

RESEARCH ARTICLE

An Integrated Framework Advancing Membrane Protein Modeling and Design

Rebecca F. Alford^{1,2}, Julia Koehler Leman¹, Brian D. Weitzner¹, Amanda M. Duran³, Drew C. Tilley⁴, Assaf Elazar⁵, Jeffrey J. Gray^{1*}

1 Department of Chemical and Biomolecular Engineering, Johns Hopkins University, Baltimore, Maryland, United States of America, **2** Department of Chemistry, Carnegie Mellon University, Pittsburgh, Pennsylvania, United States of America, **3** Center for Structural Biology, Department of Chemistry, Vanderbilt University, Nashville, Tennessee, United States of America, **4** Department of Physiology and Membrane Biology, University of California, Davis, Davis, California, United States of America, **5** Department of Biological Chemistry, Weizmann Institute of Science, Rehovot, Israel

☞ These authors contributed equally to this work.

* jgray@jhu.edu



OPEN ACCESS

Citation: Alford RF, Koehler Leman J, Weitzner BD, Duran AM, Tilley DC, Elazar A, et al. (2015) An Integrated Framework Advancing Membrane Protein Modeling and Design. *PLoS Comput Biol* 11(9): e1004398. doi:10.1371/journal.pcbi.1004398

Editor: Dennis R Livesay, UNC Charlotte, UNITED STATES

Received: February 20, 2015

Accepted: June 9, 2015

Published: September 1, 2015

Copyright: © 2015 Alford et al. This is an open access article distributed under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Data Availability Statement: The algorithm is incorporated into the Rosetta software package and is available for download at <https://www.rosettacommons.org>. The protocol was independently tested by members of the RosettaCommons and all protocols are made available in the Rosetta/demos/protocol_capture directory of the current Rosetta release.

Funding: Funding was provided by NIH R01 GM-078221 to RFA, BDW, and JGG, and Rosetta Commons to JKL. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Abstract

Membrane proteins are critical functional molecules in the human body, constituting more than 30% of open reading frames in the human genome. Unfortunately, a myriad of difficulties in overexpression and reconstitution into membrane mimetics severely limit our ability to determine their structures. Computational tools are therefore instrumental to membrane protein structure prediction, consequently increasing our understanding of membrane protein function and their role in disease. Here, we describe a general framework facilitating membrane protein modeling and design that combines the scientific principles for membrane protein modeling with the flexible software architecture of Rosetta3. This new framework, called RosettaMP, provides a general membrane representation that interfaces with scoring, conformational sampling, and mutation routines that can be easily combined to create new protocols. To demonstrate the capabilities of this implementation, we developed four proof-of-concept applications for (1) prediction of free energy changes upon mutation; (2) high-resolution structural refinement; (3) protein-protein docking; and (4) assembly of symmetric protein complexes, all in the membrane environment. Preliminary data show that these algorithms can produce meaningful scores and structures. The data also suggest needed improvements to both sampling routines and score functions. Importantly, the applications collectively demonstrate the potential of combining the flexible nature of RosettaMP with the power of Rosetta algorithms to facilitate membrane protein modeling and design.

Author Summary

Over 30% of the human proteome consists of proteins embedded in biological membranes. These proteins are critical in many processes such as transport of materials in and out of the cell and transmitting signals to other cells in the body. They are implicated in a large number of diseases; in fact, they are targeted by over 50% of pharmaceutical drugs on the

Competing Interests: The authors have declared that no competing interests exist.

market. Since the membrane environment makes experimental structure determination extremely difficult, there is a need for alternative, computational approaches. Here, we describe a new framework, RosettaMP, for computational modeling and design of membrane protein structures, integrated in the Rosetta3 software suite. This framework includes a set of tools for representing the membrane bilayer, moving the protein, altering its sequence, and estimating free energies. We demonstrate tools to predict the effects of mutations, refine atomic details of protein structures, simulate protein binding, and assemble symmetric complexes, all in the membrane bilayer. Taken together, these applications demonstrate the potential of RosettaMP to facilitate membrane protein structure prediction and design, enabling us to understand the function of these proteins and their role in human disease.

This is a *PLOS Computational Biology Methods* paper.

Introduction

Membrane proteins are critical participants in a wide variety of biological processes including cell adhesion, signaling, transport, and enzymatic activity [1]. They comprise more than 30% of open reading frames [2] and are targeted by over half of currently available pharmaceutical drugs [3,4]. Despite their importance, our knowledge of membrane protein structure and function remains severely limited, as shown by a constant 1–2% representation of structures in the Protein Data Bank [5] over the past decade [6]. The paucity of experimentally determined structures can be attributed to wide-ranging challenges in overexpression, reconstitution into membrane mimetics, and ultimately structure determination by various methods [7]. Due to these experimental challenges, computational approaches assume a pivotal role in advancing our understanding of membrane protein structure and function.

Compared to modeling soluble proteins, membrane protein modeling has the advantage of constraining the conformational search space into the two dimensions of the membrane bilayer, which imposes structural constraints onto the protein. Whereas soluble proteins exhibit enormous structural diversity, the structural motifs in the membrane environment are either α -helical bundles or β -barrels. Since these folds are formed by secondary structure elements adopting preferred orientations in the ordered environment of the lipid bilayer, the use of adapted sampling techniques could substantially increase conformational sampling efficiency. A higher sampling efficiency is required because membrane proteins are typically much larger in size, offsetting the reduction in conformational search space.

Additionally, computational methods for membrane protein modeling require reliable free energy calculations or score functions to distinguish native-like from non-native conformations. Therefore, an accurate representation of the heterogeneous environment of the lipid bilayer is needed. The membrane bilayer can be represented implicitly by using a layered, continuum solvation model, which is computationally inexpensive but unable to describe membrane fluctuations or specific membrane protein–lipid interactions. An additional challenge for the score function is that the precise location of the lipid bilayer surrounding the protein in experimental structures is unknown because the membrane mimetic evades experimental observation.

The aforementioned challenges and the lack of experimental structures have delayed the development and therefore availability of high-quality computational methods for membrane protein modeling, compared to available methods for soluble protein modeling. Whereas soluble protein modeling increasingly focuses on high-resolution structural features as in docking, design and ligand docking applications, methods for membrane protein modeling still mainly focus on obtaining models for unknown protein structures.

Four main techniques for computational modeling of membrane proteins are available: (1) Since template structures for homology modeling are unavailable for many membrane proteins of interest, ***ab initio* modeling** is an important technique (e.g. using BCL::MPFold [8–15]). *Ab initio* structure prediction is one of the most difficult of the modeling tasks, yet it also has the largest benefits because of its ability to predict novel folds. Additionally, in contrast to homology modeling where the final model can contain artifacts from the template, models from *ab initio* structure prediction are not biased by previously determined protein structures. (2) For low (~25%) to very low (~5%) sequence similarities to a known structure, **fold recognition** techniques generate a low-resolution protein model; the accuracy of these models rarely achieves better than 3–4 Å RMSD. (3) **Homology modeling** can be used to model the three-dimensional structure of a query protein if the sequence similarity between the query sequence and the sequence of a template structure is greater than ~30%. The recent increase in determined membrane protein structures (and therefore template availability) has elevated the quality and number of built homology models. Recently, GPCR homology models with an RMSD as low as 2.9 Å from the target structure were created from starting templates with a sequence identity as low as 15% [16]. (4) If the structure of the membrane protein is known, **molecular dynamics (MD) simulations** can follow time trajectories of proteins and lipids in full-atom representation with physics-based energy functions to investigate high-resolution phenomena such as ion channel gating or transport across the membrane [17–19]. With the recent increase in available membrane protein structures, high-resolution modeling methods including **protein design** have started to emerge [20–25]. Two notable achievements include a helix–helix interface design [21] and a design of a four-helix bundle that selectively transports metal ions across the membrane [20].

A limitation of many membrane protein modeling tools is high specialization to accomplish a single task; thus these methods are not easily combined with other modeling tools. The membrane protein community would benefit from an integrated tool that is able to carry out a variety of complex modeling tasks such as loop modeling, predicting the effects of mutations, design, docking, symmetric complex assembly, and ligand docking, in addition to *ab initio* structure prediction, homology modeling, and high-resolution refinement. Additionally, integrated methods would enable testing of a score function in multiple contexts to more rapidly converge on a universal score function in the bilayer environment. The Rosetta software suite offers an integrated toolset for biomolecular modeling, docking, and design, including a broadly tested and refined score function for soluble biomolecules. Moreover, Rosetta has two pioneering membrane protein modeling applications, RosettaMembrane *ab initio* [26] and relax [23].

The RosettaMembrane *ab initio* protocol [23,26] was one of the first methods for *ab initio* structure prediction of membrane proteins. It combines Rosetta's *ab initio* structure prediction protocol for soluble proteins [27] with a low-resolution score function derived from a database of structures of membrane proteins [26,28,29]. This method was later updated to include a high-resolution refinement stage [23,30] that uses an all-atom score function based on the Lazaridis implicit Gaussian-exclusion solvation model for atoms in the membrane [31]. Recently, RosettaMembrane was also used to model transmembrane helical proteins from distant homologues [16].

Since the creation of the RosettaMembrane *ab initio* protocol in 2006, Rosetta has been reorganized into a set of object-oriented libraries (“Rosetta3”) while RosettaMembrane remained in its original implementation. Rosetta3 is now a cohesive, flexible software suite that includes separate objects for conformation and scoring, interaction graphs, score functions organized by multi-body dependencies, kinematics managed through a fold tree, maps to identify flexible portions of the molecule(s), job distribution, and scripting interfaces [32]. Continuous improvements and additions to Rosetta’s extensive library of complex tools motivate its use for membrane protein modeling. However, the scientific concepts of the original RosettaMembrane require integration and generalization to be compatible with the object-oriented architecture of Rosetta3.

Here, we present a new framework, called RosettaMP, integrated in the Rosetta software suite, which enables the development of novel protocols for membrane protein modeling and design. We describe the new, central building blocks to represent the membrane bilayer, and to sample and score both conformations and sequences. We used RosettaMP to create four proof-of-concept applications: (1) prediction of free energy changes upon mutation, (2) high-resolution structural refinement, (3) protein—protein docking, and (4) assembly of symmetric complexes, all in the membrane bilayer. The protocols can be accessed via command line, PyRosetta [33], and RosettaScripts [34], with various levels of customizability for both developers and users. Using a set of test cases, we are able to obtain information on the applicability of the existing score function in these wider contexts. Collectively, the applications demonstrate how RosettaMP and existing Rosetta protocols can be combined to quickly create powerful new methods to answer a broad range of scientific questions. Because of its ability to interoperate with existing Rosetta code and reusable representation of the lipid bilayer, RosettaMP substantially lowers the barrier in complexity for the development of new protocols to model and design membrane proteins, opening the door to many new, critically needed methods.

Methods

Design of RosettaMP

RosettaMP extends the object-oriented architecture of Rosetta3. Our goal was to create a flexible software architecture for modeling membrane proteins; we therefore used object-oriented design principles [35] to encapsulate individual scientific concepts for membrane protein modeling into a set of software ‘building blocks’, or objects. Each object includes information (data) and actions (methods) required to represent a given scientific concept. Together, well-designed objects interact within the larger infrastructure to perform specific tasks.

A Rosetta simulation is centered on one or more biomolecules (protein, protein-ligand complex, nucleic acids, etc.), which are stored in an object called a *Pose*. Sampling in the conformational search space is accomplished by manipulating the *Pose* by *Movers* and evaluating the resulting conformations using a *ScoreFunction* [32]. We expanded this infrastructure by creating new objects to represent the membrane environment and enable scoring and sampling routines that account for the lipid bilayer (Fig 1). Data that describes the membrane, such as membrane location and thickness, is stored in an object encapsulated in the *Pose*. Membrane-specific *Movers* sample the conformational search space exploiting the implicit constraints imposed by the membrane bilayer. Membrane-specific *ScoreFunction* terms use the geometrical information provided by the *Pose*. This code design separates conformation and score evaluation, and preserves the score models of the original membrane score functions.

