

TriGlue: a Biology-Inspired Generative Model for Generating Molecular Glue-Induced Ternary Complex

Yuliang Yan¹, Shuo Yan¹, Haochun Tang^{2,1}, Yiqin Sun³, Enyan Dai¹

¹The Hong Kong University of Science and Technology (Guangzhou)

²Jilin University

³University of Sydney

Abstract

Molecular glue degraders have emerged as a promising strategy for targeted protein degradation by inducing ternary complex formation between an E3 ubiquitin ligase and a target protein. Despite their therapeutic potential, computational design of molecular glues remains largely unexplored. Unlike conventional structure-based drug design, molecular glue design is governed by the unknown protein-protein interface and requires the simultaneous modeling of ligand generation, protein-protein docking, and ternary complex assembly. In this work, we formulate molecular glue design as a ternary complex generation problem and propose a biology-inspired generative framework, **TriGlue**. Motivated by the mechanism of molecular glue action, we decompose ternary complex generation into two coupled stages: interface estimation and interface-conditioned complex generation. First, we develop an SE(3)-equivariant interface estimation module that predicts a geometrically constrained protein-protein interface from unbound monomer structures. Second, we introduce an interface-conditioned ternary flow matching network that jointly generates the molecular glue and predicts the rigid-body transformation required to assemble the ternary complex. Extensive experiments demonstrate that **TriGlue** generates chemically valid molecules and produces plausible ternary complexes, which highlight the potential of biology-inspired generative modeling for accelerating molecular glue discovery. Our code is available at <https://anonymous.4open.science/r/molecular-glue-design-806B>.

1 Introduction

Recently, molecular glue degraders (MGDs) have emerged as a rapidly growing strategy in targeted protein degradation [1, 2, 3, 4, 5]. As illustrated in Fig. 1, the mechanism of molecular glue degraders (MGDs) proceeds as follows: The molecular glue binds to the E3 ubiquitin ligase and remodels its interaction surface. This remodeled interface recruits and stabilizes the association of the target protein (POI), forming a ternary complex. Then, the E2 enzyme transfers ubiquitin molecules to the POI, resulting in ubiquitination, and the ubiquitinated POI is recognized by the proteasome and degraded, completing the targeted protein degradation process. Molecular glues are typically structurally simple, drug-like, and increasingly clinically validated [6]. Therefore, molecular glue development has become a rapidly emerging focus in modern drug discovery [7, 8, 9].

Accelerating the discovery of molecular glues remains a central objective. Traditional molecular glue discovery has largely relied on expensive and labor-intensive experimental screening, and to date, only

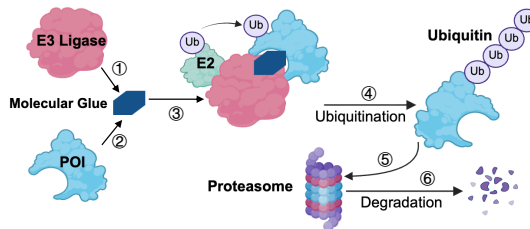


Figure 1: MGDs in targeted protein degradation.

a few hundred molecular glues have been identified [10, 11, 12]. AI-based approaches have recently propelled drug discovery [13, 14, 15, 16]. However, these methods are not suitable for molecular glue design, where ligand generation, protein docking, and ternary complex formulation must be considered simultaneously. Fundamentally, they are designed for binary protein–ligand binding and typically assume the existence of a predefined binding pocket on a single protein. DeepTernary [17] introduces an SE(3)-equivariant network for ternary complex structure prediction. However, it focuses on predicting structures for known ligands and does not address *de novo* drug design. Therefore, we pursue generative drug design tailored to ternary molecular glue systems.

Molecular glue design is to generate a small molecule that induces an E3 ubiquitin ligase and a target protein to form a productive ternary complex. Achieving this goal requires the simultaneous optimization of three tightly coupled components: the molecule-induced E3 ligase–target protein interface, the novel small molecule, and the rigid-body transformation that positions the two proteins into a coherent ternary complex. This gives rise to two major challenges. *First*, the success of molecular glue design is determined by the formation of a compatible ligand-mediated protein–protein interface, which ultimately dictates whether a stable ternary complex can form. However, this interface is unknown prior to complex assembly and cannot be directly observed from the unbound proteins. *Second*, molecular glue design requires interface-conditioned generative modeling that jointly produces a novel ligand and the rigid-body transformation aligning the two proteins simultaneously. This involves tightly coupled dependencies across three components: the ligand structure, the protein–protein docking pose, and the interface geometry that stabilizes their interaction.

To address these challenges, we propose a biology-inspired generative framework named **TriGlue**. (i) We develop an SE(3)-equivariant interface estimation module that predicts a protein–protein interface from the two monomer structures, providing a geometrically constrained and biologically meaningful structural prior. (ii) We introduce an interface-conditioned ternary flow matching network that jointly generates the molecular glue and predicts the rigid-body rotation and translation required for protein–protein docking under the guidance of the estimated interface. In summary, our main contributions are as follows:

- We investigate the novel problem of molecular glue design for ternary complexes.
- We propose a biology-inspired generative framework that decomposes ternary complex generation into interface estimation and interface-conditioned ternary flow generation.
- Extensive experiments demonstrate the effectiveness of the proposed framework in generating valid small molecules and plausible ternary complexes.

2 Preliminaries

2.1 Molecular Glue-Induced Ternary Complex Formation

Mechanistic Overview. Molecular glue drives ternary complex formation by binding to one protein and reshaping its surface to recruit a second partner, thereby inducing or stabilizing a protein–protein interaction. They remodel the interaction landscape and convert transient encounters into stable assemblies, rather than acting as simple binders. Molecular glues can modulate protein function, localization, or stability through cooperative interactions among the ligand and both proteins, which explains their growing importance in therapeutic discovery.

Molecular Glue-induced Protein-protein Interface. Molecular glue activity is determined by the creation of a ligand-mediated protein–protein interface. The ligand occupies the interfacial region and stabilizes complementary surface patches from both proteins, effectively bridging them and reducing the tendency of the complex to dissociate. Therefore, the molecular glue-induced protein–protein interface serves as an important biological prior to form the ternary complex.

Dynamic Ternary Complex Assembly. Ternary complex formation is inherently dynamic. The two proteins typically start from unbound conformations and undergo rigid-body rearrangements as ligand binding stabilizes the complex. The final ternary structure therefore emerges from a coupled process involving ligand placement, protein docking, and cooperative stabilization. This dynamic view naturally motivates a generative formulation that jointly models ligand structure and protein motion, aligning the learning objective with the underlying biological mechanism.

2.2 Problem Formulation

Formally, we formulate molecular-glue-induced ternary complex design as a generative modeling task. Specifically, a ternary complex is defined as $\mathcal{C} = (\mathcal{R}, \mathcal{M}, \mathcal{T})$, where $\mathcal{R}$ and $\mathcal{T}$ denote the receptor and target proteins, and $\mathcal{M}$ is a small molecule that induces their association. Each protein is represented as a residue-level graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where node i corresponds to residue identity AA_i with $\text{C}\alpha$ coordinate $\mathbf{X}_i \in \mathbb{R}^3$, and edges are constructed using a k -nearest-neighbor scheme. The ligand is represented as a molecular graph $\mathcal{G}_{\mathcal{M}} = (\mathcal{V}_{\mathcal{M}}, \mathcal{E}_{\mathcal{M}})$, where each node corresponds to an atom type $\mathbf{a}_i$ with coordinate $\mathbf{x}_i \in \mathbb{R}^3$. During generation, the bound ternary complex is unknown. We are given a receptor protein $\mathcal{R}$ in its bound conformation, an unbound target protein $\tilde{\mathcal{T}}$ with unknown pose, and a ligand $\mathcal{M}$ to be generated. The bound target structure is related to the unbound structure by a closed-form rigid-body transformation:

$$\mathbf{X}_{\mathcal{T}} = \mathbf{R}\mathbf{X}_{\tilde{\mathcal{T}}} + \mathbf{t}, \quad \mathbf{R} \in \text{SO}(3), \mathbf{t} \in \mathbb{R}^3. \quad (1)$$

