, , pp. 1–12

doi:

Generating Multi-state Conformations of P-type ATPases with a Diffusion Model

Jingtian Xu[1] and Yong Wang[2,*]

[1]College of Life Sciences, Zhejiang University, 310027, Hangzhou, China
[*]yongwang_isb@zju.edu.cn

Abstract

Understanding and predicting the diverse conformational states of membrane proteins is essential for elucidating their biological functions. Despite advancements in computational methods, accurately capturing these complex structural changes remains a significant challenge. In this study, we introduce a method for predicting diverse functional states of membrane protein conformations using a diffusion model. Our approach integrates forward and backward diffusion processes, incorporating state classifiers and additional conditioners to control the generation gradient of conformational states. We specifically target the P-type ATPases, a key membrane transporter, for which we curated and expanded a structural dataset. By employing a graph neural network with a custom membrane constraint, our model generates precise structures for P-type ATPases across different functional states. This approach represents a significant step forward in computational structural biology and holds great potential for studying the dynamics of other membrane proteins.

Key words: P-type ATPase, Diffusion model, AI generation of protein structure, Multi-state conformations, Membrane transporter

Introduction

Proteins are inherently dynamic rather than static entities, with their functions deeply rooted in their capacity to change their conformations. These conformational changes are crucial as they allow proteins to interact with diverse ligands, substrates, and other macromolecules, thereby driving intricate biochemical processes. Thus the research on proteins in different states plays a vital role in elucidating their functional mechanisms, regulatory pathways, and interactions within biological systems. For instance, conformational flexibility is essential for enzyme catalysis, where substrate binding induces a shift in the enzyme's structure, enhancing its catalytic efficiency [1]. Similarly, protein-protein interactions often rely on the dynamic nature of protein conformations, enabling the formation of transient complexes necessary for signal transduction and cellular regulation [2]. Analyzing these multi-state conformations is thus integral to comprehending the roles proteins play in biological systems [3, 4].

Currently, multiple methods exist for studying protein multi-state conformations, each with unique advantages and limitations. Wet lab techniques, such as X-ray crystallography, nuclear magnetic resonance spectroscopy (NMR), and cryo-electron microscopy, are primary methods for protein structure determination but often requires extensive time, effort, and financial resources. On the other hand, *in silico* molecular dynamics (MD) simulations offers a state-of-the-art and feasible alternative, employing numerical solutions of Newtonian mechanics to explore protein conformational space. This approach effectively captures the microscopic processes of protein conformational changes, particularly in smaller systems, such as mini-protein folding and ligand binding [5, 6]. However, it struggles with larger protein systems with conformational changes occurring on timescales beyond tens of microseconds[7].

To address these challenges, advanced enhanced sampling techniques have been developed, though they bring increased computational demands and complexity [8, 9]. Moreover, recent progress in deep learning has notably advanced structural biology, with initiatives to harness it for predicting multi-state protein conformations [10, 11]. Methods such as finely tuned AlphaFold2 enable adjustments in structural templates, the depth of multiple sequence alignments (MSA), and the masking of MSA details to precisely characterize specific protein conformational states [12]. For example, state-annotated templated GPCR databases assist Alphafold2 in modeling both active and inactive states of GPCR at high accuracy [13]. Nevertheless, despite these advancements, these approaches continue to grapple with issues related to interpretability and stability across diverse conditions.

Generative AI has become a prominent research field, aiming to fit the distribution of target variables and sample from it through neural networks. Initially applied to the field of computer vision, generative AI has recently achieved significant breakthroughs in life sciences, particularly in structural biology. As a prominent framework in generative AI, diffusion models are renowned for their capability to accurately represent complex distributions by emulating the underlying characteristics of the training data [14, 15]. In the context of biology, particularly in predicting protein

© The Author . Published by Oxford University Press. All rights reserved. For permissions, please e-mail: journals.permissions@oup.com

structures, diffusion models offer a powerful approach due to their ability to model intricate conformational landscapes. For instance, the AlphaFold3 model replaces the structure model in AlphaFold2 with a diffusion model, enabling the prediction of joint structures of complexes, including proteins, nucleic acids, small molecules, ions, and modified residues, without excessive specialization [16]. Chroma, another notable advancement, is a protein and protein complex generation model based on diffusion models, capable of directly designing new protein structures and sequences [17]. AlphaFlow and ESMFlow combine the flow matching framework with the AlphaFold2 and ESMFold models, accurately capturing conformational flexibility, positional distributions, and higher-order ensemble observables for proteins [18]. Distributed graphormer (DiG) is a novel deep generative method aimed at approximating equilibrium distributions and efficiently sampling diverse and chemically plausible structures of molecular systems [19]. Diffold transforms AlphaFold2 into a diffusion model, achieving efficient sampling comparable to MD simulations [20]. By introducing physical prior knowledge, ConfDiff incorporates a force-guided network with a mixture of data-based score models, enabling it to prioritize generating conformations with lower potential energy, which effectively enhances sampling quality [21]. However, these models may still require further validation to ensure their predictions closely align with experimental data, and they have not been fine-tuned for membrane proteins. Membrane proteins present unique challenges due to their complex environments and interactions within the lipid bilayer, which are not always adequately captured by current models. Additionally, these generative models may struggle to generate structures in specific conformational states, which is crucial for understanding the functional mechanisms of proteins. This highlights important areas for future research and development in the application of generative AI in structural biology.

Building on the impressive achievements of generative AI in structural biology, we have developed a method utilizing the diffusion model to predict multi-state protein conformations. Inspired by the concept of heating and annealing, we propose a forward conditional-backward process for the diffusion model. By adjusting the duration of noise scale, we can modulate the generated conformational space, thereby controlling the degree of conformational changes. Using Bayesian theorem, we decompose the score function of the backward process into multiple conditioners, enabling precise control over the gradient direction and guiding structure generation. This method is highly modular, requiring minimal training of our conditioners to accurately predict multi-state conformations of membrane proteins, thus enhancing our ability to model complex biological systems.

In this study, P-type ATPases [22] served as the model system to validate our methodology. We constructed a dataset, trained a classifier, and developed membrane constraint strategies specifically tailored for membrane proteins. These strategies provide essential contextual cues that enhance the prediction accuracy of membrane protein structures, thereby elevating the biological relevance of our computational models. Consequently, we were able to accurately generate distinct conformations of P-type ATPases in specific functional states. By comparing these generated structures to the latest cryo-EM structures of hSPCA1 [23], we found a high degree of similarity between our model-generated conformations and their experimentally observed counterparts in similar states. Additionally, our analysis of the structural domains revealed

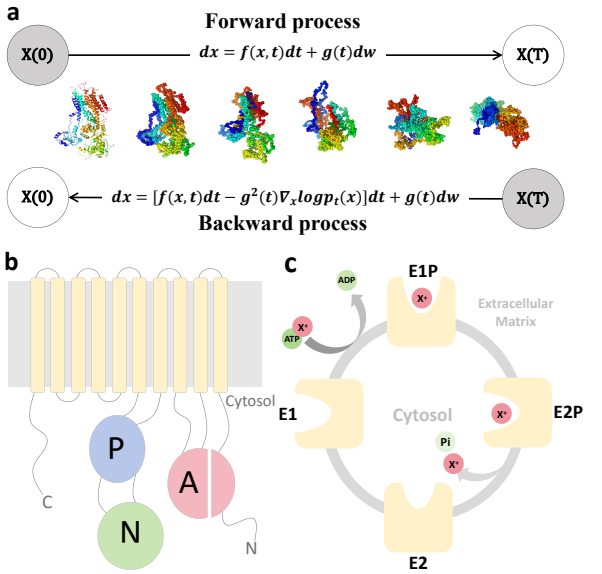


Fig. 1. Diffusion model architecture and P-type ATPase conformational cycle. a) Illustration of diffusion model on protein structure generation in the form of a stochastic differential equation with noise space on the right and data space on the left. b) Schematic representation of the topology of the core subunit of a P-type ATPase. The actuator (A) domain is colored in red, the phosphorylation (P) domain in blue, and the nucleotide-binding (N) domain in green. (c) Illustrates the typical conformational cycle of a P-type ATPase with ions ATPase as the example, showing the sequential changes that occur during its conformational cycle.

that the generated P-type ATPases authentically mirror the characteristic features of their respective states. We believe our method offers versatility and can be adapted by researchers to employ these conditioners in a forward conditional-backward process, facilitating rapid conformational transitions across a variety of membrane proteins.