Membrane data is stored centrally in the *Pose*. To extend Rosetta’s description of a biomolecule to include the lipid bilayer, we created a class called *MembraneInfo* to store all information necessary for representing a membrane *Pose*. *MembraneInfo* attaches a virtual

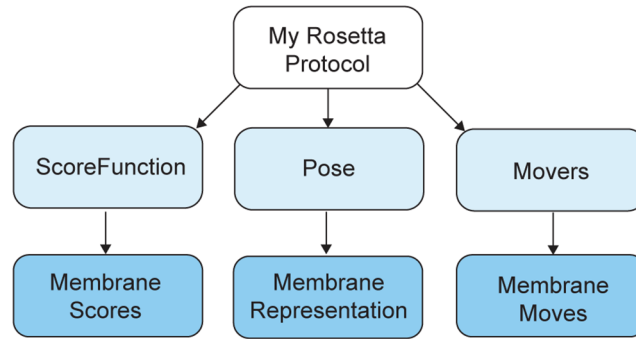


Fig 1. RosettaMP directly extends the architecture of Rosetta3. Every Rosetta protocol requires at least these three main objects for modeling or design tasks (light blue): a *Pose* to represent a biomolecule, a *ScoreFunction* to rank modeled structures and sequences, and *Movers* to sample new conformations of the *Pose*. RosettaMP directly extends this architecture (blue) by adding an element to the *Pose* representing the membrane bilayer, restructuring the original membrane *ScoreFunction* to rely on this membrane representation, and implementing a new set of *Movers* to sample the conformational search space available in the membrane bilayer.

doi:10.1371/journal.pcbi.1004398.g001

residue to the *Pose* (see below and Fig B in [S1 File](#)) that represents the membrane bilayer chemistry and geometry such as its center, normal and thickness. *MembraneInfo* also stores descriptors of sequence- and structure-based membrane protein properties, such as transmembrane spans. *MembraneInfo* is a member of the *Conformation* object, which is part of the *Pose* ([Fig 2](#) and Fig A and Table A in [S1 File](#)). Because the *Pose* is the central object in

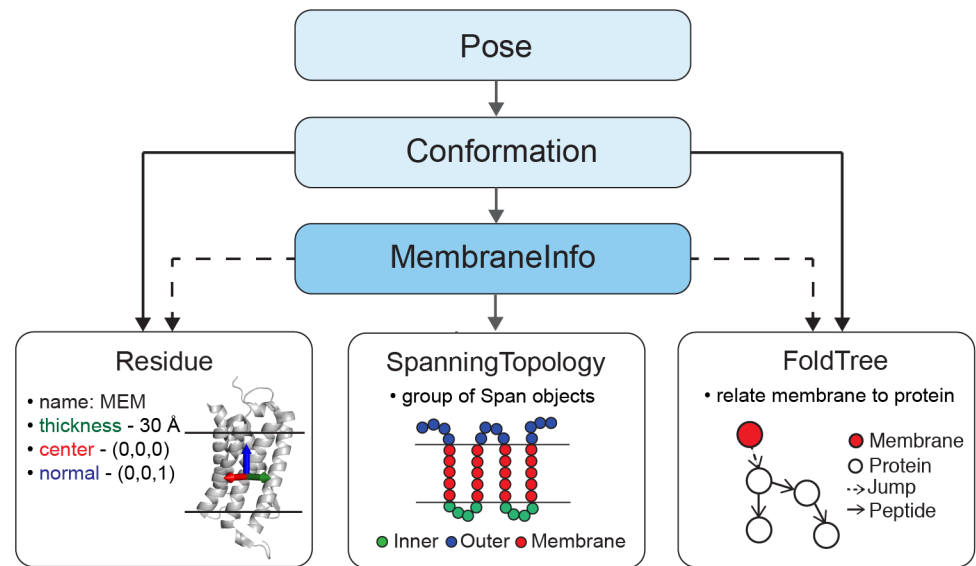


Fig 2. Detailed architecture of RosettaMP. RosettaMP represents the membrane bilayer using three main components connected to a central *MembraneInfo* object (blue). *MembraneInfo* stores information needed to represent the membrane (line arrows) and tracks information present in the *Pose* (dotted arrows). A special *Residue* type is added to the *Pose*, describing the geometry of the membrane bilayer by coordinates storing the center, normal and thickness of the bilayer. A *SpanningTopology* object describes the transmembrane regions of the *Pose*. The *FoldTree* uses a jump edge to establish the connection between the membrane residue and the protein. *MembraneInfo* is also a central repository for membrane-related features such as lipid accessibility of each residue (*LipidAccInfo*). A full Universal Markup Language (UML) diagram is presented in Fig A in [S1 File](#).

doi:10.1371/journal.pcbi.1004398.g002

Rosetta protocols, the information in `MembraneInfo` is readily accessible through the `Pose`, enabling access to the membrane representation in any Rosetta protocol.

Membrane position and thickness is stored in a `Residue` object. Molecules in Rosetta are represented as a set of `Residue` objects. For proteins or nucleic acids, the data structure corresponds to the chemical residue (e.g., an alanine or cytosine), but in Rosetta a `Residue` can be used more generally to encapsulate a set of related coordinates. Thus, to represent the membrane ‘conformation,’ we use a ‘`Residue`’ object with three ‘virtual’ atoms that define the membrane center, normal vector, and thickness. As a `Residue`, the membrane can be fixed or moveable during modeling using the same machinery as with the biomolecular residues. The default membrane thickness is 30 Å, including both the hydrophobic membrane core and the membrane-water interfacial regions.

Connection between the protein and membrane is established through the `FoldTree`. To allow either the membrane or the protein to be fixed or moveable in the coordinate frame, we use Rosetta’s `FoldTree` object. As described previously [36], the `FoldTree` is a rooted, directed, acyclic graph that describes the chemical and geometric connectivity of the `Residues` in the `Pose`. Since Rosetta uses an internal coordinate system, atomic coordinates are described relative to each other by using distances, angles, and dihedral angles for covalently bonded atoms, or rigid body transformations for non-covalently-bonded atoms. The `FoldTree` defines the order of residues (i.e. connections) in which the coordinate update takes place after a conformational sampling step. Coordinates are updated starting at a *root* and propagate along the connections in the `FoldTree`, called `Edges`. These connections can be peptide bonds, called *peptide edges*, or non-bonding connections, called *jump edges*. Jump edges in the `FoldTree` therefore maintain long-range connections, such as interactions between a protein and a ligand, positions of anchor residues for loop building, or relative positions of domains during the assembly of multi-domain proteins.

In RosettaMP, we use a jump edge in the `FoldTree` to connect the membrane residue to the protein. Adjusting the position of the root in the `FoldTree` will subsequently invert jump and peptide edges (i.e. coordinate update takes place in the opposite direction) and therefore changes which parts of the pose are fixed or movable. For instance, if the membrane residue is at the root of the `FoldTree`, it will stay fixed, whereas the protein will move in the coordinate system of the membrane. In contrast, if the root is a residue in the protein, this residue will stay fixed and the membrane residue will move in the coordinate system of the protein.

Membrane protein topology is stored in the `SpanningTopology` object. *Ab initio* structure prediction for membrane proteins often begins with an estimate of the number and location of the transmembrane spans of secondary structure. Transmembrane spans can be predicted from the protein sequence using servers such as OCTOPUS [37] or BCL::Jufo9D [38]. In RosettaMP, the start and end residue numbers of a single transmembrane span are stored in a `Span` object. All `Span` objects for the protein are gathered in a `SpanningTopology` container. If an experimental protein structure is available, the `span_from_pdb` application (S7 File) can be used to create a span file for RosettaMP protocols. The protein structure must be transformed into the membrane coordinate frame, e.g. by using the PDBTM [39] or OPM [40] databases or the TMDet [41] or PPM [42] servers.

RosettaMembrane score functions were restructured to use the new framework. The original implementation of RosettaMembrane includes a low-resolution and high-resolution score function [23,26,43]. We restructured both score functions to follow Rosetta3’s decomposition of energies into one-body, two-body, and whole protein terms [32]. The new implementation uses the membrane position defined in the membrane residue to score per-residue and residue pair interactions within the hydrophobic layers. The scientific integrity of the restructured membrane score function code was verified using continuous regression testing against

Table 1. Rosetta membrane energy terms used by RosettaMP.

Term	Resolution	Scope ¹	Description	Ref
mp_env	low	1b, z, cd	Knowledge-based potential describing propensity for a single residue to be at a given depth in the membrane and burial by residues	[26]
mp_pair	low	2b, z	Knowledge-based pairwise interaction potential between two residues some distance apart at a given depth in the membrane	[26]
mp_cbeta	low	1b, cd	Knowledge-based residue density potential based on number of neighbor residues and conditional upon number of transmembrane helices	[26,28]
mp_lipo	low	1b	Scores agreement between predicted lipophilicity (from LIPS server) and the model	[29]
mp_nonhelix	low	1b, z	Penalty for non-helical secondary structure in the membrane	[26]
mp_termini	low	1b, z	Penalty for residues outside of the hydrophobic layer of the membrane	[26]
mp_tmproj	low	ws	Penalty for transmembrane helices that project outside of the membrane	[26]
fa_mpenv	high	1b, z	Free energy of a single, isolated atom in solvent or lipid, depending on the depth in the membrane	[23,31]
fa_mpsolv	high	2b, z, cd	Atomic solvation free energy change due to the presence of surrounding atoms, modeled via Gaussian exclusion	[23,31]
fa_mpenv_smooth	high	1b, z, cd	Knowledge-based potential describing propensity for a single atom to be at a given depth in the membrane	[43]
hbond	high	2b, z	Depth-adjusted Rosetta hydrogen bonding term with stronger hydrogen bonding in the membrane	[23,30]

¹ Scope of individual energy terms. 1b indicates a per-residue or per-atom score (one-body), 2b indicates a two-body score, z indicates the score is dependent upon depth in the membrane bilayer, cd indicates the score depends on context (typically the number of surrounding residues), and ws is a score based on the whole structure.

doi:10.1371/journal.pcbi.1004398.t001

the original implementation. The new implementation is now compatible with a fixed or a movable membrane and enables facile adjustment of the score functions. The score function terms are explained in [Table 1](#) with formulas in Table B in [S1 File](#) and weights in Tables C-D in [S1 File](#).

Specific Movers invoke RosettaMP and position the protein in the membrane. In Rosetta3 protocols, `Movers` are used to manipulate the `Pose` by conformation or sequence sampling. In RosettaMP, we use `Movers` to setup required membrane information and manipulate conformations in accordance with the implicit constraints the membrane imposes on the structure of these proteins.

The Mover that invokes RosettaMP and instantiates a membrane `Pose` is the **AddMembraneMover**. The `AddMembraneMover` adds the membrane residue to the `Pose`, sets up a default `FoldTree`, and initializes `MembraneInfo`. The position of the root in the `FoldTree` for either a fixed membrane/movable protein or movable membrane/fixed protein is determined by the specific protocol.

The **SetMembranePositionMover** sets the center and normal of the membrane representation to pre-computed values for a fixed protein. These values can be computed from PPM [42] or TMDet [41], two servers that transform membrane protein structures into membrane coordinates with the membrane center at $z = 0$ and membrane normal along the z -axis.

The **MembranePositionFromTopologyMover** uses knowledge of the transmembrane spans (stored in the `SpanningTopology` object) and protein coordinates to compute the centers and normals of each transmembrane span, average them, and then set the membrane center and normal to these values. Thus, this mover provides a first estimation that can be subsequently refined (as for instance in the `MPrelax` application below).

Similar to the previous mover, the **TransformIntoMembraneMover** uses the transmembrane spans and the protein coordinates to compute the centers and normal of each transmembrane span, and averages those to estimate an initial membrane position for the whole

protein. This Mover then transforms the protein into membrane coordinates such that the centers and normals of the overall estimated values and the membrane residue coincide [default center at (0, 0, 0) and normal at (0, 0, 15), where 15 Å represents half of the bilayer thickness].

Applications can invoke real-time visualization of a protocol in PyMOL. Visualizing the membrane bilayer is useful for evaluating the protein position in the membrane during the simulation and for final models. Rosetta’s PyMOLMover transmits coordinate information from the Pose in Rosetta to PyMOL in real time [44]. We extended this mover to additionally transfer membrane information, allowing the membrane bilayer to be displayed in PyMOL [45] when RosettaMP is in use (Fig J in S1 File). The PyMOLMover extracts the center, normal and thickness of the membrane bilayer from the membrane residue and uses this information to compute the location of two parallel planes representing the membrane. The planes are drawn as compiled graphics objects (CGO) [46] and are automatically updated after a change in position or orientation of the bilayer during a simulation. The visualization is demonstrated in the protocol capture in S6 File and S1 Movie.

Results

RosettaMP Applications

Creating new membrane protein modeling and design applications is simplified. To demonstrate the flexibility and potential of RosettaMP, we developed four proof-of-concept applications that draw on the rich existing functionality of Rosetta3 and adapt them for membrane proteins. The four applications are the prediction of free energy changes upon mutation (MPddG), high-resolution structural refinement (MPrelax), protein-protein docking (MPdock), and assembly of symmetric complexes (MPsymdock; Table 2). These protocols demonstrate use of mutations (design), energetic optimization, rigid-body transformations, and use of the symmetry machinery, all in the membrane environment. They further use three interfaces with the Rosetta libraries: writing code in C++ and scripting in PyRosetta [33] and RosettaScripts [34].