Problem 1. Given a bound receptor protein $\mathcal{R}$ and an unbound target protein $\tilde{\mathcal{T}}$, molecular glue-induced ternary complex design is formulated as learning the conditional distribution:

$$p(\mathcal{C} \mid \mathcal{R}, \tilde{\mathcal{T}}). \quad (2)$$

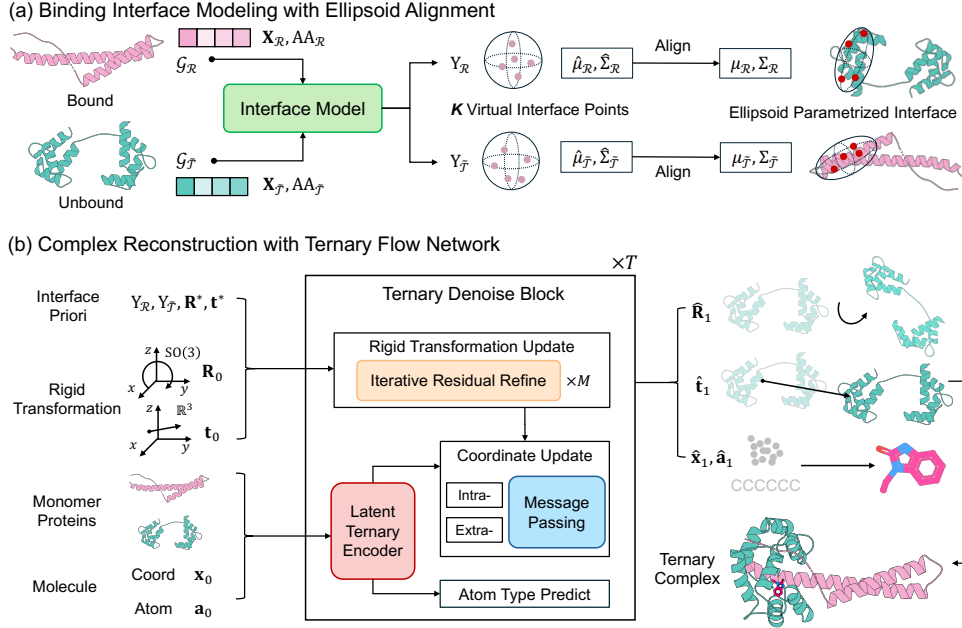


Figure 2: Framework of our interface-conditioned ternary flow network TriGlue.

3 Method

Following the biological process of molecular glue-induced ternary complex formation, we formulate TriGlue as a biology-inspired generative framework. As shown in Fig. 2, we factorize ternary complex generation into two coupled modules: an interface estimation module that infers a global virtual interface from two monomer proteins, and an interface-conditioned complex generation module that jointly generates the molecular glue and estimates the inter-protein rigid-body transformation to assemble a coherent ternary complex. This factorization aligns the generative process with the biological mechanism where a glue-mediated protein-protein interface guides subsequent ternary complex assembly. The factorized design raises two key challenges. (i) the inferred interface should be constrained in a geometrically meaningful manner to ensure its utility in ternary complex modeling; (ii) the predicted interface should effectively guide a unified generative process that simultaneously produces a novel molecular glue and drives the rigid-body assembly of the two proteins. To address these challenges, we constrain the estimated interface with an ellipsoid-parameterized geometry. We further inject interface information through an interface-conditioned ternary flow network. We next describe the factorization process and the two corresponding modules in detail.

3.1 Biology-Inspired Factorization of Ternary Complex Generation

Inspired by the mechanism of molecular glue-induced ternary complex formation, we decompose the Problem 1 into two sub-goals: interface estimation and interface-conditioned complex generation. Specifically, this decomposition leads to two coupled probabilistic components: one estimates the molecular glue-induced interface between the two proteins, and the other generates the ternary complex conditioned on the inferred interface as follows:

$$p(\mathcal{C} \mid \mathcal{R}, \tilde{\mathcal{T}}) = \underbrace{p(\mathcal{I}_{\mathcal{R}}, \mathcal{I}_{\tilde{\mathcal{T}}} \mid \mathcal{R}, \tilde{\mathcal{T}})}_{\text{Interface Estimation}} \cdot \underbrace{p((\mathcal{R}, \mathcal{M}, \mathcal{T}) \mid \mathcal{I}_{\mathcal{R}}, \mathcal{I}_{\tilde{\mathcal{T}}}, \mathcal{R}, \tilde{\mathcal{T}})}_{\text{Interface-Conditioned Complex Generation}}, \quad (3)$$

where $\mathcal{I}_{\mathcal{R}}$ and $\mathcal{I}_{\tilde{\mathcal{T}}}$ denote the sets of interface residues, whose $C\alpha$ atoms lie within $8\text{\AA}$ of the binding partner in the complex. This biology-inspired factorization brings two key benefits: (i) It faithfully matches the mechanism of molecular-glue action; (ii) It transforms the challenging joint modeling problem into two tractable sub-tasks. We next present our approach for modeling both components.

3.2 Interface Estimation under Ellipsoidal Geometric Constraints

Protein–protein interface governs the process of molecular glue design, yet such information is unavailable when only monomer structures are given. This absence prevents the direct incorporation of biologically grounded geometric constraints during generation. We introduce an interface estimation module that infers a *Global Virtual Interface* from the two monomer structures. We then align this virtual interface with an *Ellipsoid Parameterized Representation* of the physical protein–protein interface. In the following, we present the detailed formulation of this module.

Protein Encoder. Characterizing the interface first requires encoding each monomer protein to obtain latent representations. Given two proteins $\mathcal{G}_{\mathcal{R}}$ and $\mathcal{G}_{\tilde{\mathcal{T}}}$, we encode them with EGNN [18] to obtain the invariant features $\mathbf{F}_{\mathcal{R}} \in \mathbb{R}^{d \times N}$, $\mathbf{F}_{\tilde{\mathcal{T}}} \in \mathbb{R}^{d \times M}$ and SE(3)-equivariant coordinates $\mathbf{Z}_{\mathcal{R}} \in \mathbb{R}^{3 \times N}$, $\mathbf{Z}_{\tilde{\mathcal{T}}} \in \mathbb{R}^{3 \times M}$ as follows:

$$\mathbf{F}_{\mathcal{R}}, \mathbf{Z}_{\mathcal{R}}, \mathbf{F}_{\tilde{\mathcal{T}}}, \mathbf{Z}_{\tilde{\mathcal{T}}} = \text{EGNN}(\mathcal{G}_{\mathcal{R}}, \mathcal{G}_{\tilde{\mathcal{T}}}), \quad (4)$$

where N, M is the node (residue) number and d is the hidden dimension.

Global Virtual Interface. To capture protein–protein interaction regions in a compact, geometry-aware manner, we map each protein to a fixed set of K *virtual interface points* [19], denoted as $\mathbf{Y}_{\mathcal{R}}, \mathbf{Y}_{\tilde{\mathcal{T}}} \in \mathbb{R}^{3 \times K}$. The interface modeling in Eq. (3) is further approximated as: $p(\mathbf{Y}_{\mathcal{R}}, \mathbf{Y}_{\tilde{\mathcal{T}}} \mid \mathcal{R}, \tilde{\mathcal{T}})$. Specifically, we represent each virtual point as an attention-weighted aggregation of coordinates:

$$\mathbf{y}_{\mathcal{R},k} = \sum_{i=1}^N \alpha_{\mathcal{R},i}^k \mathbf{z}_{\mathcal{R},i}, \quad \mathbf{y}_{\tilde{\mathcal{T}},k} = \sum_{j=1}^M \alpha_{\tilde{\mathcal{T}},j}^k \mathbf{z}_{\tilde{\mathcal{T}},j}, \quad (5)$$

where the attention weights are computed via cross-protein feature attention:

$$\alpha_{\mathcal{R},i}^k = \text{softmax} \left(\frac{1}{\sqrt{d}} \mathbf{f}_{\mathcal{R},i}^{\top} \mathbf{W}_{\mathcal{R}}^k \phi(\mathbf{F}_{\tilde{\mathcal{T}}}) \right), \quad \alpha_{\tilde{\mathcal{T}},j}^k = \text{softmax} \left(\frac{1}{\sqrt{d}} \mathbf{f}_{\tilde{\mathcal{T}},j}^{\top} \mathbf{W}_{\tilde{\mathcal{T}}}^k \phi(\mathbf{F}_{\mathcal{R}}) \right). \quad (6)$$

Here, $\mathbf{y}_{\mathcal{R},k}$ denotes the k -th virtual point of protein $\mathcal{R}$, $\mathbf{f}_{\mathcal{R},i}$ and $\mathbf{z}_{\mathcal{R},i}$ denote the feature and coordinate of residue i , $\mathbf{W}_{\mathcal{R}}^k \in \mathbb{R}^{d \times d}$ is a learnable projection matrix for head k , and $\phi(\cdot)$ is an average pooling operator. The same formulation is applied symmetrically to the protein $\tilde{\mathcal{T}}$. Our global virtual interface summarizes a soft surface region, yielding a unified interface representation.

Ellipsoid Parametrized Interface Alignment. The underlying interaction region is expected to satisfy finer-grained geometric priors that capture its precise position and directional spread in 3D space. We introduce an ellipsoid-parametrized representation of the ground-truth interface and encourage the global virtual interface $\mathbf{Y}$ to reflect the geometric structure. Specifically, given the interface residue set $\mathcal{I}$, we summarize its spatial distribution using a Gaussian ellipsoid parameterized by a centroid $\boldsymbol{\mu}$ and a covariance matrix $\boldsymbol{\Sigma}$, which capture the location and anisotropic extent of the interaction region, respectively. More details are included in Appendix B.