Preliminary

Diffusion model on protein structure

The diffusion model can be unified under the Stochastic Differential Equation (SDE) framework [24]. As shown in the Fig. 1, X $\in$ $R^{N\times 3}$ represents the backbone structure of a protein, namely the Cartesian coordinates of all heavy atom including Cα, C, N, O; t $\in$ [0, T] denotes the time step, and W is the standard Brownian motion regarded as disturbance term. The functions $f(x,t)$ and $g(t)$ are the drift coefficient and diffusion coefficient, respectively. The term $\nabla_x logp_t(x)$ is referred to as the score function, where $p_t(x)$ represents the distribution of the protein structure at time t. Protein structure can be sampled during inference utilizing either an empirical distribution [25] or predicted through a neural network.

The forward process transforms the native protein conformation $x(0)$ into a noisy, disordered conformation $x(T)$ by adding noise. For convenience, $p_T(x)$ is chosen to be a simple, easy-to-sample distribution (usually Gaussian distribution). The backward process, derived from the given forward process, gradually denoises the noisy conformation $x(T)$ back into the physically reasonable protein conformation

$x(0)$. The score function in this framework plays a key role in generation, which is often trained through deep learning. Utilizing the exponential-logarithm map, the process can be discretized similar to the Euler–Maruyama step in Euclidean space as geodesic random walk [26].

To add conditions into backward process, it is necessary to replace $\nabla_x logp_t(x)$ with $\nabla_x logp_t(x|y)$ [27], where y represents the given condition. The precise $\nabla_x logp_t(x|y)$ can guide the gradient of structural generation during the backward process to obtain the protein structure under the specified condition.

Conformation and substrate transport cycle of P-type ATPases

The P-type ATPases are a class of membrane proteins widely present in eukaryotes, which utilize ATP hydrolysis to translocate various substrates across membranes. These ATPases harness ATP energy to transport substrates spanning from protons to phospholipids and are present in every cell type across all domains of life. Despite their overall low sequence conservation, they share a common structural fold and mechanism as shown in Fig. 1 [28]). The transmembrane (TM) region typically consists of 10 TM helices, crucial for substrate recognition and binding. The nucleotide-binding domain (N) functions as an intrinsic kinase binding ATP within a conserved amino acid motif. The phosphorylation domain (P) contains reactive aspartate residues subjected to transient phosphorylation. The actuator domain (A), serving as an internal phosphatase, houses conserved amino acid motifs that trigger dephosphorylation of the phosphorylated intermediate. Classified into five subfamilies (P1–P5), each specializing in distinct substrates, P-type ATPases generally follow the Post-Albers cycle [29]. This cycle involves transitions between two primary functional states (E1 and E2) and their phosphorylated forms (E1P and E2P). During each cycle, regulation of inward and/or outward transport occurs through the opening and closing of cytoplasmic and exoplasmic pathways, enabling access to TM binding sites with specific affinities and selectivities for transported substrates.

Materials and Methods

Structural dataset construction

We constructed our dataset using experimental structures from the IPR023298 superfamily of P-type ATPases, based on InterPro databank classifications [30]. A total of 321 P-type ATPase structures were retrieved from the Protein Data Bank (PDB) released before May 2023. We streamlined this data by retaining only one major α-subunit from each crystal structure with duplicated chains and removing heteroatoms and ligand subunits to avoid interference in later analysis.

Each structure was labeled with state tags (see Table 2), classifying P-type ATPases into four main states—E1, E1P, E2P, and E2—based on experimental conditions. Despite the presence of additional functional states (e.g., E1-ATP, E1P-ADP), we simplified the classification for consistency. For instance, E1P and E1P-ADP were grouped as E1P, and E1 and E1-ATP were categorized as E1. Structures that couldn't be clearly assigned or did not fit the typical Post-Albers cycle states, including auto-inhibited states [31, 32], were excluded. Additionally, E2Pi was considered a hybrid state, comprising 80% E2P and 20% E2.

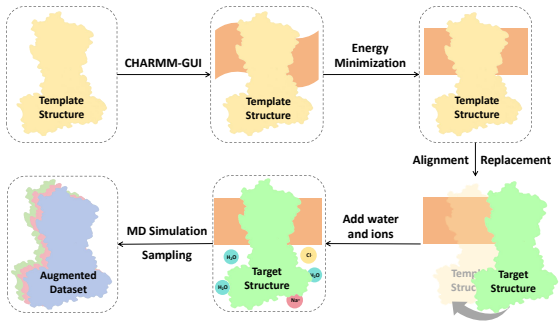


Fig. 2. Workflow for Atomistic Modeling of P-type ATPases. This flowchart outlines the process of constructing atomistic models of various P-type ATPases. Template structures were used to generate models, which were then embedded in lipid bilayers and solvated. Energy minimization and molecular dynamics simulations were performed to refine the structures.

To refine the dataset, we calculated TM-scores for all structure pairs and created a TM-score matrix. This matrix was used for hierarchical clustering to correct any incorrect state classifications.

Data augmentation using molecular dynamics simulations

To enhance and balance the dataset distribution across different states, and to ensure the physical plausibility of the augmented data, we employed all-atom molecular dynamics (MD) simulations. Given that the conformational changes of P-type ATPases between states typically occur over time scales longer than typical MD simulations [33], the nanosecond-scale simulations used here are intended only to sample conformational states in local equilibrium, rather than capturing large-scale state transitions. To accurately represent the membrane environment of P-type ATPases, we developed a protocol to model the membrane protein in a lipid bilayer, as detailed in the pipeline shown in Fig. 2.

Initially, four P-type ATPases were selected as templates: KdpFABC in the E1-ATP state, SPCA1a in Ca^{2+}-bound E1-ATP and E2 states, and yeast P4-ATPase, Dnf1, in the E2P state. These templates, with representative sizes ranging from 675 to 1158 amino acids, were used to construct atomistic models of other P-type ATPases of different sizes (see Table 1) using the CHARMM-GUI web server [34]. DOPC was used for modeling the lipid bilayer.

Next, energy minimization was performed using GROMACS-2022.5 [35] to optimize the structures. For each structure in the dataset, PyMOL (version 2.5) was used to align and position it onto the membrane of the corresponding template, while DOPC molecules within 1 $\AA$ of the protein were removed to avoid steric clashes. Water molecules and 0.15M sodium and chloride ions were subsequently added to the model using the GROMACS gmx tool. After energy minimization, temperature equilibration, and pressure equilibration, MD simulations were conducted for 5-10 ns using the CHARMM36m force field [36]. Detailed MD parameters are provided in the supplementary materials. The structures were uniformly collected along the MD trajectories at intervals of 1 ns per frame, with the number of MD structures for each state detailed in Table 2.

Table 1. Four templates for modeling P-type ATPases in different sizes.

PDB ID	P-type ATPase	Template size (a.a.)	Model size (a.a.)	PBC box
7ZRG	KdpFABC in E1-ATP state	675	(603,800)	13nm x 13nm x 16nm
7YAH	SPCA1a in Ca^{2+}-bound E1-ATP state	877	(800,900)	13.5nm x 13.5nm x 17nm
1IWO	SERCA1a in E2 state	994	(900,1085)	15nm x 15nm x 18nm
7WHV	yeast P4-ATPase, Dnf1 in E2P state	1158	(1085,1294)	16nm x 16nm x 18nm

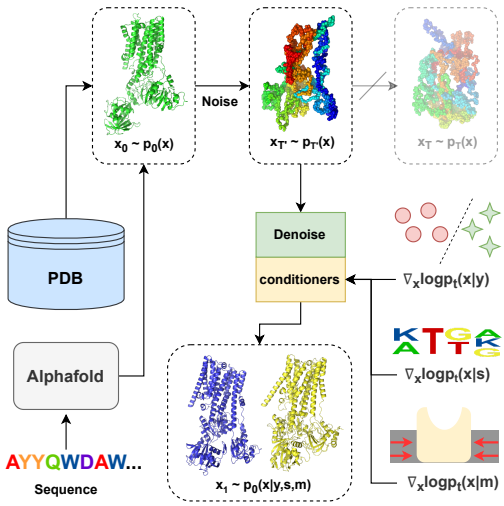


Fig. 3. Generation workflow for multi-state conformations of protein. Firstly, we need to obtain a reference structure observed through experimental methods or predicted by AlphaFold. This structure is then subjected to noise addition to x(t) during the forward process and subsequently denoised toward x(0) in a specific state through the conditional backward process with state classifier, sequence conditioner and membrane constraintor.