In Fig 3, we illustrate the simplicity of combining RosettaMP with existing Rosetta functions to create a useful application. The PyRosetta script consists of six steps: initializing Rosetta, loading a protein, adding a membrane, positioning the membrane using a predicted spanning topology, initializing a score function, and then calling PyRosetta’s $\Delta\Delta G$ calculation, a simplified version of a $\Delta\Delta G$ calculation protocol described below. This $\Delta\Delta G$ calculation simply optimizes side chain conformations and scores the protein with the wild-type and mutant residue at a given position.

Predicting free-energy changes ($\Delta\Delta G$) upon mutation in the membrane (MPddG).

Measuring the thermodynamic cost of a mutation provides insights into protein stability and serves as the starting point for protein design. Rosetta has thus far been used to estimate the free energy changes upon mutation ($\Delta\Delta G$) for proteins in solution [47] and for interfaces between soluble proteins [48,49]. To predict the $\Delta\Delta G$ of mutation in the membrane environment (protocol capture in S2 File), we combined RosettaMP with a fixed backbone $\Delta\Delta G$

Table 2. Applications developed with RosettaMP.

Application	Description	Platform
MPddG	Prediction of $\Delta\Delta G$ of mutation in the membrane	PyRosetta
MPrelax	High-resolution refinement in the membrane	RosettaScripts
MPdock	Protein-protein docking in the membrane	Rosetta3
MPsymdock	Assembly of symmetric complexes in the membrane	Rosetta3

doi:10.1371/journal.pcbi.1004398.t002


```

## PyRosetta Membrane Protocol
## Compute ddG of Mutation

# Step 1: Initialize Rosetta
from rosetta import *
rosetta.init()

# Step 2: Load protein from PDB file
pose = pose_from_pdb( "1MPF.pdb" )

# Step 3: Add Membrane Representation
add_memb = AddMembraneMover()
add_memb.apply( pose )

# Step 4: Embed protein in the membrane
init_mem_pos = MembranePositionFromTopology()
init_mem_pos.apply( pose )

# Step 5: Read score function from database
sfxn = create_score_function("mpframework_fa_smooth_2014")

# Step 6: Compute ddG of Mutation at residue 181 to alanine
ala_mut_ddG = calc_ddG( pose, sfxn, 181, 'A' )
print ala_mut_ddG

```

```

def calc_ddG( pose, sfxn, resnum, aa ):

# Score native Pose
native_score = sfxn( pose )

# Perform mutation at residue <resnum>
# to amino acid <aa>
mutated_pose = mutate_residue( pose, resnum, aa, sfxn )

# Pack at mutated residue position
pack_rot = PackRotamersMover()
pack_rot.apply( pose )

# Score mutated Pose
mutant_score = sfxn( mutated_pose )

# Return scores
return mutant_score - native_score

```

Fig 3. PyRosetta script for calculating the $\Delta\Delta G$ of mutation via RosettaMP. This example script loads Rosetta, adds the membrane representation, and uses the membrane score function to compute the $\Delta\Delta G$ of mutation in the membrane. Left: Python script used for $\Delta\Delta G$ calculations. Right: `calc_ddG` method used for computing $\Delta\Delta G$ of mutation.

doi:10.1371/journal.pcbi.1004398.g003

prediction protocol, similar to the one described in Kellogg *et al.* [47]. After mutation, side chain conformations were optimized for all residues within 8 Å of the mutated residue, and the $\Delta\Delta G$ was computed as the difference in Rosetta Energy Units (REU) between the mutant and native structure.

We tested the $\Delta\Delta G$ application's ability to reproduce experimental $\Delta\Delta G$ values from two data sets: (1) comprehensive mutations on outer membrane protein phospholipase A (OmpLA) [50] at a single position at the center of the membrane (Fig 4A), and (2) mutations from aromatic residues to alanine at several interfacial positions (Fig 4C) in outer membrane protein A (OmpA) [51]. Calculated values are compared to the experimental measurements in Fig 4B and 4D. A complete breakdown of predicted $\Delta\Delta G$ values by Rosetta score term is shown in Tables E-F in S1 File.

For OmpLA, an alanine at the center of the membrane is mutated into all 19 other amino acids. Insertion of proline is significantly overestimated because unfavorable dihedral angles cause a kink in the β -strand that cannot be resolved due to the fixed-backbone assumption. The negatively charged residues aspartic acid and glutamic acid are also over-predicted, similar to comparisons of these experimental values with published hydrophobicity scales [50]. When aspartic acid, glutamic acid, and proline are excluded, the correlation between the experimental and calculated values is $R = 0.86$, and the calculated Rosetta Energy Units (REU) correspond roughly to the measurements in kcal/mol. RosettaMP predicts the insertion of an arginine at the center of the membrane to be less disruptive than insertion of a lysine, and the side chain stretches toward the membrane interface (Fig 5A). This result matches previous experimental values [50,52,53] and occurs because the longer, positively charged side chain of arginine can snorkel further towards the interface region (Fig C in S1 File) and interact with charged lipid

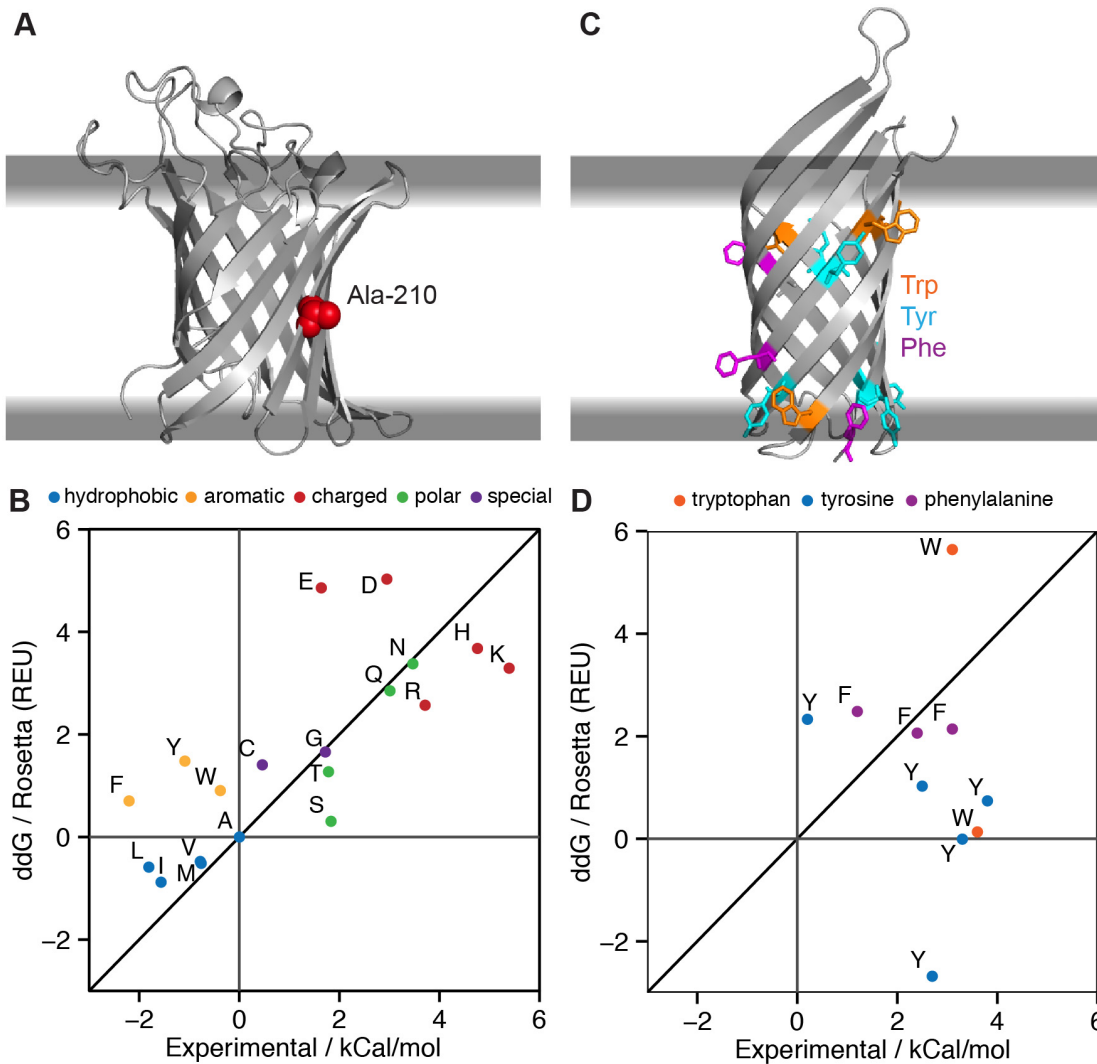


Fig 4. MPddG computes free energy changes upon mutation in the membrane environment ($\Delta\Delta G$). (A) Outer membrane protein phospholipase A (OmpLA, PDB 1qd6) with its native alanine at position 210 in red at the center of the membrane. (B) Plot of RosettaMP-calculated fixed-backbone $\Delta\Delta G$ s versus experimentally measured values of Moon & Fleming for variants at position 210 [50]. Proline is off-scale ($\Delta\Delta G_{pred} = 193.2$ REU) due to incompatible backbone torsions yielding ring closure penalties. (C) Outer membrane protein A (OmpA, PDB 1qjp) with aromatic residues mutated to alanine at various interfacial positions. (D) Plot of RosettaMP-calculated $\Delta\Delta G$ s versus experimentally measured values of Hong & Tamm [51]. The mutation W15A is off-scale ($\Delta\Delta G_{pred} = -43.0$ REU) due to the loss of repulsive interactions upon mutation to alanine. Both (B) and (D) include a line for $y = x$.

doi:10.1371/journal.pcbi.1004398.g004

head groups or interfacial water molecules [54]. The preference for arginine over lysine arises from the ~ 0.7 REU difference in the environment score (fa_mpenv_smooth) while small variations in the other score terms balance out. $\Delta\Delta G$ values for the polar residues asparagine and glutamic acid are consistent with the published values [50].

The Lazaridis membrane environment energy (fa_mpenv) correctly identifies both threonine and serine to be unfavorable, yet the $\Delta\Delta G$ value of serine is under-predicted. The $\Delta\Delta G$ of threonine is calculated to be greater than for serine due to larger van der Waals repulsive scores from a clash with leucine 225 (Fig 5B and Fig D in S1 File). Mutations into aromatic residues (phenylalanine, tryptophan and tyrosine) are calculated to be disruptive, which is in agreement with other hydrophobicity scales [55–57] but disagrees with published values [50]. In the RosettaMP calculations, the positive $\Delta\Delta G$ mainly arises from unfavorable van der Waals

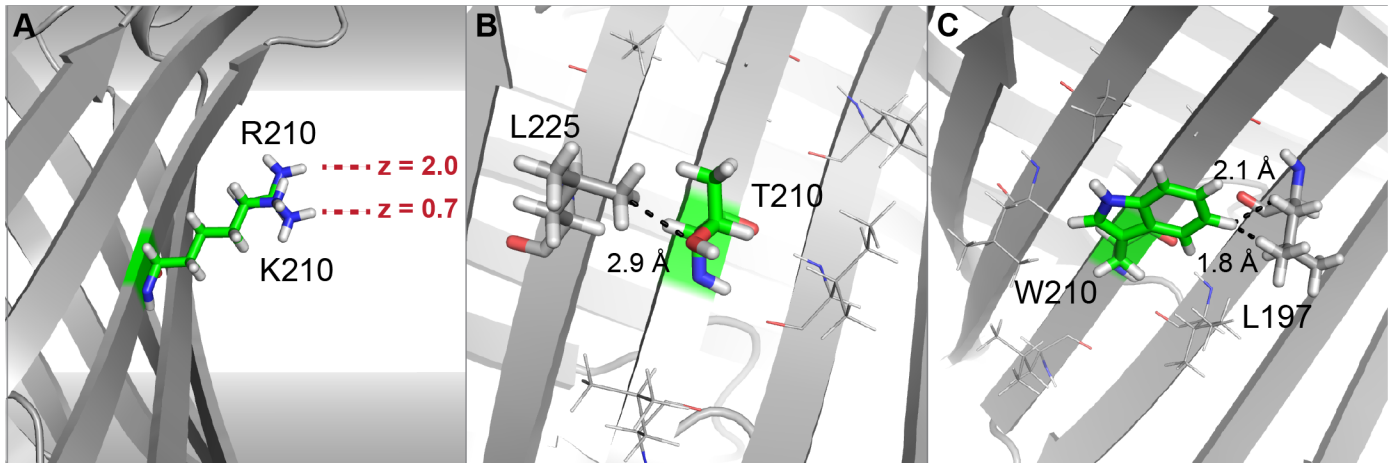


Fig 5. Structures of mutant residues at position 210 of OmpLA. (A) The charged residues arginine and lysine (superimposed) cannot reach the interface region. The z-coordinate shows the difference in membrane depth of the two charged side chains. Membrane environment scores are unfavorable for both, with lysine being slightly more unfavorable. (B) Insertion of threonine at position 210 is penalized by a mild clash from the neighboring leucine 225; serine at this position is accommodated more easily (Fig D in [S1 File](#)). (C) The tryptophan side chain is close to the neighboring leucine 197, resulting in large repulsive scores. All aromatic mutations have a comparably large repulsive van der Waals and rotamer scores, resulting in over-prediction of their $\Delta\Delta G$ values.

doi:10.1371/journal.pcbi.1004398.g005

repulsive energy (fa_rep) (Fig 5C) and the rotamer score (fa_dun) due to clashes with leucine 225 (Fig E in [S1 File](#)). Insertion of negatively charged residues, aspartic acid and glutamic acid, is significantly overestimated, accounting for the cost of placing a charged species into the membrane, which should add about 2 kcal/mol [50]. These findings suggest that further improvements to the score function are necessary.