To parameterize the geometry of the virtual interface, we compute the empirical mean and covariance of the predicted virtual points as follows:

$$\hat{\boldsymbol{\mu}} = \frac{1}{K} \sum_{k=1}^K \mathbf{y}_k, \quad \hat{\boldsymbol{\Sigma}} = \frac{1}{K} \sum_{k=1}^K (\mathbf{y}_k - \hat{\boldsymbol{\mu}})(\mathbf{y}_k - \hat{\boldsymbol{\mu}})^{\top}. \quad (7)$$

We align the virtual interface with the ground-truth ellipsoid by minimizing the discrepancy between their first- and second-order moments:

$$\mathcal{L}_{\text{ellip}} = \|\hat{\boldsymbol{\mu}} - \boldsymbol{\mu}\|_2^2 + \|\hat{\boldsymbol{\Sigma}} - \boldsymbol{\Sigma}\|_F^2. \quad (8)$$

The ellipsoid-parameterized interface provides conditioning information for subsequent generative design. We provide the proof of the equivariance in Appendix F.

3.3 Interface-Conditioned Generation with Ternary Flow Network

Ternary complex formation requires jointly determining the molecular glue structure and the rigid-body docking of the two proteins. Effectively integrating the learned interface prior into the generative process remains non-trivial. We therefore introduce an interface-conditioned generative framework that jointly models molecular conformation and protein-protein docking within a unified multi-modality flow matching formulation. Formally, conditioned on the learned interface, the complex generation distribution is further factorized as three independent components:

$$\begin{aligned} p((\mathcal{R}, \mathcal{M}, \mathcal{T}) \mid \underbrace{\mathcal{I}_{\mathcal{M}}, \mathcal{I}_{\tilde{\mathcal{T}}}, \mathcal{R}, \tilde{\mathcal{T}}}_{\text{condition}:=\star}) &= p(\mathcal{M} \mid \mathcal{I}_{\mathcal{R}}, \mathcal{I}_{\tilde{\mathcal{T}}}, \mathcal{R}, \tilde{\mathcal{T}}) \cdot p(\mathcal{T} \mid \mathcal{I}_{\mathcal{R}}, \mathcal{I}_{\tilde{\mathcal{T}}}, \mathcal{R}, \tilde{\mathcal{T}}) \\ &= p(\mathbf{x} \mid \star) \cdot p(\mathbf{a} \mid \star) \cdot p(\mathbf{R}, \mathbf{t} \mid \star). \end{aligned} \quad (9)$$

Generally, three flows are modeled, each equipped with a specialized architecture: (i) *Rigid Transformation Flow* models $p(\mathbf{R}, \mathbf{t} \mid \star)$ and predicts the rigid-body rotation and translation that place the target protein into its ternary-complex pose. (ii) *Coordinate Flow* models $p(\mathbf{x} \mid \star)$ and generates the three-dimensional coordinates of ligand atoms. (iii) *Atom Type Flow* models $p(\mathbf{a} \mid \star)$ and predicts the chemical identities of ligand atoms. Details of these components are described next.

Rigid Transformation Flow for $p(\mathbf{R}, \mathbf{t} \mid \star)$. Predicting the rigid-body transformation that restores the target protein pose is highly dependent on interface geometry, making direct estimation challenging. We therefore use the global virtual interfaces to obtain an initial docking transformation and further refine it through an iterative module to approximate the final rigid-body configuration.

Conversely, we first utilize our interface model to generate the virtual interface $\mathbf{Y}_{\mathcal{R}}$ and $\mathbf{Y}_{\tilde{\mathcal{T}}}$. Then, we predict the *first-order* rigid transformation with the *Differentiable Kabsch Model* [19]. We form the cross-covariance matrix as $\mathbf{A} = \mathbf{Y}_{\tilde{\mathcal{T}}} \mathbf{Y}_{\mathcal{R}}$, and decompose it with single value decomposition as $\mathbf{A} = \mathbf{U}_2 \mathbf{S} \mathbf{U}_1^\top$. The resulting rotation $\mathbf{R}^*$ and translation $\mathbf{t}^*$ are given by:

$$\mathbf{R}^* = \mathbf{U}_2 \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & d \end{pmatrix} \mathbf{U}_1^\top, \quad \mathbf{t}^* = \hat{\boldsymbol{\mu}}_{\tilde{\mathcal{T}}} - \mathbf{R}^* \hat{\boldsymbol{\mu}}_{\mathcal{R}}, \quad d = \text{sign}(\det(\mathbf{U}_2 \mathbf{U}_1^\top)). \quad (10)$$

The first-order transformation $(\mathbf{R}^*, \mathbf{t}^*)$ and the virtual interface $\mathbf{Y}_{\tilde{\mathcal{T}}}$ are subsequently refined through an iterative Rigid Transformation (RT) block $\Phi_{\text{RT}}(\cdot)$. At each iteration, the i th layer of the module predicts *residual updates* in the Lie algebra space and translation, conditioned on the current transformation, noised input, and the two virtual interfaces. The refinement proceeds as:

$$\Delta\omega, \Delta\mathbf{t} = \Phi_{\text{RT}}^{(i)}(\mathbf{R}_t, \mathbf{t}_t, \mathbf{R}^*, \mathbf{t}^*, \mathbf{Y}_{\mathcal{R}}, \mathbf{Y}_{\tilde{\mathcal{T}}}), \quad (11)$$

$$\mathbf{R}^* = \exp([\Delta\omega \times]) \mathbf{R}^*, \quad (12)$$

$$\mathbf{t}^* = \mathbf{t}^* + \Delta\mathbf{t}, \quad (13)$$

$$\mathbf{Y}_{\tilde{\mathcal{T}}} = \mathbf{R}^* \mathbf{Y}_{\tilde{\mathcal{T}}} + \mathbf{t}^*, \quad (14)$$

where $\Delta\omega$ is a predicted tangent vector in the tangent space $\mathfrak{so}_3$, and $\exp(\cdot)$ denotes the exponential map from $\mathfrak{so}_3$ to $\text{SO}(3)$. The noisy input $(\mathbf{R}_t, \mathbf{t}_t)$ is obtained from two conditional flows: (i) For rotations, given noise $\mathbf{R}_0 \sim U(\text{SO}(3))$ and the target rotation $\mathbf{R}_1 \sim p(\mathbf{R} \mid \star)$, the Riemannian conditional flow [20] is defined as the geodesic interpolation between the $\mathbf{R}_0$ and $\mathbf{R}_1$:

$$\mathbf{R}_t = \psi_t(\mathbf{R}_0 \mid \mathbf{R}_1) = \exp_{\mathbf{R}_0}(t \cdot \log_{\mathbf{R}_0}(\mathbf{R}_1)), \quad (15)$$

where $\exp$ and $\log$ denote the Riemannian exponential and logarithm maps on $\text{SO}(3)$. t is the timestep. (ii) For translations, given noise $\mathbf{t}_0 \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$ and ground-truth $\mathbf{t}_1 \sim p(\mathbf{t} \mid \star)$, the Euclidean conditional flow [21] is defined as a linear interpolation as follows:

$$\mathbf{t}_t = \psi_t(\mathbf{t}_0 \mid \mathbf{t}_1) = t \cdot \mathbf{t}_1 + (1 - t) \cdot \mathbf{t}_0. \quad (16)$$

The conditional flow matching is utilized to train the model, and the objective is formulated as:

$$\mathcal{L}_{\text{RT}} = \mathbb{E}_{t \sim \mathcal{U}(0,1)} [\| (\log_{\mathbf{R}_t}(\hat{\mathbf{R}}_1) - \log_{\mathbf{R}_t}(\mathbf{R}_1)) / (1-t) \|_2^2 + \|\hat{\mathbf{t}}_1 - \mathbf{t}_1\|_2^2]. \quad (17)$$

where $(\hat{\mathbf{R}}_1, \hat{\mathbf{t}}_1)$ is the final output prediction of $\Phi_{\text{RT}}(\cdot)$.

Latent Ternary Encoder. Modeling ternary complexes requires representations that remain invariant to input structures while placing all components in a unified latent space to capture their interactions. To be specific, given the input $\mathcal{G}_{\mathcal{R}}$, $\mathcal{G}_{\tilde{\mathcal{T}}}$, and $\mathcal{G}_{\tilde{\mathcal{M}}}$ (noised graph from $\mathcal{G}_{\mathcal{M}}$), we encode them with the *Invariant Point Attention* block [22]:

$$\mathbf{H}_{\mathcal{R}}, \mathbf{H}_{\tilde{\mathcal{T}}}, \mathbf{h}_{\tilde{\mathcal{M}}} = \text{IPA}(\mathcal{G}_{\mathcal{R}}, \mathcal{G}_{\tilde{\mathcal{T}}}, \mathcal{G}_{\tilde{\mathcal{M}}}), \quad (18)$$

where $\mathbf{H}_{\mathcal{R}}$, $\mathbf{H}_{\tilde{\mathcal{T}}}$, and $\mathbf{h}_{\tilde{\mathcal{M}}}$ denote invariant representations that encode both sequence and structural information, which are subsequently used by the denoising networks.

