# Template-Guided 3D Molecular Pose Generation via Flow Matching and Differentiable Optimization

**Noémie Bergues**[1] * **Arthur Carré**[1,2,3] * **Paul Join-Lambert**[1]

**Brice Hoffmann**[1] **Arnaud Blondel**[2] **Hamza Tajmouati**[1]

[1] Iktos SA, 65 rue de Prony, 75017 Paris, France
[2] Institut Pasteur, Université Paris Cité, UMR 3528, Paris, France
[3] Doctoral school MTCI (ED 563), Paris Cité University, Paris, France

## Abstract

Predicting the 3D conformation of small molecules within protein binding sites is a key challenge in drug design. When a crystallized reference ligand (template) is available, it provides geometric priors that can guide 3D pose prediction. We present a two-stage method for ligand conformation generation guided by such templates. In the first stage, we introduce a molecular alignment approach based on flow-matching to generate 3D coordinates for the ligand, using the template structure as a reference. In the second stage, a differentiable pose optimization procedure refines this conformation based on shape and pharmacophore similarities, internal energy, and, optionally, the protein binding pocket. We introduce a new benchmark of ligand pairs co-crystallized with the same target to evaluate our approach and show that it outperforms standard docking tools and open-access alignment methods, especially in cases involving low similarity to the template or high ligand flexibility.

## 1 Introduction

Drug design is a complex and resource-intensive process, with timelines that can span a decade and development costs estimated to be on the order of one billion dollars per approved drug [Wouters et al., 2020]. To address these challenges, computational approaches have emerged to accelerate early-stage drug discovery and reduce associated costs. In particular, 3D-based methods show great promise by enabling *in silico* screening and optimization of candidate molecules, thereby reducing reliance on expensive and time-consuming experimental assays. Two major 3D computational approaches are widely used: 3D Ligand-Based (LB) and 3D Structure-Based (SB). LB methods leverage the 3D structure of known active compounds to identify or design structurally and functionally similar molecules [Acharya et al., 2011, Petrovic et al., 2022, Bolcato et al., 2022], whereas SB methods use the 3D structure of the target receptor to predict ligand binding modes or affinities [Anderson, 2003]. Popular examples include LB virtual screening tools such as Rapid Overlay of Chemical Structures (ROCS) [Hawkins et al., 2007], and SB molecular docking tools like AutoDock Vina [Trott and Olson, 2010] and Glide [Friesner et al., 2004].

Small molecule 3D alignment, or superposition, is an LB approach that aims to spatially align a molecule with a known 3D template. A variety of methods have been developed for this task, reflecting the diverse ways in which molecular similarity can be defined and assessed. In their

*Equal contribution. Correspondence to: {noemie.bergues, arthur.carre}@iktos.com

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

comprehensive review, Hönig *et al*. [Hönig et al., 2023] categorize six major classes of alignment strategies: Gaussian volume overlap, field-based, graph-based, volume overlap optimization, distance-based, and shape-based methods. Despite these advances, accurately aligning molecules remains challenging, particularly when the template and query share limited structural or chemical features. Additionally, small molecule superposition is inherently an LB approach [Hönig et al., 2023], meaning it does not incorporate any structural information from the target protein, potentially overlooking critical receptor-ligand interactions that can drive binding. Given that many drug discovery projects benefit from the availability of experimentally resolved receptor-ligand complexes [Hubbard, 2005, Congreve et al., 2014], there is an opportunity to develop hybrid methods that jointly leverage both ligand and receptor information to enhance the accuracy of molecular pose prediction.

This work introduces Flow Molecular Alignment with Pose Optimization (FMA-PO), a template-guided method for 3D molecular pose generation, which combines LB alignment with structure-aware refinement. FMA-PO employs a Flow Matching (FM) model conditioned on a 3D template molecule. The model generates 3D conformers for a query molecule from its 2D structure, aligning them spatially with the template. The poses are then refined via a coordinate-level differentiable optimization procedure integrating constraints based on shape alignment, pharmacophore similarity, binding pocket complementarity, and internal energy. To evaluate the performance of our method, we introduce AlignDockBench, a new benchmark comprising 369 protein-ligand (PL) template–query pairs. Unlike existing cross-docking benchmarks such as the one proposed by FitDock [Yang et al., 2022]— which focuses on ligand pairs with high chemical similarity—AlignDockBench also includes pairs with lower similarity. This enables a more challenging and realistic assessment of LB alignment and docking methods.

The main contributions of this work are as follows:

- A template-guided 3D molecular pose generation model with FM, capable of producing accurate ligand conformations even in cases of low template similarity.
- A pose refinement protocol operating directly on all atomic coordinates, improving pose accuracy and quality.
- AlignDockBench, a benchmark for evaluating template-based docking accuracy.

## 2 Related Work

**3D Alignment Methods.** In recent years, various advanced methods have been developed to improve the accuracy and efficiency of PL docking and virtual screening, including approaches based on 3D molecular alignment. One of the earliest and most widely adopted tools in this area is ROCS, which performs molecular alignment based on the overlap of Gaussian functions that represent molecular shape and, optionally, pharmacophoric features such as hydrogen bond donors, acceptors, and hydrophobic regions [Hawkins et al., 2007]. More recently, LS-align [Hu et al., 2018] introduced a fast atom-level alignment algorithm that integrates interatomic distances, atomic mass, and chemical bond information, enabling both rigid and flexible alignments. Another notable method, FitDock [Yang et al., 2022], improves docking accuracy by using template fitting to guide initial ligand conformations, followed by refinement with a scoring function derived from AutoDock Vina [Trott and Olson, 2010]. ROSHAMBO [Atwi et al., 2024] is another recent method that combines shape and pharmacophore similarity using Gaussian volume overlap to perform 3D molecular alignment and similarity scoring.

**Flow Matching for Biomolecular Applications.** FM is an emerging generative modeling framework that learns a continuous flow, mapping from a source distribution to a target distribution. Its adaptability has led to increasing adoption in biomolecular applications, including 3D conformer generation from 2D molecular graph [Hassan et al., 2025]. FM has also been applied to molecular docking, with methods such as Harmonic Flow [Stark et al., 2024] and FlowDock [Morehead and Cheng, 2024], generating ligand conformations within protein binding pockets. AlphaFlow [Jing et al., 2024] extends FM techniques to protein structure prediction, demonstrating their applicability to macromolecular modeling. Recently, FM has been applied to Molecular Dynamics (MD) through MD-GEN [Jing et al., 2025], a method that learns to generate physically realistic MD trajectories. These advances underscore the growing potential of FM to support a wide range of tasks in computational chemistry and structural biology.

# 3 Method

## 3.1 Overview

The present work introduces a Flow-Matching Molecular Alignment model followed by a Pose Optimization protocol (FMA-PO), a novel method for generating 3D molecular poses within a protein binding site from a 2D graph representation. FMA-PO uses a 3D reference ligand as a structural template (e.g., a crystallized ligand bound to the target protein), which serves as a spatial guide for predicting the pose of a query compound. The method consists of two main stages, as illustrated in Figure 1:

1. **Flow Molecular Alignment (FMA):** an initial 3D conformation of the ligand is generated using an FM model trained to produce a conformer aligned with the reference ligand (template), given a 2D molecular graph as input.
2. **Pose Optimization (PO):** the initial pose is refined through differentiable optimization on all coordinates using the following objectives:
   - Shape and pharmacophore-based alignment optimization: the pose is optimized to maximize alignment with the reference ligand using shape and pharmacophore scoring functions. Pharmacophore is defined as a 3D arrangement of features (hydrogen bond donors/acceptors, hydrophobic groups, aromatic rings, etc.) for a molecule using the RDKit chemoinformatics library [Landrum].
   - Internal energy optimization: the ligand's geometry is adjusted to minimize its internal (strain) energy, producing chemically valid conformations.
   - Protein pocket integration (optional): binding site information is incorporated to refine the pose, improve pharmacophore complementarity, and prevent steric clashes.

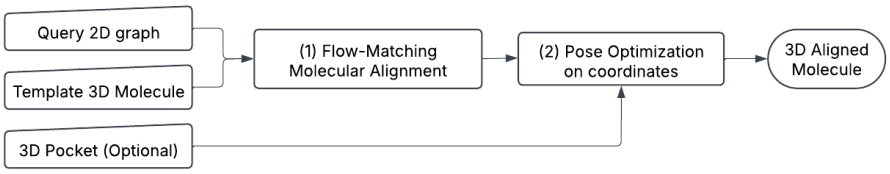


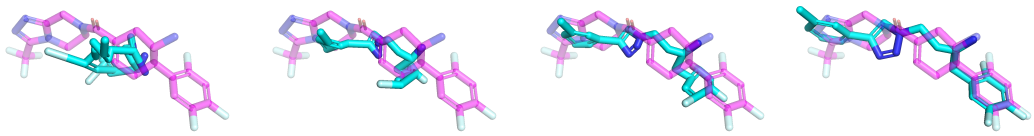

Figure 1: Overview of the FMA-PO pipeline. The method comprises two stages: (1) 3D FMA of the ligand to the template; and (2) Pose Optimization, including shape and pharmacophores scores, energy minimization, and optional refinement based on protein pocket information.

## 3.2 Flow-Matching Molecular Alignment

The FMA module generates 3D conformations of a query ligand conditioned on the 3D structure of a reference ligand. Starting from a random 3D conformation of the query 2D graph, the model progressively denoises it to produce conformations that reproduce the ground-truth pose, spatially aligned with the reference ligand. This alignment process is illustrated in Figure 2, which shows the evolution of the query conformation in blue, from initial noise ($t = 0$) to final alignment with the reference structure in pink ($t = 1$). Inspired by pocket-guided generative models like DiffDock [Corso et al., 2022], which condition ligand generation on protein pocket geometry, our method instead uses the 3D structure of a template ligand as the conditioning reference. An illustration of FMA model pipeline is provided in Figure S1.

**FM model.** FM [Lipman et al., 2022] is a generative modeling approach that constructs a time-continuous mapping $\phi : [0, 1] \times \mathbb{R}^d \to \mathbb{R}^d$ to transport a prior distribution $\rho_0$ to a target distribution $\rho_1$. This transformation is learned through a vector field (or velocity field) $v : [0, 1] \times \mathbb{R}^d \to \mathbb{R}^d$ governing the sample evolution, described by the ordinary differential equation (ODE):

$$\frac{d}{dt}\phi_t(x) = \mathbf{v}_t(\phi_t(x)), \quad \phi_0(x) = x.$$



(a) $t = 0$ (b) $t = 0.5$ (c) $t = 0.75$ (d) $t = 1$

Figure 2: Evolution of the query ligand's conformation (in blue) during the FM-MA denoising process. The template ligand is shown in pink. Starting from a noisy initial conformation ($t = 0$), the model progressively refines the 3D coordinates to align the query ligand with the template pose ($t = 1$).

That is, $\phi_t(x)$ represents the point $x$ transported along the vector field $v$ from time 0 to time $t$. To construct a feasible transport trajectory, [Tong et al., 2023] [Albergo et al., 2023] introduces a time-differentiable interpolant:

$$I_t : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^d \quad \text{such that} \quad I_0(x_0, x_1) = x_0 \quad \text{and} \quad I_1(x_0, x_1) = x_1.$$

Following [Hassan et al., 2025], we choose a linear interpolation between samples $x_0 \sim \rho_0$ and $x_1 \sim \rho_1$:

$$I_t(\mathbf{x}_0, \mathbf{x}_1) = \alpha_t \mathbf{x}_1 + \beta_t \mathbf{x}_0, \quad \text{where} \quad \alpha_t = t, \quad \beta_t = 1 - t,$$

and similarly define the conditional probability path $\rho_t(\mathbf{x}|\mathbf{x}_0, \mathbf{x}_1)$ as a Gaussian distribution centered on $I_t(\mathbf{x}_0, \mathbf{x}_1)$ with variance $\sigma_t^2$:

$$\rho_t(\mathbf{x}|\mathbf{x}_0, \mathbf{x}_1) = \mathcal{N}(x|I_t(\mathbf{x}_0, \mathbf{x}_1), \sigma_t^2 \mathbf{I}), \quad \text{where} \quad \sigma_t = \sigma\sqrt{t(1-t)}.$$

The time-dependent velocity field governing sample evolution is then given by:

$$\mathbf{v}_t(\mathbf{x}) = \frac{d}{dt}\alpha_t \mathbf{x}_1 + \frac{d}{dt}\beta_t \mathbf{x}_0 + \frac{d}{dt}\sigma_t \mathbf{z}, \quad \mathbf{z} \sim \mathcal{N}(0, \mathbf{I}).$$

This results in the explicit formulation: $\mathbf{v}_t(\mathbf{x}) = \mathbf{x}_1 - \mathbf{x}_0 + \frac{1-2t}{2\sqrt{t(1-t)}}\mathbf{z}$.

Given the probability path $\rho_t$ and the vector field $\mathbf{v}_t$, the FM training objective is defined as:

$$\mathcal{L} = \mathbb{E}_{t\sim\mathcal{U}(0,1),\mathbf{x}\sim\rho_t(\mathbf{x}_0,\mathbf{x}_1)}||\mathbf{v}_\theta(t, \mathbf{x}) - \mathbf{v}_t(\mathbf{x})||^2,$$

where $\mathbf{v}_\theta$ is the learned velocity field. The optimal solution ensures that $\mathbf{v}_\theta$ generates the probability path flow $\rho_t$ [Tong et al., 2023].

For inference, a sample is drawn from $\rho_0$, and the final conformation $\mathbf{x}_1$ is obtained by numerically integrating the ODE using the Euler method and the learned velocity field $\mathbf{v}_\theta$:

$$\mathbf{x}_{t+\Delta t} = \mathbf{x}_t + \mathbf{v}_\theta(t, \mathbf{x}_t)\Delta t.$$

More details on the training and sampling algorithms are provided in supplementary A.2.

**Harmonic Prior.** Unlike diffusion-based models that typically require a Gaussian prior, the FM framework allows flexibility in choosing the initial distribution $\rho_0$. To better capture the structural properties of molecular graphs, a harmonic prior is used, following the formulations in [Jing et al., 2023] and [Stark et al., 2024]. This prior ensures that atom positions in the initial conformation reflect molecular connectivity by encouraging bonded atoms to remain close in sampled structures. The harmonic prior is defined as:

$$\rho_0(\mathbf{x}_0) \propto \exp\left(-\frac{1}{2}\mathbf{x}_0^T \mathbf{L} \mathbf{x}_0\right).$$

where $\mathbf{L}$ is the laplacian of the molecular graph, and $x_0 \in \mathbb{R}^{1\times N}$ representing one of the three Cartesian coordinates (x, y, or z) of the $N$ atoms in the molecule. This formulation ensures that the sampled initial conformations maintain local structural coherence. The sampling algorithm is detailed in supplementary A.2.

**Molecular Graph Construction.** FMA is a molecular graph neural network that encodes a graph comprising both query and template ligands. It jointly represents their structures to predict the velocity field of the query ligand for conformation denoising. The graph representation is illustrated in Appendix A.1, and is constructed as follows:

- **Atomic-level nodes:** Each heavy atom in both ligands is represented as a node with features encoding atomic properties, including atomic number, total degree, formal charge, chiral tag, total number of hydrogen atoms, and hybridization state.
- **Functional group nodes:** To capture higher-level chemical structures, ligands are decomposed into functional groups using the BRICS fragmentation method [Degen et al., 2008]. Each resulting fragment is treated as a distinct node in the graph, with a scalar feature encoding the fragment size (number of atoms) assigned to each functional group node.
- **Covalent edges:** Covalent bonds within each ligand are represented as graph edges, with associated edge features encoding bond type (e.g. single, double, triple, or aromatic), conjugation status, ring membership, and bond stereochemistry.
- **Functional group connections:** Functional group nodes are connected to their corresponding constituent atoms, enabling feature aggregation from atomic-level representations. The edge features for these connections are scalar values set to zero.
- **Dense edges:** Functional groups within and across the query and template ligands are fully connected. Distances between functional groups are encoded using a radial basis function (RBF), following [Unke and Meuwly, 2019] and [Hassan et al., 2025].