Our second data set is from OmpA. The data points cluster from calculated values of 0–3 REU and experimental values 0–4 kcal/mol, capturing disruption of π -orbital stacking interactions with neighboring aromatic residues, as shown by large changes in the van der Waals attractive scores. Although the correlation between predictions and published values is low ($R = 0.13$), the data in [Fig 4D](#) demonstrate that the MPddG application correctly identifies 9 of 12 mutations as unfavorable, attributed to loss of van der Waals attractive energy and an unfavorable knowledge-based membrane environment score (Table F in [S1 File](#)). The two tyrosine residues and single tryptophan whose mutation to alanine are predicted to be favorable ($\Delta\Delta G_{\text{pred}} < 0$) are isolated residues without any π -stacking interactions that would be disrupted by the mutation. In each case, alanine is predicted to be more stable because the unfavorable van der Waals attractive score difference is outweighed by the sum of the favorable repulsive and rotamer score differences. To achieve high-quality predictions of $\Delta\Delta G$ s of mutation, possible improvements include consideration of backbone and side chain flexibility [47] and advances in the score function, both of which require extensive benchmarking on a large dataset, which is currently difficult to obtain. Therefore, identification of favorable vs. disruptive mutations is an important first step towards this goal.

High-resolution refinement of membrane protein structures (MPrelax). High-resolution refinement of protein structures is necessary to advance low-resolution structures to atomic level detail and to create high-resolution models as inputs for protein-protein docking, protein-ligand docking, and design. For membrane proteins, this task is complicated by lack of knowing the precise embedding of the protein in the membrane.

We combined RosettaMP with Rosetta’s high-resolution refinement protocol FastRelax [58,59], the high-resolution membrane score function [23,43], and a minimization-based technique for optimizing the protein embedding to create MPrelax (protocol capture in [S3 File](#)).

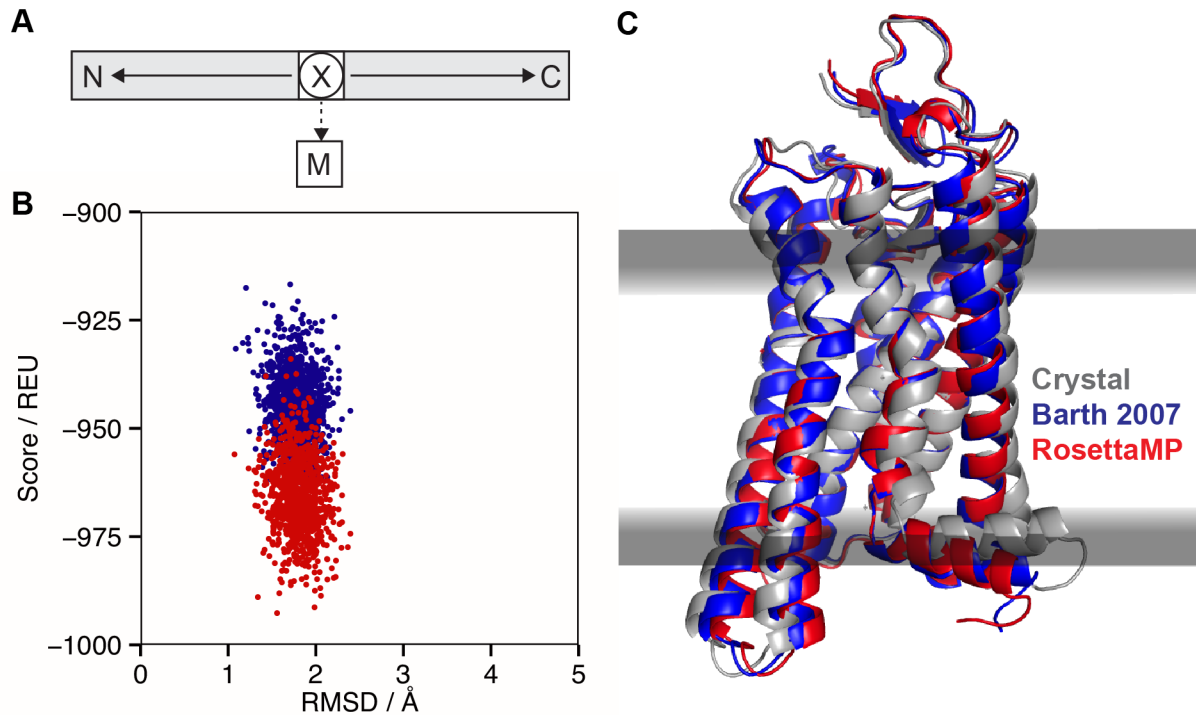


Fig 6. MPrelax for high-resolution refinement of a membrane protein. (A) *FoldTree* representation for the MPrelax protocol with the residue closest to the center-of-mass of the protein being at the root of the *FoldTree* (circled X). The membrane residue (M) is attached via a flexible jump edge (dashed arrow). Protein chains are shown as gray boxes with N- and C- termini marked and peptide edges shown as solid arrows. (B) Rosetta total score vs. backbone RMSD to the crystal structure for 1000 models of meta-rhodopsin. Models in blue are created with the original membrane relax protocol of RosettaMembrane; models in red are created with MPrelax. (C) Crystal structure of meta-rhodopsin in gray (PDB 3pxo) superimposed with the lowest scoring models from both the original RosettaMembrane protocol (blue) and the MPrelax protocol (red).

doi:10.1371/journal.pcbi.1004398.g006

The membrane residue is initially placed at the center-of-mass of the transmembrane spans and is allowed to move during the simulation by placing the *FoldTree* root in the protein (Fig 6A). The protein is then refined using FastRelax, performing eight iterations of rotamer trials and minimization of the backbone and side chains. During these iterations, the position and orientation of the membrane is optimized by gradient-based minimization.

We tested MPrelax on four test cases (Fig F in S1 File). As a representative example, Fig 6B and 6C show results for meta-rhodopsin II (PDB 3pxo) in the apo form [60] (calculations with retinal present are shown in Fig G in S1 File). For comparison, we also refined the structures with the original RosettaMembrane relax application [23]. The relaxed structures from both protocols are similar to each other and to the crystal coordinates (Fig 6C), and the new MPrelax protocol samples conformations at lower energies than the original RosettaMembrane relax (Fig 6B). The RosettaMP models have slightly different positions and orientations (embedding) in the membrane, since this method allows optimization of membrane embedding through minimization across the jump between the protein and the membrane residue. When RosettaMP outperforms RosettaMembrane, favorable hydrogen bonds result in lower scores (Table G in S1 File). Interestingly, the lower hydrogen bonding scores do not arise from an increase in the number of hydrogen bonds, but from better positioning within the membrane since the hydrogen bonding energy is dependent upon membrane embedding. In the original RosettaMembrane relax application, the membrane embedding is computed at the beginning of the protocol and then kept constant. For MPrelax, the minimization-based routine uses the

membrane score function to optimize the position and orientation of the membrane based on the evolving protein conformation.

Similar to the meta-rhodopsin case, the new relax protocol also leads to lower scores in methyltransferase (PDB 4a2n) and histidine kinase receptor QseC (PDB 2kse) as shown in Supplementary Fig F and Tables H and I in [S1 File](#). One more case (disulfide bond protein B, PDB 2leg, Table J in [S1 File](#)) shows higher scores with MPrelax than with the original relax protocol. Since this protein does not fully span the hydrophobic thickness of the membrane, this data suggests the minimization routine used by RosettaMP requires adjustment to sample a larger conformational space for embedding. For all cases, the range of sampled structures typically covers 1–6 Å RMSD, but often with a lack of structures close to native (within 1–2 Å). This may indicate that Rosetta scores the near native structures poorly, perhaps due to clashes in these structures, errors in the score function, or artifacts of the artificial lipid environment in the experiments that preclude these structures from being the low-energy conformation in an ideal, implicit, slab-like membrane.

Protein-protein docking in the membrane bilayer (MPdock). Structure determination of protein-protein complexes in the membrane bilayer is extraordinarily difficult due to the requirement for the membrane mimetic to maintain stability of the complex and because many complexes exceed the molecular weight limit for NMR spectroscopy. We combined RosettaMP with the RosettaDock algorithm [[61,62](#)] to predict structures of protein-protein complexes in the membrane bilayer (protocol capture in [S4 File](#)). The protocol consists of two steps: (1) a prepacking step to create a starting structure, and (2) protein-protein docking in the membrane bilayer. In the pre-packing step, the two partners are first separated by a large distance (keeping their membrane embedding constant), the side chains are repacked using rotamer trials, and the partners are moved back together. Next, the docking step samples random interface conformations using a score function that is created by combining the standard docking score functions with the membrane score terms (both in the low-resolution and all-atom stages, see Tables C-D in [S1 File](#)). The membrane bilayer is kept fixed during this simulation ([Fig 7A](#)).

MPdock was tested on five protein-protein complexes in the membrane. We created 1000 models for each complex and examined the structures, interface scores, fraction of native residue contacts, and RMSD values to the crystal structures ([Fig H in S1 File](#)). As an example, [Fig 7](#) shows data for vitamin B12 importer BtuCD (PDB ID: 2qi9, docking partners are chains A and B), a dimer with 10 transmembrane helices per subunit. The score vs. RMSD plot in [Fig 7B](#) shows that the score function is able to distinguish near-native from non-native conformations; the model with the lowest interface score has an RMSD of 1.8 Å. Across our test set, the score function is able to distinguish native-like from non-native conformations in about half of the targets ([Fig H in S1 File](#)). In these cases, it recovers greater than 50% of native contacts meeting the CAPRI criteria for high quality predictions [[63](#)]. Scoring failures are seen for glycoporphin A and for the SecYG complex, where the refined crystal structures have higher scores than the models with the lowest interface score. The native conformation for the methionine transporter has higher scores than the lowest scoring models, which have larger interfaces. These models suggest either valid alternate conformations of the transporter or imbalances in the score function. While these preliminary data are encouraging, it is important to note that these proof-of-concept calculations were restricted to a local vicinity of the interface in the crystal structure and that crystal backbone coordinates were used. Improvements to both the sampling routine as well as the score function are needed for docking of unbound complexes or homology models, or to carry out blind, global docking in the membrane environment while simultaneously moving both partners in the membrane.