Coordinate Flow for $p(\mathbf{x} \mid \star)$. Generating ligand atomic coordinates from random noise is particularly challenging when the binding pocket is unknown. We first leverage the predicted rigid transformation to pre-align the target protein, providing a geometry that is consistent with the inferred interface. Then, we integrate the unified latent representations with the current coordinates and design two complementary message passing: one capturing interactions between the proteins and the ligand, and the other modeling intra-ligand geometric dependencies, to progressively update coordinates.

Specifically, the first-order transformation is utilized to correct the coordinate of the target protein:

$$\hat{\mathbf{X}}_{\mathcal{T}} = \underbrace{\mathbf{R}^* \mathbf{X}_{\tilde{\mathcal{T}}} + \mathbf{t}^*}_{\text{Interface-Correction}}. \quad (19)$$

Then, a bi-level message passing network is constructed to denoise the molecular coordinates. Without loss of generality, for each $\mathbf{x}_i \in \mathbf{x}$, we adopt the linear interpolation in Euclidean space to obtain the noised coordinate as: $\mathbf{x}_{i,t} = t\mathbf{x}_{i,1} + (1-t)\mathbf{x}_{i,0}$, where $\mathbf{x}_{i,0} \sim \mathcal{N}(0, I_3)$ is the prior and $\mathbf{x}_{i,1} \sim p(\mathbf{x}_i)$ is the data distribution. Then, we update the coordinate via the intrinsic self-update and the extrinsic protein-molecule update with the Coordinate block $\Phi_{\mathbf{x}}(\cdot)$:

$$\begin{aligned} \hat{\mathbf{x}}_{i,1} = \mathbf{x}_{i,t} &+ \underbrace{\sum_{j \in \mathcal{V}_I} (\mathbf{x}_{i,t} - \mathbf{x}_{j,t}) \phi_{\text{intra}}(\mathbf{h}_i, \mathbf{h}_j, \|\mathbf{x}_{i,t} - \mathbf{x}_{j,t}\|_2^2, t)}_{\text{Intrinsic Update}} \\ &+ \underbrace{\sum_{k \in \mathcal{V}_E} (\mathbf{x}_{i,t} - \mathbf{X}_k) \phi_{\text{extra}}(\mathbf{h}_i, \mathbf{H}_k, \|\mathbf{x}_{i,t} - \mathbf{X}_k\|_2^2, t)}_{\text{Extrinsic Update}}, \end{aligned} \quad (20)$$

where $\phi(\cdot)$ is MLP. $\mathcal{V}_I$ and $\mathcal{V}_E$ denote the sets of neighbors for node i corresponding to intra-molecular connections within the ligand and inter-molecular interactions with the two proteins, respectively. The $\Phi_{\mathbf{x}}$ block outputs the coordinate $\hat{\mathbf{x}}_1$, and the objective is formulated as follows:

$$\mathcal{L}_{\text{coord}} = \mathbb{E}_{t \sim \mathcal{U}(0,1)} [\|\hat{\mathbf{x}}_{i,1} - \mathbf{x}_{i,1}\|_2^2]. \quad (21)$$

Atom Type Flow for $p(\mathbf{a} \mid \star)$. Recovering ligand atom identities is formulated as a continuous flow on the probability simplex. Each atom’s identity $\mathbf{a}_i \in \mathbf{a}$ is mapped to a scaled soft one-hot logit vector [23], which lies on the probability simplex $\Delta^{|\mathcal{A}|-1}$, where $|\mathcal{A}|$ denotes the total number of atom types. The conditional flow is defined as a linear interpolation between the prior $\mathbf{a}_{i,0} \sim \mathcal{N}(\mathbf{0}, K^2 I_{|\mathcal{A}|})$ and the target $\mathbf{a}_{i,1}$: $\psi_t(\mathbf{a}_{i,0} \mid \mathbf{a}_{i,1}) = t\mathbf{a}_{i,1} + (1-t)\mathbf{a}_{i,0}$, and K is a scale constant. The atom type is predicted through the Atom Type block $\Phi_{\mathbf{a}}(\cdot)$ and the training objective is:

$$\hat{\mathbf{a}}_1 = \Phi_{\mathbf{a}}(\mathbf{H}_{\mathcal{R}}, \mathbf{H}_{\tilde{\mathcal{T}}}, \mathbf{H}_{\tilde{\mathcal{M}}}, \mathbf{a}_t, t), \quad \mathcal{L}_{\text{type}} = \mathbb{E}_{t \sim \mathcal{U}(0,1)} [\|\hat{\mathbf{a}}_1 - \mathbf{a}_1\|_2^2]. \quad (22)$$

Training Objective. Finally, we balance the gradients of all flows using weighting coefficients λ_i and jointly train the entire ternary denoising framework as follows:

$$\mathcal{L}_{\text{flow}} = (\lambda_1, \lambda_2) \cdot \mathcal{L}_{\text{RT}} + \lambda_3 \cdot \mathcal{L}_{\text{coord}} + \lambda_4 \cdot \mathcal{L}_{\text{type}}. \quad (23)$$

We provide the details of the model architecture and the flow matching in the Appendix C and D.

Sampling. We provide the detailed sampling algorithm in the Appendix E.

4 Experiment

In this section, we conduct experiments to answer the following research questions:

- **RQ1:** Can TriGlue generate high-quality molecular glues while accurately docking the two proteins into a reasonable ternary complex?
- **RQ2:** How accurate and geometrically meaningful are the ellipsoid interfaces predicted by TriGlue?
- **RQ3:** How does interface conditioning affect the performance of TriGlue?

4.1 Experimental Setup

Dataset. We utilize **TernaryDB** [17] for training and evaluation, which contains 22,303 ternary complexes curated from the Protein Data Bank [24], covering diverse proteins and drug-like ligands. Moreover, to prevent data leakage, complexes are clustered by protein sequence similarity using MMseqs2 [25], and clusters overlapping with known molecular glue complexes are excluded from training to construct non-overlapping validation and test splits. Detailed statistics are in Appendix G.1

Baselines. Since no existing method jointly performs *de novo* molecular glue generation and ternary complex reconstruction, we evaluate representative baselines separately for each sub-task. For molecular generation, we compare with three structure-based design methods, **AR** [26], **TargetDiff** [13], and **PocketXMol** [16]. For protein-protein docking, we adopt **DeepTernary** [17], **EquiDock** [19], and **DiffDock-PP** [27]. Detailed baseline descriptions are provided in Appendix G.2.

Training Details. For the interface modeling module, the EGNN encoder consists of 4 layers with a hidden dimension of 128. The model is trained using the AdamW optimizer [28] with an initial learning rate of 5×10^{-4} and a cosine annealing schedule for 1200 epochs. For the ternary denoising network, we stack two IPA blocks as the encoder with a hidden dimension of 128. Training is performed using the AdamW optimizer with a linear learning rate decay from 5×10^{-4} to 4×10^{-4} for 800 epochs. The loss weights for the joint training objective are set to $\{0.008, 1.0, 1.0, 10.0\}$. All experiments are conducted on 8 NVIDIA L40 GPUs. Appendix G.3 provides more details.

Evaluation Metrics. We evaluate performance from two perspectives. For molecular generation, we report AutoDock Vina-based affinity metrics (**Vina**, **Vina min**, **Vina dock**) [29], the high-affinity ratio (**Aff**), and molecular property metrics (**QED**, **SA**). For ternary-complex reconstruction, we measure ligand pose accuracy using **RMSD** and protein docking quality using **DockQ** [30] and its success rate ($\text{DockQ} > 0.23$). We provide more details in Appendix G.4.

Table 1: Performance comparison on generated molecules.

Method	RMSD ($\downarrow$)		Vina score ($\downarrow$)		Vina min ($\downarrow$)		Vina dock ($\downarrow$)		QED ($\uparrow$)		SA ($\uparrow$)		Aff ($\uparrow$)
	Avg		Avg	Med	Avg	Med	Avg	Med	Avg	Med	Avg	Med	Avg
AR	8.16	2.17	8.76	9.04	3.62	3.69	-4.39	-4.87	0.51	0.54	0.54	0.53	0.53
TargetDiff	6.27	1.99	-2.11	-6.09	-4.83	-6.74	-7.55	-7.86	0.47	0.53	0.55	0.58	0.65
PocketXMol	6.74	0.91	-6.17	-6.14	-6.88	-6.88	-7.69	-7.65	0.47	0.49	0.79	0.78	0.69
TriGlue	5.22	1.41	-6.23	-6.79	-6.77	-7.37	-8.28	-8.85	0.67	0.77	0.58	0.66	0.79

4.2 Ternary Complex Generation

To answer **RQ1**, the evaluation is conducted from two complementary perspectives:

- **The quality and placement of the generated molecules.** Here, all baselines are provided with pocket information and use their best sampling strategies. As shown in Tab. 1, even without access to binding pockets, TriGlue achieves the most accurate ligand placement in the reconstructed complexes, yielding the lowest average RMSD (5.22). Meanwhile, TriGlue generates ligands with strong predicted binding affinity: all affinity-related metrics outperform the baselines except for Vina min, which is slightly lower than PocketXMol. In terms of molecular properties, TriGlue produces more drug-like molecules (QED 0.67), while the SA score is worse than PocketXMol. We attribute this to PocketXMol being trained on a substantially larger dataset ($\sim 10^8$ molecules), which provides broader coverage of chemical space.

