_{\text{near}}.n_{\text{fail}} + 1$;
end

iteration, the motion of active parts is computed first. The function `SampleConf` receives as argument the list of active parts P_{act} and samples only the associated parameters $\mathbf{q}^{\text{act}}$. Thus, this function generates a conformation $\mathbf{q}^{\text{act}}_{\text{rand}}$ in a sub-manifold of the conformational space involving the active parameters, $\boldsymbol{C}^{\text{act}}$. The function `NearestNeighbor` selects the node to be expanded $\mathbf{q}_{\text{near}}$ using a distance metric in $\boldsymbol{C}^{\text{act}}$ (i.e. involving the ligand pose and its bond torsions). Then, `Expand` performs the expansion of the selected conformation by only changing the active parameters. The returned conformation $\mathbf{q}_{\text{new}}$ corresponds to the last valid point (i.e. satisfying all the geometric constraints) computed along the straight-line path from $\mathbf{q}_{\text{near}}$ toward $\{\mathbf{q}^{\text{act}}_{\text{rand}}, \mathbf{q}^{\text{pas}}_{\text{near}}\}$. If the expansion succeeds (i.e. the distance from $\mathbf{q}_{\text{near}}$ to $\mathbf{q}_{\text{new}}$ is not negligible), a new node and the corresponding edge are added to the tree. The function `Expand` analyzes the collision pairs yielding the stop of the expansion process. If active parts in P_{act} collide with potentially mobile passive parts in P_{pas}, the list of the involved passive parts $P^{\text{col}}_{\text{pas}}$ is returned. This information is used in the second stage of the algorithm, which generates the motion of passive parts.

The function `PartsToMove` determines the list $P^{\text{mov}}_{\text{pas}}$ of passive parts to be moved at one iteration. This function receives as argument the list of colliding passive parts $P^{\text{col}}_{\text{pas}}$, and constructs a list with all the parts indirectly involved in the collision based on the kinematic diagram of the molecular model. Figure 5 illustrates three typical situations. If the ligand motion is hindered by a side-chain in a secondary structure element (Case 1 in Figure 5), then, the list involves this side-chain and the corresponding group G_i. When the colliding side-chain is on a flexible loop, then the list involves the side-chain, the loop backbone, and the group G_i for an intra-group loop iL_i (Case 2), or the groups G_i and G_{i+1} for an inter-group loop $eL_{i,i+1}$ (Case 3). In all the cases, when a group G_i is involved in $P^{\text{mov}}_{\text{pas}}$, then the backbone of inter-group loops $eL_{i-1,i}$ and $eL_{i,i+1}$ (if any) is also considered into the list, since the conformation of

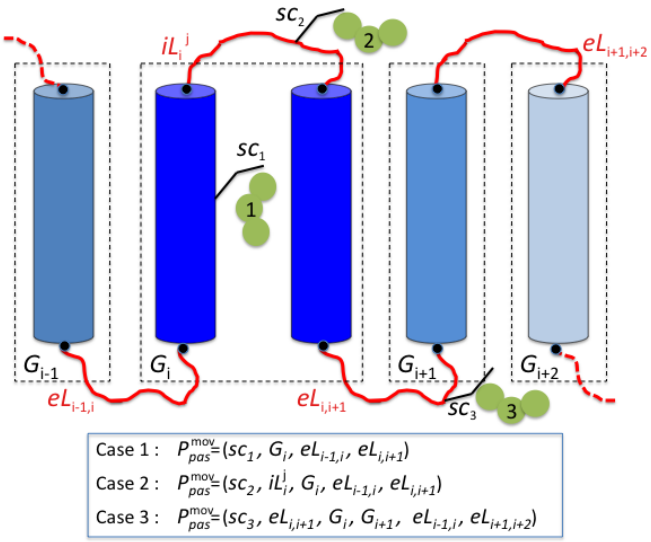

Figure 5: Determination of the the list of passive parts to be moved $P^{\text{mov}}_{\text{pas}}$ based on the contacts with active parts and on the kinematic diagram of the protein model. Three typical cases are illustrated.

these loops needs to be sampled together with the group pose in order to maintain the chain integrity.