**Model Architecture.** The constructed molecular graph, which includes both the query and template ligands, is first processed by a multilayer perceptron (MLP) to embed node and edge features. Next, a Multi-Head Attention with Edge Bias (MHAwithEdgeBias) encoder [Qiao et al., 2024] is applied to extract topological and contextual information from the molecular graph. The attention mechanism facilitates effective information exchange between the two ligands, ensuring that the query ligand remains informed by the reference ligand. Finally, a time-dependent Vector Field Network (VFN), inspired by [Qiao et al., 2024], predicts the velocity field of the query ligand, based on the molecular graph, the positions of the reference ligand, and positions of the query ligand at time $t$. Further architectural and algorithmic details are provided in Appendix A.3, and training details and hyperparameters in Appendix A.4.

### 3.3 Pose Optimization

The initial pose of the query molecule, predicted by FMA, is refined by directly optimizing its atomic coordinates. This process is formulated as a differentiable optimization problem, where gradient descent minimizes a weighted sum of scoring functions. These functions include ligand-based terms comparing the query ligand $\mathcal{M}_{\text{query}}$ to the reference ligand $\mathcal{M}_{\text{template}}$, and optionally, receptor-based terms involving the template pocket $\mathcal{P}_{\text{template}}$. Unlike traditional methods that optimize rigid-body transformations or torsion angles, PO updates atomic positions directly, enabling flexible and fine-grained conformation refinement.

**Shape-based Tanimoto Score (STS).** Each heavy atom in the molecular structure is modeled as a spherical Gaussian function, following the molecular volume representation introduced by Grant and Pickup [1995], Grant et al. [1996]. This formulation enables the computation of the molecular volume $V_A$, and the overlapping volume $V_{AB}$ of two molecules $A$ and $B$, as detailed in Appendix B.1. Tanimoto (or Jaccard) score based on atoms Gaussian volumes provides a normalized measure of shape similarity:

$$Tanimoto_a(A, B) = \frac{V_{AB}^a}{V_A^a + V_B^a - V_{AB}^a},$$

where $a$ refers to atom-based volumes. The STS between the template $\mathcal{M}_{template}$ and the query $\mathcal{M}_{query}$ is defined as the $Tanimoto_a$ similarity between atoms:

$$\mathcal{S}_{STS}(\mathcal{M}_{query}, \mathcal{M}_{ref}) = Tanimoto_s(\mathcal{M}_{query}, \mathcal{M}_{ref}),$$

and is differentiable with respect to the query ligand coordinates.

**Pharmacophore-based Tanimoto Score (PTS).** Pharmacophoric features are represented using spherical Gaussian functions in a similar fashion. Each pharmacophoric feature type (e.g., hydrogen bond donor, acceptor, aromatic) is treated independently in the similarity computation. We define a pharmacophore-based alignment score that quantifies the degree of overlap between the pharmacophoric volumes of two ligands—higher values indicate that ligands share similar 3D pharmacophoric profile. This approach does not require explicit matching or pairing of features between the template and query molecules. Instead, the overlap between Gaussian spheres with the same pharmacophoric label is directly accumulated into the final pharmacophoric overlapping volume score. The PTS computed on the pharmacophoric volumes between the query $\mathcal{M}_{query}$ and the reference $\mathcal{M}_{ref}$ is then defined as:

$$\mathcal{S}_{PTS}(\mathcal{M}_{query}, \mathcal{M}_{template}) = Tanimoto_p(\mathcal{M}_{query}, \mathcal{M}_{template}),$$

where $p$ refers to pharmacophore-based volumes with details provided in Appendix B.2.

**Protein Pocket Score (Optional).** Receptor-based terms are optionally included in optimization. Both atoms and pharmacophoric features of the template pocket $\mathcal{P}_{\text{template}}$ are modeled using Gaussian functions. Two scores are computed between the query ligand and the template pocket:

- A shape clash penalty based on atomic overlap, using STS to measure overlap between $\mathcal{P}_{template}$ and $\mathcal{M}_{query}$.
- A pharmacophoric complementarity, using PTS to evaluate the alignment between the query ligand and template pocket pharmacophores. For pocket pharmacophore types, an inversion is applied to model complementary interactions. More details are provided in Appendix B.2.

The resulting receptor-based pocket score is defined as:

$$\mathcal{S}_{pocket} = \mathcal{S}_{PTS}(\mathcal{M}_{query}, \mathcal{P}_{template}) - \mathcal{S}_{STS}(\mathcal{M}_{query}, \mathcal{P}_{template}).$$

This formulation encourages pharmacophoric alignment while penalizing steric clashes between the ligand and the template binding pocket.

**Internal Energy.** Optimizing atomic coordinates directly, rather than restricting moves to translations, rotations, and torsions, offers greater flexibility but can easily lead to physically unrealistic molecular conformations. To ensure physically plausible structures, we incorporate the ligand's internal energy into the optimization process, thereby mitigating artifacts such as unrealistic bond lengths and angles that can arise from other scoring terms. The internal energy, $\mathcal{E}_{internal}$, is computed using molecular force fields, specifically General Amber Force Field 2 (GAFF2) [Wang et al., 2004], which defines energy functions based on atomic types and spatial configurations. The detailed formulation is provided in Appendix B.3.

**Optimization Objective.** The optimization objective combines all scores into a weighted loss function defined as follows:

$$\begin{aligned}\mathcal{L}_{\text{optim}} = & -\alpha\, \mathcal{S}_{\text{STS}}(\mathcal{M}_{\text{query}}, \mathcal{M}_{\text{template}}) - \beta\, \mathcal{S}_{\text{PTS}}(\mathcal{M}_{\text{query}}, \mathcal{M}_{\text{template}}) \\ & - \omega\, \mathcal{S}_{\text{pocket}}(\mathcal{M}_{\text{query}}, \mathcal{P}_{\text{template}}) + \gamma\, \mathcal{E}_{\text{internal}}(\mathcal{M}_{\text{query}}).\end{aligned}$$

Further details on the optimization algorithm and hyperparameters are provided in Appendix B.

### 3.4 Benchmark and training dataset construction

**AlignDockBench.** AlignDockBench is a curated benchmark designed to evaluate template-based molecular docking and 3D molecular alignment methods. Unlike common docking benchmarks such as PoseBusters [Buttenschoen et al., 2024], which do not include reference ligands, AlignDockBench provides co-crystallized template ligands for each query. This enables the evaluation of methods that leverage known references to guide 3D conformation prediction. The benchmark includes 369 PL query structures, each associated with a corresponding PL template structure. To construct AlignDockBench, we selected 61 diverse PL templates from the DUD-E [Mysinger et al., 2012] and DEKOIS [Bauer et al., 2013] datasets, spanning major target classes such as kinases, proteases, nuclear receptors, and GPCRs. For each template, we searched the Protein Data Bank (PDB) for query PL complexes whose binding pockets could be structurally aligned to the pocket template.

Binding pockets were defined as all residues within $12$Å of the ligand. Using TM-align [Zhang and Skolnick], we rigidly aligned (via translation and rotation) query pocket to the pocket template and retained only those with a backbone Root Mean Square Deviation (RMSD) below $1.2$Å. To ensure sufficient 3D structural similarity between ligands, we further filtered query-template pairs with an STS score greater than $0.5$. The mapping of queries and associated templates is provided in Appendix C.1, along with PL complex preparation details in Appendix C.2. Protein class diversity is summarized in Figure S6. The benchmark is available at Zenodo [2].

**Training set.** Training FMA requires a dataset of molecular pairs with ground-truth alignments. Following the protocol used in AlignDockBench, we selected ligand pairs from the PDB that bind to the same protein pocket, assuming that co-binding implies structural and functional similarity. We further restricted to pairs in which both ligands have molecular weights above 170 Da. Additionally, the training set was augmented with compounds from the ChEMBL database[3], following a protocol similar to that used for constructing the BindingNet and BindingNet2 datasets [Li et al., 2024, Zhu et al., 2025]. To prevent train–test leakage, we excluded any training molecule with a Morgan fingerprint [Rogers and Hahn, 2010] Tanimoto similarity greater than $0.5$ to any ligand in AlignDockBench (Figure S7). The resulting training set contains $301,348$ complex pairs covering $111,678$ unique molecules.

## 4 Experiments

**Experimental Setup.** We evaluate FMA-PO on AlignDockBench, comparing its performance to both traditional docking tools and state-of-the-art open-access alignment methods. Docking baselines include Vina (v1.2.5) [Trott and Olson, 2010] and rDock (v24.04.204-legacy) [Ruiz-Carmona et al., 2014], while alignment baselines include FitDock (v1.0.9) [Yang et al., 2022], LS-align (Version J201704171741) [Hu et al., 2018], and ROSHAMBO [Atwi et al., 2024]. Detailed configurations and hyperparameters used for each method are provided in Appendix D.1. Alignment-based methods receive the 2D graph of the query molecule along with the 3D structure of the template ligand and, optionally, the protein. In contrast, in our experiments, docking methods do not use the template ligand information. For each method, we generate $10$ candidate poses and evaluate the top-ranked one using the scoring function originally designed for that method. The selected pose is then assessed by computing the RMSD to the experimental ligand structure. Pose selection of our method is guided by a scoring function defined as:

$$\mathcal{S}_{\text{score}} = \mathcal{S}_{STS}(\mathcal{M}_{\text{query}}, \mathcal{M}_{\text{template}}) + \mathcal{S}_{PTS}(\mathcal{M}_{\text{query}}, \mathcal{M}_{\text{template}}) - \mathcal{S}_{\text{Vina}}(\mathcal{M}_{\text{query}}, \mathcal{P}_{template}),$$

where $\mathcal{S}_{\text{Vina}}$ represents the Vina docking score [Trott and Olson, 2010], rescaled to $[0, 1]$ interval using Min-Max normalization across the samples. Two pose selection strategies are explored:

- FMA-PO: The top-ranked pose predicted by FMA, according to $\mathcal{S}_{\text{score}}$, is selected and subsequently refined via PO.
- FMA-PO+: All sampled poses are first refined through PO, and the final pose is selected according to $\mathcal{S}_{\text{score}}$.

**Results.** Table 1 summarizes the performance of each method in terms of mean RMSD, the proportion of molecules with RMSD below $2$Å, and the average runtime per molecule. Both FMA-PO and FMA-PO+ outperform competing methods, achieving the lowest mean RMSD and the highest proportion of alignments with RMSD below $2$Å. Examples of poses generated with each method are provided in Figure S13.

To assess the effect of template similarity on alignment performance, AlignDockBench was divided into two similarity bins ($[0, 0.5[$ and $[0.5, 1.0]$), based on Tanimoto similarity between the query and template compounds. The similarity was computed using two approaches: first, with standard Morgan fingerprints [Rogers and Hahn, 2010] derived from the complete molecular structures (Figure 3a); and second, with Morgan fingerprints computed on the molecules' Generic Murcko scaffolds [Bemis and Murcko, 1996], which capture the core chemical framework by removing side chains and

[2] `https://anonymous.4open.science/r/AlignDockBench-6756/`

[3] `https://www.ebi.ac.uk/chembl/`

Table 1: Performance comparison of 3D molecular alignment and docking methods on AlignDock-Bench in a crossdocking scenario. Methods marked with an (*) use GPU. For methods that did not align all 369 molecules, percentages are reported as X/Y, where the first value is calculated over aligned molecules only, and the second over the total set of 369 molecules.

| **Method** | # of Molecules Aligned | Mean RMSD (Å ↓) | % of Molecules with RMSD $< 2$Å (↑) | Average Runtime (s) |
|---|---|---|---|---|
| FMA* | 369/369 | 1.97 ± 1.36 | 64.77 | 0.83 |
| FMA-PO* | 369/369 | 1.86 ± 1.42 | 69.38 | 3.66 |
| FMA-PO+* | 369/369 | **1.62 ± 1.33** | **77.78** | 27.96 |
| FitDock | 313/369 | 2.93 ± 3.73 | 53.67 / 45.53 | 19.71 |
| LSalign | 368/369 | 2.54 ± 2.00 | 54.35 / 54.2 | 5.67 |
| ROSHAMBO* | 369/369 | 2.87 ± 2.09 | 30.35 | 4.90 |
| rDock | 369/369 | 4.52 ± 3.28 | 34.42 | 20.68 |
| Vina | 368/369 | 3.39 ± 2.81 | 47.28 / 47.15 | 72.98 |

replacing all atoms with carbons (Figure 3b). We further analyzed how molecular complexity affects performance by evaluating results with respect to the number of atoms (Figure 3c) and the number of rotatable bonds (Figure 3d) in the query molecules. For each bin and each method, we plotted the fraction of molecules achieving RMSD below 2Å (Figure 3).

As expected, alignment accuracy generally declines with decreasing molecular similarity to the template or increasing structural complexity. Nonetheless, FMA-PO and FMA-PO+ consistently outperform all baselines across all similarity and complexity bins, demonstrating their robustness in challenging scenarios. Indeed, FMA generates poses without relying on explicit structural similarity, instead leveraging global spatial priors derived from the reference conformation. When the query molecule shares high similarity with the template, 3D LB alignment methods such as LS-align and FitDock tend to outperform Vina and rDock SB approaches, highlighting the value of exploiting reference ligand information to guide pose generation in these scenarios.

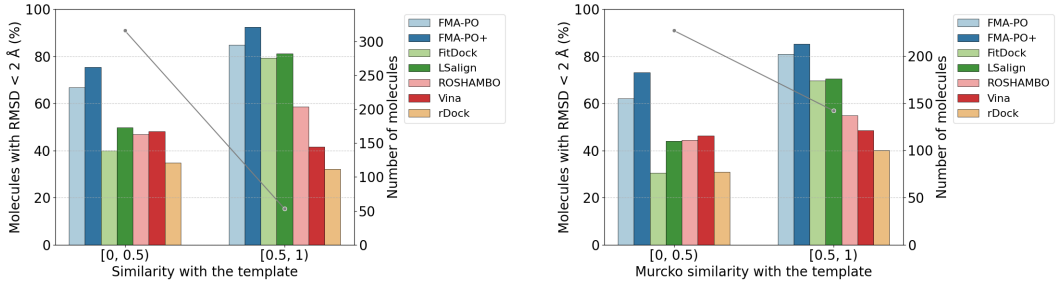
All results were obtained under a cross-docking setup, where the pocket structure of the template ligand is used, simulating the realistic scenario in which the query ligand's pocket structure is unavailable. Additional experiments were conducted under a redocking setup, using the query ligand's own crystal pocket. Results are provided in the Table S4. We further assessed the quality of generated poses using the PoseBusters test suite [Buttenschoen et al., 2024], with detailed results available in Appendix D.3. Ablation studies are presented in Appendix D.4. Comparison to Harmonic Flow Stark et al. [2024] is provided in D.2.3.

**Runtime.** In terms of computational efficiency, FMA-PO achieves a competitive average runtime of 3.66 seconds per molecule. As expected, FMA-PO+ incurs a higher runtime (27.96 s per molecule) due to the additional PO steps involving all samples but resulting in improved alignment accuracy. Both FMA-PO and FMA-PO+ benefit from GPU [4] acceleration during the FMA stage, whereas conventional methods such as rDock and FitDock were executed using a single CPU core, and Vina was run with four CPU cores. More details are provided in D.2.4.

## 5 Discussion

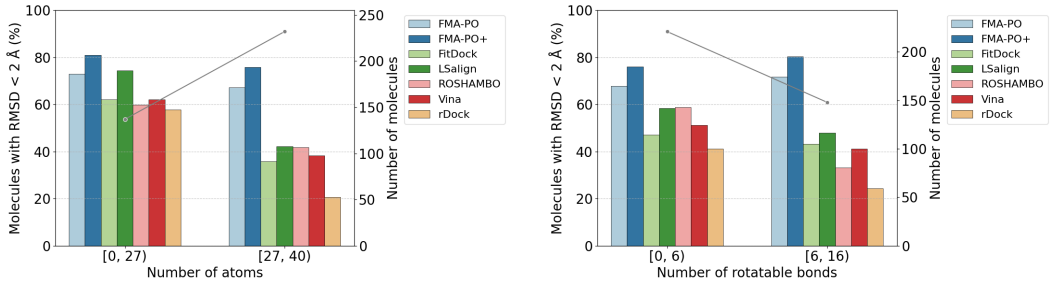
Ligand alignment methods followed by 3D LB similarity calculations are commonly used in virtual screening for hit identification [Jiang et al., 2021]. FMA-PO+, combined with a scoring function that integrates STS, PTS, and optionally $\mathcal{S}_{pocket}$, fits naturally within this framework. Furthermore, the proposed method could be integrated into a *de novo* generative design workflow as a reward signal, encouraging the generation of compounds that exhibit shape and pharmacophore similarity to a reference ligand, as well as complementarity to the protein binding pocket. Indeed, biasing molecular design towards compounds similar to known actives is particularly relevant during the hit discovery and hit-to-lead optimization stages [Bolcato et al., 2022].