Assembly of symmetric membrane protein complexes (MPsymdock). Many membrane proteins assemble into symmetric complexes in the membrane environment. We developed an

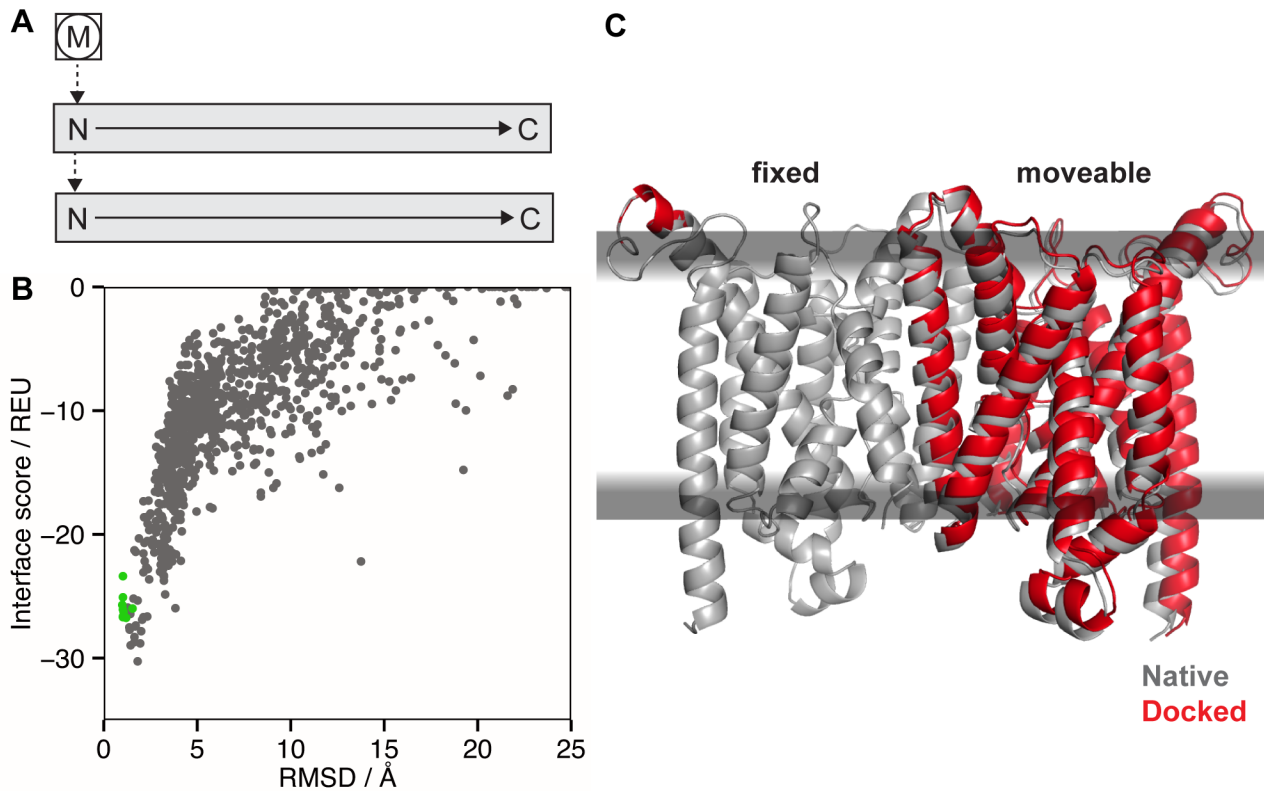


Fig 7. Protein-protein docking in the membrane bilayer using MPdock. (A) FoldTree representation used in MPdock with the membrane residue (M) being fixed at the root (circled) of the FoldTree and the protein chains as docking partners attached via jump edges. (B) Interface score vs. backbone RMSD to the native structure for 1000 models of the vitamin B12 importer BtuCD. The RMSD is the 'ligand' RMSD, which is computed only over the moving partner after superimposing the fixed partner and membrane. The green dots represent ten models created by minimizing the crystal structure. The interface score of the crystal structure (180.5 REU) is outside of the plotting range due to clashes. (C) Native structure of the vitamin B12 importer (gray, PDB 2qi9) superimposed with the model having the lowest interface score (red).

doi:10.1371/journal.pcbi.1004398.g007

application for symmetric assembly of complexes in the membrane bilayer; we achieved this by combining Rosetta's symmetric docking protocol [64] with RosettaMP (Fig 8 with protocol capture in S5 File). The FoldTree maintains internal symmetry of the complex and its position in the membrane bilayer, with the membrane residue being at its root, hence keeping it fixed (Fig 8A). The subunits are arranged in C_n symmetry around the membrane normal axis (defined consistently in this protocol as the z -axis), where n is the number of subunits in the complex. To account for symmetry in the protein, the FoldTree uses two additional virtual residues per subunit, $V_{1,i}$ and $V_{2,i}$, where i is the number of the subunit and $1 \leq i \leq n$. The jump from $V_{1,i}$ to $V_{1,i+1}$ describes the rotation and translation required to transform the i^{th} subunit to the $(i+1)^{\text{th}}$ subunit based on the C_n symmetry, and the jump from $V_{1,i}$ to $V_{2,i}$ describes the rotation and translation between the $V_{1,i}$ and the protein subunit root residue. This setup allows the protocol to respect both symmetry and the membrane environment while allowing efficient sampling moves, side chain packing, and scoring.

MPsymdock was tested on four examples (Fig I in S1 File) with one example shown in Fig 8: the crystal structure of the homo-tetrameric KcsA potassium channel (PDB 1bl8). Because the membrane protein docking score function had a low tolerance for initial clashes in the structure, each complex was first refined using the MPrelax protocol described above. Ten models were generated, and one asymmetric unit of the lowest scoring model was used as input to the

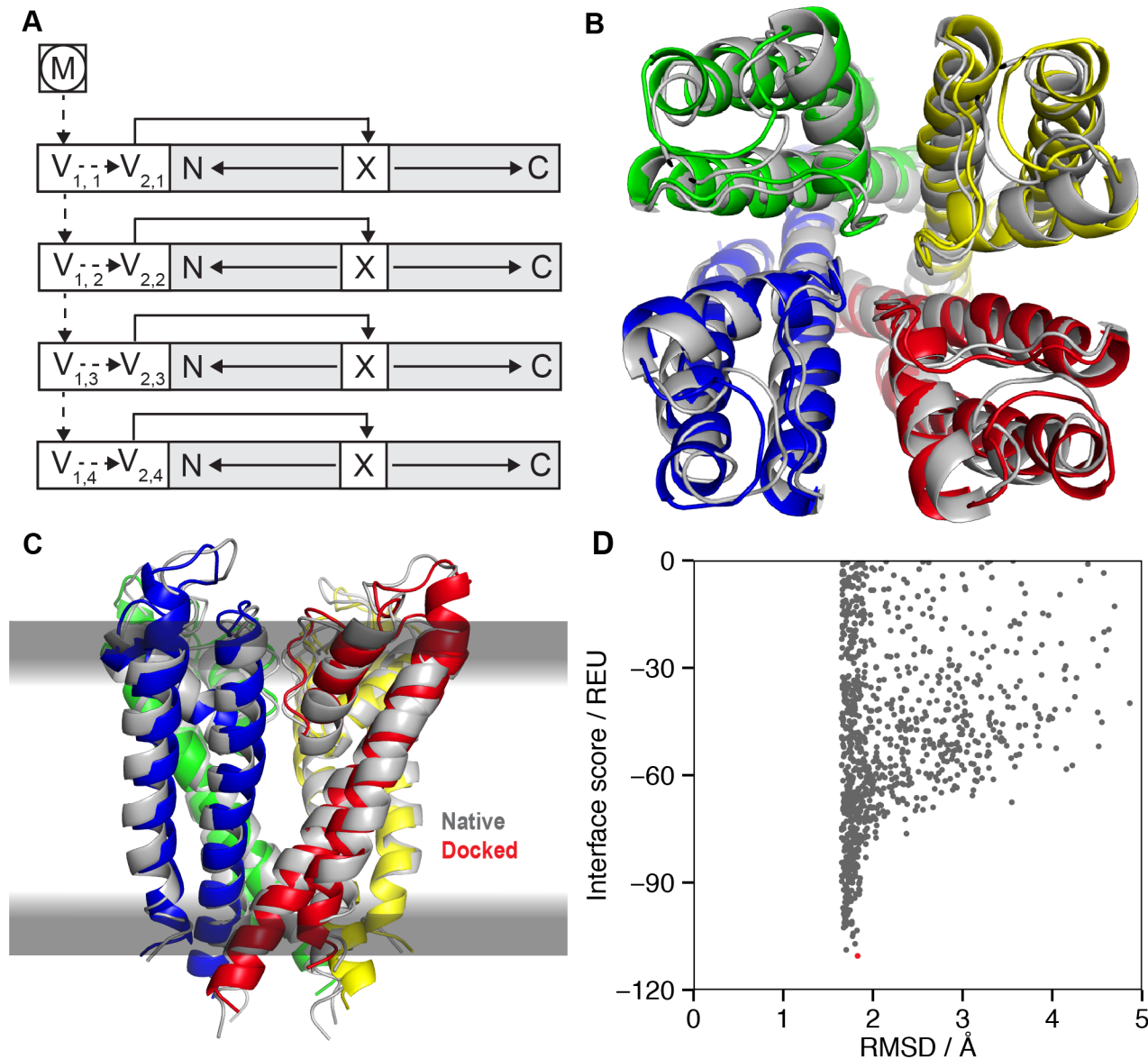


Fig 8. Assembly of symmetric protein complexes in the membrane using MPsymdock. (A) FoldTree representation of the homo-tetrameric KcsA potassium channel with the membrane residue (M) at the root (circled). The virtual residues $V_{1,i}$ and $V_{2,i}$ required for the symmetry machinery are described in the text. (B) Native structure in gray (PDB 1bl8) superimposed with the model from MPsymdock with the lowest interface score. The view is from the extracellular side of the membrane. (C) Membrane plane view of (B). (D) Interface score vs. backbone RMSD to the native structure for 1000 models of the KcsA potassium channel. The lowest scoring model, shown in (B) and (C), is indicated in red.

doi:10.1371/journal.pcbi.1004398.g008

symmetric docking protocol. From the asymmetric unit, the full complex was reassembled by the symmetric docking routine and the interface score was computed as the cumulative score of all symmetric interfaces [65]. The interface score vs. RMSD plot (Fig 8D) indicates that the high-resolution membrane score function, adapted for symmetric docking, is able to distinguish near-native from non-native conformations; the model with the lowest interface score has an RMSD of 1.8 Å to the native crystal structure. This is an encouraging result, especially since for this target the interface in the selectivity filter contains a number of proximal carbonyl oxygen atoms. The large interface size between the subunits may compensate for adverse

effects in the selectivity filter. This example suggests the requirement for a more sophisticated membrane model accounting for channels in the membrane. Glycophorin A is similarly successful (Fig I in [S1 File](#)). In both cases, the RMSD is elevated by the asymmetry in tail regions of each subunit in the native reference structure; calculations without the tails may be more physically realistic and produce lower RMSDs.

While the low-RMSD docked complexes of KcsA and glycophorin A are encouraging, the remaining two examples were unable to form native-like complexes with well-packed interfaces; instead the subunits separated during the search to relieve clashes (Fig I in [S1 File](#)). To obtain high-quality models, several improvements are necessary. A new symmetric refinement protocol could generate symmetric starting structures while removing clashes [66]. Another approach to remove clashes would be through backbone flexibility in the protocol itself; this approach will require a gradual transition to introduce repulsive van der Waals energies to avoid the high gradients that currently cause the subunits to separate.

Discussion

Here, we describe the use of a software suite for biomolecular modeling, docking, and design to enable rapid development of new applications targeted at membrane proteins, a class for which structure determination efforts are notoriously difficult. We used the scientific concepts in the RosettaMembrane structure prediction and refinement applications and created a modular framework within Rosetta3's object-oriented architecture [32]. The four proof-of-concept applications demonstrate flexibility, generality, and simplicity of RosettaMP. The new framework enables combination of the membrane environment with a variety of Rosetta features: with the fold tree [36], jumps in membrane proteins can be used to model multiple protein chains, flexible loops, and ligands; and with symmetry [64], symmetric protein structure prediction, refinement and design will be feasible. RosettaMP will serve as the starting point for future protocol development, and each new application can be extensively tested and benchmarked. Our preliminary results show that RosettaMP has the potential to answer long-standing questions involving membrane proteins and lays the groundwork for the challenges that still remain.

RosettaMP complements many existing tools for membrane protein modeling. MPrelax can be used to refine proteins inserted into the membrane using tools such as iMembrane [67]. MPrelax can also be used in combination with a contact prediction method to predict structures with low sequence similarities to their template (similar to I-TASSER [68,69]). MPddG can directly be used for alanine scanning and extended for membrane protein- and interface-design. Homology models from MEDELLER [70] and Rosetta [16] can be used as input to MPdock and MPsymdock for modeling of large membrane protein complexes. In principle, the MPsymdock protocol can also be used to distinguish biological interfaces from non-native crystal contacts (similar to COMP [71] or PISA for soluble proteins [72]). The variety of these potential applications shows that RosettaMP forms an important basis for new protocol development.