Forward conditional-backward process

Inspired by the forward process and backward process of diffusion model, we propose a generation workflow for multi-state conformations of protein as shown in Fig. 3, which is based on a novel method named Forward Conditional-Backward Process. The process can be described by the following integral equation:

$$\hat{x}_0 = x_0 + \int_0^{T'} [f(x_t, t)dt + g(t)dw] + \int_{T'}^{2T'} \{-[f(x_\tau, \tau) + g^2(\tau)\nabla_x logp_\tau(x_\tau|y)]d\tau + g(\tau)dw\} \tag{1}$$

In Eq. 1, $\tau = 2T' - t$, y represents the specific conditions, $\hat{x}_0$ is the predicted conformation, x_0 is the reference structure (obtained from experiment observation or sequence prediction), and $T' \in (0, 1)$ denotes the noise duration. This equation describes the forward noise injection and backward denoising process toward the specific domain for a known structure. The noise time step T' limits the scale of perturbation, thus preventing the complete elimination of information from the reference structure x_0. Increasing T' appropriately can enhance diversity but may require more reverse steps, while decreasing T' can ensure the authenticity of the results but narrows the sampling space. Intuitively, it simulates the heating and annealing process. Here, the heating process aims to strengthen exploration, while the annealing process ensures authenticity.

For the reverse process, we recommend the probability flow ODE scheme [24] as follows:

$$dx = [f(x, \tau) - \frac{1}{2}g^2(\tau)\nabla_x logp_\tau(x|y)]d\tau \tag{2}$$

The ODE approach is a special case where the noise variance is zero. By eliminating the noise in the backward process, the model is reformulated as a continuous normalizing flow, which allows for efficient and precise likelihood computation using the adjoint method [37]. This is reasonable when our goal is to improve model accuracy rather than diversity.

The score function in the formulation can treat arbitrary combinations of conditions if we model the joint event y as factorizing into independent sub-events $y_1, ..., y_N$ [17]. Then we have the posterior score:

$$\begin{aligned}\nabla_x logp_t(x|y_1, ..., y_N) &= \nabla_x logp_t(x) + \nabla_x logp_t(y_1, ..., y_N|x) \\ &\quad - \nabla_x logp_t(y_1, ..., y_N) \\ &= \nabla_x logp_t(x) + \sum_{i=1}^{N} \nabla_x logp_t(y_i|x)\end{aligned} \tag{3}$$

Thus the posterior score can be split into the sum of origin score and different scores of conditioners. Specifically, the backward process involves three conditioners in our system: state conditioner, sequence conditioner, and membrane constraintor. State conditioner refers to the specified state for generated conformations of protein. Sequence conditioner implies the constraint of sequence given by the reference structure. The membrane constraintor is specific to membrane proteins, representing the constraints imposed by the plasma membrane on the TM domain. The final posterior score should be:

$$\begin{aligned}\nabla_x logp_t(x|y, s, m) =& \nabla_x logp_t(x) + \nabla_x logp_t(y|x) \\ &+ \nabla_x logp_t(s|x) + \nabla_x logp_t(m|x)\end{aligned} \tag{4}$$

where x represents the structural coordinates, y represents the specified state, s represents the given sequence, and m represents the membrane constraint. The Eq. 4 includes the assumption that condition y, s, and m are mutually independent. The final expression consists of four terms. The first term $p_t(x)$ and the third term $p_t(s|x)$ represent the stable distribution of the protein structure at time t and the distribution of the protein sequence given the structure x at time t, respectively. We utilize the models of these two terms from the open-source project Chroma to avoid unnecessary training. The second term, $p_t(y|x)$, represents the conformational state distribution of the protein structure x at time t. It is therefore referred to as the state classifier, which requires self-training. The final term, $p_t(m|x)$, represents the distribution of the protein structure x under membrane constraints at time t, which is thus referred to as the membrane constraint. The design of this term will be discussed later.

Architecture and training of state classifier

The architecture is implemented based on PyTorch. Given the limited data, we selected a simple architecture comprising GNNEncoder, Attention Pooling layer, and MLP, for classifier training.

In GNNEncoder, we first encoded the diffusion time using random Fourier featurization. Next, backbone coordinates were utilized to extract 2-mer and chain-based distances and orientations [17]. The sum of these components was passed to the later layer. We constructed a k-NN graph where heavy atoms on the backbone serve as nodes and interactions between atoms as edges. Specifically, in our model, we set that each atom is linked to its 20 nearest neighbors. The Graph Convolutional Networks architecture [38] was adopted for message passing, which is defined as follows:

$$m_i = \sum_{j \in N(i)} \frac{m_{i,j,k+1}}{\sqrt{deg(i) \cdot deg(j)}}, m_{i,j} = \Phi_{m,k}([h_{i,k}, h_{j,k}, e_{i,j,k}]) \tag{5}$$

$$h_{i,k+1} = h_{i,k} + \Phi_{h,k}([h_{i,k}, m_i]) \tag{6}$$

$$e_{i,j,k+1} = e_{i,j,k} + \Phi_{e,k}([h_{i,k}, h_{j,k}, e_{i,j,k}]) \tag{7}$$

where $h_{i,k}$ represents the SE(3)-invariant hidden representation of protein structure and nodes of graph after k iterations; $e_{i,j,k}$ represents the edge of graph between $h_{i,k}$ and $h_{j,k}$ after k iterations; $\Phi_{m,k}$, $\Phi_{e,k}$ and $\Phi_{h,k}$ are various MLP at the k iteration; $N(i)$ represents all the neighbors of node i; $deg(i)$ represents the number of neighbors of node i. Our GNNEncoder model comprises four layers, with a node feature dimension of 512 and an edge feature dimension of 192. The $\Phi_{m,k}$ has two hidden layers, each with a hidden dimension of 128. The $\Phi_{h,k}$ includes two hidden layers, each with a hidden dimension of 256. The $\Phi_{e,k}$ also has two hidden layers, each with a hidden dimension of 128. We set the activation function of all MLP above to ReLU.

Attention pooling layer is required to aggregate feature representations based on importance weights and enhance model focus on relevant information for downstream tasks [39]. We define the attention pooling as follow:

$$head_i = softmax(\frac{w_i^k (w_i^Q)^T X^T}{\sqrt{h}}) X W_i^v \tag{8}$$

$$output = Concat[head_1, ..., head_n] W_o \tag{9}$$

Here, m is the input dimension, h is the output dimension and n is the number of head; $W_i^k \in R^{1\times h}, W_i^Q \in R^{m\times h}, W_i^V \in R^{m\times h}, W_i^O \in R^{n\times h}$ are the learnable matrices. In our model, we set n = 8 and h =512. Leveraging the attention mechanism helps the model to capture the relational information between residues.

Finally, the classification MLP has one hidden layer, with a hidden dimension of 64 and ELU as activation function. The MLP outputs a four-dimensional vector representing four states for classification, with cross-entropy as the loss function to supervise the model learning.

We introduce the weights of GNNEncoder from Chroma as our encoder section. Then We divided the dataset into a training set and a validation set in a ratio of 0.8:0.2. The classifier predicts the following labels: E1, E1P, E2P, and E2. We used one-hot encoding for label processing and quantified the loss of each label prediction through cross-entropy. Subsequently, we optimized the parameters using the Adam optimizer [24] with default momentum settings (beta = (0.9, 0.999)). We set the batch size to 8, linearly decreased the learning rate from 1e-5 to 1e-6, and trained for a total of 200 epochs. During training, we first uniformly sampled a time t ∈ (0,0.65) and sampled noise from the global covariance distribution [17]. Then we injected this noise into the backbone coordinates, and fed the results back to the classifier. Next, we predicted labels, calculated the loss, and updated the parameters.