[4] Experiments were run on a single NVIDIA T4 Tensor Core GPU.




(a) Stratified by the structure similarity between query and template.

(b) Stratified by the Murcko scaffold similarity between query and template.




(c) Stratified by the number of atoms in the query ligand.

(d) Stratified by the number of rotatable bonds in the query ligand.

Figure 3: Performance of molecular alignment and docking methods on AlignDockBench in a cross-docking scenario. Bar plots (left y-axis) indicate the percentage of molecules with RMSD below 2Å, while the line plot (right y-axis) shows the number of molecules in each bin. Each figure stratifies the results based on a different structural or chemical property of the query ligand.

FMA-PO+ is capable of generating 3D poses of candidate compounds that are directly suitable for downstream activity prediction tasks. Indeed, both deep learning-based scoring functions [Shen et al., 2023, Valsson et al., 2025] and physics-based approaches [Greenidge et al., 2013] rely on a 3D representation of the receptor-ligand complex. In addition, using FMA-PO+ generated alignments to augment training datasets represents a promising strategy to enhance the performance of predictive or generative models. This aligns with recent efforts such as BindingNet V2 [Zhu et al., 2025], which uses alignment-based data augmentation to boost the performance of deep learning models. In particular, FMA itself could benefit from such augmented data to improve its performance in an iterative training process. Future work could explore conditioning FMA on either a single or a set of reference ligands, along with the protein binding pocket, thereby incorporating more structural context into the pose generation process Guan et al. [2025]. Additionally, addressing flexible receptor scenarios, where both the 3D receptor and ligand are predicted, could be interesting. It is worth noting that, compared to non-deep learning methods, FMA-PO remains computationally more intensive, suggesting a potential direction for efficiency improvements.

## 6 Conclusion

This work presents a template-guided framework for 3D molecular pose generation that combines the strengths of FM generative models with differentiable optimization. By conditioning the model on a reference ligand, the proposed FMA-PO approach achieves accurate molecular alignment, outperforming both classical docking tools and open-access alignment methods on the AlignDockBench template-based cross-docking benchmark. It remains robust even in challenging scenarios with low molecular similarity to the template and high ligand flexibility. These results highlight the potential of integrating known ligand geometries into generative frameworks to enhance pose prediction in structure-based drug design. Beyond performance gains, our framework enables sampling for data augmentation and direct integration with downstream predictive models. Future directions include incorporating receptor context into the generative process, extending the approach to flexible receptor scenarios (e.g., induced-fit), and assessing its impact on molecular design applications.

## Acknowledgment

We would like to express our sincere gratitude to the Iktos team for their exceptional contributions to this project. The authors would like to thank IKTOS for supporting this study. We are also grateful to the reviewers for their valuable comments and suggestions, which have improved the manuscript. We further thank Ennys Gheyouche, Maoussi Lhuillier-Akakpo, and Juan Sanz García for their feedback and review of an early draft of this work.

## References

Chayan Acharya, Andrew Coop, James E Polli, and Alexander D MacKerell. Recent advances in ligand-based drug design: relevance and utility of the conformationally sampled pharmacophore approach. *Current computer-aided drug design*, 7(1):10–22, 2011.

Michael S Albergo, Nicholas M Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions. *arXiv preprint arXiv:2303.08797*, 2023.

Amy C Anderson. The process of structure-based drug design. *Chemistry & biology*, 10(9):787–797, 2003.

Rasha Atwi, Ye Wang, Simone Sciabola, and Adam Antoszewski. ROSHAMBO: Open-source molecular alignment and 3d similarity scoring. 64(21):8098–8104, 2024. ISSN 1549-9596. doi: 10.1021/acs.jcim.4c01225. Publisher: American Chemical Society.

Benoit Baillif, Jason Cole, Patrick McCabe, and Andreas Bender. Benchmarking structure-based three-dimensional molecular generative models using genbench3d: ligand conformation quality matters. *arXiv preprint arXiv:2407.04424*, 2024.

Matthias R. Bauer, Tamer M. Ibrahim, Simon M. Vogel, and Frank M. Boeckler. Evaluation and optimization of virtual screening workflows with DEKOIS 2.0 – a public library of challenging docking benchmark sets. 53(6):1447–1462, 2013. ISSN 1549-9596. doi: 10.1021/ci400115b. Publisher: American Chemical Society.

Guy W Bemis and Mark A Murcko. The properties of known drugs. 1. molecular frameworks. *Journal of medicinal chemistry*, 39(15):2887–2893, 1996.

Giovanni Bolcato, Esther Heid, and Jonas Bostrom. On the value of using 3d shape and electrostatic similarities in deep generative methods. *Journal of chemical information and modeling*, 62(6): 1388–1398, 2022.

Martin Buttenschoen, Garrett M Morris, and Charlotte M Deane. Posebusters: Ai-based docking methods fail to generate physically valid poses or generalise to novel sequences. *Chemical Science*, 15(9):3130–3139, 2024.

David A Case, Hasan Metin Aktulga, Kellon Belfon, David S Cerutti, G Andrés Cisneros, Vinícius Wilian D Cruzeiro, Negin Forouzesh, Timothy J Giese, Andreas W Gotz, Holger Gohlke, et al. Ambertools. *Journal of chemical information and modeling*, 63(20):6183–6191, 2023. doi: 10.1021/acs.jcim.3c01153.

Sun-Shin Cha, Dennis Lee, Jerry Adams, Jeffrey T Kurdyla, Christopher S Jones, Lisa A Marshall, Brian Bolognese, Sherin S Abdel-Meguid, and Byung-Ha Oh. High-resolution x-ray crystallography reveals precise binding interactions between human nonpancreatic secreted phospholipase a2 and a highly potent inhibitor (fpl67047xx). *Journal of medicinal chemistry*, 39(20):3878–3881, 1996.

John M Clements, R Paul Beckett, Anthony Brown, Graham Catlin, Mario Lobell, Shilpa Palan, Wayne Thomas, Mark Whittaker, Stephen Wood, Sameeh Salama, et al. Antibiotic activity and characterization of bb-3497, a novel peptide deformylase inhibitor. *Antimicrobial agents and chemotherapy*, 45(2):563–570, 2001.

Miles Congreve, João M Dias, and Fiona H Marshall. Structure-based drug design for g protein-coupled receptors. *Progress in medicinal chemistry*, 53:1–63, 2014. doi: 10.1016/B978-0-444-63380-4.00001-9.

Gabriele Corso, Hannes Stärk, Bowen Jing, Regina Barzilay, and Tommi Jaakkola. Diffdock: Diffusion steps, twists, and turns for molecular docking. *arXiv preprint arXiv:2210.01776*, 2022.

Jorg Degen, Christof Wegscheid-Gerlach, Andrea Zaliani, and Matthias Rarey. On the art of compiling and using'drug-like'chemical fragment spaces. *ChemMedChem*, 3(10):1503, 2008.

Todd J Dolinsky, Paul Czodrowski, Hui Li, Jens E Nielsen, Jan H Jensen, Gerhard Klebe, and Nathan A Baker. Pdb2pqr: expanding and upgrading automated preparation of biomolecular structures for molecular simulations. *Nucleic acids research*, 35(suppl_2):W522–W525, 2007. doi: 10.1093/nar/gkm276.

Peter Eastman, Raimondas Galvelis, Raúl P Peláez, Charlles RA Abreu, Stephen E Farr, Emilio Gallicchio, Anton Gorenko, Michael M Henry, Frank Hu, Jing Huang, et al. Openmm 8: molecular dynamics simulation with machine learning potentials. *The Journal of Physical Chemistry B*, 128 (1):109–116, 2023.

Richard A Friesner, Jay L Banks, Robert B Murphy, Thomas A Halgren, Jasna J Klicic, Daniel T Mainz, Matthew P Repasky, Eric H Knoll, Mee Shelley, Jason K Perry, et al. Glide: a new approach for rapid, accurate docking and scoring. 1. method and assessment of docking accuracy. *Journal of medicinal chemistry*, 47(7):1739–1749, 2004.

Johann Gasteiger and Mario Marsili. Iterative partial equalization of orbital electronegativity—a rapid access to atomic charges. *Tetrahedron*, 36(22):3219–3228, 1980. doi: 10.1016/0040-4020(80) 80168-2.

J Andrew Grant and BT Pickup. A gaussian description of molecular shape. *The Journal of Physical Chemistry*, 99(11):3503–3510, 1995.

J Andrew Grant, Maria A Gallardo, and Barry T Pickup. A fast method of molecular shape comparison: A simple application of a gaussian description of molecular shape. *Journal of computational chemistry*, 17(14):1653–1666, 1996.

Paulette A Greenidge, Christian Kramer, Jean-Christophe Mozziconacci, and Romain M Wolf. Mm/gbsa binding energy prediction on the pdbbind data set: successes, failures, and directions for further improvement. *Journal of chemical information and modeling*, 53(1):201–209, 2013.

Jiaqi Guan, Jiahan Li, Xiangxin Zhou, Xingang Peng, Sheng Wang, Yunan Luo, Jian Peng, and Jianzhu Ma. Group ligands docking to protein pockets. *arXiv preprint arXiv:2501.15055*, 2025.

Jean-Pierre Guilloteau, Magali Mathieu, Carmela Giglione, Véronique Blanc, Alain Dupuy, Miline Chevrier, Patricia Gil, Alain Famechon, Thierry Meinnel, and Vincent Mikol. The crystal structures of four peptide deformylases bound to the antibiotic actinonin reveal two distinct types: a platform for the structure-based design of antibacterial agents. *Journal of molecular biology*, 320(5): 951–962, 2002.

Thomas A Halgren. Merck molecular force field. i. basis, form, scope, parameterization, and performance of mmff94. *Journal of computational chemistry*, 17(5-6):490–519, 1996. doi: 10.1002/(SICI)1096-987X(199604)17:5/6<490::AID-JCC1>3.0.CO.

Karl A Hansford, Robert C Reid, Chris I Clark, Joel DA Tyndall, Michael W Whitehouse, Tom Guthrie, Ross P McGeary, Karl Schafer, Jennifer L Martin, and David P Fairlie. D-tyrosine as a chiral precusor to potent inhibitors of human nonpancreatic secretory phospholipase a2 (iia) with antiinflammatory activity. *Chembiochem*, 4(2-3):181–185, 2003.

Majdi Hassan, Nikhil Shenoy, Jungyoon Lee, Hannes Stärk, Stephan Thaler, and Dominique Beaini. Et-flow: Equivariant flow-matching for molecular conformer generation. *Advances in Neural Information Processing Systems*, 37:128798–128824, 2025.

Paul CD Hawkins, A Geoffrey Skillman, and Anthony Nicholls. Comparison of shape-matching and docking as virtual screening tools. *Journal of medicinal chemistry*, 50(1):74–82, 2007.

Sophia MN Hönig, Christian Lemmen, and Matthias Rarey. Small molecule superposition: A comprehensive overview on pose scoring of the latest methods. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 13(2):e1640, 2023. doi: 10.1002/wcms.1640.

Jun Hu, Zi Liu, Dong-Jun Yu, and Yang Zhang. LS-align: an atom-level, flexible ligand structural alignment algorithm for high-throughput virtual screening. 34(13):2209–2218, 2018. ISSN 1367-4803. doi: 10.1093/bioinformatics/bty081.

Roderick E Hubbard. 3d structure and the drug-discovery process. *Molecular BioSystems*, 1(5-6): 391–406, 2005. doi: 10.1039/b514814f.

Araz Jakalian, David B Jack, and Christopher I Bayly. Fast, efficient generation of high-quality atomic charges. am1-bcc model: Ii. parameterization and validation. *Journal of computational chemistry*, 23(16):1623–1641, 2002. doi: 10.1002/jcc.10128.

Zhenla Jiang, Jianrong Xu, Aixia Yan, and Ling Wang. A comprehensive comparative assessment of 3d molecular similarity tools in ligand-based virtual screening. *Briefings in bioinformatics*, 22(6): bbab231, 2021.

Bowen Jing, Ezra Erives, Peter Pao-Huang, Gabriele Corso, Bonnie Berger, and Tommi Jaakkola. Eigenfold: Generative protein structure prediction with diffusion models. *arXiv preprint arXiv:2304.02198*, 2023.

Bowen Jing, Bonnie Berger, and Tommi Jaakkola. Alphafold meets flow matching for generating protein ensembles. *arXiv preprint arXiv:2402.04845*, 2024.

Bowen Jing, Hannes Stärk, Tommi Jaakkola, and Bonnie Berger. Generative modeling of molecular dynamics trajectories. *Advances in Neural Information Processing Systems*, 37:40534–40564, 2025.

Greg Landrum. Rdkit: Open-source cheminformatics. URL `https://www.rdkit.org/`.

Xuelian Li, Cheng Shen, Hui Zhu, Yujian Yang, Qing Wang, Jincai Yang, and Niu Huang. A high-quality data set of protein–ligand binding interactions via comparative complex structure modeling. *Journal of Chemical Information and Modeling*, 64(7):2454–2466, 2024.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Alex Morehead and Jianlin Cheng. Flowdock: Geometric flow matching for generative protein-ligand docking and affinity prediction. *arXiv preprint arXiv:2412.10966*, 2024.

Michael M. Mysinger, Michael Carchia, John. J. Irwin, and Brian K. Shoichet. Directory of useful decoys, enhanced (DUD-e): Better ligands and decoys for better benchmarking. 55(14):6582–6594, 2012. ISSN 0022-2623. doi: 10.1021/jm300687e. Publisher: American Chemical Society.

Noel M O'Boyle, Michael Banck, Craig A James, Chris Morley, Tim Vandermeersch, and Geoffrey R Hutchison. Open babel: An open chemical toolbox. *Journal of cheminformatics*, 3:1–14, 2011. doi: 10.1186/1758-2946-3-33.

Conor D Parks, Zied Gaieb, Michael Chiu, Huanwang Yang, Chenghua Shao, W Patrick Walters, Johanna M Jansen, Georgia McGaughey, Richard A Lewis, Scott D Bembenek, et al. D3r grand challenge 4: blind prediction of protein–ligand poses, affinity rankings, and relative binding free energies. *Journal of computer-aided molecular design*, 34:99–119, 2020.

Dusan Petrovic, James S Scott, Michael S Bodnarchuk, Olivier Lorthioir, Scott Boyd, George M Hughes, Jordan Lane, Allan Wu, David Hargreaves, James Robinson, et al. Virtual screening in the cloud identifies potent and selective ros1 kinase inhibitors. *Journal of Chemical Information and Modeling*, 62(16):3832–3843, 2022.

Zhuoran Qiao, Weili Nie, Arash Vahdat, Thomas F Miller III, and Animashree Anandkumar. State-specific protein–ligand complex structure prediction with a multiscale deep generative model. *Nature Machine Intelligence*, 6(2):195–208, 2024.

Sereina Riniker and Gregory A Landrum. Better informed distance geometry: using what we know to improve conformation generation. *Journal of chemical information and modeling*, 55(12): 2562–2574, 2015.

David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *Journal of chemical information and modeling*, 50(5):742–754, 2010. doi: 10.1021/ci100050t.

Sergio Ruiz-Carmona, Daniel Alvarez-Garcia, Nicolas Foloppe, A Beatriz Garmendia-Doval, Szilveszter Juhos, Peter Schmidtke, Xavier Barril, Roderick E Hubbard, and S David Morley. rdock: a fast, versatile and open source program for docking ligands to proteins and nucleic acids. *PLoS computational biology*, 10(4):e1003571, 2014.

Chao Shen, Xujun Zhang, Chang-Yu Hsieh, Yafeng Deng, Dong Wang, Lei Xu, Jian Wu, Dan Li, Yu Kang, Tingjun Hou, et al. A generalized protein–ligand scoring framework with balanced scoring, docking, ranking and screening powers. *Chemical Science*, 14(30):8129–8146, 2023.

Hannes Stärk, Bowen Jing, Regina Barzilay, and Tommi Jaakkola. Harmonic self-conditioned flow matching for multi-ligand docking and binding site design. *arXiv preprint arXiv:2310.05764*, 2023.