The function `PerturbConf` acts on passive parameters. The conformational parameters associated with parts in the list $P^{\text{mov}}_{\text{pas}}$ are sampled with a probability that depends on their mobility coefficient δ, and on the difficulty for expanding $\mathbf{q}_{\text{near}}$, which is estimated by the number of previous expansion failures n_{fail}. A parameter is sampled if the following condition is satisfied:

$$\texttt{NormalRand}(\mu, \sigma^2) \geq 1 - \delta$$

Where `NormalRand` returns a random positive real number sampled from a normal distribution with mean $\mu = 0$ and variance $\sigma^2 = 0.1 \times n_{\text{fail}}$. Such a selection strategy maintains a low probability of moving parts with small mobility coefficient (e.g. protein domains) when the diffusion tree grows easily, while the probability is increased when required to unblock the exploration.

The value of the selected passive parameters is perturbed by randomly sampling in a ball centered at $\mathbf{q}_{\text{near}}$. Then, an attempt is made to further expand $\mathbf{q}_{\text{new}}$ toward $\{\mathbf{q}^{\text{act}}_{\text{new}}, \mathbf{q}^{\text{pas}}_{\text{rand}}\}$. Note that only parts in $P^{\text{mov}}_{\text{pas}}$ associated with the perturbed parameters move during this tree expansion. The function `Expand` returns a list P'$^{\text{col}}_{\text{pas}}$ of blocking parts involved in collisions with moving passive parts. If this list contains new passive parts (not contained in $P^{\text{col}}_{\text{pas}}$), the process generating passive part motions is iterated. Such a possible cascade of passive part motions is needed to solve problems where passive parts indirectly hinder the motion of the active ones because they block other passive parts.

The algorithm is iterated until the problem is solved, or when a `StopCondition` determines that the solution cannot be found. The problem is considered to be solved when a conformation with the ligand outside the protein is reached. Failure is returned if a solution is not found after a given maximum number of iterations. Once the random diffusion tree is constructed, the solution path is simply obtained by tracing back the edges from the goal node ("disassembled" state) to the root node ("assembled" state). Finally, a randomized path smooth-

ing post-processing[1] is performed in the composite space of all the parameters, so that simultaneous motions of the ligand and the protein are obtained in the final path, instead of the alternate motions resulting from the Manhattan-like exploration strategy.

Geometric constraints verification

During the conformational exploration, a set of geometric constraints have to be checked (e.g. collision avoidance, hydrogen/disulfide bond integrity) or reinforced (e.g. loop closure). These constraints are explained below.

Collision avoidance: The main geometric constraint to be verified during the conformational exploration is the avoidance of atom overlaps. The atoms are represented by rigid spheres with a percentage of van der Waals radii. Considering a percentage of the van der Waals equilibrium distance ensures that only energetically infeasible conformations are rejected by the collision checker. The value of 80% is often used in techniques that geometrically check atom overlaps[22]. Collisions are checked between the ligand and the protein, as well as internal collisions between mobile parts of each molecule. The collision test is done inside the function `Expand`, which performs the local expansion motion. Our implementation builds on the efficient BioCD algorithm[23], specially designed for articulated molecular models. BioCD uses hierarchical data structures to approximate the shape of the molecules at successive levels of detail, making the number of atom pairs tested for collision to be significantly reduced.

Loop closure: The functions `SampleConf` and `PerturbConf` perform a specific sampling procedure of loop conformations, taking into account loop closure constraints. Once the pose parameters of all groups G_i have been sampled, the Random Loop Generator (RLG) algorithm[24] is applied to sample the backbone torsions of the N-terminal and C-terminal segments of each loop. This iterative algorithm, based on simple geometric operations, biases the sampling of these chain segments toward conformations with a high probability of satisfying the loop closure constraint. The constraint is reinforced within the function `Expand`, which applies an inverse kinematics method[25] to compute the bond torsions of the tripeptide in the middle loop part (M) for the conformations along the local expansion motion.