The key components of any structure prediction or design algorithm are sampling and scoring. Conformational sampling routines are improved via RosettaMP through the connection of the membrane bilayer to the modeled biomolecule. This representation allows flexibility in choosing which object should be fixed vs. movable (protein or membrane) by representing the membrane as a 'residue' and using a jump in the fold tree. For example, a fixed bilayer enables sampling of membrane-embedded docking conformations in the MPdock and MPsymdock protocols, whereas a movable membrane decreases the computational cost of the MPrelax protocol. The latter allows optimizing the membrane position and orientation using minimization algorithms, resulting in lower scores for three of four cases. Moreover, the flexible linkage now

permits constraining spans, chains, or proteins to the membrane in various depths and orientations, features that could not be modeled previously. The framework also simplifies implementation of enhanced sampling protocols through specialized movers that favor meaningful protein conformations in the membrane, for instance for *ab initio* prediction of α -helical and β -barrel membrane proteins using particular fragment types or favoring appropriate orientations and pairings of the secondary structure elements.

For scoring, the new framework allows us to test Rosetta's membrane score functions in new contexts. The four applications collectively demonstrate that the existing low- and high-resolution membrane protein score functions are generally effective, yet require further optimization. The preliminary MPddG application is able to identify favorable vs. disruptive mutations in the two tested cases and even produces a reasonable correlation for predicted vs. experimental $\Delta\Delta G$ values for an intra-membrane residue in OmpLA. Refinement, which is minimally modified from its original, tested implementation, captures the same minima. The naïve score functions for docking and symmetric docking exhibit minima for native-like interfaces in about half of the asymmetric and symmetric cases. These results are encouraging, especially since we have not made any changes to the RosettaMembrane score functions originally developed for folding and refinement.

For future work, several improvements to the score function seem possible. Since the number of determined membrane protein structures has increased substantially in recent years, the low-resolution, knowledge-based score functions can be updated to reduce statistical errors. Further, existing score functions were solely derived from α -helical membrane proteins, and data from β -barrels could be used to create a distinct score function that could be tested with large-scale folding and refinement of these proteins. For instance, a recently derived hydrophobic potential for outer membrane β -barrels has been found to be condensed compared to that for α -helical membrane proteins, since bacterial outer membranes have a smaller membrane thickness [55,57]. An updated formulation of Rosetta's distance-dependent dielectric electrostatic score [73] is needed to accommodate the low dielectric constant in the membrane. It is also now feasible in principle to sample protonated and deprotonated forms of ionizable residues with Rosetta pH [74,75]; however parameterization is needed to account for the insertion of charged species in the membrane. An advantage of RosettaMP is that new score functions can now be more easily derived than previously by using the score function machinery in Rosetta3.

The current membrane model is a flat bilayer model of fixed thickness, i.e. a slab model. RosettaMP could be a stepping-stone for tackling complex biological questions with more sophisticated membrane models. Effects needed to create the next generation of this model include intrinsic curvature, charge asymmetry [76–78], and variable thickness [79], attributed to the diverse repertoire of lipids that constitute the membrane environment [80]. Membrane thickness might play a role in the accurate estimation of $\Delta\Delta G$ values for interfacial aromatic residues [51] and in reproducing snorkeling of arginine and lysine to the membrane interface [54,81]. More sophisticated membrane models will be required for proteins that form pores or toroidal pores [82]. Another challenge is the modeling of membrane-anchored proteins or peptides [83] especially for small and/or unstructured peptides or half-helices inserting into the bilayer that are not identified as such by sequence-based prediction methods.

RosettaMP will also enable the development of design protocols, an important yet challenging task with potential impact in synthetic biology and gene therapy. The latest membrane protein design efforts focused on helix-helix interfaces [20], protein chimeras [84], and used protein display [85] and combinatorial libraries [86] to identify promising designs. These efforts require manual processes and use full-atom energy functions derived mostly from MD force fields. With robust score functions and sampling routines, methods developed with RosettaMP will add to these emerging tools and complement MD simulation packages,

enabling investigation of membrane protein structure, dynamics, and function from low- to high-resolution representations.

In summary, we anticipate new progress by combining the power of existing Rosetta applications with RosettaMP. By making membrane protein modeling and design accessible to the broad scientific community, we hope to drive understanding of membrane protein structure, function and ultimately enable drug design for this essential class of proteins.

Supporting Information

S1 File. Object-oriented design, detailed methods, and results. This file contains a discussion of the object-oriented design of the RosettaMP framework. Both the low-resolution and high-resolution score functions are described in detail. Finally, we discuss detailed methods and results for development and testing of each RosettaMP application.
(PDF)

S2 File. Protocol capture for MPddG. This protocol capture contains the steps, input files and example output files necessary to run the MPddG protocol described in this manuscript. While both mutations in OmpLA and OmpA are discussed, we only describe the protocol for computing $\Delta\Delta G$ s for OmpLA for simplification. The supplementary files are included with the Rosetta3 software suite under the directory Rosetta/demos/protocol_capture/MPddG.
(GZ)

S3 File. Protocol capture for MPrelax. This protocol capture contains the steps, input files, and example output files necessary to run the MPrelax protocol described in this manuscript. For simplification, we only describe refinement of meta-rhodopsin II in these files. The supplementary files are included with the Rosetta3 software suite under the directory Rosetta/demos/protocol_capture/MPrelax.
(GZ)

S4 File. Protocol capture for MPdock. This protocol capture contains the steps, input files, and example output files necessary to run the MPdock protocol described in this manuscript. For simplification, we only describe refinement of Glycophorin A in these files. The supplementary files are included with the Rosetta3 software suite under the directory Rosetta/demos/protocol_capture/MPdock.
(GZ)

S5 File. Protocol capture for MPsymdock. This protocol capture contains the steps, input files, and example output files necessary to run the MPdock protocol described in this manuscript. For simplification, we only describe docking of Glycophorin A in these files. The supplementary files are included with the Rosetta3 software suite under the directory Rosetta/demos/protocol_capture/MPdock.
(GZ)

S6 File. Protocol capture for MPPyMOLViewer. This protocol capture contains the steps, input files, and example output files necessary to run the MPPyMOLViewer protocol described in this manuscript. For simplification, we only describe visualization of bacteriorhodopsin in these files. The supplementary files are included with the Rosetta3 software suite under the directory Rosetta/demos/protocol_capture/MPPyMOLViewer.
(GZ)

S7 File. Protocol capture for MPspanFromPDB. This protocol capture contains the steps, input files, and example output files necessary to run the MPPyMOLViewer protocol described

in this manuscript. For simplification, we only describe visualization of bacteriorhodopsin. The supplementary files are included with the Rosetta3 software suite under the directory Rosetta/demos/protocol_capture/MPSpanFromPDB.

(GZ)

S1 Movie. Real-Time Visualization of Membrane Simulation in PyMOL. Visualization of a symmetric docking simulation of the protein complex potassium channel KcsA (PDB ID 1bl8). For every conformational change in the simulation, Rosetta sends an updated structure to PyMOL via the PyMOL Mover. The new structure is displayed in real-time in PyMOL. Asymmetric subunits colored independently. The complex is first assembled from a single asymmetric subunit, docked in low-resolution, and then refined in high-resolution.

(MP4)

Acknowledgments

The authors thank Vladimir Yarov-Yarovoy, Frank DiMaio, Evan Baugh, Justin Porter, Andrew Leaver-Fay, Jens Meiler, Sarel Fleishman and the RosettaCommons for scientific discussions and comments on code design. We also thank Sergey Lyskov for support with automated testing of the code and Karen Fleming and Sarah McDonald for helpful discussions.

Author Contributions

Conceived and designed the experiments: JKL RFA BDW JYG. Analyzed the data: RFA JKL BDW JYG. Wrote the paper: RFA JKL BDW JYG. Designed and implemented code: RFA JKL BDW JYG. Tested protocols and reviewed documentation: AMD DCT AE.

References

1. Von Heijne G. The membrane protein universe: what's out there and why bother? *J Intern Med.* 2007; 261: 543–557. doi: [10.1111/j.1365-2796.2007.01792.x](https://doi.org/10.1111/j.1365-2796.2007.01792.x) PMID: [17547710](https://pubmed.ncbi.nlm.nih.gov/17547710/)
2. Tan S, Tan HT, Chung MCM. Membrane proteins and membrane proteomics. *Proteomics.* 2008; 8: 3924–3932. doi: [10.1002/pmic.200800597](https://doi.org/10.1002/pmic.200800597) PMID: [18763712](https://pubmed.ncbi.nlm.nih.gov/18763712/)
3. Bakheet TM, Doig AJ. Properties and identification of human protein drug targets. *Bioinformatics.* 2009; 25: 451–457. doi: [10.1093/bioinformatics/btp002](https://doi.org/10.1093/bioinformatics/btp002) PMID: [19164304](https://pubmed.ncbi.nlm.nih.gov/19164304/)
4. Overington JP, Al-Lazikani B, Hopkins AL. How many drug targets are there? *Nat Rev Drug Discov.* 2006; 5: 993–996. doi: [10.1038/nrd2199](https://doi.org/10.1038/nrd2199) PMID: [17139284](https://pubmed.ncbi.nlm.nih.gov/17139284/)
5. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, et al. The Protein Data Bank. *Nucleic Acids Res.* 2000; 28: 235–242. doi: [10.1093/nar/28.1.235](https://doi.org/10.1093/nar/28.1.235) PMID: [10592235](https://pubmed.ncbi.nlm.nih.gov/10592235/)
6. Koehler Leman J, Ulmschneider MB, Gray JJ. Computational modeling of membrane proteins. *Proteins Struct Funct Bioinforma.* 2014; 1–24. doi: [10.1002/prot.24703](https://doi.org/10.1002/prot.24703)
7. Roos C, Kai L, Proverbio D, Ghoshdastider U, Filipek S, Dötsch V, et al. Co-translational association of cell-free expressed membrane proteins with supplied lipid bilayers. *Mol Membr Biol.* 2013; 30: 75–89. doi: [10.3109/09687688.2012.693212](https://doi.org/10.3109/09687688.2012.693212) PMID: [22716775](https://pubmed.ncbi.nlm.nih.gov/22716775/)
8. Weiner BE, Woetzel N, Karakaş M, Alexander N, Meiler J. BCL::MP-Fold: Folding Membrane Proteins through Assembly of Transmembrane Helices. *Structure.* 2013; 21: 1107–1117. doi: [10.1016/j.str.2013.04.022](https://doi.org/10.1016/j.str.2013.04.022) PMID: [23727232](https://pubmed.ncbi.nlm.nih.gov/23727232/)
9. Michel M, Hayat S, Skwark MJ, Sander C, Marks DS, Elofsson A. PconsFold: improved contact predictions improve protein models. *Bioinforma Oxf Engl.* 2014; 30: i482–488. doi: [10.1093/bioinformatics/btu458](https://doi.org/10.1093/bioinformatics/btu458)
10. Marks DS, Hopf TA, Sander C. Protein structure prediction from sequence variation. *Nat Biotechnol.* 2012; 30: 1072–1080. doi: [10.1038/nbt.2419](https://doi.org/10.1038/nbt.2419) PMID: [23138306](https://pubmed.ncbi.nlm.nih.gov/23138306/)
11. Jones DT, Buchan DWA, Cozzetto D, Pontil M. PSICOV: precise structural contact prediction using sparse inverse covariance estimation on large multiple sequence alignments. *Bioinforma Oxf Engl.* 2012; 28: 184–190. doi: [10.1093/bioinformatics/btr638](https://doi.org/10.1093/bioinformatics/btr638)