Hannes Stark, Bowen Jing, Regina Barzilay, and Tommi Jaakkola. Harmonic self-conditioned flow matching for joint multi-ligand docking and binding site design. In *Forty-first International Conference on Machine Learning*, 2024.

Ryoji Suno, Sangbae Lee, Shoji Maeda, Satoshi Yasuda, Keitaro Yamashita, Kunio Hirata, Shoichiro Horita, Maki S Tawaramoto, Hirokazu Tsujimoto, Takeshi Murata, et al. Structural insights into the subtype-selective antagonist binding to the m2 muscarinic receptor. *Nature chemical biology*, 14(12):1150–1158, 2018. doi: 10.1038/s41589-018-0152-y.

Alexander Tong, Kilian Fatras, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *arXiv preprint arXiv:2302.00482*, 2023.

Oleg Trott and Arthur J Olson. Autodock vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *Journal of computational chemistry*, 31(2):455–461, 2010. doi: 10.1002/jcc.21334.

Oliver T Unke and Markus Meuwly. Physnet: A neural network for predicting energies, forces, dipole moments, and partial charges. *Journal of chemical theory and computation*, 15(6):3678–3693, 2019.

Ísak Valsson, Matthew T Warren, Charlotte M Deane, Aniket Magarkar, Garrett M Morris, and Philip C Biggin. Narrowing the gap between machine learning scoring functions and free energy perturbation using augmented data. *Communications Chemistry*, 8(1):41, 2025.

David Van Der Spoel, Erik Lindahl, Berk Hess, Gerrit Groenhof, Alan E Mark, and Herman JC Berendsen. Gromacs: fast, flexible, and free. *Journal of computational chemistry*, 26(16): 1701–1718, 2005. doi: 10.1002/jcc.20291.

Junmei Wang, Romain M Wolf, James W Caldwell, Peter A Kollman, and David A Case. Development and testing of a general amber force field. *Journal of computational chemistry*, 25(9): 1157–1174, 2004. doi: 10.1002/jcc.20035.

Olivier J Wouters, Martin McKee, and Jeroen Luyten. Estimated research and development investment needed to bring a new medicine to market, 2009-2018. *Jama*, 323(9):844–853, 2020.

Xiaocong Yang, Yang Liu, Jianhong Gan, Zhi-Xiong Xiao, and Yang Cao. FitDock: protein–ligand docking by template fitting. 23(3):bbac087, 2022. ISSN 1477-4054. doi: 10.1093/bib/bbac087.

Yan Zhang, Joel Desharnais, Thomas H Marsilje, Chenglong Li, Michael P Hedrick, Lata T Gooljarsingh, Ali Tavassoli, Stephen J Benkovic, Arthur J Olson, Dale L Boger, et al. Rational design, synthesis, evaluation, and crystal structure of a potent inhibitor of human gar tfase: 10-(trifluoroacetyl)-5, 10-dideazaacyclic-5, 6, 7, 8-tetrahydrofolic acid. *Biochemistry*, 42(20): 6043–6056, 2003.

Yang Zhang and Jeffrey Skolnick. TM-align: a protein structure alignment algorithm based on the TM-score. 33(7):2302–2309. ISSN 0305-1048. doi: 10.1093/nar/gki524. URL `https://doi.org/10.1093/nar/gki524`.

Hui Zhu, Xuelian Li, Baoquan Chen, and Niu Huang. Augmented bindingnet dataset for enhanced ligand binding pose predictions using deep learning. *npj Drug Discovery*, 2(1):1, 2025.

## NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract claims a two-stage approach for template-guided ligand pose generation, including flow-matching-based alignment and differentiable pose optimization, which are detailed in the section 3. The performance advantage over standard docking and open-access alignment methods is demonstrated in the section 4.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss limitations and explore future work in section 5 and Apendix D.3.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: This paper relies on established theoretical frameworks, such as flow-matching theory, with all relevant assumptions and proofs cited accordingly. No new theoretical results are introduced.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Yes, the paper provides all the necessary details to reproduce the main experimental results, including the description of the methods, benchmark data, and evaluation protocols, ensuring reproducibility.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: No, we do not provide open access to the code, as it is proprietary and developed within a commercial context. However, we will release the AlignDockBench benchmark to support reproducibility and comparative evaluation.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Datasets are detailled in section 3.4 and models hyperparameters are provided in Appendix A.4.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Our results focus on the novel method introduced (first time applying deep learning methods to perform molecular alignment), and the experiments were conducted on a representative test set of data ($n > 300$). We did not feel necessary to report error bars due to computational resource and time constraints, but we are open to provide them upon request.

   Guidelines:

   - The answer NA means that the paper does not include experiments.

- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Details are provided in Appendix A.4.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Yes, our research aligns with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [No]

    Justification: This paper aims to advance the field of AI for health. While there are potential societal implications, both positive (e.g., accelerating therapeutic development) and negative (e.g., potential misuse for harmful compound design), none that we feel require specific acknowledgment have been identified at this stage.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [No]

    Justification: This paper aims to advance the field of AI for health. While there are potential risks of misuse, none that we feel require specific safeguards have been identified, given the scientific focus of our work.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: : All creators and original owners of work are cited in the references of the paper.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.

- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We will release the AlignDockBench benchmark, the details are provided in section 3.4, along with an access link.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: No crowdsourcing or research with human subjects is done in this work.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: No IRB Approvals or equivalent for research with human subjects are done in this work.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: No, LLMs were not used as an important, original, or non-standard component of the core methods in this research, so no declaration is required.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

# A Flow-Matching Molecular Alignment model details

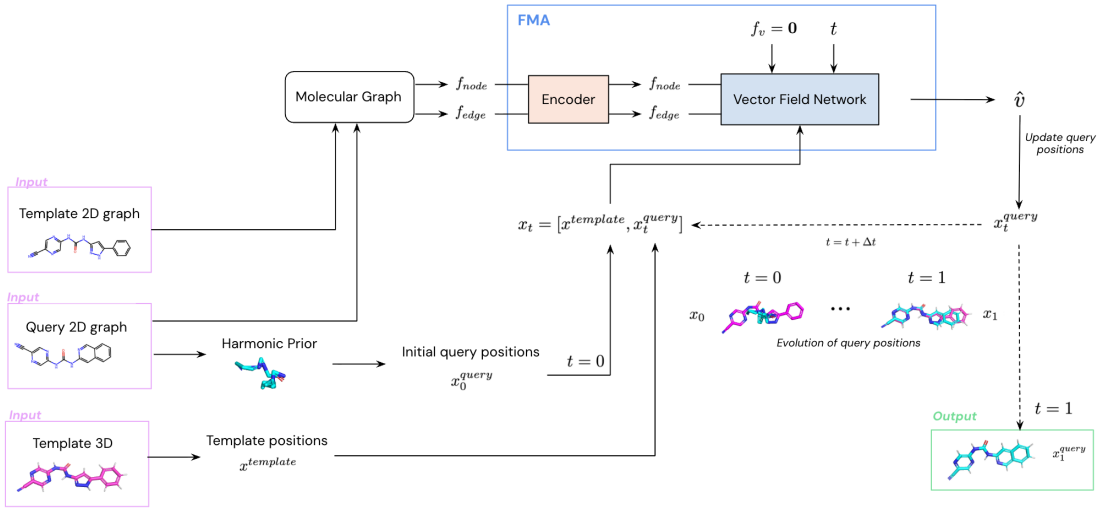



Figure S1: Illustration of FMA model pipeline.

## A.1 Molecular graph construction

The FMA model operates on a hybrid molecular graph $\mathcal{G}$ constructed from both the query and template ligands. The total number of atoms is denoted as $N_{\text{atoms}} = N_{\text{atoms}}^{\text{query}} + N_{\text{atoms}}^{\text{template}}$, where $N_{\text{atoms}}^{\text{query}}$ and $N_{\text{atoms}}^{\text{template}}$ are the numbers of atoms in the query and template molecules, respectively. The graph contains two types of nodes: atomic-level nodes and functional group nodes. Let $N_{\text{nodes}}$ denote the total number of nodes in the graph, including both atoms and functional groups. Edges include covalent bonds, dense inter-ligand functional group connections, and hierarchical links between atoms and their corresponding functional groups. The total number of edges is denoted by $N_{\text{edges}}$.

## A.2 Flow-Matching Training and Inference Algorithms

The inference and training procedures of the FMA model are outlined in Algorithm S1 and Algorithm S2, respectively. In addition, the harmonic sampling process used to generate an initial random conformation from the reference structure is described in Algorithm S3.

**Algorithm S1:** INFERENCE

**Input:** Query 2D graph $G_{query}$, template molecule $\mathcal{M}_{template}$, number of sampling steps $N$, number of samples $K$

$\mathcal{G} \leftarrow$ Construct graph with $G_{query}$ and $\mathcal{M}_{template}$

**for** $i \leftarrow 1$ *to* $K$ **do**

  Sample query positions $C_i \sim \text{HARMONICSAMPLING}(G_{query})$

  Center $C_i$ on $\mathcal{M}_{template}$

  **for** $n \leftarrow 0$ *to* $N-1$ **do**

    $t \leftarrow \frac{n}{N}$

    $\Delta t \leftarrow \frac{1}{N}$

    Predict vector field $\hat{v} \leftarrow \text{FMA}(\mathcal{G}, t, C_i)$

    Update positions $C_i \leftarrow C_i + \hat{v} \times \Delta t$

**return** $C_1, ..., C_K$

## A.3 FMA Molecular Alignment model

FMA (Algorithm S4) takes as input a molecular graph $\mathcal{G}$ representing both query and template ligands, along with their atom coordinates. Positional features $x$ concatenates the ground-truth positions of

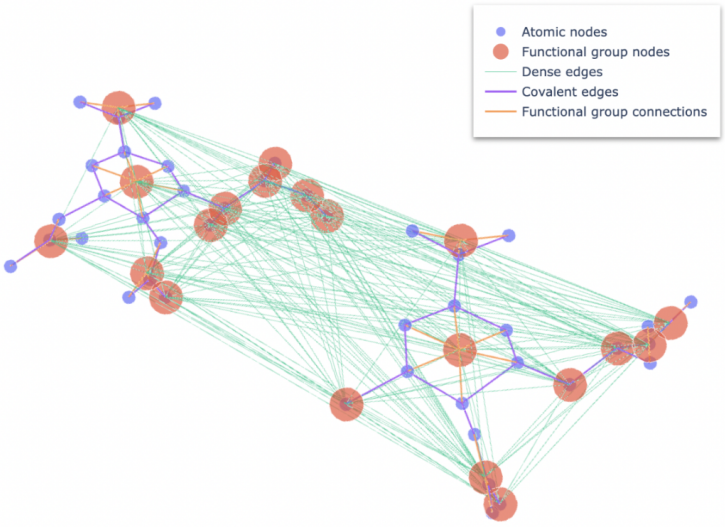



Figure S2: Molecular graph $\mathcal{G}$ with query and template ligand. Blue nodes represent atoms, while orange nodes denote functional groups. The graph incorporates multiple edge types: purple edges for covalent bonds, green edges for dense interconnections, and orange edges linking functional groups to their constituent atoms.

**Algorithm S2:** TRAINING

**Input:** Query molecules with 2D graphs $[G^1_{query}, ..., G^K_{query}]$, true coordinates $[C^1, ..., C^K]$ and associated template molecules $[\mathcal{M}^1_{template}, ..., \mathcal{M}^K_{template}]$, number of epochs $N_{epochs}$, learning rate $\alpha$.

**for** $n \leftarrow 1$ *to* $N_{epochs}$ **do**
  **for** $i \leftarrow 1$ *to* $K$ **do**
    Sample time $t \sim \mathcal{U}([0, 1])$
    $\mathcal{G}^i \leftarrow$ Construct graph with $G^i_{query}$ and $\mathcal{M}^i_{template}$
    Sample query positions $C^i_0 \sim$ HARMONICSAMPLING$(G^i_{query})$
    Center $C_0$ on $\mathcal{M}^i_{template}$
    Sample $C^i_t = t \times C^i + (1 - t) \times C^i_0 + \sigma^2 t(1 - t)\mathbf{z}, \quad z \sim \mathcal{N}(0, \mathbf{I})$
    Compute vector field $u_t \leftarrow C^i - C^i_0 + \frac{1-2t}{2\sqrt{t(1-t)}}\mathbf{z}$
    Predict vector field $\hat{v}_\theta \leftarrow$ FMA$(\mathcal{G}^i, t, C^i_t)$
    Compute loss $\mathcal{L} \leftarrow ||\hat{v}_\theta - u_t||^2$
    Take gradient step $\theta \leftarrow \theta - \alpha \times \Delta_\theta \mathcal{L}$
**return** Trained $v_\theta$

**Algorithm S3:** HARMONICSAMPLING

**Input:** 2D Molecular graph $G$
$\mathbf{L} \leftarrow$ Laplacian of $G$ — $\mathbf{L} \in \mathbb{R}^{N_{atoms} \times N_{atoms}}$
$\mathbf{D}, \mathbf{P} \leftarrow$ Diagonalize Laplacian $\mathbf{L}$ — $\mathbf{D}, \mathbf{P} \in \mathbb{R}^{N_{atoms} \times N_{atoms}}$
Sample $x \sim \mathcal{N}(0, \mathbf{I})$ — $x \in \mathbb{R}^{N_{atoms} \times 3}$
$x \leftarrow \mathbf{P}\mathbf{D}^{-\frac{1}{2}}x$
**return** $x$

the template atoms with the query atom coordinates at diffusion step $t$. The model outputs the vector field $v^{query}$ used for query pose prediction. Initially, node and edge features of $\mathcal{G}$ are independently embedded into vectors of dimensions $c_n$ and $c_e$, respectively, using separate MLPs. All node and

edge embeddings are projected into a common feature space with identical dimensionality, thereby transforming the heterogeneous graph into a homogeneous representation. To preserve type-specific information, each node and edge is annotated with its corresponding type attribute. The resulting representation is processed by a stack of $N_{\text{blocks}}^{MHA}$ MHAWITHEDGEBIAS layers (Algorithm S3 in [Qiao et al., 2024]), jointly updating node and edge embeddings. The processed node embeddings are then passed to the VECTORFIELDNETWORK model (Algorithm S5), which applies a stack of $N_{\text{blocks}}^{VFN}$ layers of POINTSETATTENTIONWITHEDGEBIAS (Algorithm S8 in [Qiao et al., 2024]), followed by a GATEDUPDATE layer (Algorithm S6), adapted from [Qiao et al., 2024]. These blocks operate directly on scalar and vector node representations, denoted $f_s$ and $f_v$ respectively, incorporating positional information $x$ and diffusion time $t$ to produce the final learned vector field $v^{query}$.

**Algorithm S4:** FMA

**Input:** Molecular graph $\mathcal{G}$ with adajency matrix $\mathbf{X}$, nodes features $f_{node}$, edges features $f_{edge}$, time $t$, template-query positions $x \in \mathbb{R}^{N_{atoms} \times 3}$ at time $t$, $N_{\text{blocks}}^{MHA}, N_{\text{heads}}^{MHA}, N_{\text{blocks}}^{VFN}, N_{\text{heads}}^{VFN}$.