Membrane constraintors

The lipid bilayer thickness of the cytoplasmic membrane is 4-5 nm [40], which can constrain membrane proteins through various interactions with the transmembrane domain, including hydrophobic interactions, electrostatic interactions, and hydrogen bonds [41]. To simulate the interaction of membrane proteins with the plasma membrane in a realistic environment, it is necessary to abstract these constraints and apply them during the backward process of the Diffusion Model. Here, we propose two schemes: a coarse-grained constraint based on Root Mean Square Deviation (RMSD) and a fine-grained constraint based on radius of gyration (Rg).

RMSD is a commonly used metric in structural biology, primarily for comparing the similarity between two different protein structures. In the RMSD scheme, we adopt a form of reconstruction guidance based on the optimal RMSD of the TM domain in the current denoised structure. Specifically, at time t, we define the membrane constraint $logp_t(m|x)$ as:

$$logp_t(m|x) = ReLU(min_{T\in SE(3)}(||x_0^M - T \cdot \hat{x}_0^M||_2) - \rho) \tag{10}$$

$$\hat{x}_0 = \frac{x_t}{\alpha_t} - \frac{\sqrt{1-\alpha_t^2}}{\alpha_t} RZ, Z \sim N(0, I) \tag{11}$$

In Eq. 10 and Eq. 11, $\hat{x}_0^M$ and x_0^M represent the TM domain of predicted structure at t=0 and the reference structure respectively. We set the z-axis as the normal to the plasma membrane, and define z_{min} as the minimum z-component of all atoms in the structure. Atoms within the range z ∈ [z_{min}+5, z_{min}+45] are selected as $\hat{x}_0^M$ and x_0^M. T is the optimal transformation matrix that minimizes the difference $\hat{x}_0^M$ and x_0^M. ρ is the threshold controlling the constraint range. This approach references the true structure to create a gradient that restricts the TM domain, effectively simulating membrane constraints. The advantage of this method lies in its simplicity and computational efficiency. Although this approach might incorporate information from the original state, the influence can be mitigated by adjusting weights.

Rg is a parameter that describes the size of a protein in space, reflecting the average distribution of all atoms relative to its center of mass. It is used to measure the compactness and spatial extension of the protein's structure. In the Rg scheme, we constrain the diffusion of non-physical conformations in the TM domain by a penalty function based on the Rg of the TM region. Specifically, we define the constraint as $p_t(Rg^2 < k|X_t)$, where k is the threshold, and Rg^2 is defined as follows:

$$Rg^2 = \sum_{i=1}^{n} (\bar{X} - \hat{X}_{0,j}^M)^2, \bar{X} = \frac{1}{n}\sum_{i=1}^{n} \hat{X}_{0,j}^M \tag{12}$$

In Eq. 12, the $\hat{X}_0^M$ representing TM domain at time t=0 is predicted by X_t (see Appendix for details). The advantage of this approach is that it avoids the bias towards specific state structures and only imposes constraints based on the threshold k. However, our calculations revealed that Rg^2

follows a generalized χ^2 distribution (see Appendix), indicating that the distribution of Rg^2 does not have a closed-form solution. Consequently, specific values can only be computed using simulation methods such as Monte Carlo simulations. With limited computational power, the results of Monte Carlo simulations are unsatisfactory, and gradient vanishing issues are likely to occur during backpropagation. Therefore, this approach is more suitable for detailed simulations when ample computational resources and time are available.

Structure generation of P-type ATPases in specific state

In conditional generation models, adjusting the weight of the classifier is a common strategy [42]. Modulating the classifier weight can balance output diversity and adherence to classification guidance. If the weight is too high, the generated structures may overly depend on the classifier's guidance, resulting in homogeneous or unnatural content. Conversely, if the weight is too low, the generated content may deviate from the target category. In practical applications, we do not strictly follow Eq. 4 but adopt the following form:

$$\nabla_x logp_t(x) + w_1 \nabla_x logp_t(y|x) + w_2 \nabla_x logp_t(s|x) + w_3 \nabla_x logp_t(m|x) \tag{13}$$

By adjusting various weights, we can control the diversity, reliability, and membrane constraint strength of the results. Ultimately, when generating protein conformations with the model, we used the following parameters: (1) Chroma parameters set step=500, $\lambda_0 = 10$, $\varphi = 2$, noise time step $T' = 0.65$. Additionally, to ensure quality over diversity, we applied the probability flow ODE in the backward process. (2) Classifier weight w1=1, gradient clipping value max_norm=10. (3) Sequence conditioner weight w2=0.75. (4) Membrane constrainer using RMSD scheme, weight w3=0.1, threshold ρ set to 4. The generated structures currently contain only the backbone without side chains, which however can be modeled later using other algorithms, such as Modeller [43], if needed.

Structure analysis and visualization

We used PyMOL for structural visualization; TMtools [44] to calculate TM scores; Matplotlib to plot line graphs, violin plots, and bar charts; the R package Bio3D [45] for PCA; and improved scripts from Bozzi et al. [46] to calculate and average the distance difference matrices of all pairwise structures between two states.

Results

Structural dataset construction and its augmentation using molecular dynamics simulations

We constructed a structural dataset of P-type ATPases by curating PDB structures (released before 2023-05-01). Rigorous quality control was applied to ensure data integrity (see Methods). To navigate the complexity of P-type ATPase conformational space, we adopted a simplified classification scheme, categorizing structures into four primary states: E1, E1P, E2P, and E2. An intermediate state, E2Pi, was defined to represent a hybrid of E2P and E2 states. Structures lacking clear state assignment or deviating from the canonical Post-Albers cycle were excluded. The initial dataset, comprising 57 E1, 45 E2, 86 E2P, 47 E1P, and 66 E2Pi structures, was augmented through molecular dynamics simulations, resulting in a final dataset of 1247 structures.

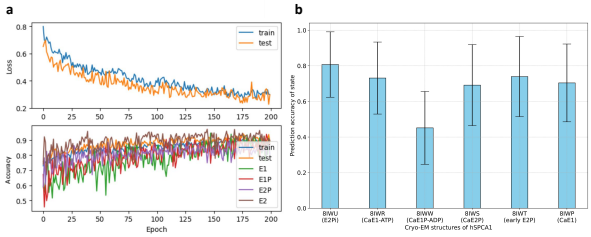


Fig. 4. Training and evaluation of functional state classifier. (a) Learning curves illustrating the evolution of loss and accuracy during classifier training and evaluation on a separate test set. (b) Classifier accuracy on experimentally determined structures excluded from the training set. Error bars indicate the standard deviation in predictions.

Development and Validation of a Functional State Classifier

We began by developing a functional state classifier to predict protein states from their structural data. As illustrated in Figure 4, the classifier's performance on a validation set stabilized with a loss of 0.3 and an accuracy of 0.85. This suggests that the model can reliably identify state-specific structural features even in the presence of noise. However, the classifier exhibited lower accuracy for the E1 state compared to other states. We hypothesize that this is due to the substantial structural variability within the E1 state (e.g., apo and Ca^{2+}-bound E1), making it challenging for the model to learn common features shared by all E1 substrates.

To assess the generalizability of our classifier, we tested it to a recently published dataset of human secretory-pathway Ca^{2+}-ATPases (hSPCA1) cryo-EM structures [23], which were not included in the training dataset. hSPCA1 plays a critical role in maintaining Ca^{2+} homeostasis, and these cryo-EM structures represent a series of intermediate states, revealing a near-complete conformational cycle. During prediction, we normalized the four-dimensional vector (representing the probability of these four functional states) output of the classifier using softmax and took the value at the corresponding state position as the prediction accuracy. To account for the inherent randomness of neural network-based classifiers, we repeated the prediction for each conformation state 100 times.

As shown in Fig. 4, the classifier accurately predicted the majority of conformations with accuracies ranging from 65% to 80%. However, the performance was significantly lower for the Ca^{2+}-bound E1P-ADP state of hSPCA1 (PDB ID: 8IWW). We attribute this to the underrepresentation of early E1P states (including E1P-ADP) in the training data, which comprises only 2.1% of E1P structures. This result highlights a potential bias in our classifier towards certain metastable states.

Generation of conformations of hSPCA1 in specific states

To evaluate the capability of our model, we generated a series of conformations of hSPCA1 using our diffusion model, starting with the cryo-EM structure of hSPCA1 in the Ca^{2+}-bound E1 state (PDB ID: 8IWP) as the input structure. By altering the classifier's targets, we directed the model to sample 15 times for each of the four states: E1, E1P, E2P, and E2, resulting

Table 2. Summary of structure dataset of P-type ATPases.