Hydrogen bonds and disulfide bonds: These structural constraints can be considered within the mechanistic molecular model. Indeed, they are modeled as distance and angle constraints between the bonded atoms. For hydrogen bonds, the distance d between the donor and the acceptor atoms, and the bond angle θ, must remain within a given range. For instance, for O-H$\cdots$N bonds: $d_{\text{O-N}} \in [2.5\ \text{Å}, 3.8\ \text{Å}]$ and $\theta_{\text{O-H-N}} \in [110°, 180°]$. Disulfide bonds also imply bond length and bond angle constraints between the involved S and C atoms. Additionally, the S-S bond torsion γ is restricted around 90°. The ranges by default are $d_{\text{S-S}} \in [1.8\ \text{Å}, 2.2\ \text{Å}]$, $\theta_{\text{C-S-S}} \in [100°, 130°]$, and $\gamma_{\text{S-S}} \in [60°, 120°]$. All these constrains are checked within the function `Expand`.

[1] The *probabilistic path shortening* method[21] was used for path smoothing.

Results and discussion

This section presents results obtained with the proposed method on two biologically interesting systems involving ligand-induced conformational changes. In the first one, the mechanism of sugar permeation through LacY involves a large-amplitude relative motion of transmembrane domains. In the second system, the access/exit of a ligand to the active site of the β_2-adrenergic receptor is related with side-chain motions, loop motions and transmembrane domain rearrangements. The presented results are not aimed to provide new insights into these biological systems, but to serve as a proof of concept and to show the interest of the proposed approach.

The method was implemented within our software prototype BioMove3D. PyMOL[26] was used for viewing molecular models. The computing times reported below correspond to tests run on a single AMD Opteron 148 processor at 2.6 GHz.

Lactose permease

Lactose permease (LacY) is a transport protein that transduces electrochemical proton gradient into sugar concentration gradient across the cell inner membrane[4]. LacY is composed of two main domains[9]: the N-domain involving helices I-VI, and the C-domain involving helices VII-XII. The two domains are connected by a long loop containing more than 20 residues. For carrying out its function, LacY is supposed to alternate between two conformational states: the inward-open state, where the substrate is accessible from the cytoplasm, and the outward-open state, where the access is possible from the periplasmic side. However, only the structure of the inward-open conformational state of LacY has been solved by X-ray crystallography.

Different approaches have been used to analyze the conformational transition pathway toward the outward-open state. In particular, experimental studies using double electron-electron resonance (DEER)[27] suggest that the conformational transition can be mainly described as a rigid-body rotation of the C-domain and the N-domain. Based on such structural knowledge, the mechanistic model of LacY was simplified by considering a rigid backbone for the C- and N- domains. Flexibility was allocated to the loop between helices VI and VII, and to all the protein side-chains. Thus, the mechanical model contains two main groups G_1 and G_2 associated with the C-domain and the N-domain respectively, and an inter-domain loop $eL_{1,2}$. The X-ray structure of LacY of *Escherichia coli*[9] (PDB ID 1PV7), corresponding to the inward-open conformation, and used as starting point in this work, contains a bound substrate homologue TDG (see Figure 1.a). The substrate molecule was modeled with full flexibility, and it could freely rotate and translate by 50 Å in any direction excepting the direction to the cytoplasm (only 5 Å were permitted in this direction in order to force the exit toward the periplasmic side). Overall, the mechanistic model of LacY-TDG contains 775 degrees of freedom: 12 correspond to the rigid-body motion of the C- and N- domains, 75 to the backbone torsions of the inter-domain loop, 678 to the protein side-chains, and 10 to the substrate mobility and flexibility.

The ML-RRT algorithm was applied to compute the exit pathway of TDG toward the periplasmic side, which involves the conformational transition of LacY. The computing time of

Residue pair	Inward-open	Outward-open (experimental [27])	Outward-open (simulation)
73-401	41 Å	27 Å	36.9(±1.0) Å
73-340	36 Å	21 Å	31.0(±1.3) Å
136-340	34 Å	17 Å	28.7(±1.4) Å
137-340	32 Å	16 Å	26.7(±1.4) Å
136-401	40 Å	24 Å	35.6(±1.3) Å
137-401	38 Å	22 Å	33.5(±1.4) Å
105-310	34 Å	41 Å	38.0(±1.6) Å
164-310	27 Å	43 Å	32.8(±1.4) Å
164-375	33 Å	49 Å	35.8(±1.2) Å

Table 1: Distance variation between residue pairs in LacY.