**for** *j in nodes types* **do**
  $f_{node}^j \leftarrow \text{MLP}(f_{node}^j)$ $\quad f_{node}^j \in \mathbb{R}^{N_{nodes}^j \times c_n}$
**for** *j in edge types* **do**
  $f_{edge}^j \leftarrow \text{MLP}(f_{edge}^j)$ $\quad f_{edge}^j \in \mathbb{R}^{N_{edges}^j \times c_e}$
$\mathcal{G} \leftarrow$ TOHOMOGENEOUS$(\mathcal{G})$ ; `// Convert to Homogeneous Graph`
**for** $k = 1$ *to* $N_{blocks}^{MHA}$ **do**
  $f'_{node}, \_, z \leftarrow \text{MHAWITHEDGEBIAS}_k(f_{node}, f_{node}, f_{edge}, \mathbf{X}, N_{\text{heads}}^{MHA})$
  $f'_{node} \in \mathbb{R}^{N_{nodes} \times c_n}$
  $f_{node} \leftarrow \text{MLP}_k(f_{node} + f'_{node}) + f_{node}$ $\quad f_{node} \in \mathbb{R}^{N_{nodes} \times c_n}$
  $f_{edge} \leftarrow \text{MLP}_k(f_{edge} + \text{LINEARNOBIAS}_k(z)) + f_{edge}$ $\quad f_{edge} \in \mathbb{R}^{N_{edges} \times c_n}$
$f_v \leftarrow \mathbf{0} \in \mathbb{R}^{N_{nodes} \times 3 \times c_n}$ ; `// Initialize vector features as zero tensors`
$x \leftarrow x - \text{mean}(x)$ ; `// Center positions`
$v \leftarrow$ VECTORFIELDNETWORK$(f_{node}, f_v, f_{edge}, \text{x}, t, N_{\text{blocks}}^{VFN}, N_{\text{heads}}^{VFN})$ $\quad v \in \mathbb{R}^{N_{nodes} \times 3}$
$v \leftarrow$ Extract atomic nodes from $v$ $\quad v \in \mathbb{R}^{N_{atoms} \times 3}$
$v^{query} \leftarrow$ Extract Query vector from $v$ $\quad v^{query} \in \mathbb{R}^{N_{atoms}^{query} \times 3}$
**return** $v^{query}$

**Algorithm S5:** VECTORFIELDNETWORK

**Input:** Scalar features $fs \in \mathbb{R}^{N_{nodes} \times c_n}$, vector features $fv \in \mathbb{R}^{N_{nodes} \times 3 \times c_n}$, edges features $fe \in \mathbb{R}^{N_{edges} \times c_e}$, query-template positions x, time t, $N_{\text{blocks}}$, $N_{\text{heads}}$

$fs \leftarrow \text{CONCAT}(fs, t)$ $\quad fs \in \mathbb{R}^{N_{nodes} \times (c_n+1)}$
**for** $k = 1$ *to* $N_{blocks}$ **do**
  $fs, fv \leftarrow \text{POINTSETATTENTIONWITHEDGEBIAS}_k(fs, fv, fe, x, N_{\text{heads}})$
  $fs \in \mathbb{R}^{N_{nodes} \times (c_n+1)}, fv \in \mathbb{R}^{N_{nodes} \times 3 \times (c_n+1)}$
  $fs, fv \leftarrow \text{GATEDUPDATE}_k(fs, fv, (c_n + 1))$
  $fs \in \mathbb{R}^{N_{nodes} \times (c_n+1)}, fv \in \mathbb{R}^{N_{nodes} \times 3 \times (c_n+1)}$
$\_, v \leftarrow \text{GATEDUPDATE}(fs, fv, 1)$ $\quad v \in \mathbb{R}^{N_{nodes} \times 3}$
**return** $v$

### A.4 Training Details and Hyperparameters

We trained FMA for 100 epochs with a batch size of 128 on a single NVIDIA GeForce RTX 2080 Ti GPU. We used the AdamW optimizer with a learning rate of $3 \times 10^{-4}$ and a weight decay of $10^{-5}$. Hyperparameters are detailed in Table S1.

**Algorithm S6:** GATEDUPDATE

**Input:** Scalar features $fs \in \mathbb{R}^{N_{nodes} \times c_n}$, vector features $fv \in \mathbb{R}^{N_{nodes} \times 3 \times c_n}$, output dimension $d$

$f_{loc} \leftarrow \text{CONCAT}(fs, \|fv\|_2)$ $\quad f_{loc} \in \mathbb{R}^{N_{nodes} \times (c_n + c_n)}$

$f_{loc} \leftarrow \text{LAYERNORMMLP}(f_{loc})$ $\quad f_{loc} \in \mathbb{R}^{N_{nodes} \times (c_n + d)}$

$fs, f_{gate} \leftarrow \text{SPLIT}(f_{loc})$ $\quad fs \in \mathbb{R}^{N_{nodes} \times c}, f_{gate} \in \mathbb{R}^{N_{nodes} \times 1 \times d}$

$f_{gate} \leftarrow \text{SIGMOID}(f_{gate})$

$fv \leftarrow \text{LINEARNOBIAS}(fv)$ $\quad fv \in \mathbb{R}^{N_{nodes} \times 3 \times d}$

$fv \leftarrow fv \odot f_{gate}$ ; `// Element wise multiplication`

**return** $fv$

Table S1: Hyperparameters used for the training of FMA.

| **Hyperparameter** | |
|---|---|
| Nodes embedding dimension ($c_n$) | 128 |
| Edges embedding dimension ($c_e$) | 32 |
| MHAWITHEDGEBIAS heads ($N_{\text{heads}}^{\text{MHA}}$) | 6 |
| MHAWITHEDGEBIAS blocks ($N_{\text{blocks}}^{\text{MHA}}$) | 6 |
| VECTORFIELDNETWORK heads ($N_{\text{heads}}^{\text{VFN}}$) | 6 |
| VECTORFIELDNETWORK blocks ($N_{\text{blocks}}^{\text{VFN}}$) | 6 |
| Noise scale ($\sigma$) | 0.1 |
| Total parameters | 3.5M |

# B Pose Optimization details

## B.1 Shape-based Tanimoto Score

Following the molecular volume representation proposed in [Grant and Pickup, 1995, Grant et al., 1996], each atom in the molecular structure is modeled as a spherical Gaussian function. An atom located at coordinates $R_i = (x_i, y_i, z_i)$ with a radius $\sigma_i$ is represented by a Gaussian density function:

$$\rho_i(r_i) = p_i \exp(-\alpha_i r_i^2) \quad with \quad \alpha_i = \frac{k_i}{\sigma_i^2},$$

where $r_i = |r - R_i|$ is defined as a distance vector from the atomic center $R_i$, and parameters $p_i$, $k_i$ are atomic-type-specific chosen to ensure that the Gaussian atomic volume matches the van der Waals hard-sphere volume: $V_i^a = \frac{4\pi}{3}\sigma_i^3$, where the exponent $a$ refers to atom-based gaussian volumes.

According to the Gaussian product theorem, the product of two Gaussian spheres results in another Gaussian function. Utilizing this property, explicit formulas can be derived to compute the intersection volume between two atomic Gaussians, enabling the calculation of both molecular volumes and overlapping volumes between molecules [Grant and Pickup, 1995, Grant et al., 1996]. The intersection volume between atoms $i$ and $j$ is expressed as:

$$V_{ij}^a = p_i p_j K_{ij} (\frac{\pi}{\alpha_i + \alpha_j})^{3/2} \quad with \quad K_{ij} = \exp(-\frac{\alpha_i \alpha_j R_{ij}^2}{\alpha_i + \alpha_j}),$$

where $R_{ij}$ is the distance between atoms $i$ and $j$. The total volume of a molecule is expressed as:

$$V^g = \sum_i V_i^a - \sum_{i<j} V_{ij}^a + \sum_{i<j<k} V_{ijk}^a - \ ...$$

Similarly, the overlapping volume between molecules $A$ and $B$ is given by:

$$V_{AB}^g = \sum_{i \in A, j \in B} V_{ij}^a - \sum_{i,j \in A, k \in B} V_{ijk}^a - \sum_{i \in A, j,k \in B} V_{ijk}^a + \ ...$$

For computational tractability, we approximate the overlapping volume by truncating this series at second order, thus including only pairwise and triplet intersections.

### B.2 Pharmacophore scoring

**Pharmacophore-based Tanimoto Score**. Pharmacophores of a molecule are 3D arrangements of specific chemical features, such as hydrogen bond donors, hydrogen bond acceptors, hydrophobic groups, and aromatic rings. These features are identified using RDKit. Each pharmacophore feature is positioned at the barycenter of the corresponding atomic group and assigned a specific type (e.g., hydrogen donor, hydrogen acceptor, aromatic, etc.). Pharmacophores are modeled as spherical Gaussian functions with a fixed radius of 1.7Å, comparable to the van der Waals radius of a carbon atom. The overlapping pharmacophore volume between molecules $A$ and $B$ is given by:

$$
\begin{aligned}
V_{AB}^{p} = & \sum_{i \in A,\, j \in B} V_{ij}^{p}\, \mathbb{1}_{\{T(i)=T(j)\}} \\
& - \sum_{i,j \in A,\, k \in B} V_{ijk}^{p}\, \mathbb{1}_{\{T(i)=T(j)=T(k)\}} \\
& - \sum_{i \in A,\, j,k \in B} V_{ijk}^{p}\, \mathbb{1}_{\{T(i)=T(j)=T(k)\}} \\
& + \ldots
\end{aligned}
$$

where $V_{ij}^{p}$ represents the intersection volume between two Gaussians pharmacophores of types $T(i)$ and $T(j)$ from molecules $A$ and $B$. The indicator function $\mathbb{1}_{\{T(i)=T(j)\}}$ ensures that only pharmacophores of the same type are considered for overlapping volume calculations. In practice, we approximated the overlapping pharmacophore volume by truncating this series at the first order, including only pairwise interactions.

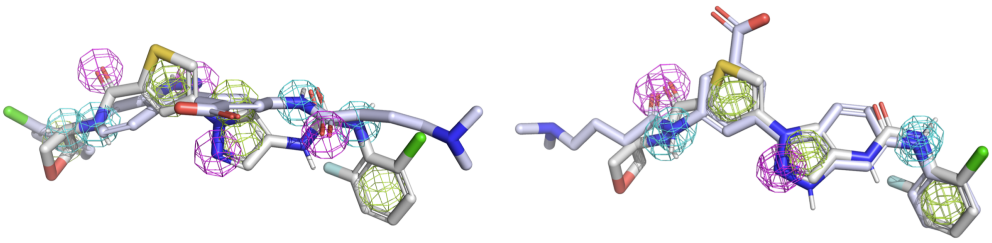


Figure S3: **Aligned ligands with pharmacophores overlapping**. Ligands carbons are shown in grey. Gaussian pharmacophore spheres are depicted as sphere mesh. The color of each sphere encodes a specific pharmacophoric feature: cyan for hydrogen bond donors, magenta for hydrogen bond acceptors, red for negatively charged groups, blue for positively charged groups, and limon for aromatic groups.

**Pharmacophoric complementarity**. The concept of pharmacophoric complementarity in our method stems from Gaussian-based shape similarity modeling. Specifically, once a Gaussian sphere is labeled with a pharmacophoric feature (e.g., hydrogen bond donor (HBD), hydrophobic, etc.), it becomes possible to compute a pharmacophoric similarity between two ligands based on the overlap of these labeled Gaussian volumes.

When introducing the protein pocket into the PO process, we extended this idea to model complementarity between ligand and pocket by maximizing the overlap between complementary pharmacophoric Gaussians. For instance, Hydrogen Bond Donor (HBD) and Hydrogen Bond Acceptor (HBA) were defined as complementary, as well as cationic and anionic features (Figure S4). During the PO, the $\mathcal{S}_{\mathrm{PTS}}$ term of the objective function increases the overlap between such complementary features from the ligand and the pocket, whereas the shape $-\mathcal{S}_{\mathrm{STS}}$ term prevents steric clashes by penalizing excessive overlap between atoms.

Regarding hydrogen bonding in particular, we model the HBD pharmacophoric center as a Gaussian sphere located at the position of the donor hydrogen atom. If this hydrogen atom (and thus its associated Gaussian sphere) is positioned near an HBA-labeled Gaussian from the pocket, the overlap between these spheres is maximized, effectively favoring geometries consistent with hydrogen bond formation. The geometry naturally tends toward a favorable $\mathrm{X}-\mathrm{H}\cdots\mathrm{X}$ angle close to $180°$, since this configuration maximizes Gaussian overlap. Simultaneously, the internal energy term in the PO loss encourages the donor group (e.g., $\mathrm{C}-\mathrm{X}-\mathrm{H}$) to remain in energetically favorable conformations.

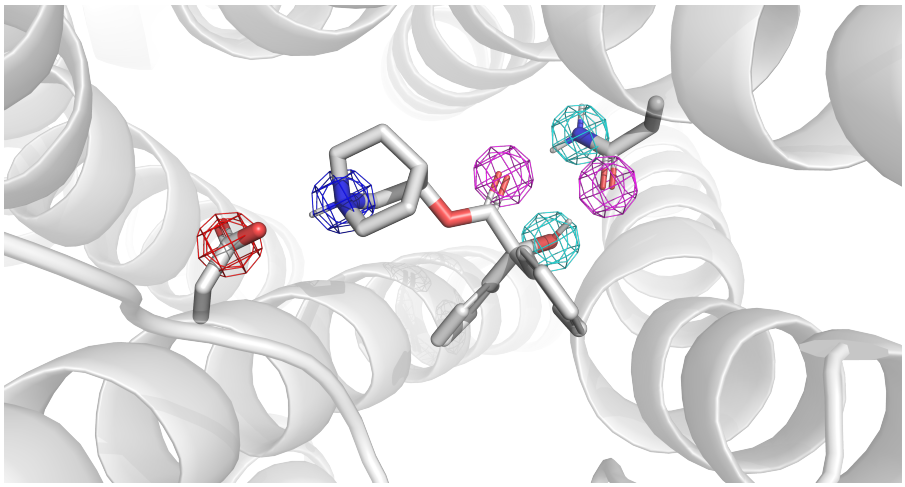


Figure S4: **Illustration of pharmacophoric complementarity within a PL binding site**. 3D structure of the human M2 muscarinic receptor in complex with the ligand QNB (PDB ID: 5ZK3) [Suno et al., 2018], shown in grey. Gaussian pharmacophore spheres are depicted as sphere mesh to illustrate key interaction features within the binding site. The color of each sphere encodes a specific pharmacophoric feature: cyan for hydrogen bond donors, magenta for hydrogen bond acceptors, red for negatively charged groups, and blue for positively charged groups.

As an illustrative example, consider the protein–ligand complex (PDB: `1NJS`) shown in Fig. S5. The ligand KEU (sequence number 510, chain A) contains two hydroxyl groups, `C5-OA1` and `C5-OA2`, both positioned near the carboxylate group of `ASP 144` (chain A):

- The hydroxyl oxygen `OA1` belongs to a donor group (`C5-OA1-H1`) and lies near the acceptor `OD1` of `ASP 144`.
- The second hydroxyl oxygen `OA2` (in `C5-OA2-H2`) is close to the second acceptor `OD2`.

At the beginning of PO, hydrogen atoms `H1` and `H2` are added by RDKit without considering the protein pocket. However, once optimization begins, complementary HBD Gaussians are assigned to these hydrogens, and their overlap with HBA Gaussians on `OD1` and `OD2` is maximized.

### B.3 Internal Energy components

Internal energy components are computed using GAFF2 [Wang et al., 2004] force field parameters that depend on atomic and chemical bond properties. These parameters are parsed using the OpenMM[Eastman et al., 2023] package and energy equations are implemented in PyTorch, ensuring differentiability with respect to ligand coordinates. Following OpenMM formulation, the internal energy is decomposed into two main components:

- **Bonded forces:** These include harmonic bond forces, harmonic angle forces, and periodic torsion forces, which account for bond stretching, angle bending, and torsional rotations.
- **Non-bonded interactions:** These include Lennard-Jones interactions, modeling van der Waals forces, and Coulomb interactions, which account for electrostatic forces between partial atomic charges.