State	Ligands	No. of experimental structures	No. of simulated structures	Total No.
E1	AMPPCP/ATP	57	228	285
E1P	AMPPCP, AlF_4^-	47	235	282
E2P	BeF_3^-, VO_3^-, AlF_4^-	86	258	344
E2Pi[a]	AlF_4^-, MgF_4^{2-}	66	0	66
E2	Thapsigargin, CPA, BHQ	45	225	270
Total		301	946	1247

[a] E2Pi was modeled as a composite state, incorporating 80% of E2P features and 20% of E2 features.

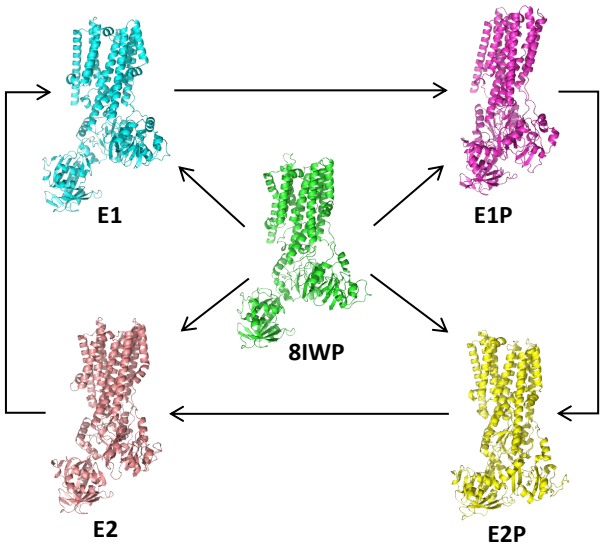


Fig. 5. Diffusion Model-Generated Structures of hSPCA1. Our diffusion model utilized the cryo-EM structure of hSPCA1 in the Ca^{2+}-bound E1 state (PDB ID: 8IWP) as a template to generate representative structures for four functional states. These states, E1 (cyan), E1P (magenta), E2P (yellow), and E2 (red), are depicted in cartoon representation.

in a total of 60 protein conformations. The structures were generated using a single NVIDIA GeForce RTX 3090, with each structure taking approximately 10-20 minutes to generate, amounting to a total of 18 hours. The representative structures are well-compacted (Fig. 5).

To further assess the quality of these generated structures, we first used RMSD as a metric to examine the conformational similarity of the cytoplasmic domains, which exhibit high structural identity in different states. As shown in Fig. 6, the RMSD for most domains remained around 2 $\AA$, indicating stability during the generation process.

We then compared the generated structures with the experimental structures of hSPCA1, considered the ground truth, using TM scores to evaluate precision and validity. The TM score matrix in Fig. 6 shows that the TM scores between the generated and experimental structures at corresponding states (8IWP, 8IWR, 8IWW, 8IWS, 8IWT, and 8IWU, representing the states of CaE1, CaE1-ATP, CaE1P-ADP, CaE2P, early E2P, and E2≈P, respectively) remained above 0.70. This indicates that our method accurately predicts the structures in different states. Additionally, the lower TM scores between the generated structures and non-corresponding state structures reflect the state-specific characteristics of the generated structures.

To further explore structural differences and similarities, we performed PCA on the generated conformations. The resulting PCA plot in Fig. 6 shows distinct clustering of different states, indicating that the model effectively captured state-dependent conformational variations. We observed some diversity among the samples, particularly during the E1 state, characterized by domain activities in the cytoplasm, resulting in a broader sampling space. However, there is some overlap between different states, partly due to the inherent stochastic diffusion properties of the diffusion model. Despite using a flow model to eliminate noise during the backward process, the deep learning-based denoising network still exhibits some uncertainty. Additionally, heterogeneity among structures in the same state within the training set, such as E1, which can be further divided into metastable states like E1(apo) and E1-2Ca^{2+}, results in some divergence in the classifier's guidance of model generation. This can be attributed to the inherent stochasticity of diffusion models, training set heterogeneity, and the complexity of representing metastable states.

Subsequently, we used the eigenvector of PCA to characterize the structural features. The eigenvectors PC1 and PC2 were extracted to plot the line graph shown in Fig. 6. Note that we took the absolute values of the eigenvectors when plotting the line graph to avoid influencing the weight judgment of the sites. We observed that the weights exhibit a step-like pattern and are closely related to the distribution of domains: the weights of the regions corresponding to the A, P, and N domains are generally at higher levels, indicating that these cytoplasmic domains are active during conformational changes and undergo significant variations. In contrast, most of the TM regions exhibit generally lower weights, suggesting that these TM helices are relatively stable during conformational changes. We also noticed that TM helices 1-2 and 9-10 have higher weights compared to other TM regions, suggesting they may be the main factors controlling ATPase cavity changes, which is related to their vertical membrane displacement during the transition from E1P to E2P [22].

Generated structures capture the subtle conformational differences between functional states

Based on structural biology studies [22, 47, 48, 49], we selected specific features to depict structural details across various states, assessing whether the generated structures conform to reasonable conformational changes: (1) Distance between cytoplasmic domains, as significant displacements of these domains occur during phosphorylation and dephosphorylation phases; (2) Arrangement of TM regions, which is closely related to the binding and release of substrates.

As shown in Fig. 7, we computed the distances between the centers of mass of the cytoplasmic domains. We observed that the distance between domains A and N is the longest in state E1, decreases during the E1 to E1P transition, and gradually