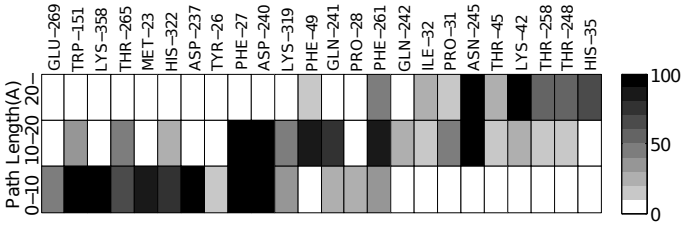

Figure 6: List of residues whose side-chain are encountered by the substrate during its diffusion toward the periplasm. For facilitating interpretation, the pathway is divided into three segments. The grey-scale represents the percentage of times that the contact appears over the set of 10 runs.

a run was about 1 hour on a single processor. Such high computational performance is worth to be noted since it represents an important feature of the proposed approach compared to the very long computing times required by other simulation methods such as molecular dynamics. The algorithm was run 10 times in order to analyze a possible variability of results associated with the randomized exploration procedure. All the runs yielded very similar results with regard to the protein conformational change. The obtained "disassembled" conformation, with the ligand outside the protein and LacY in a outward-open state, is represented in Figure 1.b[2]. As it has been pointed out by prior studies [27], the substrate exit requires the rotation of the two domains. In our results, the observed rotation between the domains is around 20°. Although this is smaller that the 60° suggested by DEER experiments, the overall motion is alike. The comparison of the variation of distances between some residue pairs in the inward- and outward- faces of LacY (see Table 1) shows an approximate overall ratio of 1/3 between the values measured by DEER and our results. The explanation to this quantitative difference is that ML-RRT tends to produce the minimal conformational change required for the molecular disassembly, while larger motions may occur in reality. Interestingly, the distance between residues Ile40 and Asn245 in the outward-open conformation computed by ML-RRT is of approximately 15Å, which has been shown by cross-linking experiments [28] to be the minimal distance between these residue positions for guaranteeing the activity of LacY.

In other recent studies [29], steered molecular dynamics (SMD) simulations have been carried out to better understand the physical mechanisms of lactose permeation at the atomic level. SMD results provide detailed information about the interactions between lactose and LacY residues during permeation. Such kind of information cannot be directly provided by our method, since it does not consider accurate energy functions. However, a straightforward geometric analysis of the paths obtained by ML-RRT can provide the list of residues that the ligand has encountered during its diffusion. The diagram in Figure 6 represents the residues encountered by the ligand along the path toward the periplasm. A contact between the ligand and a residue side-chain was recorded if the distance between the surface of van der Waals spheres modeling their atoms was below 1 Å. The diagram shows the percentage of times that a contact appeared over the set of 10 paths. Contacts were recorded for three segments of the path: the beginning (0-10 Å), where the ligand is close to its location in the crystal structure, the middle part (10-20 Å), and the final part (above 20 Å), where TDG is near the periplasm. Remarkably, all the residues identified by SMD simulations [29] as interacting residues (throught side-chain hydrogen bonds or hydrophobic interactions) appear in the diagram, with the exception of Asp36. Note however that this residue is on the periplasmic surface of the protein. On the other side, only one residue (Thr265) appearing in the contact diagram with a significant percentage is not reported in the referred work. Such an impressive consistency with results of SMD simulations confirms the validity and the potential interest of our approach.

[2]A movie of the computed conformational transition can be seen at www.laas.fr/∼jcortes/tmp/LacY_TDG_pp_exit.mov).

β_2-adrenergic receptor

The β_2-adrenergic receptor (β_2-AR) is a membrane protein belonging to the superfamily of the G-protein-coupled receptors (GPCRs) [30], which activate signal transduction inside the cell in response to the binding of hormones and neurotransmitters in the extracellular region. GPCRs are important therapeutic targets for a large class of diseases. Therefore, numerous studies have been devoted to this family of proteins, aiming to better understand their activation/deactivation mechanism. However, many questions remain. In particular, little is known about the functional role of extracellular loops, and about their possible conformational coupling to ligand binding [31]. One major difficulty comes from the lack of structural information inherent to membrane proteins.