- **The docking accuracy of the unbound target protein.** DeepTernary predicts rigid transformations from complex sequences, whereas DiffDock and EquiDock take the two protein structures as input. As shown in Fig. 3, compared with DeepTernary, TriGlue achieves better overall performance with a higher acceptance rate (35.2% vs. 30.8%). Although DeepTernary attains a higher best-case performance, its results exhibit a heavier long-tail distribution.

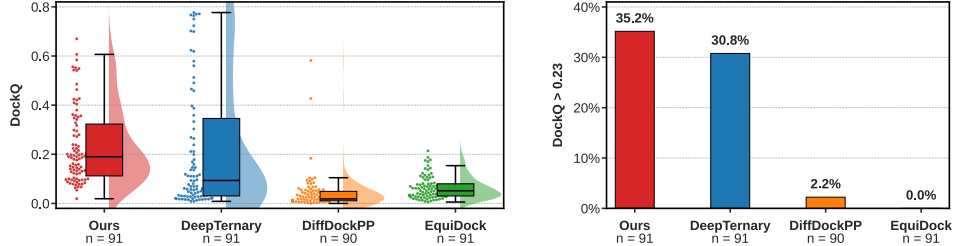


Figure 3: DockQ results.

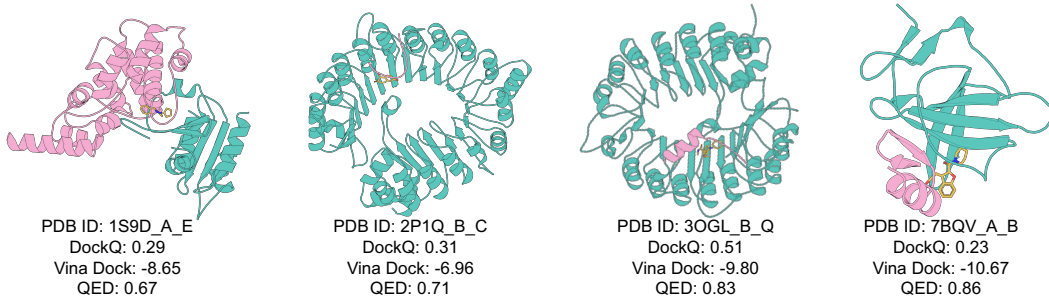


Figure 4: Serval ternary complexes generated by TriGlue.

4.3 Ellipsoid-Parameterized Interface Prediction

To further answer **RQ2**, we evaluate estimated interface ellipsoids on the held-out TernaryDB test split and compare against **PeSTo** [31] and **ScanNet** [32]. Residue-level interface predictions from these methods are converted into ellipsoids. We report geometry- and distribution-aware metrics, including JSD, Wasserstein-2 distance (W2), Center-L2, and LogDet error (see Appendix G.5). As shown in Tab. 2, our method consistently outperforms both baselines across all metrics, achieving substantially lower divergence and improved geometric alignment with ground-truth interface ellipsoids.

Table 2: Ellipsoid interface estimation on the test set.

Method	JSD ↓	W2 Dist ↓	Center-L2 ↓	LogDet Error ↓
Ours	0.47 ± 0.15	12.65 ± 7.89	10.99 ± 7.47	3.83 ± 5.01
PeSTo	0.68 ± 0.04	58.26 ± 46.03	57.39 ± 46.25	8.00 ± 11.54
ScanNet	0.68 ± 0.04	58.96 ± 45.34	57.57 ± 45.85	10.20 ± 12.92

Interface Model Captures the Potential Protein-Protein Interface Pattern. To further answer **RQ2**, we analyze the learned interface representations. We first cluster the ternary complexes in the test set by structural and functional similarity, yielding several groups represented by one or two complexes (e.g., 2098_A_P_FSC). Six representative groups are selected, and the two proteins in each complex are encoded by the interface model. As shown in Fig. 5, the resulting features are visualized using t-SNE [33]. Interestingly, the embeddings form compact and well-separated clusters that closely align with the structural grouping of the complexes, indicating that the model captures transferable, high-level interface geometry and interaction patterns rather than fitting samples.

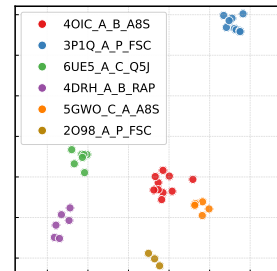


Figure 5: T-SNE of the interface model’s hidden states.

4.4 Ablation Study

To answer **RQ3**, we conduct an in-depth ablation study to evaluate the effect of interface injection in TriGlue. We consider two ablated variants: (i) **TriGlue w/o elli**, which removes the ellipsoid interface prior (Eq. (11)); and (ii) **TriGlue w/o corr**, which disables the interface correction (Eq. (19)). We report the RMSD win rate of TriGlue against the two ablated variants, and the DockQ scores of all three models. As shown in Fig. 6, we observe that:

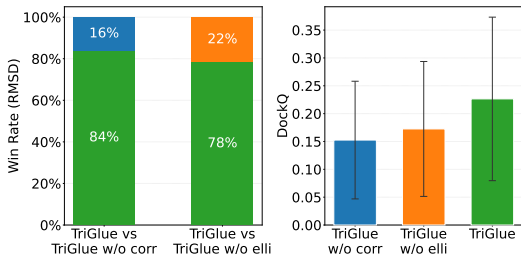


Figure 6: Ablation study.

- Both **TriGlue w/o elli** and **TriGlue w/o corr** exhibit substantially weaker docking performance for the unbound target protein, reflected by lower RMSD win rates and reduced DockQ scores. This observation is consistent with our hypothesis that accurate modeling of the protein-protein interface is critical for ternary complex generation.
- Notably, removing the interface correction module results in a more pronounced drop in performance. The average DockQ decreases from 0.22 to 0.15, and ligand placement accuracy is also significantly affected. We attribute this behavior to the increased structural noise introduced by the unconstrained rigid-body pose of the unbound protein. Without interface correction, the stochasticity in protein positioning hampers the model’s ability to reliably interpret and exploit interface cues during generation.

5 Related Work

Flow Matching has emerged as an efficient paradigm for generative modeling by learning a continuous-time vector field that transports samples from a simple prior to the data distribution [34], which demonstrates strong performance in Euclidean domains, particularly in applications such as image [35], video generation [36], and robotics [37]. Euclidean flow matching has been further generalized to non-Euclidean space, e.g., Riemannian manifolds [20], enabling principled modeling of structured objects like protein backbone [38], and discrete DNA sequence [39].

Deep Learning for Protein Design has advanced rapidly following the emergence of AlphaFold [22], which revolutionized computational biology through highly accurate protein structure prediction. This breakthrough has catalyzed a broad spectrum of generative approaches, including RFDiffusion for enzyme design [40, 41], SE(3)-Diffusion for protein backbone generation [42], Boltzgen for binder design [43], and peptide design [44, 23, 45]. Subsequent efforts have extended these advances to protein complexes, exemplified by AlphaMultimer [46] and AlphaFold3 [47], enabling accurate modeling of multi-chain assemblies. However, these approaches primarily focus on pairwise interactions and do not explicitly address ternary complex design, such as molecular glue degrader or PROTAC-mediated systems. DeepTernary [17] predicts ternary structures from given sequences, whereas our work targets the *de novo* design of molecular glue-induced ternary complexes.

6 Conclusion

In this work, we investigate the problem of molecular glue design from a generative modeling perspective. Unlike conventional structure-based drug design, molecular glue discovery requires the coordinated modeling of ligand generation, protein-protein docking, and ternary complex assembly under unknown interface conditions. To address this challenge, we propose TriGlue, a biology-inspired generative framework that decomposes ternary complex generation into interface estimation and interface-conditioned complex generation. Specifically, an SE(3)-equivariant interface module is introduced to infer geometrically meaningful protein-protein interfaces from unbound structures, while an interface-conditioned ternary flow network jointly generates the molecular glue and predicts the rigid-body transformation required for ternary complex assembly. Extensive experiments demonstrate that TriGlue achieves strong performance in both molecular generation and ternary complex reconstruction, producing chemically valid molecules and structurally plausible complexes. Overall, our results highlight the promise of biology-inspired generative modeling for molecular glue discovery and provide a new direction for AI-driven targeted protein degradation.