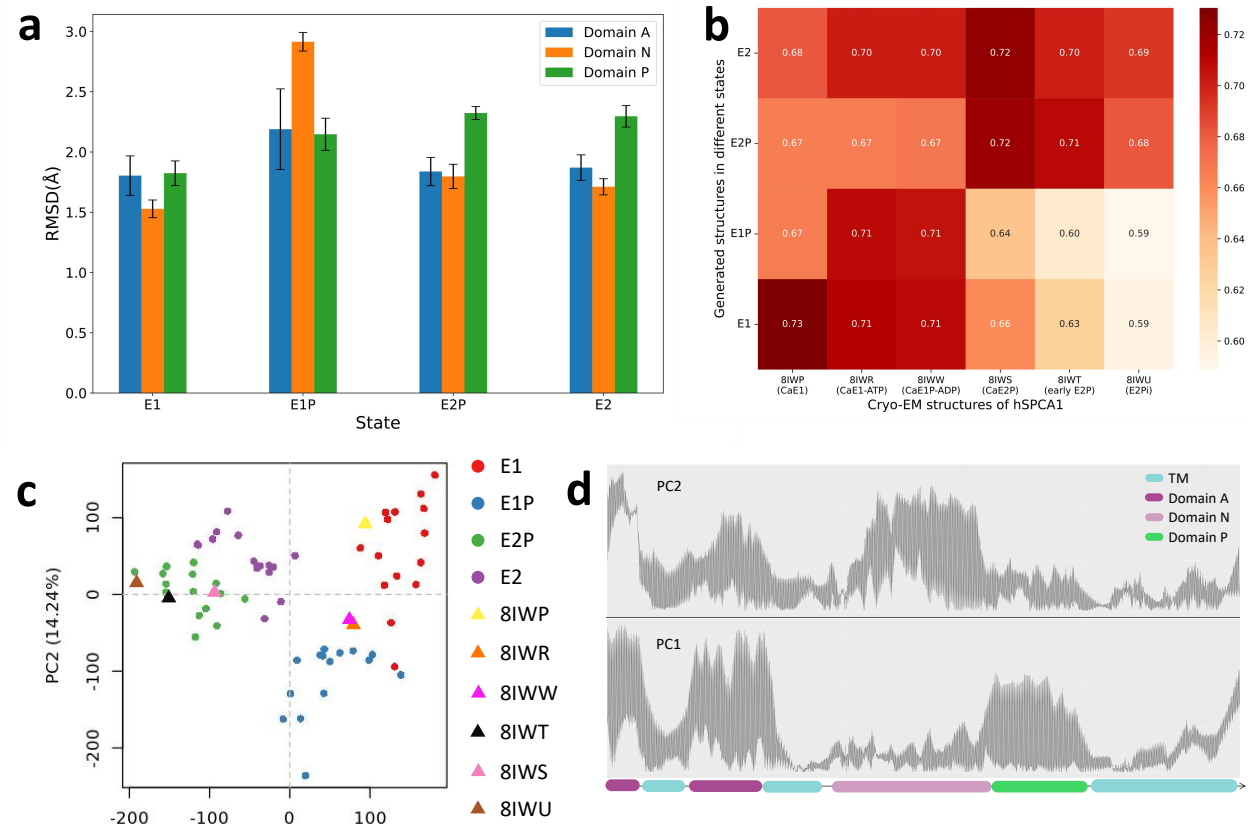


Fig. 6. Structural diversity and conformational dynamics of generated hSPCA1 across different states. a) RMSD of individual domains (A: residues 136-248, N: 358-547, P: 548-687) between generated and reference (8IWP) structures. Error bars indicate standard deviation. b) TM score matrix comparing generated and experimental structures. Rows represent experimental structures, columns represent averaged generated structures per state. c) Principal component analysis (PCA) of generated hSPCA1 conformations. States E1, E1P, E2P, and E2 are represented by red, blue, green, and purple dots, respectively. Experimental structures, including 8IWP (CaE1), 8IWR (CaE1-ATP), 8IWW (CaE1P-ADP), 8IWS (CaE2P), 8IWT (early E2P), 8IWU (E2≈P), are shown as triangles. Clustering of generated conformations with experimental structures is observed for E1, E1P, and E2P states, but not for E2 due to lack of experimental data. d) Line plot of PCA eigenvector, highlighting residue contributions to the principal component. The horizontal axis represents domains, with the vertical axis indicating weight magnitude.

increases in subsequent states to its original distance, forming a cycle. The small fluctuation in the distance between domains A and N during conformational changes indicates stable relative distances between A and N domains. Small changes between domains P and N correspond to the presence of peptide chain linkages between domains. Distances between domains in the reference structures also demonstrate the rationality of the generated structures.

During the transition from E1P to E2P, the TM regions rearrange to accommodate substrate loading. We used a distance difference matrix to analyze these rearrangements in the TM helices. As shown in Fig. 7, the generated structures closely match the ground-truth structures. TM1-2, TM3-4, and TM5-10 display low internal RMSD, indicating stability within these helices. In contrast, we observed substantial RMSD fluctuations between TM1-2, TM3-4, and TM5-10, aligning with their role as the substrate loading site [50].

Conclusion

In this study, we propose a forward conditional-backward process based on diffusion models to predict multi-state conformations of proteins, using P-type ATPase as a case study to validate the feasibility of our method. The results demonstrate that our approach is highly efficient, achieving speedups of several orders of magnitude compared to traditional MD simulations. Moreover, our results are data-driven rather than subjective, which is crucial for an objective understanding of multi-state conformations. Importantly, our approach shows promise for generalizing to explore various states within the protein conformational space.

Our approach excels in generating specific states efficiently, which provides a valuable starting point for further exploration. However, for a comprehensive understanding of novel mechanisms and dynamic behaviors, it is essential to complement our method with physics-based approaches such as MD simulations. MD simulations can offer deeper insights into the detailed dynamics and interactions within the generated states, enabling a more thorough investigation of the underlying mechanisms. Thus, while our model is effective for rapidly exploring conformational spaces, future work should integrate it with physical rule-based methods to achieve a more complete and mechanistic understanding of protein behavior.

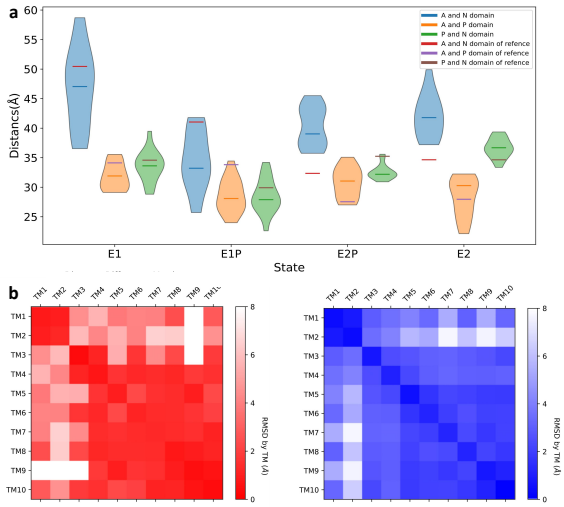


Fig. 7. Conformational differences between the generated structures in functional states. a) Inter-domain distances within hSPCA1 cytoplasmic domains. Distances were calculated between domain centers of mass, using 8IWP, 8IWW, 8IWS, and 8IWU as reference structures for E1, E1P, E2P, and E2 states, respectively. b) Distance difference matrices comparing E1P and E2P states. The left panel shows the real structure comparison using 8IWW (E1P-ADP) and 8IWS (E2P) as references. The right panel depicts the average distance difference matrix calculated from generated E1P and E2P structures.

Membrane constraintor scheme using radius of gyration

The objective function $p_t(Rg^2 < k|X_t)$ will calculate the probability that, under the given structure X_t at a specific time t, the radius of gyration of the final structure at time 0 is less than k. The formula for calculating Rg is:

$$Rg^2 = \sum_{i=1}^{n}(\bar{X} - \hat{X}_{0,i})^2, \bar{X} = \frac{1}{n}\sum_{j=1}^{n}\hat{X}_{0,i} \tag{14}$$

In this equation, $X_0 = [X_{0,1}, X_{0,2}, ..., X_{0,n}]$ represents the Cartesian coordinates of the heavy atoms in the structure at time t = 0. Here, the focus is on the positional relationships between the atoms rather than their physical and chemical properties. Therefore, for simplicity, all heavy atoms are considered to have the same mass. According to the forward process of the diffusion model:

$$X_t = \alpha_t X_0 + \sqrt{1-\alpha_t^2}RZ, Z \sim N(0, I) \tag{15}$$

In this equation, X_0 represents the true structure, and we can use this equation to estimate X_t from $\hat{X}_0$. Rearranging the terms, we get $\hat{X}_0 \sim N(\frac{X_t}{\alpha_t}, \frac{1-\alpha_t^2}{\alpha_t^2}RR^T)$, where R follows the globular covariance model [17]:

$$R = a\begin{bmatrix} vb^0 & & & & & \\ vb^1 & b^0 & & & & \\ vb^2 & b^1 & b^0 & & & \\ \vdots & & \ddots & \ddots & & \\ vb^{n-2} & & & b^1 & b^0 & \\ vb^{n-1} & & \cdots & b^2 & b^1 & b^0 \end{bmatrix} \tag{16}$$

where n is the number of atoms, $v = \frac{1}{\sqrt{1-b^2}}$,a and b are constants greater than 0. Suppose $Z = [Z_1, Z_2, ..., Z_n]$ are mutually independent standard normal distributions, then the noise $X_T = RZ$can be given by Z through the following formula:

$$X_{T,i} = a\sum_{k=2}^{i} b^{i-k}Z_k + a\frac{b^{i-1}}{\sqrt{1-b^2}}Z_1 \tag{17}$$

According to the property of normal distributions, it is known that $\bar{X} - \hat{X}_{0,i}$ still follows a normal distribution. Now, let's calculate the covariance between $\bar{X} - \hat{X}_{0,i}$ and $\bar{X} - \hat{X}_{0,j}$:

$$Cov(\bar{X} - \hat{X}_{0,i}, \bar{X} - \hat{X}_{0,j}) \tag{18}$$

$$= Cov(\bar{X}, \bar{X}) - Cov(\hat{X}_{0,i}, \bar{X}) - Cov(\bar{X}, \hat{X}_{0,j}) + Cov(\hat{X}_{0,i}, \hat{X}_{0,i}) \tag{19}$$

$$= \frac{1}{n^2}\sum_{m=1}^{n}\sum_{l=1}^{n}Cov(X_{0,m}, X_{0,l}) - \frac{1}{n}\sum_{l=1}^{n}Cov(X_{0,i}, X_{0,l}) \tag{20}$$

$$- \frac{1}{n}\sum_{m=1}^{n}Cov(X_{0,m}, X_{0,j}) + Cov(X_{0,i}, X_{0,j}) \tag{21}$$

For any j ¡ i, we have:

$$Cov(X_{0,i}, X_{0,j}) = \frac{1-\alpha_t^2}{\alpha_t^2}a^2(\frac{b^{i+j-2}}{1-b^2} + \sum_{k=2}^{j}b^{i+j-k}) \tag{22}$$

Obviously, $Cov(\bar{X}-\hat{X}_{0,i}, \bar{X}-\hat{X}_{0,j}) \neq 0$. Therefore, Rg^2 follows a generalized chi-squared distribution. The $p_t(Rg^2 < k|X_t)$ cannot be obtained in a closed-form solution and can only be approximated using methods such as Monte Carlo simulation.