A high-resolution crystal structure of β_2-AR has been recently obtained [32]. The crystal structure also contains a molecule of carazolol, a partial inverse agonist, in the protein active site. This receptor-ligand structure is the starting point of the conformational analysis presented below. The structure is represented in Figure 7, using standard notation for the structural elements. Like all GPCRs, β_2-AR contains seven transmembrane helices, which were modeled as rigid groups G_i. The intracellular and extracellular loops were modeled as flexible elements $eL_{i,i+1}$. All the side-chains and the ligand were considered to be fully flexible. The number of degrees of freedom of the whole model is 703: 42 of them correspond to the rigid-body motion of the seven transmembrane helices, 159 to the backbone torsions of the five loops, 490 to the protein side-chains, and 12 to the ligand mobility and flexibility.

The ML-RRT algorithm was applied to compute the exit pathway of carazolol from the active site of β_2-AR. A first set of 10 runs revealed some variability on the trajectories followed by the ligand. Thus, the algorithm was run 60 times in order to do a more accurate statistical analysis of results. The 60

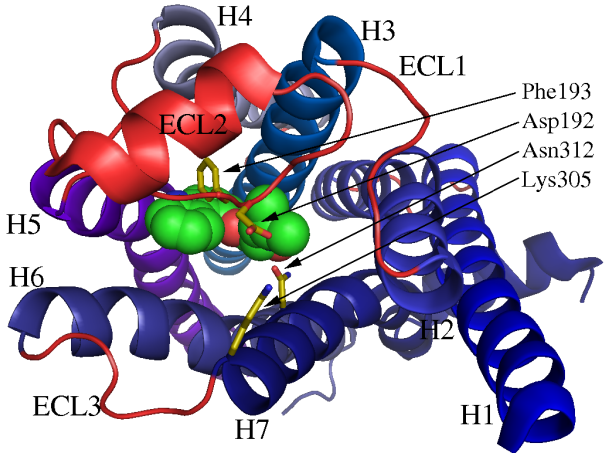

Figure 7: Structure of β_2-AR with carazolol bound in the protein active site viewed from the extracellular side. The secondary structure elements and important residues are displayed on the image.

paths were obtained in 2 hours of computing time (each run takes an average of 2 minutes on a single processor). These exit paths can be divided into two main clusters. In one class of paths, which we refer to as "left-hand" paths, carazolol exits between transmembrane helices H5, H6 and H7. In the other class, called "right-hand" paths, the ligand exits between H2, H3 and H7. The two clusters can be separated by an axis traced between residues Asp192 and Lys305, which form a salt bridge in the crystal structure. Figure 8 shows snapshots of the ligand exit path for each path class[3]. Interestingly, these two classes of exit paths have also been observed in prior studies[33] based on random acceleration molecular dynamics (RAMD) simulations. A quantitative comparison can be done between results obtained with ML-RRT and RAMD. The most significant comparable result is that both approaches suggest that left-hand and right-hand exit paths are approximately equiprobable. Indeed, 31/60 of the ML-RRT solutions correspond to left-hand, and 29/60 to right-hand paths. Another result from RAMD simulations concerns the recurrent breakage of the salt bridge Asp192-Lys305 during ligand exit. Paths computed with ML-RRT show a significant motion of the side-chains of these two residues, which lead to the salt bridge breakage for most of the 60 paths. However, in some of the left-hand paths, the ligand exits with only a slight perturbation in the conformation of Asp192 and Lys305. The interpretation is that it is geometrically possible for the ligand to exit between helices H5, H6 and H7 without breaking the salt bridge.

A further comparison between left-hand and right-hand paths obtained with ML-RRT displays other interesting differences. The first one concerns the orientation of the ligand. In most of the left-hand paths, the ring head of carazolol reaches first the protein surface (see Figure 8.a). Contrarily, the ring and the alkylamine-alcohol tail exit almost simultaneously in most of the right-hand paths (Figure 8.b). A possible interpretation may be that one of the pathways could be preferred for the exit of the ligand, while the other could be more suited to the access. A more accurate analysis of the paths computed by ML-RRT would be required to reinforce such a suggestion. Note however that RAMD simulations from a putative ligand-free model

[3]Movies of these paths can be seen at www.laas.fr/∼jcortes/tmp/beta2AR_carazolol_exit_L.mov and www.laas.fr/∼jcortes/tmp/beta2AR_carazolol_exit_R.mov.