References

- [1] Janet M Sasso, Rumiana Tenchov, DaSheng Wang, Linda S Johnson, Xinmei Wang, and Qiongqiong Angela Zhou. Molecular glues: the adhesive connecting targeted protein degradation to the clinic. *Biochemistry*, 2022.
- [2] Pamela Y Ting, Sneha Borikar, John Ryan Kerrigan, Noel M Thomsen, Eamon Aghania, Amelia E Hinman, Alejandro Reyes, Nicolas Pizzato, Barna D Fodor, Fabian Wu, et al. A molecular glue degrader of the wiz transcription factor for fetal hemoglobin induction. *Science*, 2024.
- [3] Oliver Hsia, Matthias Hinterndorfer, Angus D Cowan, Kentaro Iso, Tasuku Ishida, Ramasubramanian Sundaramoorthy, Mark A Nakasone, Hana Imrichova, Caroline Schätz, Andrea Rukavina, et al. Targeted protein degradation via intramolecular bivalent glues. *Nature*, 627(8002):204–211, 2024.
- [4] Georg Petzold, Pablo Gainza, Stefano Annunziato, Ilaria Lamberto, Peter Trenh, Laura A McAllister, Bradley DeMarco, Laura Schwander, Richard D Bunker, Mary Zlotosch, et al. Mining the crbn target space redefines rules for molecular glue-induced neosubstrate recognition. *Science*, 389(6755):eadt6736, 2025.
- [5] Yanfen Liu, Jieyun Bai, Dong Li, and Yong Cang. Routes to molecular glue degrader discovery. *Trends in Biochemical Sciences*, 2025.
- [6] Hojong Yoon, Justine C Rutter, Yen-Der Li, Benjamin L Ebert, et al. Induced protein degradation for therapeutics: past, present, and future. *The Journal of Clinical Investigation*, 2024.
- [7] Jan Krönke, Namrata D Udeshi, Anupama Narla, Peter Grauman, Slater N Hurst, Marie McConkey, Tanya Svinkina, Dirk Heckl, Eamon Comer, Xiaoyu Li, et al. Lenalidomide causes selective degradation of ikzf1 and ikzf3 in multiple myeloma cells. *Science*, 2014.
- [8] Gang Lu, Richard E Middleton, Huahang Sun, MarkVic Naniong, Christopher J Ott, Constantine S Mitsiades, Kwok-Kin Wong, James E Bradner, and William G Kaelin Jr. The myeloma drug lenalidomide promotes the cereblon-dependent destruction of ikaros proteins. *Science*, 2014.
- [9] Junichi Yamamoto, Tetsufumi Suwa, Yuki Murase, Shumpei Tateno, Hirotaka Mizutome, Tomoko Asatsuma-Okumura, Nobuyuki Shimizu, Tsutomu Kishi, Shuji Momose, Masahiro Kizaki, et al. Arid2 is a pomalidomide-dependent crl4crbn substrate in multiple myeloma cells. *Nature Chemical Biology*, 2020.
- [10] Emil Bulatov and Alessio Ciulli. Targeting cullin–ring e3 ubiquitin ligases for drug discovery: structure, assembly and small-molecule modulation. *Biochemical Journal*, 2015.
- [11] Alice Shi Ming Li, Serah Kimani, Brian Wilson, Mahmoud Noureldin, Héctor González-Álvarez, Ahmed Mamai, Laurent Hoffer, John P Guilingier, Ying Zhang, Moritz von Rechenberg, et al. Discovery of nanomolar dcaf1 small molecule ligands. *Journal of Medicinal Chemistry*, 2023.
- [12] Hai-Tsang Huang, Ryan J Lumpkin, Ryan W Tsai, Shuyao Su, Xu Zhao, Yuan Xiong, James Chen, Nada Mageed, Katherine A Donovan, Eric S Fischer, et al. Ubiquitin-specific proximity labeling for the identification of e3 ligase substrates. *Nature Chemical Biology*, 20(9):1227–1236, 2024.
- [13] Jiaqi Guan, Wesley Wei Qian, Xingang Peng, Yufeng Su, Jian Peng, and Jianzhu Ma. 3d equivariant diffusion for target-aware molecule generation and affinity prediction. In *International Conference on Learning Representations (ICLR)*, 2023.
- [14] Jiaqi Guan, Xiangxin Zhou, Yuwei Yang, Yu Bao, Jian Peng, Jianzhu Ma, Qiang Liu, Liang Wang, and Quanquan Gu. Decompdiff: Diffusion models with decomposed priors for structure-based drug design. In *International Conference on Machine Learning (ICML)*. PMLR, 2023.
- [15] Sirine Ayadi, Leon Hetzel, Johanna Sommer, Fabian J Theis, and Stephan Günnemann. Unified guidance for geometry-conditioned molecular generation. In *Neural Information Processing Systems (NeurIPS)*, 2024.

- [16] Xingang Peng, Ruihan Guo, Fenglin Guo, Ziyi Wang, Jiayu Sun, Jiaqi Guan, Yinjun Jia, Yan Xu, Yanwen Huang, Muhan Zhang, et al. Unified modeling of 3d molecular generation via atomic interactions with pocketxmol. *Cell*, 2026.
- [17] Fanglei Xue, Meihan Zhang, Shuqi Li, Xinyu Gao, James A Wohlschlegel, Wenbing Huang, Yi Yang, and Weixian Deng. Se (3)-equivariant ternary complex prediction towards target protein degradation. *Nature Communications*, 16(1):5514, 2025.
- [18] Victor Garcia Satorras, Emiel Hoogetboom, and Max Welling. E (n) equivariant graph neural networks. In *International Conference on Machine Learning (ICML)*, 2021.
- [19] Octavian-Eugen Ganea, Xinyuan Huang, Charlotte Bunne, Yatao Bian, Regina Barzilay, Tommi S Jaakkola, and Andreas Krause. Independent se (3)-equivariant models for end-to-end rigid protein docking. In *International Conference on Learning Representations (ICLR)*, 2022.
- [20] Ricky T. Q. Chen and Yaron Lipman. Flow matching on general geometries. In *International Conference on Learning Representations (ICLR)*, 2024.
- [21] Xingchao Liu, Chengyue Gong, et al. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations (ICLR)*, 2022.
- [22] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.
- [23] Jiahan Li, Chaoran Cheng, Zuofan Wu, Ruihan Guo, Shitong Luo, Zhizhou Ren, Jian Peng, and Jianzhu Ma. Full-atom peptide design based on multi-modal flow matching. In *International Conference on Machine Learning (ICML)*, 2024.
- [24] Helen M Berman, John Westbrook, Zukang Feng, Gary Gilliland, Talapady N Bhat, Helge Weissig, Ilya N Shindyalov, and Philip E Bourne. The protein data bank. *Nucleic Acids Research*, 28(1):235–242, 2000.
- [25] Martin Steinegger and Johannes Söding. Mmseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature Biotechnology*, 35(11):1026–1028, 2017.
- [26] Shitong Luo, Jiaqi Guan, Jianzhu Ma, and Jian Peng. A 3d generative model for structure-based drug design. In *Neural Information Processing Systems (NeurIPS)*, 2021.
- [27] Mohamed Amine Ketata, Cedrik Laue, Ruslan Mammadov, Hannes Stark, Menghua Wu, Gabriele Corso, Céline Marquet, Regina Barzilay, and Tommi S Jaakkola. Diffdock-pp: Rigid protein-protein docking with diffusion models. In *ICLR 2023-Machine Learning for Drug Discovery Workshop*, 2023.
- [28] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.
- [29] O Trott and AJ 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.
- [30] Sankar Basu and Björn Wallner. Dockq: a quality measure for protein-protein docking models. *PloS one*, 2016.
- [31] Lucien F Krapp, Luciano A Abriata, Fabio Cortés Rodríguez, and Matteo Dal Peraro. Pesto: parameter-free geometric deep learning for accurate prediction of protein binding interfaces. *Nature Communications*, 14(1):2175, 2023.
- [32] Jérôme Tubiana, Dina Schneidman-Duhovny, and Haim J Wolfson. Scannet: an interpretable geometric deep learning model for structure-based protein binding site prediction. *Nature Methods*, 19(6):730–739, 2022.