Hence, internal energy is computed as follows:

$$\mathcal{E}_{internal} = \mathcal{E}_{bond} + \mathcal{E}_{angle} + \mathcal{E}_{torsion} + \mathcal{E}_{lj} + \mathcal{E}_{coulomb},$$

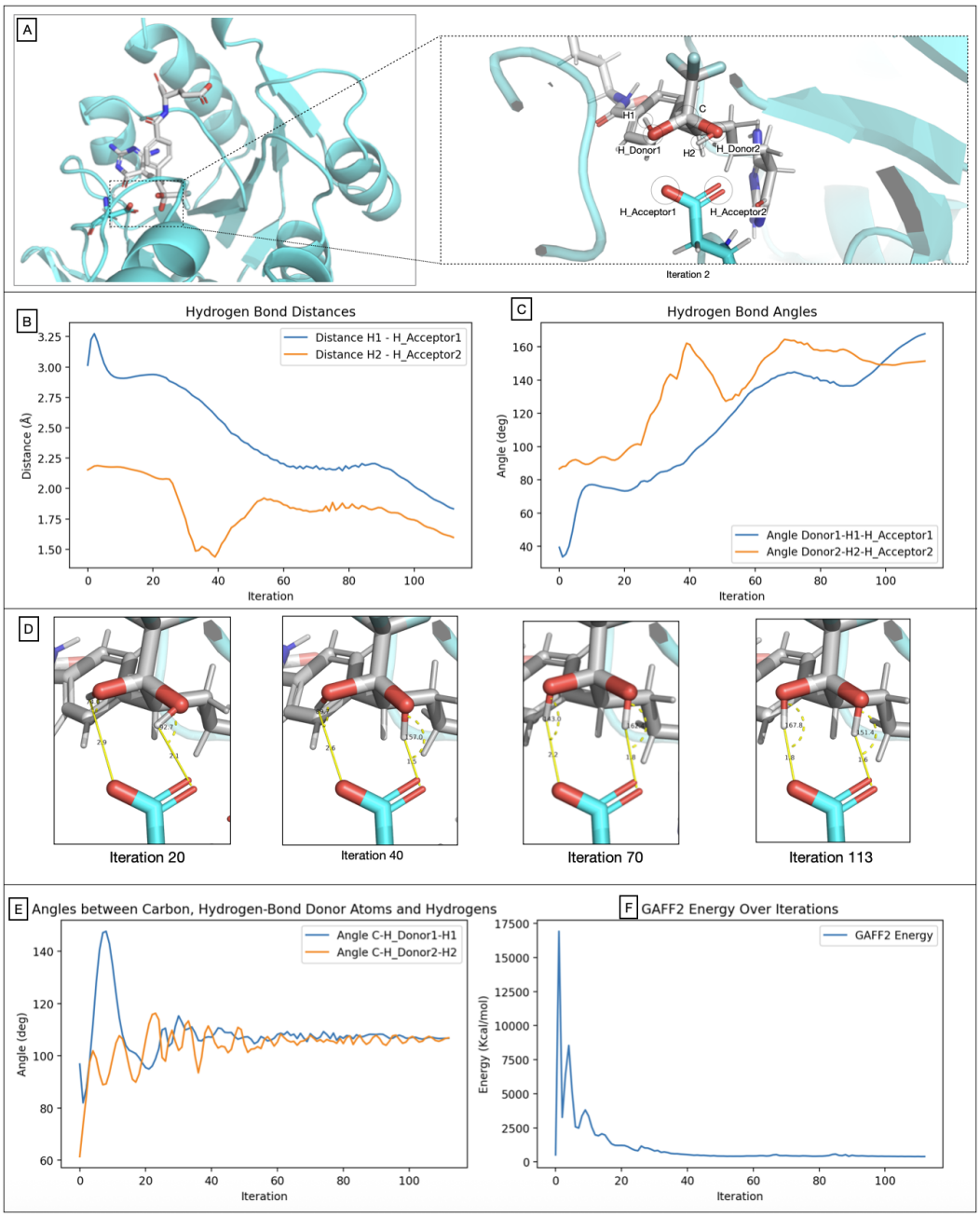



Figure S5: **Illustration of iterative optimization of hydrogen bonding interactions in a protein–ligand complex**. 3D structure of the human GAR Tfase (carbons in cyan) in complex with the hydrolyzed form of 10-trifluoroacetyl-5,10-dideaza-acyclic-5,6,7,8-tetrahydrofolic acid Zhang et al. [2003] (carbons in grey). (A) The protein–ligand binding site shows two HBD (Donor1, Donor2) and two HBA (Acceptor1, Acceptor2). (B) Time evolution of donor–acceptor distances (H1–Acceptor1, H2–Acceptor2) over the optimization iterations. (C) Time evolution of hydrogen-bond angles (Donor1–H1–Acceptor1, Donor2–H2–Acceptor2). (D) Representative ligand conformations at selected iterations (20, 40, 70, 113), illustrating the progressive stabilization of hydrogen bonds in the binding pocket. (E) Evolution of the angles between the carbon atoms bound to hydrogen-bond donors and their respective hydrogens (C–$H_{Donor1}$–H1, C–$H_{Donor2}$–H2). (F) GAFF2 energy profile across iterations, showing convergence from initially high-energy conformations toward a stabilized minimum.

where each term is described below.

**Harmonic Bond Energy** models the interaction between two bonded atoms $i, j$ using an harmonic potential:

$$\mathcal{E}_{\text{bond}}^{ij} = \frac{1}{2}k(d_{ij} - d_0)^2,$$

where $d_{ij}$ is the bond length between atoms $i, j$, $d_0$ is the equilibrium bond length, and $k$ is the bond force constant.

**Harmonic Angle Energy** describes the energy cost of deviating from the equilibrium bond angle between 3 atoms $i, j, k$:

$$\mathcal{E}_{\text{angle}}^{ijk} = \frac{1}{2}k(\theta_{ijk} - \theta_0)^2,$$

where $\theta_{ijk}$ is the bond angle between $i, j, k$, $\theta_0$ is the equilibrium bond angle, and $k$ is the angle force constant.

**Periodic Torsion Energy** accounts for dihedral interactions between four atoms $i, j, k, l$ using a periodic function:

$$\mathcal{E}_{\text{torsion}}^{ijkl} = k(1 + \cos(n\phi_{ijkl} - \gamma)),$$

where $\phi_{ijkl}$ is the dihedral angle between $i, j, k, l$, $n$ is the periodicity, $\gamma$ is the phase offset, and $k$ is the torsional force constant.

**Lennard-Jones Energy** models van der Waals interactions between atoms $i, j$ using the Lennard-Jones potential:

$$\mathcal{E}_{\text{lj}}^{ij} = 4\epsilon_{ij}\left[\left(\frac{\sigma_{ij}}{d_{ij}}\right)^{12} - \left(\frac{\sigma_{ij}}{d_{ij}}\right)^{6}\right],$$

where $\epsilon_{ij}$ defines the interaction strength, $\sigma_{ij}$ is the distance at which the potential is zero, and $d_{ij}$ is the distance between atoms $i$ and $j$.

**Coulomb Energy** describes electrostatic interactions between atoms $i, j$ using Coulomb's law:

$$\mathcal{E}_{\text{coulomb}}^{ij} = \frac{q_i q_j}{4\pi\varepsilon_0 d_{ij}},$$

where $q_i, q_j$ are the atomic charges, $d_{ij}$ is the distance between atoms $i$ and $j$, and $\varepsilon_0$ is the permittivity of vacuum ($\approx 8.85 \times 10^{-12} F/m$).

### B.4 Pose Optimization algorithm

The PO algorithm is described in Algorithm S7. We used the Adam optimizer with a learning rate of 0.2. The refinement protocol is divided into two main phases, with weights that can vary across iterations to progressively adjust the optimization focus. The two-phase protocol is structured as follows:

- **Phase 1: Hydrogen-free refinement**
  - *Step 1*: Focus on minimizing internal energy.
  - *Step 2*: Reinforce ligand alignment with a focus on shape and pharmacophore overlap.
  - *Step 3*: Add steric clashes and internal energy penalties while maintaining overlap improvements.
  - *Step 4*: Fine-tune steric clashes and internal energy, with reduced shape weighting.
- **Phase 2: Hydrogen-reintroduced refinement**
  - *Step 1*: Focus on internal energy minimization after hydrogen addition.
  - *Step 2*: Add shape and pharmacophore constraints for improved fit.
  - *Step 3*: Emphasize pocket volume interactions alongside energy adjustments to refine binding.
  - *Step 4*: Perform final fine-tuning of both overlap interactions and energetic stability.

The weights of the loss terms (Table S2) were empirically tuned using a validation set of ten diverse protein–ligand complexes from the training data. For each term—shape similarity, pharmacophoric complementarity, steric clash penalty, and internal energy regularization—we explored multiple weightings. The final values were chosen based on convergence behavior, pose quality, RMSD and alignment performance across the validation set.

**Algorithm S7:** POSEOPTIMIZATION

**Input:** Ligand conformation with coordinates $x$, number of optimization steps $n_{iterations}$, learning rate $lr$

**for** $n_{iterations}$ **do**

  Compute loss: $\mathcal{L}_{\text{optim}} = -\alpha\,\mathcal{S}_{\text{STS}} - \beta\,\mathcal{S}_{\text{PTS}} - \omega\,\mathcal{S}_{\text{pocket}} + \gamma\,\mathcal{E}_{\text{internal}}$

  Update atoms positions $x = x - lr \times \nabla_x \mathcal{L}_{optim}$

Table S2: Hyperparameters for two-phase pose optimization protocol. All weights were tuned empirically. $(x \rightarrow y)$ means that the weight increase linearly over iterations from $x$ to $y$.

| **Phase** | **Step** | max_iterations | $\alpha$ | $\beta$ | $\omega$ | $\gamma$ |
|---|---|---|---|---|---|---|
| Without H | 1 | 50 | 1.0 | 0.0 | 0.0 | 1.0 |
| | 2 | 200 | (1.0→20.0) | (1.0→50.0) | 10.0 | 10.0 |
| | 3 | 50 | 30.0 | 10.0 | 200.0 | (1.0→50.0) |
| | 4 | 200 | 1.0 | 50.0 | 200.0 | 50.0 |
| With H | 1 | 25 | 0.0 | 0.0 | 0.0 | 1.0 |
| | 2 | 200 | 10.0 | 10.0 | 25.0 | 10.0 |
| | 3 | 50 | 10.0 | 10.0 | 150.0 | (10.0→25.0) |
| | 4 | 200 | 10.0 | 10.0 | 25.0 | 10.0 |

# C AlignDockBench and training dataset

## C.1 AlignDockBench

Table S3: PDB IDs of template and query structures used in AlignDockBench.

| **Template PDB ID** | **Query PDB IDs** |
|---|---|
| 1A4G | 2QWJ, 1INF, 1VCJ, 4HZW, 1INV, 2QWI, 1IVB, 1B9S, 1XOG, 3K37, 1F8E, 1F8C, 4MJV, 1INW, 5JYY |
| 2B1P | 7KSJ, 7KSI, 4W4V, 4WHZ |
| 1UY6 | 1UYD, 3HZ5, 1UYH, 5LR1, 1UY9, 7D24, 3FT8, 2H55, 7D22, 1UY8, 2FWZ, 6OLX, 1UYI, 4XIR, 1UYF, 4U93, 4XIQ, 6LR9, 1UY7, 1UYC, 7D26, 2FWY |
| 3BIZ | 3CR0, 2Z2W, 3BI6, 3CQE, 1X8B, 2IO6, 2IN6 |
| 2AA2 | 6W9M, 1ZUC, 6W9K, 5MWY, 4QL8, 5HCV, 1XQ3, 6W9L, 2A3I, 2OAX, 2AMB, 2Q1V, 1GS4 |
| 3QKL | 3QKM, 3QKK |
| 3EML | 5IUB, 5IUA, 5IU7, 5IU8 |
| 3SFF | 7ZZW, 3SFH, 7ZZU |
| 3D4Q | 3PPK, 3PSB, 3PPJ, 7SHV |
| 2ICA | 3M6F |
| 3LBK | 4OQ3, 5LAV, 1T4E |

| | |
|---|---|
| 3BQD | 7PRV, 6W9M, 6W9L, 5NFP |
| 3KBA | 3HQ5, 3G8O, 1ZUC |
| 2I0E | 8U37, 8UAK, 2JED |
| 1XOI | 2IEG, 2ZB2 |
| 3SKC | 4XV1, 4EHG, 5ITA, 4E4X, 5FD2, 3TV4, 4PP7 |
| 3KL6 | 2J34, 2VVU, 5VOF, 2P93, 2Y7Z, 2Y80, 2VVC, 2VWO, 4Y71, 1WU1, 2UWP, 2J38, 2UWO, 2J4I, 2P95, 1IOE, 1NFW, 1IQI, 2UWL, 2J2U, 1IQ,L 2VWL, 2VVV, 2J94, 1IQK, 2VWN, 1IQN, 2PR3, 4Y79, 1IQG, 2CJI, 4Y7A, 2J95, 4Y7B, 2D1J, 1J17, 4ZH8, 1IQH, 4Y76, 3SW2, 2P94 |
| 2XCH | 3OT8, 2R7B, 2IN6, 2PE1, 2IO6 |
| 1E3G | 3G0W, 1ZUC, 2NW4, 2AAX, 5T8J, 2HVC, 4QL8, 4ZN7, 8E1A 5V8Q 5T8E, 2IHQ, 2A3I, 2AMB, 5VO4, 5CJ6, 3D90, 1XNN, 1GS4 |
| 2E1W | 1WXZ, 1V7A, 1NDY |
| 3LAN | 3LAL, 1C1C, 2JLE, 1TL3, 3DRP, 3TAM, 3LAK, 4NCG, 2YNG, 8U6E, 1C1B, 2WON, 1RT1, 8U6P, 7SLR, 8U6C, 4WE1, 2B5J, 1RT2, 8U6B, 3DYA, 8U6O, 7SLS, 2BE2, 3LAM, 2YNI, 5TUQ, 8U6Q, 3T19 |
| 3LPB | 6VNI |
| 1UYG | 2QG2, 2QF6, 1UY9, 4CWR, 3HZ5, 8SSV, 4U93, 4R3M, 1UYI, 7D24 6N8Y 3B26 7D26 5LR1 7D22 1UYC 2H55 1UYK 3O0I, 1UY7, 1UYD, 3B25, 1UY8, 6LR9, 3FT8, 1UYM, 1UYH, 4L91, 6EL5, 2FWY |
| 3BKL | 3BKK, 6TT1 |
| 1W7X | 4JYU, 5PAM, 4X8V, 4ZXY, 1YGC, 1W0Y, 4YT7, 4JYV, 4YT6, 4NGA, 5PAQ, 4NG9, 1W2K |
| 1S3B | 2VZ2, 1GOS, 2C66, 2C64, 2C65, 1S3E, 5MRL |
| 3F9M | 4MLH, 3FR0, 6E0E, 3GOI, 3A0I, 4MLE |
| 2ZEC | 3V7T, 2ZA5, 2BM2, 2ZEB |
| 2GTK | 3FEJ |
| 3LN1 | 3N8X, 6COX, 5KIR, 3QMO |
| 2RGP | 3W33, 3BEL |
| 3CJO | 2FME, 2FL2, 6HKX, 2FKY, 2G1Q, 1YRS, 2FL6 |
| 3I4B | 5BVF 3I60 |
| 1Q4X | 3IMY |
| 2VWZ | 5MJA, 4YJS, 8BK0, 4YJP, 4P4C, 2VWY, 4G2F |
| 3HMM | 6B8Y, 3KCF, 1VJY, 1PY5, 3FAA, 5QTZ, 1RW8 |
| 5L7H | 2OAX, 5MWY, 5L7G |
| 2OI0 | 3B92 |
| 2AZR | 2BGE, 7FQU, 2HB1 |
| 3FRG | 5C2E, 5SHB, 4FCD, 5SF7, 3GWT, 6MSC, 5SHG, 5SJF, 5TKB, 4PM0, 5SKF, 3SNI |
| 1LRU | 1BSJ |
| 1ZW5 | 4OGU, 4DXJ, 3ICZ, 4DWB, 4KPD |
| 3L3M | 4RV6, 5XSR, 5A00, 5XST, 2RCW, 6I8M |
| 1EVE | 6TT0, 6EZH, 7D9Q, 6EZG, 7D9P, 7D9O, 5NAP |
| 3CHP | 7AV1 |

| | |
|---|---|
| 1F0R | 1F0S |
| 3NY8 | 2YCZ, 7DHI, 5A8E, 6PS1, 4BVN |
| 3KK6 | 3N8X, 5KIR |
| 3M2W | 3FYJ |
| 1L2S | 6DPY, 1XGJ, 1XGI, 4KZ4, 4JXS, 6DPX |
| 1B9V | 3CKZ, 1B9T 1VCJ, 3K39, 4DGR |
| 2QD9 | 6M95, 3ZSH |
| 3NW7 | 3NW6, 3I81 |
| 3K5E | 1X88, 2FL6, 2FME, 2X7C, 2X7D |
| 1J4H | 1J4I |
| 1S3V | 1MVT |
| 2ZDT | 2GMX, 2ZDU |
| 2VT4 | 6PS1, 5A8E, 2YCZ, 6PS4, 2Y04 |
| 1CX2 | 5KIR |
| 3KC3 | 3R30 |
| 1KVO | 1KQU, 1J1A |

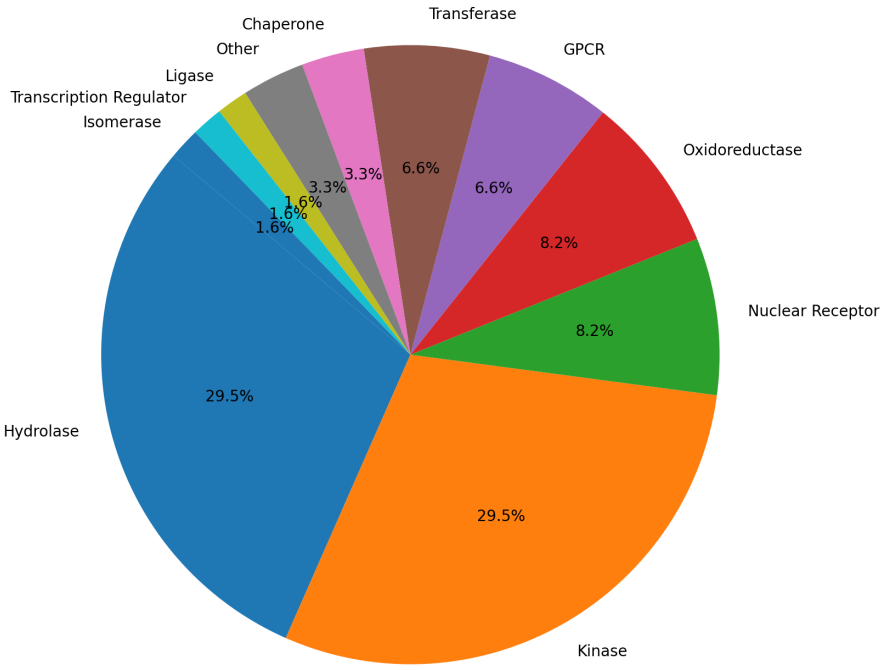



Figure S6: Overview of protein diversity in AlignDockBench.