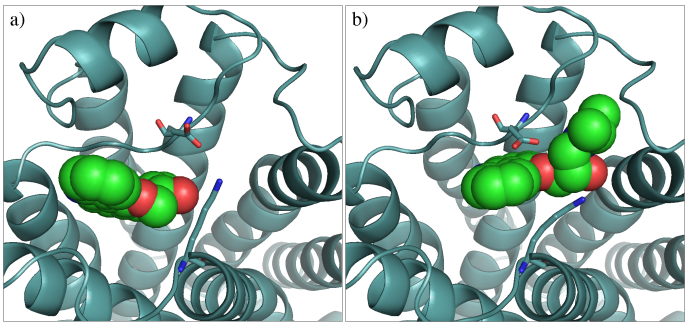

Figure 8: Snapshots of the ligand exit from β_2-AR following the left-hand pathway (a), and the right-hand pathway (b).

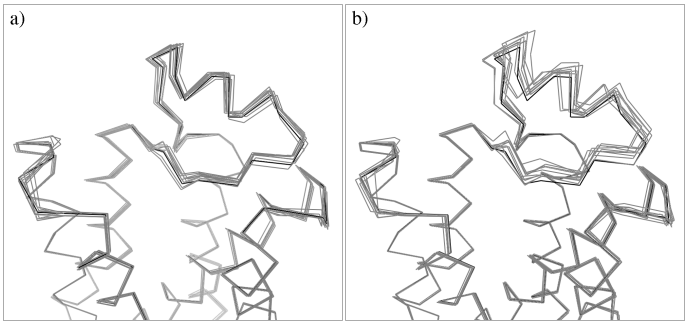

Figure 9: Superposition of the initial structure of β_2-AR (black) and conformations induced by the ligand exit (grey) following the left-hand pathway (a), and the right-hand pathway (b).

of β_2-AR[33] suggest that carazolol enters the receptor between helices H2, H3 and H7, with its ring head diving first.

Another interesting difference between the two classes of exit paths concerns the conformational changes of the extracellular loop ECL2 induced by the ligand exit. As shown in Figure 9, right-hand paths imply, in average, a more significant motion of ECL2 than left-hand paths. Note that although the loop ECL2 of β_2-AR is very long, its conformation is constrained by two disulfide bonds, one between residues in the loop (Cys184-Cys190), and one between the loop and H3 (Cys106-Cys191). Thus, in any case, this loop cannot undergo large conformational changes. The observed relationship between right-hand paths and ECL2 flexibility has been confirmed by tests performed on a model of β_2-AR only considering side-chain flexibility. Using this rigid-backbone model, the ligand exited through the left-hand pathway in 90% of the ML-RRT runs. These results suggest that right-hand access/exit paths involve a more important interaction between the ligand and ECL2 than left-hand paths. Note that recent studies on GPCRs show important roles of ECL2. Indeed, it can be required for ligand binding[34], and its motion can be involved in the activation mechanism[35].

The analysis of contacts between carazolol and β_2-AR residues along the set of 60 exit pathways computed with ML-RRT was performed using the technique described above for the study of LacY. Figure 10 shows the list of residues whose side-chains contacted the ligand. For clarity reasons, the figure only reports contacts that appeared in more than 30% of the paths. Four residues are clearly highlighted in the diagram: Asp192, Phe193, Lys305, and Asn312. The positions of these residues are indicated in Figure 7. Two of them, Asp192 and Lys305, form the aforementioned salt bridge, which is bro-

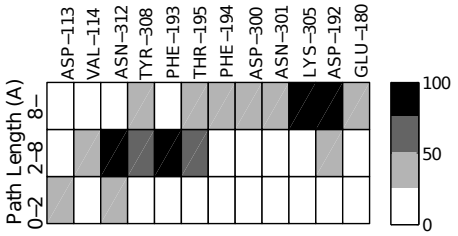

Figure 10: List of residues whose side-chain are encountered by the ligand along the exit pathway. For facilitating interpretation, the pathway is divided into three segments. The grey-scale represents the percentage of times that the contact appears over the set of 60 runs. Only contacts that appeared in more than 30% of the paths are displayed.