12. Wang Y, Barth P. Evolutionary-guided de novo structure prediction of self-associated transmembrane helical proteins with near-atomic accuracy. *Nat Commun.* 2015; 6: 7196. doi: [10.1038/ncomms8196](https://doi.org/10.1038/ncomms8196) PMID: [25995083](https://pubmed.ncbi.nlm.nih.gov/25995083/)
13. Kamisetty H, Ovchinnikov S, Baker D. Assessing the utility of coevolution-based residue-residue contact predictions in a sequence- and structure-rich era. *Proc Natl Acad Sci U S A.* 2013; 110: 15674–15679. doi: [10.1073/pnas.1314045110](https://doi.org/10.1073/pnas.1314045110) PMID: [24009338](https://pubmed.ncbi.nlm.nih.gov/24009338/)
14. Nugent T, Jones DT. Accurate de novo structure prediction of large transmembrane protein domains using fragment-assembly and correlated mutation analysis. *Proc Natl Acad Sci U S A.* 2012; 109: E1540–1547. doi: [10.1073/pnas.1120036109](https://doi.org/10.1073/pnas.1120036109) PMID: [22645369](https://pubmed.ncbi.nlm.nih.gov/22645369/)
15. Barth P, Wallner B, Baker D. Prediction of membrane protein structures with complex topologies using limited constraints. *Proc Natl Acad Sci U S A.* 2009; 106: 1409–1414. doi: [10.1073/pnas.0808323106](https://doi.org/10.1073/pnas.0808323106) PMID: [19190187](https://pubmed.ncbi.nlm.nih.gov/19190187/)
16. Chen K-YM, Sun J, Salvo JS, Baker D, Barth P. High-Resolution Modeling of Transmembrane Helical Protein Structures from Distant Homologues. *PLoS Comput Biol.* 2014; 10: e1003636. doi: [10.1371/journal.pcbi.1003636](https://doi.org/10.1371/journal.pcbi.1003636) PMID: [24854015](https://pubmed.ncbi.nlm.nih.gov/24854015/)
17. McNulty R, Ulmschneider JP, Luecke H, Ulmschneider MB. Mechanisms of molecular transport through the urea channel of *Helicobacter pylori*. *Nat Commun.* 2013; 4: 2900. doi: [10.1038/ncomms3900](https://doi.org/10.1038/ncomms3900) PMID: [24305683](https://pubmed.ncbi.nlm.nih.gov/24305683/)
18. Ulmschneider MB, Bagn eris C, McCusker EC, Decaen PG, Delling M, Clapham DE, et al. Molecular dynamics of ion transport through the open conformation of a bacterial voltage-gated sodium channel. *Proc Natl Acad Sci U S A.* 2013; 110: 6364–6369. doi: [10.1073/pnas.1214667110](https://doi.org/10.1073/pnas.1214667110) PMID: [23542377](https://pubmed.ncbi.nlm.nih.gov/23542377/)
19. Hulse RE, Sachleben JR, Wen P-C, Moradi M, Tajkhorshid E, Perozo E. Conformational dynamics at the inner gate of KcsA during activation. *Biochemistry (Mosc).* 2014; 53: 2557–2559. doi: [10.1021/bi500168u](https://doi.org/10.1021/bi500168u)
20. Joh NH, Wang T, Bhate MP, Acharya R, Wu Y, Grabe M, et al. De novo design of a transmembrane Zn²⁺-transporting four-helix bundle. *Science.* 2014; 346: 1520–1524. doi: [10.1126/science.1261172](https://doi.org/10.1126/science.1261172) PMID: [25525248](https://pubmed.ncbi.nlm.nih.gov/25525248/)
21. Caputo GA, Litvinov RI, Li W, Bennett JS, DeGrado WF, Yin H. Computationally Designed Peptide Inhibitors of Protein-Protein Interactions in Membranes. *Biochemistry (Mosc).* 2008; 47: 8600–8606. doi: [10.1021/bi800687h](https://doi.org/10.1021/bi800687h)
22. Senes A. Computational design of membrane proteins. *Curr Opin Struct Biol.* 2011; 21: 460–466. doi: [10.1016/j.sbi.2011.06.004](https://doi.org/10.1016/j.sbi.2011.06.004) PMID: [21763125](https://pubmed.ncbi.nlm.nih.gov/21763125/)
23. Barth P, Schonbrun J, Baker D. Toward high-resolution prediction and design of transmembrane helical protein structures. *Proc Natl Acad Sci.* 2007; 104: 15682–15687. doi: [10.1073/pnas.0702515104](https://doi.org/10.1073/pnas.0702515104) PMID: [17905872](https://pubmed.ncbi.nlm.nih.gov/17905872/)
24. Chen K-YM, Zhou F, Fryszczyn BG, Barth P. Naturally evolved G protein-coupled receptors adopt metastable conformations. *Proc Natl Acad Sci U S A.* 2012; 109: 13284–13289. doi: [10.1073/pnas.1205512109](https://doi.org/10.1073/pnas.1205512109) PMID: [22847407](https://pubmed.ncbi.nlm.nih.gov/22847407/)
25. Naveed H, Xu Y, Jackups R, Liang J. Predicting three-dimensional structures of transmembrane domains of β -barrel membrane proteins. *J Am Chem Soc.* 2012; 134: 1775–1781. doi: [10.1021/ja209895m](https://doi.org/10.1021/ja209895m) PMID: [22148174](https://pubmed.ncbi.nlm.nih.gov/22148174/)
26. Yarov-Yarovoy V, Schonbrun J, Baker D. Multipass membrane protein structure prediction using Rosetta. *Proteins Struct Funct Bioinforma.* 2006; 62: 1010–1025. doi: [10.1002/prot.20817](https://doi.org/10.1002/prot.20817)
27. Bonneau R, Tsai J, Ruczinski I, Chivian D, Rohl C, Strauss CEM, et al. Rosetta in CASP4: Progress in ab initio protein structure prediction. *Proteins Struct Funct Bioinforma.* 2001; 45: 119–126. doi: [10.1002/prot.1170](https://doi.org/10.1002/prot.1170)
28. Rohl CA, Strauss CEM, Misura KMS, Baker D. Protein Structure Prediction Using Rosetta. In: Brand Ludwig and Johnson Michael L., editor. *Methods in Enzymology.* Academic Press; 2004. pp. 66–93. Available: <http://www.sciencedirect.com/science/article/pii/S0076687904830040> PMID: [15063647](https://pubmed.ncbi.nlm.nih.gov/15063647/)
29. Adamian L, Nanda V, DeGrado WF, Liang J. Empirical lipid propensities of amino acid residues in multi-span alpha helical membrane proteins. *Proteins Struct Funct Bioinforma.* 2005; 59: 496–509. doi: [10.1002/prot.20456](https://doi.org/10.1002/prot.20456)
30. Kortemme T, Morozov AV, Baker D. An orientation-dependent hydrogen bonding potential improves prediction of specificity and structure for proteins and protein-protein complexes. *J Mol Biol.* 2003; 326: 1239–1259. PMID: [12589766](https://pubmed.ncbi.nlm.nih.gov/12589766/)
31. Lazaridis T. Effective energy function for proteins in lipid membranes. *Proteins.* 2003; 52: 176–192. doi: [10.1002/prot.10410](https://doi.org/10.1002/prot.10410) PMID: [12833542](https://pubmed.ncbi.nlm.nih.gov/12833542/)

32. Leaver-Fay A, Tyka M, Lewis SM, Lange OF, Thompson J, Jacak R, et al. Rosetta3: An Object-Oriented Software Suite for the Simulation and Design of Macromolecules. Johnson Michael L. and Brand Ludwig, editor. *Methods Enzymol.* 2011; Volume 487: 545–574.
33. Chaudhury S, Lyskov S, Gray JJ. PyRosetta: a script-based interface for implementing molecular modeling algorithms using Rosetta. *Bioinformatics.* 2010; 26: 689–691. doi: [10.1093/bioinformatics/btq007](https://doi.org/10.1093/bioinformatics/btq007) PMID: [20061306](https://pubmed.ncbi.nlm.nih.gov/20061306/)
34. Fleishman SJ, Leaver-Fay A, Corn JE, Strauch E-M, Khare SD, Koga N, et al. RosettaScripts: A Scripting Language Interface to the Rosetta Macromolecular Modeling Suite. *PLoS ONE.* 2011; 6: e20161. doi: [10.1371/journal.pone.0020161](https://doi.org/10.1371/journal.pone.0020161) PMID: [21731610](https://pubmed.ncbi.nlm.nih.gov/21731610/)
35. Holland IM, Lieberherr KJ. Object-oriented Design. *ACM Comput Surv.* 1996; 28: 273–275. doi: [10.1145/234313.234421](https://doi.org/10.1145/234313.234421)
36. Wang C, Bradley P, Baker D. Protein-protein docking with backbone flexibility. *J Mol Biol.* 2007; 373: 503–519. doi: [10.1016/j.jmb.2007.07.050](https://doi.org/10.1016/j.jmb.2007.07.050) PMID: [17825317](https://pubmed.ncbi.nlm.nih.gov/17825317/)
37. Viklund H, Elofsson A. OCTOPUS: improving topology prediction by two-track ANN-based preference scores and an extended topological grammar. *Bioinformatics.* 2008; 24: 1662–1668. doi: [10.1093/bioinformatics/btn221](https://doi.org/10.1093/bioinformatics/btn221) PMID: [18474507](https://pubmed.ncbi.nlm.nih.gov/18474507/)
38. Koehler Leman J, Mueller R, Karakas M, Woetzel N, Meiler J. Simultaneous prediction of protein secondary structure and transmembrane spans. *Proteins Struct Funct Bioinforma.* 2013; 81: 1127–1140. doi: [10.1002/prot.24258](https://doi.org/10.1002/prot.24258)
39. Kozma D, Simon I, Tusnády GE. PDBTM: Protein Data Bank of transmembrane proteins after 8 years. *Nucleic Acids Res.* 2012; gks1169. doi: [10.1093/nar/gks1169](https://doi.org/10.1093/nar/gks1169)
40. Lomize MA, Lomize AL, Pogozheva ID, Mosberg HI. OPM: orientations of proteins in membranes database. *Bioinforma Oxf Engl.* 2006; 22: 623–625. doi: [10.1093/bioinformatics/btk023](https://doi.org/10.1093/bioinformatics/btk023)
41. Tusnády GE, Dosztányi Z, Simon I. TMDet: web server for detecting transmembrane regions of proteins by using their 3D coordinates. *Bioinforma Oxf Engl.* 2005; 21: 1276–1277. doi: [10.1093/bioinformatics/bti121](https://doi.org/10.1093/bioinformatics/bti121)
42. Lomize MA, Pogozheva ID, Joo H, Mosberg HI, Lomize AL. OPM database and PPM web server: resources for positioning of proteins in membranes. *Nucleic Acids Res.* 2012; 40: D370–D376. doi: [10.1093/nar/gkr703](https://doi.org/10.1093/nar/gkr703) PMID: [21890895](https://pubmed.ncbi.nlm.nih.gov/21890895/)
43. Yarov-Yarovoy V, DeCaen PG, Westenbroek RE, Pan C-Y, Scheuer T, Baker D, et al. Structural basis for gating charge movement in the voltage sensor of a sodium channel. *Proc Natl Acad Sci.* 2012; 109: E93–E102. doi: [10.1073/pnas.1118434109](https://doi.org/10.1073/pnas.1118434109) PMID: [22160714](https://pubmed.ncbi.nlm.nih.gov/22160714/)
44. Baugh EH, Lyskov S, Weitzner BD, Gray JJ. Real-Time PyMOL Visualization for Rosetta and PyRosetta. *PLoS ONE.* 2011; 6: e21931. doi: [10.1371/journal.pone.0021931](https://doi.org/10.1371/journal.pone.0021931) PMID: [21857909](https://pubmed.ncbi.nlm.nih.gov/21857909/)
45. DeLano. PyMOL molecular graphics system. 2002.
46. DeLano W. The PyMOL Manual: Compiled Graphics Objects (CGOs) and Molscript Ribbons [Internet]. Available: <http://pymol.sourceforge.net/newman/user/toc.html>
47. Kellogg EH, Leaver-Fay A, Baker D. Role of conformational sampling in computing mutation-induced changes in protein structure and stability. *Proteins.* 2011; 79: 830–838. doi: [10.1002/prot.22921](https://doi.org/10.1002/prot.22921) PMID: [21287615](https://pubmed.ncbi.nlm.nih.gov/21287615/)
48. Humphris EL, Kortemme T. Design of Multi-Specificity in Protein Interfaces. *PLoS Comput Biol.* 2007; 3: e164. doi: [10.1371/journal.pcbi.0030164](https://doi.org/10.1371/journal.pcbi.0030164) PMID: [17722975](https://pubmed.ncbi.nlm.nih.gov/17722975/)
49. Kortemme T, Baker D. A simple physical model for binding energy hot spots in protein–protein complexes. *Proc Natl Acad Sci.* 2002; 99: 14116–14121. doi: [10.1073/pnas.202485799](https://doi.org/10.1073/pnas.202485799) PMID: [12381794](https://pubmed.ncbi.nlm.nih.gov/12381794/)
50. Moon CP, Fleming KG. Side-chain hydrophobicity scale derived from transmembrane protein folding into lipid bilayers. *Proc Natl Acad Sci.* 2011; doi: [10.1073/pnas.1103979108](https://doi.org/10.1073/pnas.1103979108)
51. Hong H, Park S, Flores Jiménez RH, Rinehart D, Tamm LK. Role of Aromatic Side Chains in the Folding and Thermodynamic Stability of Integral Membrane Proteins. *J Am Chem Soc.* 2007; 129: 8320–8327. doi: [10.1021/ja068849c](https://doi.org/10.1021/ja068849c) PMID: [17564441](https://pubmed.ncbi.nlm.nih.gov/17564441/)
52. Hessa T, Meindl-Beinker NM, Bernsel A, Kim H, Sato Y, Lerch-Bader M, et al. Molecular code for transmembrane-helix recognition by the Sec61 translocon. *Nature.* 2007; 450: 1026–1030. doi: [10.1038/nature06387](https://doi.org/10.1038/nature06387) PMID: [18075582](https://pubmed.ncbi.nlm.nih.gov/18075582/)
53. White SH, von Heijne G. How translocons select transmembrane helices. *Annu Rev Biophys.* 2008; 37: 23–42. doi: [10.1146/annurev.biophys.37.032807.125904](https://doi.org/10.1146/annurev.biophys.37.032807.125904) PMID: [18573071](https://pubmed.ncbi.nlm.nih.gov/18573071/)
54. Mishra VK, Palgunachari MN, Segrest JP, Anantharamaiah GM. Interactions of synthetic peptide analogs of the class A amphipathic helix with lipids. Evidence for the snorkel hypothesis. *J Biol Chem.* 1994; 269: 7185–7191. PMID: [8125930](https://pubmed.ncbi.nlm.nih.gov/8125930/)