Competing interests

No competing interest is declared.

Author contributions statement

Y.W. conceived the experiment(s), J.T.X. conducted the experiment and analysed the results. J.T.X. and Y.W. wrote and reviewed the manuscript.

Code availability

The source code and related scripts are available at GitHub.

Acknowledgements

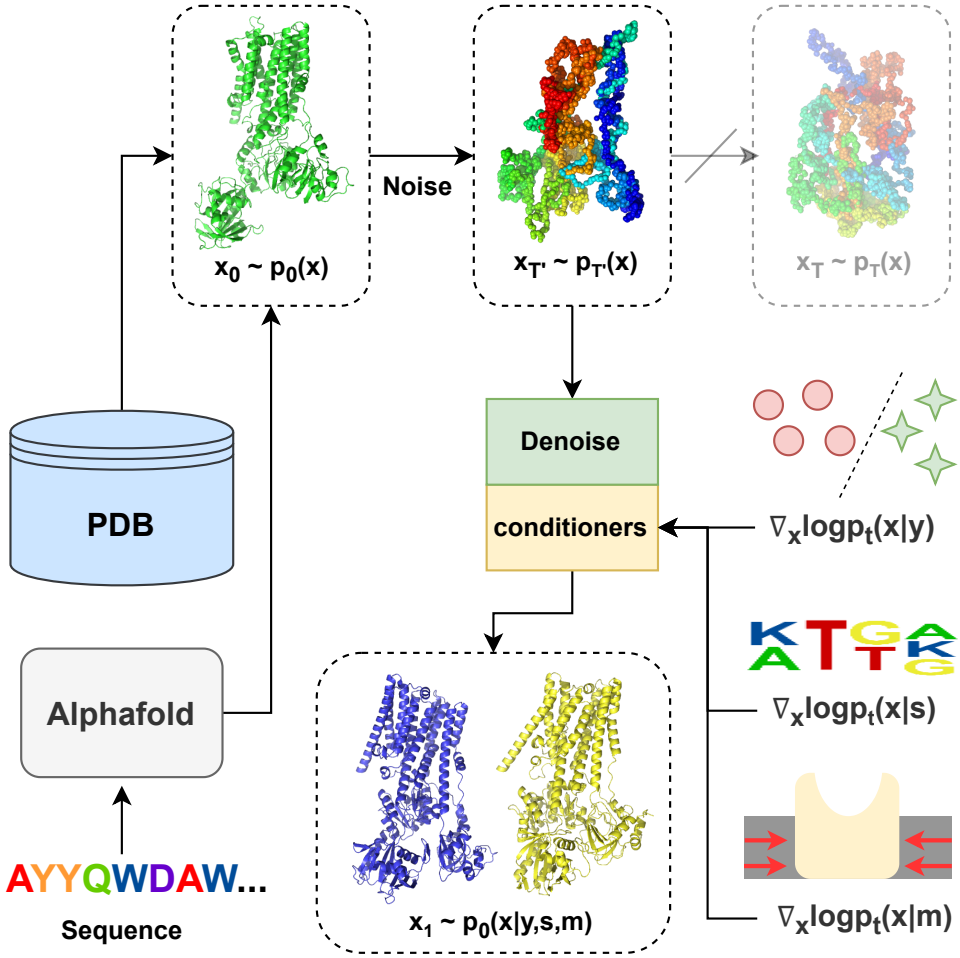
Yong Wang acknowledges the financial support of the National Natural Science Foundation of China (No. 32371300), the Zhejiang Provincial National Science Foundation of China (No. LZ24C050003), the National Key Research and Development Program of China (No. 2021YFF1200404), and computational support of the Information Technology Center and State Key Lab of CAD&CG at Zhejiang University.

References

1. Sharon Hammes-Schiffer. Catalytic efficiency of enzymes: A theoretical analysis. *Biochemistry*, 52(12):2012–2020, 2013.
2. M. Karplus and J. Kuriyan. Molecular dynamics and protein function. *Proceedings of the National Academy of Sciences*, 102(19):6679–6685, 2005.

3. A. Ramelot Theresa, Tejero Roberto, and T. Montelione Gaetano. Representing structures of the multiple conformational states of proteins. *Current Opinion in Structural Biology*, 83:102703, 2023.
4. Wang Yong, Bugge Katrine, B. Kragelund Birthe, and Lindorff-Larsen Kresten. Role of protein dynamics in transmembrane receptor signalling. *Current Opinion in Structural Biology*, 48:74–82, 2018.
5. Kresten Lindorff-Larsen, Stefano Piana, Ron O. Dror, and David E. Shaw. How fast-folding proteins fold. *Science*, 334, 2011.
6. Yong Wang, Elena Papaleo, and Kresten Lindorff-Larsen. Mapping transiently formed and sparsely populated conformations on a complex energy landscape. *eLife*, 5, 2016.
7. A. Hollingsworth Scott and O. Dror Ron. Molecular dynamics simulation for all. *Neuron*, 99(6):1129–1143, 2018.
8. Giovanni Bussi and Alessandro Laio. Using metadynamics to explore complex free-energy landscapes. *Nature Reviews Physics*, 2(4):200–212, 2020.
9. Wenhui Shen, Tong Zhou, and Xinghua Shi. Enhanced sampling in molecular dynamics simulations and their latest applications—a review. *Nano Research*, 16(12):13474–13497, 2023.
10. D. Sala, F. Engelberger, H. S. McHaourab, and J. Meiler. Modeling conformational states of proteins with alphafold. *Current Opinion in Structural Biology*, 81:102645, 2023.
11. B. Wallner. Afsample: improving multimer prediction with alphafold using massive sampling. *Bioinformatics*, 39(9), 2023.
12. Hannah K. Wayment-Steele, Adedolapo Ojoawo, Renee Otten, Julia M. Apitz, Warintra Pitsawong, Marc Hömberger, Sergey Ovchinnikov, Lucy Colwell, and Dorothee Kern. Predicting multiple conformations via sequence clustering and alphafold2. *Nature*, 2023.
13. Lim Heo and Michael Feig. Multi-state modeling of g-protein coupled receptors at experimental accuracy. *Proteins: Structure, Function, and Bioinformatics*, 90(11):1873–1885, 2022.
14. Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 2020-December, 2020.
15. Masatoshi Uehara, Yulai Zhao, Tommaso Biancalani, and Sergey Levine. Understanding reinforcement learning-based fine-tuning of diffusion models: A tutorial and review. 7 2024.
16. Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J. Ballard, Joshua Bambrick, Sebastian W. Bodenstein, David A. Evans, Chia Chun Hung, Michael O'Neill, David Reiman, Kathryn Tunyasuvunakool, Zachary Wu, Akvilė Žemgulytė, Eirini Arvaniti, Charles Beattie, Ottavia Bertolli, Alex Bridgland, Alexey Cherepanov, Miles Congreve, Alexander I. Cowen-Rivers, Andrew Cowie, Michael Figurnov, Fabian B. Fuchs, Hannah Gladman, Rishub Jain, Yousuf A. Khan, Caroline M.R. Low, Kuba Perlin, Anna Potapenko, Pascal Savy, Sukhdeep Singh, Adrian Stecula, Ashok Thillaisundaram, Catherine Tong, Sergei Yakneen, Ellen D. Zhong, Michal Zielinski, Augustin Žídek, Victor Bapst, Pushmeet Kohli, Max Jaderberg, Demis Hassabis, and John M. Jumper. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature 2024 630:8016*, 630:493–500, 5 2024.
17. J. B. Ingraham, M. Baranov, Z. Costello, K. W. Barber, W. Wang, A. Ismail, V. Frappier, D. M. Lord, C. Ng-Thow-Hing, E. R. Van Vlack, S. Tie, V. Xue, S. C. Cowles, A. Leung, J. V. Rodrigues, C. L. Morales-Perez, A. M. Ayoub, R. Green, K. Puentes, F. Oplinger, N. V. Panwar, F. Obermeyer, A. R. Root, A. L. Beam, F. J. Poelwijk, and G. Grigoryan. Illuminating protein space with a programmable generative model. *Nature*, 623(7989):1070–1078, 2023.
18. Bowen Jing, Bonnie Berger, and Tommi Jaakkola. Alphafold meets flow matching for generating protein ensembles, 2024.
19. Shuxin Zheng, Jiyan He, Chang Liu, Yu Shi, Ziheng Lu, Weitao Feng, Fusong Ju, Jiaxi Wang, Jianwei Zhu, Yaosen Min, He Zhang, Shidi Tang, Hongxia Hao, Peiran Jin, Chi Chen, Frank Noé, Haiguang Liu, and Tie-Yan Liu. Predicting equilibrium distributions for molecular systems with deep learning. *Nature Machine Intelligence*, 2024.
20. Jiahao Fan, Ziyao Li, Eric Alcaide, Guolin Ke, Huaqing Huang, and E. Weinan. Accurate conformation sampling via protein structural diffusion. *bioRxiv*, page 2024.05.20.594916, 2024.
21. Yan Wang, Lihao Wang, Yuning Shen, Yiqun Wang, Huizhuo Yuan, Yue Wu, and Quanquan Gu. Protein conformation generation via force-guided se(3) diffusion models, March 01, 2024 2024.
22. M. Dyla, M. Kjærgaard, H. Poulsen, and P. Nissen. Structure and mechanism of p-type atpase ion pumps. *Annu Rev Biochem*, 89:583–603, 2020.
23. M. Wu, C. Wu, T. Song, K. Pan, Y. Wang, and Z. Liu. Structure and transport mechanism of the human calcium pump spca1. *Cell Res*, 33(7):533–545, 2023.
24. Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations, November 01, 2020 2020.
25. Emiel Hoogeboom, Victor Garcia Satorras, Clément Vignac, and Max Welling. Equivariant diffusion for molecule generation in 3d, 2022.
26. Jiarui Lu, Bozitao Zhong, Zuobai Zhang, and Jian Tang. Str2str: A score-based framework for zero-shot protein conformation sampling, June 01, 2023 2023.
27. Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, June 01, 2020 2020.
28. Charlott Stock, Tomáš Heger, Sara Basse Hansen, Sigrid Thirup Larsen, Michael Habeck, Thibaud Dieudonné, Ronja Driller, and Poul Nissen. Fast-forward on p-type atpases: recent advances on structure and function. *Biochemical Society Transactions*, 51(3):1347–1360, 2023.
29. R. W. Albers. Biochemical aspects of active transport. *Annu Rev Biochem*, 36:727–56, 1967.
30. Typhaine Paysan-Lafosse, Matthias Blum, Sara Chuguransky, Tiago Grego, Beatriz Lázaro Pinto, Gustavo A. Salazar, Maxwell L. Bileschi, Peer Bork, Alan Bridge, Lucy Colwell, Julian Gough, Daniel H. Haft, Ivica Letunić, Aron Marchler-Bauer, Huaiyu Mi, Darren A. Natale, Christine A. Orengo, Arun P. Pandurangan, Catherine Rivoire, Christian J.A. Sigrist, Ian Sillitoe, Narmada Thanki, Paul D. Thomas, Silvio C.E. Tosatto, Cathy H. Wu, and Alex Bateman. Interpro in 2022. *Nucleic Acids Research*, 51, 2023.