ken during the ligand diffusion. Phe193, which is located on ECL2, has also been identified as an important residue in related works. Results of RAMD simulations[33] suggest that this aromatic residue may participate to the ligand entry and stabilization in the active site of β_2-AR. Recent NMR experiments[31] have shown that inverse agonists induce a conformational change of this residue. Finally, Asn312 is an important residue for the stabilization of carazolol in the active site through a polar interaction with its alkylamine-alcohol tail.

Overall, the presented results show that structural information on the access/exit of carazolol to the active site of β_2-AR provided by ML-RRT is in agreement with results of other experimental and computational studies.

Conclusion

The results in this paper show that a mechanistic approach to molecular simulations may lead to the development of efficient computational methods, able to provide relevant information on the interaction of biological molecules. The proposed algorithm, ML-RRT, is a novel and fast conformational search method for simulating ligand diffusion inside flexible models of proteins. Indeed, ML-RRT generates long (20-30 Å) diffusion paths within tens of minutes of computing time on a single processor, which is remarkably short compared to the time required by MD-based methods. Such a high computational performance is achieved thanks to the efficiency of the conformational search method that operates on geometric models of molecules. Geometrically feasible paths are a reasonably good approximation that provides itself very useful information. Furthermore, as shown in prior work[13], the approximate solution path can also be efficiently refined with standard molecular modeling tools (e.g. energy minimization) in order to perform a more accurate energetic analysis. As future work, we intend to further improve the method to better deal with full molecular flexibility during protein-ligand interactions. We also expect to extend the method for its application to the modeling of protein-protein interactions.

Acknowledgments

This work has been partially supported by the French National Agency for Research (ANR) under project GlucoDesign, and by the Région Midi-Pyrénées under project Amylo.

References

[1] H. Frauenfelder, S. G. Sligar and P. G. Wolynes, *Science*, 1991, **254**, 1598–1603.

[2] M. F. Lensink and R. Méndez, *Curr. Pharm. Biotechnol.*, 2008, **9**, 77–86.

[3] H. A. Carlson, *Curr. Pharm. Des.*, 2002, **8**, 1571–1578.

[4] H. R. Kaback, M. Sahin-Tóth and A. B. Weinglass, *Nature Rev. Moll. Cell. Biol.*, 2001, **2(8)**, 610–620.

[5] G. Katona, P. Carpentier, V. N. , P. Amara, V. Adam, J. Ohana, N. Tsanov and D. Bourgeois, *Science*, 2007, **316**, 449–453.

[6] P. Schanda, V. Forge and B. Brutscher, *PNAS*, 2007, **104**, 11257–11262.

[7] S. Izrailev, S. Stepaniants, B. Isralewitz, D. Kosztin, H. Lu, F. Molnar, W. Wriggers and K. Schulten, *Computational Molecular Dynamics: Challenges, Methods, Ideas. Vol. 4 of Lecture Notes in Computational Science and Engineering*, Springer-Verlag, Berlin, 1998, pp. 39–65.

[8] S. K. Ludemann, V. Lounnas and R. C. Wade, *J. Mol. Biol.*, 2000, **303**, 797–811.

[9] J. Abramson, I. Smirnova, V. Kasho, G. Verner, H. R. Kaback and S. Iwata, *Science*, 2003, **301**, 610–615.

[10] S. M. LaValle, *Planning Algorithms*, Cambridge University Press, New York, 2006.

[11] E. Ferré and J.-P. Laumond, *Proc. IEEE Int. Conf. Robot. Automat.*, 2004, 3149–3154.

[12] D. T. Le, J. Cortés and T. Siméon, *Proc. IEEE Int. Conf. Automat. Sci. Eng.*, 2009.

[13] J. Cortés, T. Siméon, V. Ruiz, D. Guieysse, M. Remaud and V. Tran, *Bioinformatics*, 2005, **21**, i116–i125.

[14] A. Enosh, S. J. Fleishman, N. Ben-Tal and D. Halperin, *Bioinformatics*, 2007, **23**, e212–e218.