55. Hsieh D, Davis A, Nanda V. A knowledge-based potential highlights unique features of membrane α -helical and β -barrel protein insertion and folding. *Protein Sci.* 2012; 21: 50–62. doi: [10.1002/pro.758](https://doi.org/10.1002/pro.758) PMID: [22031179](https://pubmed.ncbi.nlm.nih.gov/22031179/)
56. White SH. Membrane Protein Insertion: The Biology—Physics Nexus. *J Gen Physiol.* 2007; 129: 363–369. doi: [10.1085/jgp.200709741](https://doi.org/10.1085/jgp.200709741) PMID: [17438116](https://pubmed.ncbi.nlm.nih.gov/17438116/)
57. Senes A, Chadi DC, Law PB, Walters RFS, Nanda V, DeGrado WF. Ez, a Depth-dependent Potential for Assessing the Energies of Insertion of Amino Acid Side-chains into Membranes: Derivation and Applications to Determining the Orientation of Transmembrane and Interfacial Helices. *J Mol Biol.* 2007; 366: 436–448. doi: [10.1016/j.jmb.2006.09.020](https://doi.org/10.1016/j.jmb.2006.09.020) PMID: [17174324](https://pubmed.ncbi.nlm.nih.gov/17174324/)
58. Khatib F, Cooper S, Tyka MD, Xu K, Makedon I, Popović Z, et al. Algorithm discovery by protein folding game players. *Proc Natl Acad Sci.* 2011; doi: [10.1073/pnas.1115898108](https://doi.org/10.1073/pnas.1115898108)
59. Tyka MD, Keedy DA, Andre I, DiMaio F, Song Y, Richardson DC, et al. Alternate states of proteins revealed by detailed energy landscape mapping. *J Mol Biol.* 2011; 405: 607–618. doi: [10.1016/j.jmb.2010.11.008](https://doi.org/10.1016/j.jmb.2010.11.008) PMID: [21073878](https://pubmed.ncbi.nlm.nih.gov/21073878/)
60. Choe H-W, Kim YJ, Park JH, Morizumi T, Pai EF, Krauss N, et al. Crystal structure of metarhodopsin II. *Nature.* 2011; 471: 651–655. doi: [10.1038/nature09789](https://doi.org/10.1038/nature09789) PMID: [21389988](https://pubmed.ncbi.nlm.nih.gov/21389988/)
61. Chaudhury S, Berrondo M, Weitzner BD, Muthu P, Bergman H, Gray JJ. Benchmarking and Analysis of Protein Docking Performance in Rosetta v3.2. *PLoS ONE.* 2011; 6: e22477. doi: [10.1371/journal.pone.0022477](https://doi.org/10.1371/journal.pone.0022477) PMID: [21829626](https://pubmed.ncbi.nlm.nih.gov/21829626/)
62. Gray JJ, Moughon S, Wang C, Schueler-Furman O, Kuhlman B, Rohl CA, et al. Protein-protein docking with simultaneous optimization of rigid-body displacement and side-chain conformations. *J Mol Biol.* 2003; 331: 281–299. PMID: [12875852](https://pubmed.ncbi.nlm.nih.gov/12875852/)
63. Méndez R, Leplae R, Lensink MF, Wodak SJ. Assessment of CAPRI predictions in rounds 3–5 shows progress in docking procedures. *Proteins Struct Funct Bioinforma.* 2005; 60: 150–169. doi: [10.1002/prot.20551](https://doi.org/10.1002/prot.20551)
64. André I, Bradley P, Wang C, Baker D. Prediction of the structure of symmetrical protein assemblies. *Proc Natl Acad Sci.* 2007; 104: 17656–17661. doi: [10.1073/pnas.0702626104](https://doi.org/10.1073/pnas.0702626104) PMID: [17978193](https://pubmed.ncbi.nlm.nih.gov/17978193/)
65. DiMaio F, Leaver-Fay A, Bradley P, Baker D, André I. Modeling Symmetric Macromolecular Structures in Rosetta3. *PLoS ONE.* 2011; 6: e20450. doi: [10.1371/journal.pone.0020450](https://doi.org/10.1371/journal.pone.0020450) PMID: [21731614](https://pubmed.ncbi.nlm.nih.gov/21731614/)
66. Kc DB. Improving consensus structure by eliminating averaging artifacts. *BMC Struct Biol.* 2009; 9: 12. doi: [10.1186/1472-6807-9-12](https://doi.org/10.1186/1472-6807-9-12) PMID: [19267905](https://pubmed.ncbi.nlm.nih.gov/19267905/)
67. Kelm S, Shi J, Deane CM. iMembrane: homology-based membrane-insertion of proteins. *Bioinformatics.* 2009; 25: 1086–1088. doi: [10.1093/bioinformatics/btp102](https://doi.org/10.1093/bioinformatics/btp102) PMID: [19237449](https://pubmed.ncbi.nlm.nih.gov/19237449/)
68. Roy A, Kucukural A, Zhang Y. I-TASSER: a unified platform for automated protein structure and function prediction. *Nat Protoc.* 2010; 5: 725–738. doi: [10.1038/nprot.2010.5](https://doi.org/10.1038/nprot.2010.5) PMID: [20360767](https://pubmed.ncbi.nlm.nih.gov/20360767/)
69. Zhang Y. I-TASSER server for protein 3D structure prediction. *BMC Bioinformatics.* 2008; 9: 40. doi: [10.1186/1471-2105-9-40](https://doi.org/10.1186/1471-2105-9-40) PMID: [18215316](https://pubmed.ncbi.nlm.nih.gov/18215316/)
70. Kelm S, Shi J, Deane CM. MEDELLER: homology-based coordinate generation for membrane proteins. *Bioinformatics.* 2010; 26: 2833–2840. doi: [10.1093/bioinformatics/btq554](https://doi.org/10.1093/bioinformatics/btq554) PMID: [20926421](https://pubmed.ncbi.nlm.nih.gov/20926421/)
71. Tsuchiya Y, Nakamura H, Kinoshita K. Discrimination between biological interfaces and crystal-packing contacts. *Adv Appl Bioinforma Chem AABC.* 2008; 1: 99–113.
72. Krissinel E, Henrick K. Inference of Macromolecular Assemblies from Crystalline State. *J Mol Biol.* 2007; 372: 774–797. doi: [10.1016/j.jmb.2007.05.022](https://doi.org/10.1016/j.jmb.2007.05.022) PMID: [17681537](https://pubmed.ncbi.nlm.nih.gov/17681537/)
73. Morozov AV, Kortemme T, Baker D. Evaluation of Models of Electrostatic Interactions in Proteins. *J Phys Chem B.* 2003; 107: 2075–2090. doi: [10.1021/jp0267555](https://doi.org/10.1021/jp0267555)
74. Kilambi KP, Gray JJ. Rapid Calculation of Protein pKa Values Using Rosetta. *Biophys J.* 2012; 103: 587–595. doi: [10.1016/j.bpj.2012.06.044](https://doi.org/10.1016/j.bpj.2012.06.044) PMID: [22947875](https://pubmed.ncbi.nlm.nih.gov/22947875/)
75. Kilambi KP, Reddy K, Gray JJ. Protein-protein docking with dynamic residue protonation states. *PLoS Comput Biol.* 2014; 10: e1004018. doi: [10.1371/journal.pcbi.1004018](https://doi.org/10.1371/journal.pcbi.1004018) PMID: [25501663](https://pubmed.ncbi.nlm.nih.gov/25501663/)
76. Slusky JSG, Dunbrack RL. Charge asymmetry in the proteins of the outer membrane. *Bioinforma Oxf Engl.* 2013; 29: 2122–2128. doi: [10.1093/bioinformatics/btt355](https://doi.org/10.1093/bioinformatics/btt355)
77. Zhang S-Q, Jiang J, Ulas G, Wu Y, Tong W, DeGrado WF. Deciphering Regulatory Mechanism of the Juxtamembrane Region in Thrombopoietin Receptor Activation. *Biophys J.* 2014; 106: 103a. doi: [10.1016/j.bpj.2013.11.639](https://doi.org/10.1016/j.bpj.2013.11.639)
78. Lazaridis T. Implicit solvent simulations of peptide interactions with anionic lipid membranes. *Proteins.* 2005; 58: 518–527. doi: [10.1002/prot.20358](https://doi.org/10.1002/prot.20358) PMID: [15609352](https://pubmed.ncbi.nlm.nih.gov/15609352/)

79. Andersen OS, Koeppe RE. Bilayer Thickness and Membrane Protein Function: An Energetic Perspective. *Annu Rev Biophys Biomol Struct.* 2007; 36: 107–130. doi: [10.1146/annurev.biophys.36.040306.132643](https://doi.org/10.1146/annurev.biophys.36.040306.132643) PMID: [17263662](https://pubmed.ncbi.nlm.nih.gov/17263662/)
80. Van Meer G, Voelker DR, Feigenson GW. Membrane lipids: where they are and how they behave. *Nat Rev Mol Cell Biol.* 2008; 9: 112–124. doi: [10.1038/nrm2330](https://doi.org/10.1038/nrm2330) PMID: [18216768](https://pubmed.ncbi.nlm.nih.gov/18216768/)
81. Strandberg E, Killian JA. Snorkeling of lysine side chains in transmembrane helices: how easy can it get? *FEBS Lett.* 2003; 544: 69–73. PMID: [12782292](https://pubmed.ncbi.nlm.nih.gov/12782292/)
82. Wiedman G, Fuselier T, He J, Searson PC, Hristova K, Wimley WC. Highly Efficient Macromolecule-Sized Poration of Lipid Bilayers by a Synthetically Evolved Peptide. *J Am Chem Soc.* 2014; 136: 4724–4731. doi: [10.1021/ja500462s](https://doi.org/10.1021/ja500462s) PMID: [24588399](https://pubmed.ncbi.nlm.nih.gov/24588399/)
83. Wimley WC, Hristova K. Interfacially active peptides and proteins. *Biochim Biophys Acta BBA—Bio-membr.* 2014; 1838: 2139. doi: [10.1016/j.bbamem.2014.06.001](https://doi.org/10.1016/j.bbamem.2014.06.001)
84. Sapay N, Estrada-Mondragon A, Moreau C, Vivaudou M, Crouzy S. Rebuilding a macromolecular membrane complex at the atomic scale: Case of the Kir6.2 potassium channel coupled to the muscarinic acetylcholine receptor M2. *Proteins Struct Funct Bioinforma.* 2014; 82: 1694–1707. doi: [10.1002/prot.24521](https://doi.org/10.1002/prot.24521)
85. Bagriantsev SN, Chatelain FC, Clark KA, Alagem N, Reuveny E, Minor DL. Tethered protein display identifies a novel Kir3.2 (GIRK2) regulator from protein scaffold libraries. *ACS Chem Neurosci.* 2014; 5: 812–822. doi: [10.1021/cn5000698](https://doi.org/10.1021/cn5000698) PMID: [25028803](https://pubmed.ncbi.nlm.nih.gov/25028803/)
86. Rathinakumar R, Walkenhorst WF, Wimley WC. Broad-Spectrum Antimicrobial Peptides by Rational Combinatorial Design and High-Throughput Screening: The Importance of Interfacial Activity. *J Am Chem Soc.* 2009; 131: 7609–7617. doi: [10.1021/ja8093247](https://doi.org/10.1021/ja8093247) PMID: [19445503](https://pubmed.ncbi.nlm.nih.gov/19445503/)