### C.2 Structure Preparation

Each structure retrieved from the PDB was repaired and protonated using the PDB2PQR program [Dolinsky et al., 2007] with the AMBER force field. Subsequently, energy minimization was performed using the GROMACS molecular dynamics engine [Van Der Spoel et al., 2005] with the AMBER03 force field in implicit solvent. Small molecules were parametrized using GAFF2 [Wang

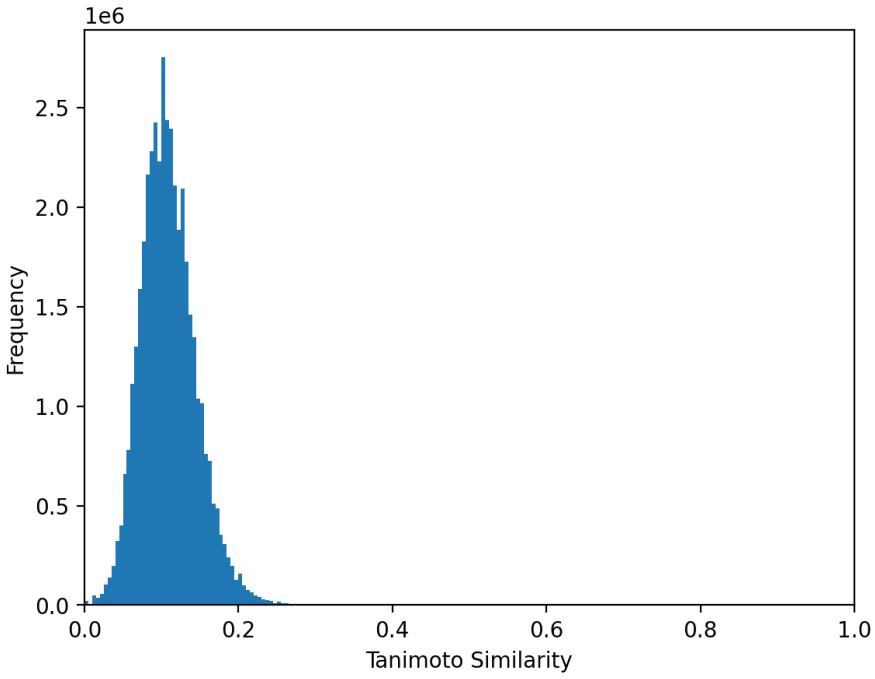



Figure S7: Morgan fingerprints tanimoto similarity histogram between AlignDockBench and the training set molecules.

et al., 2004] with the ambertools suite [Case et al., 2023], and partial charges were assigned using AM1-BCC [Jakalian et al., 2002]. The minimization was performed in two stages: initially, 5,000 steps (time step = 2 fs) were conducted with positional restraints of 10,000 kJ/mol·nm² applied to all heavy atoms. In the second stage, another 5,000 steps were performed with restraints of 1,000 kJ/mol·nm² applied to the protein backbone and all heavy atoms of the small molecules.

# D Experimental details

## D.1 Experimental setup

**Vina and rDock.** Docking simulations were performed as follows: a random conformer of each query molecule was generated using RDKit ETKDGv2 [Riniker and Landrum, 2015] and minimized with MMFF94 forcefields [Halgren, 1996]. Protonation states were assigned using the internal proprietary small-molecule protonation model at physiological pH (7.4). Ten docking poses were generated per molecule. In rDock, docking was carried out in "dock" mode using the default "STANDARD SCORE" scoring function. For AutoDock Vina, docking was performed with an exhaustiveness parameter set to 15, using the default Vina scoring function.

**FitDock and LSalign.** To prepare ligand inputs for both FitDock [Yang et al., 2022] and LS-align [Hu et al., 2018], we generated 10 random conformers per query molecule using the same protocol described above, based on RDKit's ETKDGv2 method followed by MMFF94 energy minimization. Each minimized conformer was exported as an individual SDF file, then converted to MOL2 format using Open Babel [O'Boyle et al., 2011], with partial atomic charges assigned using the Gasteiger method [Gasteiger and Marsili, 1980]. For each template–query pair, FitDock was provided with the MOL2 representations of both the template and query ligands, as well as the corresponding PDB files of the template and query proteins. LSalign requires only the MOL2 ligand files; we used its flexible alignment mode with the "-rf" flag enabled, allowing torsions to adjust during superposition. After alignment, we selected the top-ranked conformer based on each software's primary scoring metric: the "Binding Score" for FitDock, and the "RMSD lb" output for LSalign.

**ROSHAMBO.** For the ROSHAMBO software [Atwi et al., 2024], the query ligand was provided as a single 2D SDF file, from which ten aligned poses were generated. Shape volumes were modeled using the "Gaussian" option. The resulting poses were ranked according to the "ComboTanimoto" score, and the highest-scoring pose was retained.

### D.2 Supplementary results

#### D.2.1 Cumulative distribution of RMSD values

In Figure S8 we report the cumulative distribution of RMSD values on AlignDockBench for each method.

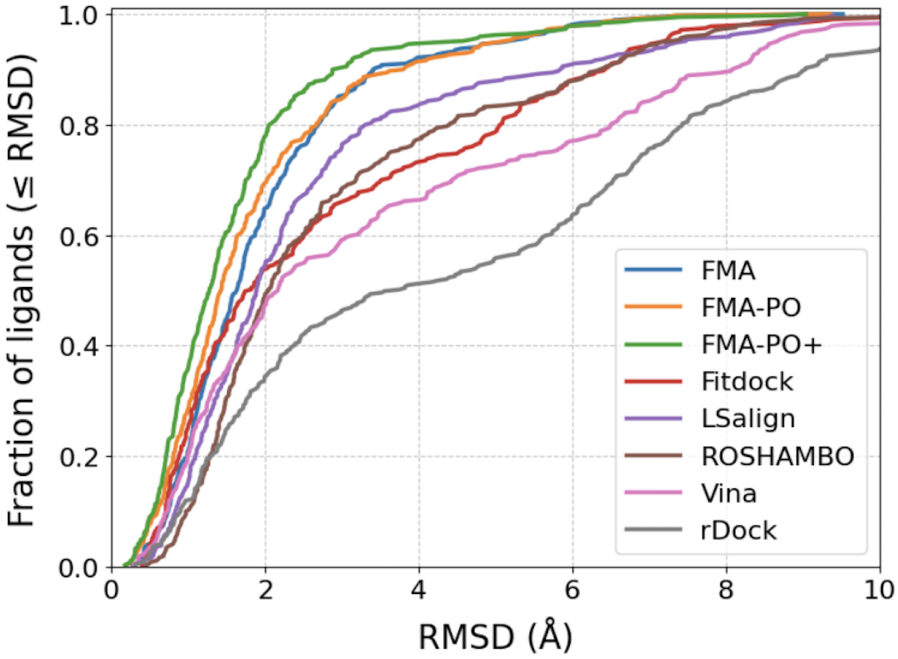



Figure S8: Cumulative distribution of RMSD values: percentage of molecules with RMSD below different thresholds

#### D.2.2 Redocking experiments

Results for each method in a redocking scenario, where the query ligand is aligned within its own crystal protein, are presented in Table S4. Methods like ROSHAMBO and LSalign, which do not incorporate protein information, have identical performance in both redocking and crossdocking contexts.

Figure S9 compares the performance of pocket-aware methods in both crossdocking and redocking scenarios, highlighting the gain of using true protein context for pose accuracy. Traditional docking approaches, such as Vina and rDock, exhibit significant improvements in redocking due to the availability of the correct pocket environment. FMA-PO+ also shows a slight performance gain in the redocking scenario whereas FMA-PO shows no notable difference.

#### D.2.3 Comparison to HarmonicFlow

We compared FMA to the recent flow-based docking method HarmonicFlow Stärk et al. [2023] using the official settings from the authors' repository. As shown in Table S5, FMA significantly outperforms HarmonicFlow in both mean RMSD and success rate ($< 2$Å), even when reporting HarmonicFlow's best RMSD across the 10 sampled poses. Applying PO to HarmonicFlow improves results but still underperforms FMA, highlighting the higher quality of FMA's pose distribution and the benefit of using the template ligand to guide docking.

Table S4: Performance comparison of 3D molecular alignment and docking methods on AlignDock-Bench in a redocking scenario. Methods marked with an (*) use GPU. For methods that did not align all 369 molecules, percentages are reported as X/Y, where the first value is calculated over aligned molecules only, and the second over the total set of 369 molecules.

| **Method** | # of Molecules Aligned | Mean RMSD (Å ↓) | % of Molecules with RMSD $<$ 2Å (↑) |
|---|---|---|---|
| FMA-PO* | 369/369 | 1.86 ± 1.43 | 68.56 |
| FMA-PO+* | 369/369 | **1.52 ± 1.24** | **82.11** |
| FitDock | 292/369 | 3.27 ± 4.64 | 54.11 / 42.82 |
| LSalign | 368/369 | 2.54 ± 2.00 | 54.35 / 54.2 |
| ROSHAMBO* | 369/369 | 2.87 ± 2.09 | 48.51 |
| rDock | 369/369 | 2.79 ± 2.93 | 60.43 |
| Vina | 368/369 | 1.95 ± 2.54 | 75.82 / 75.61 |

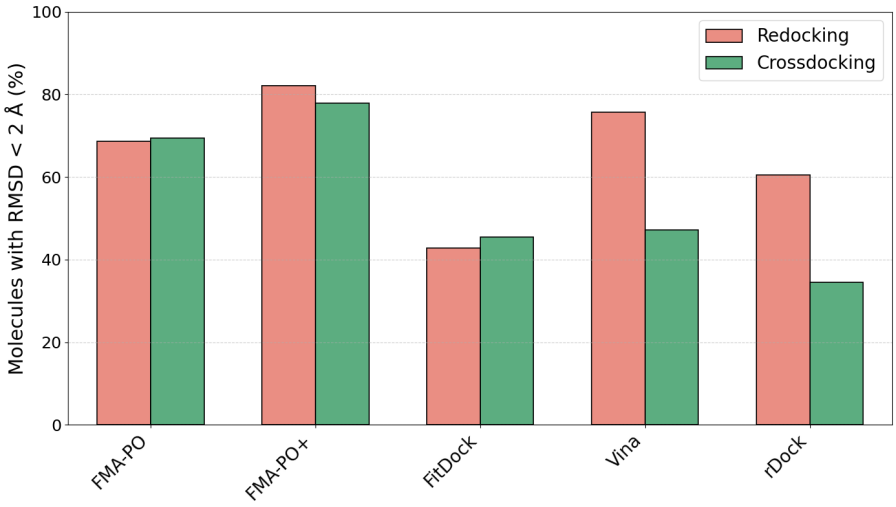



Figure S9: Pose accuracy comparison in redocking and crossdocking scenarios.

Table S5: Comparison to Harmonic Flow

| **Method** | # of Molecules with successful pose generation | Mean RMSD (Å ↓) | % of Molecules with RMSD $<$ 2Å (↑) |
|---|---|---|---|
| FMA (Top score) | 369/369 | 1.97 ± 1.36 | 64.77 |
| FMA (Top RMSD) | 369/369 | 1.39 ± 0.85 | 84.55 |
| FMA + PO+ (Top score) | 369/369 | 1.62 ± 1.33 | 77.78 |
| FMA + PO+ (Top RMSD) | 369/369 | 1.11 ± 0.85 | 91.6 |
| Harmonic-Flow (Top RMSD) | 356/369 | 4.47 ± 2.07 | 2.81/2.71 |
| Harmonic-Flow + PO +(Top RMSD) | 356/369 | 3.71 ± 2.22 | 19.94/19.24 |

### D.2.4 Discussion about runtime

We analyzed runtime as a function of ligand size for both FMA and PO. As shown in Table S6, PO (2.7 s per pose, run on CPU) is slower than FMA (0.84 s for 10 poses, run on GPU). In both cases, runtime increases moderately with the number of heavy atoms and rotatable bonds (Pearson ∼0.2), without signs of exponential growth.

Runtime is a critical factor for practical applications. Each molecule is processed independently, so scaling across multiple compute instances is straightforward. To illustrate this, we report in Table S7

Table S6: Time, * GPU

| **Method** | Average runtime (s) | Pearson correlation with the number of heavy atoms | Pearson correlation with the number of rotatable bonds |
|---|---|---|---|
| FMA$*$ | 0.84 s (for 10 poses) | 0.20 | 0.23 |
| PO | 2.7 s (for 1 pose) | 0.19 | 0.23 |

the estimated runtime and cost of processing various dataset sizes using 100 AWS `g4dn.2xlarge` instances in parallel. This setup could be use within a de novo drug design pipeline, where screening around 50,000 molecules is a realistic and relevant scale. For ultra-large virtual screening campaigns (e.g., tens or hundreds of millions of molecules), further acceleration and simplification of PO would be required. For example, moving to a batched optimization strategy, rather than optimizing one molecule at a time, could significantly reduce total runtime.

Table S7: Estimated runtime (in hours) and cost (USD) using 100 AWS `g4dn.2xlarge` instances in parallel for different numbers of molecules.

| **Number of molecules** | **50,000** | $\mathbf{10^5}$ | $\mathbf{10^6}$ |
|---|---|---|---|
| Time FMA+PO (h) | 0.49 | 0.98 | 9.83 |
| Cost FMA+PO ($) | 41 | 82 | 826 |

### D.3 Quality of predictions

Figure S10 reports of the strain energy as defined by GenBench3D Baillif et al. [2024], for each method. FMA yields higher strain energies, due to the absence of the PO module. While unrealistic conformations are a known limitation of deep learning models [Buttenschoen et al., 2024], this was one of the key motivations for introducing our PO module. Both FMA-PO and FMA-PO+ reduce strain energy.

Table S8 reports the number of molecules exhibiting clashes with protein, per method. FMA suffers from significantly more clashes, due to the absence of pocket conditioning. FMA-PO and FMA-PO+ reduce the number of clashes, confirming the effectiveness of the PO module in improving structural plausibility.

To further evaluate the quality of generated poses, we employed the PoseBusters test suite [Buttenschoen et al., 2024], which checks a range of geometric and chemical criteria to ensure physically plausible ligand conformations within the binding pocket. We observed that some of the $369$ ground-truth crystallographic query poses from AlignDockBench did not fully satisfy the PoseBusters criteria, particularly the minimum distance to the protein. As a result, we restricted our evaluation to a subset of 263 molecules for which the ground-truth poses passed all PoseBusters checks.