[15] S. Kirillova, J. Cortés, A. Stefaniu and T. Siméon, *Proteins*, 2008, **70**, 131–143.

[16] J. Cortés, L. Jaillet and T. Siméon, *IEEE Transactions on Robotics*, 2008, **24**, 475–481.

[17] D. Guieysse, J. Cortés, S. Puech-Guenot, S. Barbe, V. Lafaquière, P. Monsan, T. Siméon, I. André and M. Remaud-Siméon, *ChemBioChem*, 2008, **9**, 1308–1317.

[18] V. Lafaquière, S. Barbe, S. Puech-Guenot, D. Guieysse, J. Cortés, P. Monsan, T. Siméon, I. André and M. Remaud-Siméon, *ChemBioChem*, 2009, **10**, 2760–2771.

[19] S. M. LaValle and J. J. Kuffner, *Algorithmic and Computational Robotics: New Directions (WAFR2000)*, A.K. Peters, Boston, 2001, pp. 293–308.

[20] S. Wells, S. Menor, B. Hespenheide and M. F. Thorpe, *Phys. Biol.*, 2005, **2**, 127–136.

[21] S. Sekhavat, P. Svestka, J.-P. Laumond and M. H. Overmars, *Int. J. Robot. Res.*, 1998, **17(8)**, 840–857.

[22] M. A. DePristo, P. I. W. de Bakker, S. C. Lovell and T. L. Blundell, *Proteins*, 2003, **51**, 41–55.

[23] V. Ruiz de Angulo, J. Cortés and T. Siméon, *Robotics: Science and Systems*, MIT Press, Cambridge, 2005, pp. 6–11.

[24] J. Cortés, T. Siméon, M. Remaud-Siméon and V. Tran, *J. Comput. Chem.*, 2004, **25(7)**, 956–967.

[25] M. Renaud, *Current Advances in Mechanical Design and Production VII*, Pergamon, New York, 2000, pp. 57–66.

[26] W. L. DeLano, http://www.pymol.org.

[27] I. Smirnova, V. Kasho, J.-Y. Choe, C. Altenbach, W. L. Hubbell and H. R. Kaback, *Proc. Natl. Acad. Sci. USA*, 2007, **104**, 16504–16509.

[28] Y. Zhou, L. Guan, J. A. Freites and H. R. Kaback, *Proc. Natl. Acad. Sci. USA*, 2008, **105**, 3774–3778.

[29] M. Ø. Jensen, Y. Yin, E. Tajkhorshid and K. Schulten, *Biophys J.*, 200, **93**, 92–102.

[30] G. Vauquelin and B. von Mentzer, *G Protein-Coupled Receptors: Molecular Pharmacology from Academic Concept to Pharmaceutical Research*, John Wiley & sons Ltd, Chichester, 2007.

[31] M. P. Bokoch, Y. Zou, S. G. F. Rasmussen, C. W. Liu, R. Nygaard, D. M. Rosenbaum, J. J. Fung, H.-J. Choi, F. S. Thian, T. S. Kobilka, J. D. Puglisi, W. I. Weis, L. Pardo, R. S. Prosser, L. Mueller and B. K. Kobilka, *Nature*, 2010, **463**, 108–112.

[32] V. Cherezov, D. M. Rosenbaum, M. A. Hanson, S. G. Rasmussen, F. S. Thian, T. S. Kobilka, H. J. Choi, P. Kuhn, W. I. Weis, B. K. Kobilka and R. C. Stevens, *Science*, 2007, **318**, 1258–1265.

[33] T. Wang and Y. Duan, *J. Mol. Biol.*, 2009, **392**, 1102–1115.

[34] V. A. Avlani, K. J. Gregory, C. J. Morton, M. W. Parker, P. M. Sexton and A. Christopoulos, *J. Biol. Chem.*, 2007, **282**, 25677–25686.

[35] S. Ahuja, V. Hornak, E. C. Yan, N. Syrett, J. A. Goncalves, A. Hirshfeld, M. Ziliox, T. P. Sakmar, M. Sheves, P. J. Reeves, S. O. Smith and M. Eilers, *Nat. Struct. Mol. Biol.*, 2009, **16**, 168–175.