- [33] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research (JMLR)*, 9(86):2579–2605, 2008.
- [34] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *International Conference on Learning Representations (ICLR)*, 2023.
- [35] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *International Conference on Machine Learning (ICML)*, 2024.
- [36] Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, et al. Movie gen: A cast of media foundation models. *arXiv preprint arXiv:2410.13720*, 2024.
- [37] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. π_0 : A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.
- [38] Joey Bose, Tara Akhound-Sadegh, Guillaume Huguet, Kilian FATRAS, Jarrid Rector-Brooks, Cheng-Hao Liu, Andrei Cristian Nica, Maksym Korablyov, Michael M. Bronstein, and Alexander Tong. SE(3)-stochastic flow matching for protein backbone generation. In *International Conference on Learning Representations (ICLR)*, 2024.
- [39] Hannes Stark, Bowen Jing, Chenyu Wang, Gabriele Corso, Bonnie Berger, Regina Barzilay, and Tommi Jaakkola. Dirichlet flow matching with applications to dna sequence design. In *International Conference on Machine Learning (ICML)*, pages 46495–46513, 2024.
- [40] Joseph L Watson, David Juergens, Nathaniel R Bennett, Brian L Trippe, Jason Yim, Helen E Eisenach, Woody Ahern, Andrew J Borst, Robert J Ragotte, Lukas F Milles, et al. De novo design of protein structure and function with rfdiffusion. *Nature*, 620(7976):1089–1100, 2023.
- [41] Woody Ahern, Jason Yim, Doug Tischer, Saman Salike, Seth M Woodbury, Donghyo Kim, Indrek Kalvet, Yakov Kipnis, Brian Coventry, Han Raut Altae-Tran, et al. Atom-level enzyme active site scaffolding using rfdiffusion2. *Nature Methods*, pages 1–10, 2025.
- [42] Jason Yim, Brian L Trippe, Valentin De Bortoli, Emile Mathieu, Arnaud Doucet, Regina Barzilay, and Tommi Jaakkola. Se (3) diffusion model with application to protein backbone generation. In *International Conference on Machine Learning (ICML)*, 2023.
- [43] Hannes Stark, Felix Faltings, MinGyu Choi, Yuxin Xie, Eunsu Hur, Timothy John O’Donnell, Anton Bushuiev, Talip Uçar, Saro Passaro, Weian Mao, Mateo Reveiz, Roman Bushuiev, Tomáš Pluskal, Josef Sivic, Karsten Kreis, Arash Vahdat, Shamayeeta Ray, Jonathan T. Goldstein, Andrew Savinov, Jacob A. Hambalek, Anshika Gupta, Diego A. Taquiri-Diaz, Yaotian Zhang, A. Katherine Hatstat, Angelika Arada, Nam Hyeong Kim, Ethel Tackie-Yarboi, Dylan Boselli, Lee Schnaider, Chang C. Liu, Gene-Wei Li, Denes Hnysz, David M. Sabatini, William F. DeGrado, Jeremy Wohlgend, Gabriele Corso, Regina Barzilay, and Tommi Jaakkola. Boltzgen: Toward universal binder design. *bioRxiv*, 2025.
- [44] Haitao Lin, Odin Zhang, Huifeng Zhao, Dejun Jiang, Lirong Wu, Zicheng Liu, Yufei Huang, and Stan Z Li. Ppflow: Target-aware peptide design with torsional flow matching. In *International Conference on Machine Learning (ICML)*, 2024.
- [45] Xiangzhe Kong, Rui Jiao, Haowei Lin, Ruihan Guo, Wenbing Huang, Wei-Ying Ma, Zihua Wang, Yang Liu, and Jianzhu Ma. Peptide design through binding interface mimicry with pepmimic. *Nature Biomedical Engineering*, pages 1–16, 2025.
- [46] Richard Evans, Michael O’Neill, Alexander Pritzel, Natasha Antropova, Andrew Senior, Tim Green, Augustin Židek, Russ Bates, Sam Blackwell, Jason Yim, et al. Protein complex prediction with alphafold-multimer. *bioRxiv*, 2021.

- [47] Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J Ballard, Joshua Bambrick, et al. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, 630(8016):493–500, 2024.
- [48] Shanshan Gu, Danrui Cui, Xiaoyu Chen, Xiufang Xiong, and Yongchao Zhao. Protacs: an emerging targeting technique for protein degradation in drug discovery. *BioEssays*, 2018.
- [49] Stuart L Schreiber. The rise of molecular glues. *Cell*, 2021.
- [50] Steven M Banik, Kayvon Pedram, Simon Wisnovsky, Green Ahn, Nicholas M Riley, and Carolyn R Bertozzi. Lysosome-targeting chimaeras for degradation of extracellular proteins. *Nature*, 2020.
- [51] Adam D Cotton, Duy P Nguyen, Josef A Gramespacher, Ian B Seiple, and James A Wells. Development of antibody-based protacs for the degradation of the cell-surface immune checkpoint protein pd-11. *Journal of the American Chemical Society*, 2021.
- [52] Matthieu Schapira, Matthew F Calabrese, Alex N Bullock, and Craig M Crews. Targeted protein degradation: expanding the toolbox. *Nature Reviews Drug Discovery*, 2019.
- [53] Kusal TG Samarasinghe and Craig M Crews. Targeted protein degradation: A promise for undruggable proteins. *Cell Chemical Biology*, 2021.
- [54] Lin Zhao, Jia Zhao, Kunhong Zhong, Aiping Tong, and Da Jia. Targeted protein degradation: mechanisms, strategies and application. *Signal Transduction and Targeted Therapy*, 2022.
- [55] Guangcai Zhong, Xiaoyu Chang, Weilin Xie, and Xiangxiang Zhou. Targeted protein degradation: advances in drug discovery and clinical practice. *Signal Transduction and Targeted Therapy*, 2024.
- [56] Jeffrey A Dewey, Clémence Delalande, Saara-Anne Azizi, Vivian Lu, Dionysios Antonopoulos, and Gyorgy Babnigg. Molecular glue discovery: current and future approaches. *Journal of Medicinal Chemistry*, 2023.
- [57] Hossam Nada, Yongseok Choi, Sungdo Kim, Kwon Su Jeong, Nicholas A Meanwell, and Kyeong Lee. New insights into protein–protein interaction modulators in drug discovery and therapeutic advance. *Signal Transduction and Targeted Therapy*, 2024.
- [58] AEA Lopez-Girona, D Mendy, T Ito, K Miller, AK Gandhi, J Kang, S Karasawa, G Carmel, P Jackson, M Abbasian, et al. Cereblon is a direct protein target for immunomodulatory and antiproliferative activities of lenalidomide and pomalidomide. *Leukemia*, 2012.
- [59] Haijuan Yang, Derek G Rudge, Joseph D Koos, Bhamini Vaidialingam, Hyo J Yang, and Nikola P Pavletich. mtor kinase structure, mechanism and regulation. *Nature*, 2013.
- [60] Robin CE Deutscher, Christian Meyners, Maximilian L Repity, Wisely Oki Sugiarto, Jürgen M Kolos, Edvaldo VS Maciel, Tim Heymann, Thomas M Geiger, Stefan Knapp, Frederik Lermyte, et al. Discovery of fully synthetic fkbp12-mtor molecular glues. *Chemical Science*, 2025.
- [61] James P Griffith, Joseph L Kim, Eunice E Kim, Michael D Sintchak, John A Thomson, Matthew J Fitzgibbon, Mark A Fleming, Paul R Caron, Kathy Hsiao, and Manuel A Navia. X-ray structure of calcineurin inhibited by the immunophilin-immunosuppressant fkbp12-fk506 complex. *Cell*, 82(3):507–522, 1995.

Appendix Contents

A Terminology	15
B Ellipsoid Construction	16
C Architecture	17
D Flow Matching	19
E Sample Algorithm	20
F Proofs	21
G Experimental Details	21
G.1 Dataset Details	21
G.2 Baseline Details	22
G.3 Training Details	22
G.4 Evaluation Metric Details	22
G.5 Ellipsoid Interface Prediction Metric Details	24
H Group Analysis	25
I Broader Impacts	25

A Terminology

Table 3 provides supplementary introductions to the biological background and abbreviations of the technical terms that appear in this paper.

Table 3: Terminology list.

Terminology	Abbreviation	Description
Target Protein	POI	The specific protein intended to be modulated, inhibited, stabilized, or degraded by a therapeutic molecule or biological mechanism in order to achieve a desired biological or therapeutic effect.
Targeted Protein Degradation	TPD	A therapeutic strategy that eliminates disease-related proteins by harnessing the cell's natural protein disposal machinery rather than inhibiting protein activity directly.
Proteasome	–	A large multisubunit protease complex that degrades ubiquitin-tagged proteins and maintains protein quality control in cells.
Molecular Glue Degradader	MGD	A small molecule that induces interaction between an E3 ligase and a target protein, promoting ubiquitination and subsequent proteasomal degradation.
Proteolysis Targeting Chimeras	PROTACs	Heterobifunctional molecules that bind a protein of interest via a target warhead and recruit an E3 ligase via a second ligand, connected by a chemical linker that controls ternary complex geometry.
E3 Ligase	–	An enzyme that recognizes substrate proteins and catalyzes ubiquitin transfer from E2 enzymes, determining substrate specificity in the ubiquitin proteasome system.
Ubiquitin	–	A conserved regulatory protein that can be attached to substrates to mark them for degradation or regulate signaling pathways.
Ubiquitination	–	A post-translational modification in which ubiquitin, a small regulatory protein, is covalently attached to lysine residues on a substrate protein through an enzymatic cascade involving E1, E2, and E3 enzymes, commonly marking the protein for proteasomal degradation or altering its cellular function.