31. Milena Timcenko, Joseph A. Lyons, Dovile Januliene, Jakob J. Ulstrup, Thibaud Dieudonné, Cédric Montigny, Miriam Rose Ash, Jesper Lykkegaard Karlsen, Thomas Boesen, Werner Kühlbrandt, Guillaume Lenoir, Arne Moeller, and Poul Nissen. Structure and autoregulation of a p4-atpase lipid flippase. *Nature*, 571, 2019.
32. Yong Wang, Joseph A. Lyons, Milena Timcenko, Felix Kümmerer, Bert L. De Groot, Poul Nissen, Vytautas Gapsys, and Kresten Lindorff-Larsen. Substrate transport and specificity in a phospholipid flippase. *bioRxiv*, 2020.
33. C. Kobayashi, Y. Matsunaga, J. Jung, and Y. Sugita. Structural and energetic analysis of metastable intermediate states in the e1p-e2p transition of ca(2+)-atpase. *Proc Natl Acad Sci U S A*, 118(40), 2021.
34. Sunhwan Jo, Taehoon Kim, Vidyashankara G. Iyer, and Wonpil Im. Charmm-gui: A web-based graphical user interface for charmm. *Journal of Computational Chemistry*, 29, 2008.
35. Mark James Abraham, Teemu Murtola, Roland Schulz, Szilárd Páll, Jeremy C. Smith, Berk Hess, and Erik Lindah. Gromacs: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. *SoftwareX*, 1-2, 2015.
36. Jing Huang, Sarah Rauscher, Grzegorz Nawrocki, Ting Ran, Michael Feig, Bert L. de Groot, Helmut Grubmüller, and Alexander D. MacKerell. Charmm36m: an improved force field for folded and intrinsically disordered proteins. *Nature Methods*, 14(1):71–73, 2017.
37. Will Grathwohl, Ricky T. Q. Chen, Jesse Bettencourt, Ilya Sutskever, and David Duvenaud. Ffjord: Free-form continuous dynamics for scalable reversible generative models, 2018.
38. Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks, September 01, 2016 2016.
39. Peter Meltzer, Marcelo Daniel Gutierrez Mallea, and Peter J. Bentley. Pinet: Attention pooling for graph classification, August 01, 2020 2020.
40. H. Sprong, P. van der Sluijs, and G. van Meer. How proteins move lipids and lipids move proteins. *Nat Rev Mol Cell Biol*, 2(7):504–13, 2001.
41. O. G. Mouritsen and M. Bloom. Models of lipid-protein interactions in membranes. *Annu Rev Biophys Biomol Struct*, 22:145–71, 1993.
42. Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis, May 01, 2021 2021.
43. Benjamin Webb and Andrej Sali. Comparative protein structure modeling using modeller. *Current Protocols in Bioinformatics*, 2016, 2016.
44. Y. Zhang and J. Skolnick. Scoring function for automated assessment of protein structure template quality. *Proteins*, 57(4):702–10, 2004.
45. Lars Skjaerven, Xin Qiu Yao, Guido Scarabelli, and Barry J. Grant. Integrating protein structural dynamics and evolutionary analysis with bio3d. *BMC Bioinformatics*, 15, 2014.
46. Aaron T. Bozzi, Christina M. Zimanyi, John M. Nicoludis, Brandon K. Lee, Casey H. Zhang, and Rachelle Gaudet. Structures in multiple conformations reveal distinct transition metal and proton pathways in an nramp transporter. *eLife*, 8:e41124, 2019.
47. Mateusz Dyla, Sara Basse Hansen, Poul Nissen, and Magnus Kjaergaard. Structural dynamics of p-type atpase ion pumps. *Biochemical Society Transactions*, 47(5):1247–1257, 2019.
48. Y. He, J. Xu, X. Wu, and L. Li. Structures of a p4-atpase lipid flippase in lipid bilayers. *Protein Cell*, 11(6):458–463, 2020.
49. R. L. López-Marqués. Mini-review: Lipid flippases as putative targets for biotechnological crop improvement. *Front Plant Sci*, 14:1107142, 2023.
50. M. Hiraizumi, K. Yamashita, T. Nishizawa, and O. Nureki. Cryo-em structures capture the transport cycle of the p4-atpase flippase. *Science*, 365(6458):1149–1155, 2019.


Noise
$x_0 \sim p_0(x)$
$x_{T'} \sim p_{T'}(x)$
$x_T \sim p_T(x)$
PDB
Denoise
conditioners
$\nabla_x \log p_t(x|y)$
Alphafold
$\nabla_x \log p_t(x|s)$
AYYQWDAW...
Sequence
$x_1 \sim p_0(x|y,s,m)$
$\nabla_x \log p_t(x|m)$