Figure S11 presents, for each method, the percentage of molecules achieving an RMSD below $2$Å, along with the proportion that also satisfies the PoseBusters criteria. Traditional docking methods tend to yield a higher proportion of PoseBusters-valid poses. In contrast, purely LB methods like LSalign and ROSHAMBO exhibit a substantial drop in valid poses, reflecting the importance of protein context in guiding pose generation. FMA (without PO) shows a larger drop in PoseBusters-valid poses compared to FMA-PO+, highlighting the critical role of PO and the importance of sampling multiple initial poses. Nevertheless, FMA-PO+ still exhibits a $12\%$ drop, primarily due to the minimum distance-to-protein check. This suggests that the method could be further improved by incorporating additional pocket-specific constraints or assigning greater weight to the pocket score during the PO stage. Additionally, integrating protein context directly into the initial pose generation with the FMA model may enhance the physical plausibility of the generated poses.

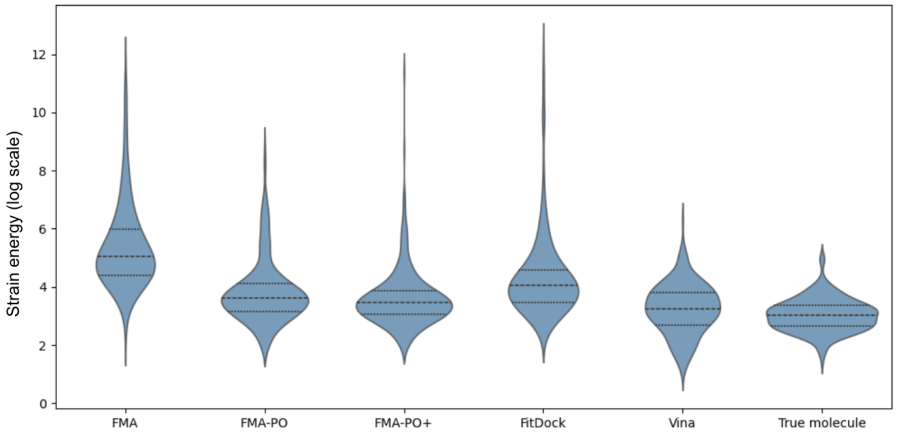



Figure S10: Strain energies of poses across methods (log scale).

Table S8: Number of molecules with protein clashes with the protein, per method.

| **Method** | **1 clash** | **2 clashes** | **$\geq$ 3 clashes** |
|---|---|---|---|
| FMA | 37 | 16 | 20 |
| FMA-PO | 35 | 9 | 7 |
| FMA-PO+ | 21 | 6 | 2 |
| Vina | 1 | 0 | 0 |
| FitDock | 7 | 5 | 4 |
| Co-crystallized Ligand | 0 | 0 | 0 |

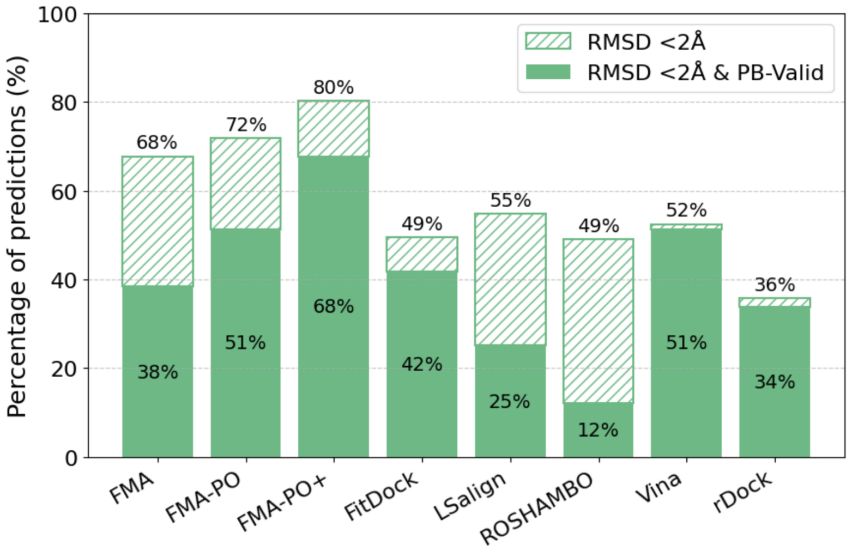



Figure S11: Percentage of accurate poses (RMSD $<$ 2Å) and PoseBusters-valid (PB-valid) poses for each method, evaluated on a subset of 263 molecules whose ground-truth crystallographic poses pass the PoseBusters checks.

### D.4 Ablation studies

#### D.4.1 FMA's contribution

We generated 10 candidate poses per method (FMA, RDKit-centroid, Vina, LS-align, FitDock-LB), applied PO to each, and selected the top-ranked pose. Results are reported in In Table S9. The RDKit-centroid baseline places a random conformer at the binding-site center. FMA-PO+ consistently yields the lowest RMSD among all alternatives, indicating that FMA is the best initialization strategy for PO and that its accuracy is crucial.

Table S9: PO applied to different base alignments. All methods use flexible search mode and identical PO hyperparameters.

| **Method + PO** | # of Molecules Aligned | Mean RMSD (Å ↓) | % of Molecules with RMSD $<$ 2Å (↑) |
|---|---|---|---|
| FMA + PO+ | 369/369 | 1.62 ± 1.33 | 77.78 |
| rdkit + PO+ | 369/369 | 4.63 ± 1.94 | 10.33 |
| Vina + PO+ | 368/369 | 1.96 ± 1.87 | 71.47/71.27 |
| LS-align + PO+ | 357/369 | 2.31 ± 2.07 | 63.31/61.25 |
| Fitdock LB + PO+ | 365/369 | 2.78 ± 2.41 | 52.6/52.03 |

#### D.4.2 PO's contribution

In Table S10, we evaluate the impact of the PO stage and its objectives on pose prediction. FMA, exhibits the poorest performance across all metrics, highlighting the importance of the pose refinement. Moreover, including the pocket during PO enhances the quality of the generated poses, as evidenced by lower mean RMSD values and higher percentages of accurate alignments for both FMA-PO and FMA-PO+.

Table S10: Ablation studies of PO loss terms on AlignDockBench in a crossdocking scenario.

| **Method** | Mean RMSD (Å ↓) | % of Molecules with RMSD $<$ 2Å (↑) |
|---|---|---|
| FMA | 1.97 ± 1.36 | 64.77 |
| FMA-PO | 1.86 ± 1.42 | 69.38 |
| FMA-PO w/o $\mathcal{S}_{STS}$ | 1.98 ± 1.38 | 63.96 |
| FMA-PO w/o $\mathcal{S}_{PTS}$ | 1.86 ± 1.38 | 67.48 |
| FMA-PO w/o $\mathcal{S}_{\text{pocket}}$ | 1.88 ± 1.41 | 67.75 |
| FMA-PO w/o $\mathcal{E}_{\text{internal}}$ | 2.8 ± 0.91 | 13.82 |
| FMA-PO+ | 1.62 ± 1.33 | 77.78 |
| FMA-PO+ w/o $\mathcal{S}_{STS}$ | 1.85 ± 1.32 | 66.67 |
| FMA-PO+ w/o $\mathcal{S}_{PTS}$ | 1.71 ± 1.33 | 74.25 |
| FMA-PO+ w/o $\mathcal{S}_{\text{pocket}}$ | 1.70 ± 1.39 | 75.61 |
| FMA-PO+ w/o $\mathcal{E}_{\text{internal}}$ | 2.98 ± 1.26 | 14.09 |

In Table S11) we applied PO to top poses from other methods. PO consistently reduced RMSD supporting its general effectiveness as a refinement stage. PO also improves chemical plausibility and geometrical validity as demonstrated in Figure S10 and Table S8

#### D.4.3 Vina contribution to $\mathcal{S}_{\text{score}}$

We evaluated the influence of the $S_{\text{Vina}}$ component in the total score $\mathcal{S}_{\text{score}}$ by systematically ablating it across three key configurations: FMA, FMA-PO(i.e., scoring of FMA poses and PO applied to the top-ranked pose), and FMA-PO+(i.e., PO applied to all poses, and scoring of optimized poses). The

Table S11: Impact of PO on FMA, Vina, LS-align and FitDock LB compared to the base aligners alone.

| **Method** | # Aligned | Mean RMSD (Å ↓) | % RMSD $<$ 2Å (↑) |
|---|---|---|---|
| FMA | 369/369 | 1.97 ± 1.36 | 64.77 |
| FMA + PO | 369/369 | 1.86 ± 1.42 | 69.38 |
| Vina | 368/369 | 3.39 ± 2.81 | 47.28 / 47.15 |
| Vina + PO | 368/369 | 3.24 ± 2.85 | 52.17 / 52.03 |
| LS-align | 368/369 | 2.54 ± 2.00 | 54.35 / 54.20 |
| LS-align + PO | 357/369 | 2.47 ± 2.15 | 59.66/57.72 |
| FitDock LB | 366/369 | 2.93 ± 2.36 | 49.73/ 49.32 |
| Fitdock LB + PO | 366/369 | 2.75 ± 2.38 | 52.88/52.3 |

results are summarized in Table S12. These results indicate that Vina scoring provides limited benefit when applied to unoptimized poses (FMA or FMA-PO), but can help select better candidates among refined poses (FMA-PO+). Our method still outperforms all baselines without using the Vina score in $\mathcal{S}_{\text{score}}$.

Table S12: Vina contribution to $\mathcal{S}_{\text{score}}$

| **Method** | Vina term in $\mathcal{S}_{\text{score}}$ | Mean RMSD (Å ↓) | % of Molecules with RMSD $<$ 2Å (↑) |
|---|---|---|---|
| FMA | Yes | 1.97 ± 1.36 | 64.77 |
| FMA | No | 1.96 ± 1.35 | 64.77 |
| FMA-PO | Yes | 1.86 ± 1.42 | 69.38 |
| FMA-PO | No | 1.81 ± 1.40 | 69.38 |
| FMA-PO+ | Yes | 1.62 ± 1.33 | 77.78 |
| FMA-PO+ | No | 1.68 ± 1.38 | 75.61 |

### D.4.4 Impact of the number of generated conformations

We assessed the impact of the number of generated poses with FMA on final pose accuracy. Figure S12 shows the percentage of molecules with RMSD below 2Å and 1.5Å for the best (i.e., lowest RMSD) pose of FMA-PO and FMA-PO+ as a function of the number of sampled poses prior to PO. Performance increases with the number of generated poses, with higher sampling yielding a greater proportion of low RMSD predictions.

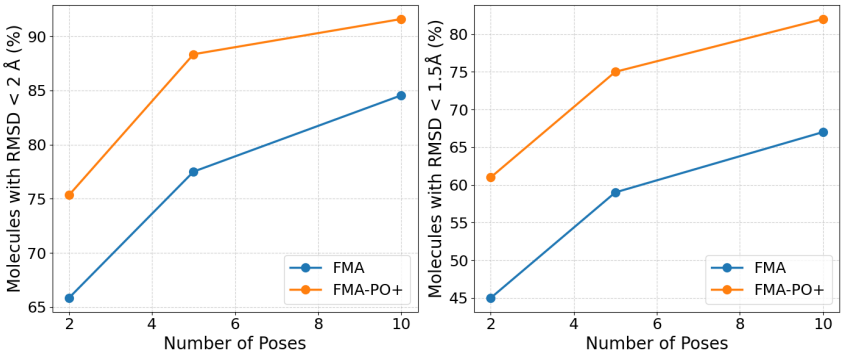



Figure S12: Effect of the number of generated poses prior to PO on the pose accuracy of FMA-PO and FMA-PO+ on AlignDockBench.

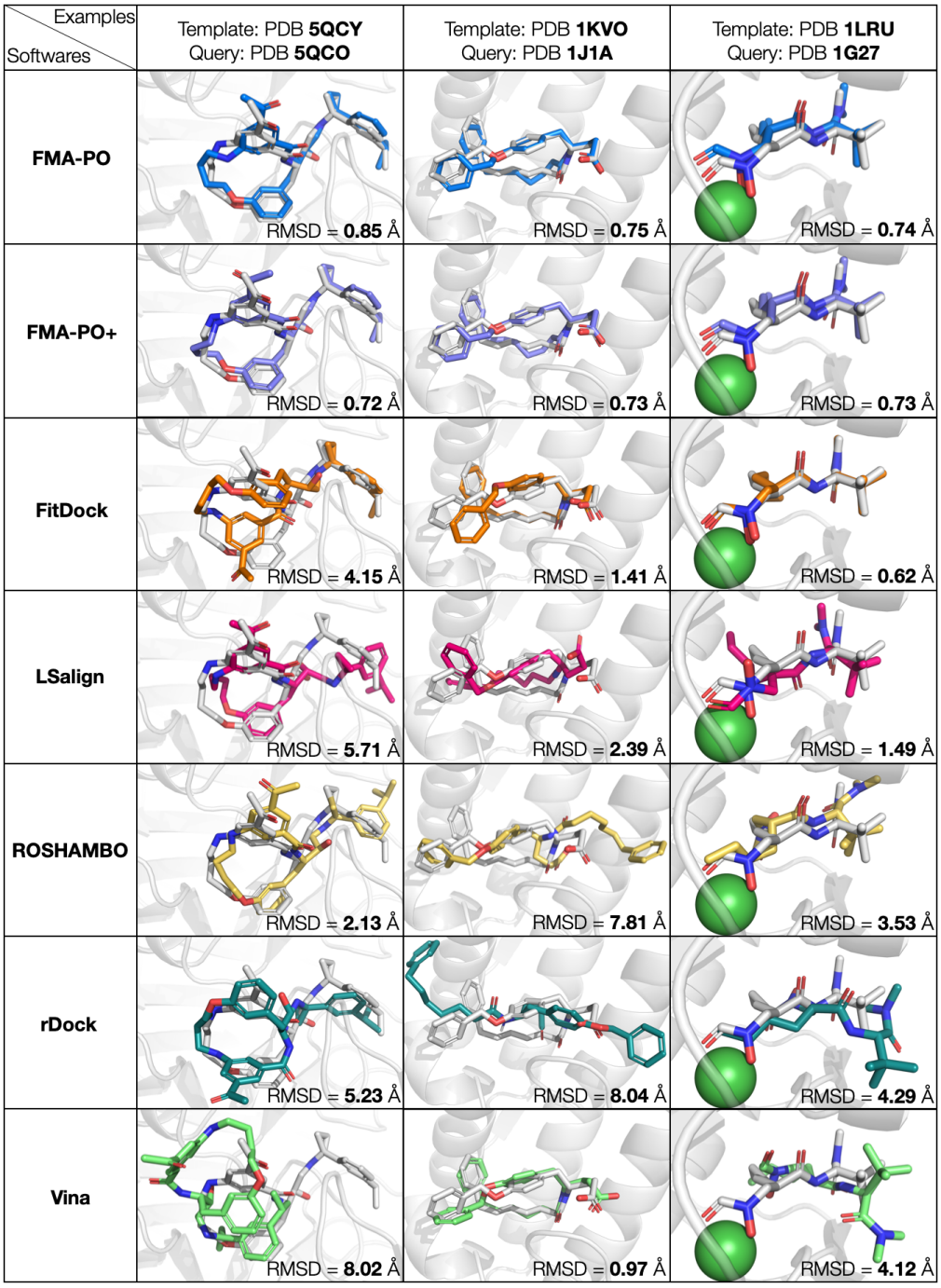



Figure S13: **Comparison of predicted binding poses of various ligands using 3D molecular alignment and docking methods.** The crystallized protein and ligand are shown in grey. The first example is based on PDB entry 5QCO, corresponding to the human Beta-secretase 1 (BACE1), co-crystallized with a macrocyclic compound [Parks et al., 2020], using the PDB entry 5QCY [Parks et al., 2020] as template. The second structure is based on PDB entry 1J1A, corresponding to the pancreatic secretory phospholipase A2 co-crystallized with an inhibitor [Hansford et al., 2003], using the PDB entry 1KVO [Cha et al., 1996] as template. The third structure is based on PDB entry 1G27, corresponding to a polypeptide deformylase from *Escherichia coli* co-crystallized with an inhibitor [Clements et al., 2001], using the PDB entry 1LRU [Guilloteau et al., 2002] as template. The quality of the predicted poses was assessed by computing the RMSD relative to the crystallized ligand conformation.