Traditional drug discovery has long relied on inhibitors to modulate protein activity directly, yet this approach faces limitations when targeting certain disease-related proteins. Targeted protein degradation (TPD) [48, 49, 50, 51] has emerged as a transformative therapeutic paradigm, which harnesses the cell's intrinsic proteolytic machinery, such as the proteasome and lysosomal pathways, to selectively degrade disease-associated proteins, in contrast to traditional small-molecule inhibitors that merely block protein function [52, 53, 54, 55].

Here, we provide additional information related to molecular glues.

Molecular Glue is a class of small molecules that modulate protein–protein interactions by enhancing the affinity between specific proteins within cells. These compounds can either induce novel protein interactions (type I) or stabilize existing ones (type II), providing an alternative therapeutic approach to traditional drug discovery. Notably, molecular glues offer a means to target proteins once considered

“undruggable” by conventional strategies. They function through diverse mechanisms, e.g., promoting targeted protein degradation or disrupting essential protein functions, and hold potential for treating a range of diseases, including cancer and neurodegenerative disorders.

Molecular glues can be functionally classified based on their mechanism of action in modulating protein–protein interactions: type I glues stabilize non-native or novel PPIs, while type II glues enhance the stability of native, pre-existing interactions.

- **Type I** Molecular glues induce non-native protein–protein interactions that effectively block or “shield” a protein’s endogenous activity. Rather than promoting protein degradation, these compounds typically stabilize inactive conformations or obscure critical functional regions of the target protein. This interference prevents the protein from participating in its normal biological processes, such as by blocking active sites, disrupting ligand binding, or inhibiting native protein–protein interactions [54, 56].
- **Type II** Molecular glues stabilize endogenous protein–protein interactions by modulating protein conformation or dynamics. Depending on the context, they can either inhibit or enhance protein activity by locking interacting partners into specific functional states. A notable example is lenalidomide, an immunomodulatory drug that binds cereblon (CRBN) and reprograms it to target transcription factors such as IKZF1 and IKZF3 for proteasomal degradation in multiple myeloma [54].

Generally, Molecular Glues employ two primary mechanisms to modulate protein-protein interactions: allosteric regulation and direct bridging.

- **Allosteric Regulation.** In the context of allosteric regulation, molecular glues interact with a primary protein, triggering conformational rearrangements that generate or reinforce novel binding interfaces. These newly formed interfaces facilitate the recruitment of a secondary protein [57]. E.g., lenalidomide associates with the E3 ubiquitin ligase cereblon (CRBN), altering its structural configuration to enable the recruitment and subsequent ubiquitination of neo-substrates like IKZF1 and IKZF3, leading to their degradation [58].
- **Direct Bridging.** Unlike allosteric modulation, direct bridging entails the molecular glue simultaneously binding to two proteins at their interface, effectively connecting them. A classic example is rapamycin, which binds both FKBP12 and mTOR to form a stable ternary complex, thereby suppressing mTOR’s kinase function [59]. Although such direct bridging mechanisms do occur, they are relatively rare compared to allosteric modulation. The latter is more prevalent among molecular glues because it leverages the inherent flexibility of protein conformations to induce new interaction surfaces, even in the absence of pre-defined binding sites [60].

B Ellipsoid Construction

Given an interface residue set $\mathcal{I}$ (either $\mathcal{I}_{\mathcal{R}}$ or $\mathcal{I}_{\mathcal{T}}$), we approximate its spatial distribution using a Gaussian ellipsoid parameterized by a centroid $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. This representation provides a compact description of the interface location, spatial extent, and principal orientation in 3D space. Formally, the centroid is computed as the mean of the $C\alpha$ coordinates:

$$\boldsymbol{\mu} = \frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} \mathbf{X}_i, \quad (24)$$

and the covariance matrix is estimated as:

$$\boldsymbol{\Sigma} = \frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} (\mathbf{X}_i - \boldsymbol{\mu})(\mathbf{X}_i - \boldsymbol{\mu})^\top, \quad (25)$$

where $\mathbf{X}_i \in \mathbb{R}^3$ denotes the $C\alpha$ coordinate of residue i .

The resulting Gaussian ellipsoid $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ captures the global geometry of the interface: the mean encodes the interface center, while the eigenstructure of $\boldsymbol{\Sigma}$ reflects its spatial spread and principal axes. This compact parameterization serves as the geometric representation used throughout our framework for interface alignment and evaluation.

C Architecture

Algorithm 1 Invariant Point Attention (IPA)

```

1: Input: single feature  $\{s_i\}$ , pair features  $\{z_{ij}\}$ , residue frame  $\{T_i\}$ 
2:  $q_i^h, k_i^h, v_i^h \leftarrow \text{LinearNoBias}(s_i)$   $\triangleright h = 1, \dots, N_{\text{head}}$ 
3:  $\tilde{q}_i^{hp}, \tilde{k}_i^{hp} \leftarrow \text{LinearNoBias}(s_i)$   $\triangleright p = 1, \dots, N_{\text{query points}}$ 
4:  $\tilde{v}_i^{hp} \leftarrow \text{LinearNoBias}(s_i)$   $\triangleright p = 1, \dots, N_{\text{point values}}$ 
5:  $b_{ij}^h \leftarrow \text{LinearNoBias}(z_{ij})$ 
6:  $w_C \leftarrow \sqrt{\frac{2}{9N_{\text{query}}}}, \quad w_L \leftarrow \sqrt{\frac{1}{c}}$ 
7:  $a_{ij}^h \leftarrow \text{softmax}_j \left( w_L \cdot (q_i^h)^\top k_j^h - \frac{w_C^2}{2} \sum_p \|T_i \tilde{q}_{i,p}^h - T_j \tilde{k}_{j,p}^h\|_2^2 \right)$ 
8:  $\tilde{o}_i^h \leftarrow \sum_j a_{ij}^h z_{ij}$ 
9:  $o_i^h \leftarrow \sum_j a_{ij}^h v_j^h$ 
10:  $\tilde{o}_i^{h,p} \leftarrow T_i^{-1} \left( \sum_j a_{ij}^h T_j \tilde{v}_{j,p}^h \right)$ 
11:  $\tilde{s}_i \leftarrow \text{Linear} \left( \text{concat}_{h,p} (\tilde{o}_i^h, o_i^h, \tilde{o}_i^{h,p}, \|\tilde{o}_i^{h,p}\|) \right)$ 
12: return  $\{\tilde{s}_i\}$ 

```

In this section, we provide the implementation details of our neural network architecture.

Latent Ternary Encoder. We adopt Invariant Point Attention (IPA) [22] to encode the proteins and ligand in the ternary flow network. As shown in Algorithm 1, the IPA block jointly attends over scalar features, pairwise relational features, and 3D coordinates. The key design principle is SE(3) invariance: the model’s predictions remain unchanged under global rotations and translations of the input structure. IPA augments attention with geometric queries and keys represented as points in 3D space. During attention computation, the similarity between residues depends not only on feature similarity but also on the relative spatial configuration between these points. The resulting attention weights therefore encode both biochemical context and geometric compatibility, allowing the network to iteratively refine spatial representations.

Node (single) features. For each protein’s residue, we form the single feature by concatenating: (i) a learned amino-acid embedding; (ii) heavy-atom coordinates expressed in the local frame defined by the N–C α –C; and (iii) backbone dihedral descriptors with chain-aware validity masking. For the ligand, each atom is embedded by a learned atom-type embedding as the single feature. The ligand does not have the residue frame. So we set the rotation matrix as the identity matrix and the translation as the atom’s coordinate to construct the local coordinate system.

Pair (edge) features. For protein residue pairs (i, j) , the pair feature encodes: (i) a learned embedding of the ordered amino-acid pair; (ii) sequence separation along the same chain (clipped integer offset, zeroed across chains); (iii) all-pairs heavy-atom distances summarized by Gaussian radial-basis activations whose scale is conditioned on the amino-acid pair; and (iv) pairwise backbone dihedral channels. For ligand atom pairs (i, j) , we use Gaussian-bump encodings of the interatomic distance together with an MLP on the normalized displacement $\mathbf{r}_{ij}/\|\mathbf{r}_{ij}\|$.

All of the single (pair) features are mapped into a unified dimension. Given the input $\mathcal{G}_{\mathcal{R}}, \mathcal{G}_{\tilde{\mathcal{T}}}$, and $\mathcal{G}_{\tilde{\mathcal{M}}}$ (noised graph from $\mathcal{G}_{\mathcal{M}}$), we encode them with the *Invariant Point Attention* block [22]:

$$\mathbf{H}_{\mathcal{R}}, \mathbf{H}_{\tilde{\mathcal{T}}}, \mathbf{h}_{\tilde{\mathcal{M}}} = \text{IPA}(\mathcal{G}_{\mathcal{R}}, \mathcal{G}_{\tilde{\mathcal{T}}}, \mathcal{G}_{\tilde{\mathcal{M}}}), \quad (26)$$

where $\mathbf{H}_{\mathcal{R}}, \mathbf{H}_{\tilde{\mathcal{T}}}$, and $\mathbf{h}_{\tilde{\mathcal{M}}}$ denote invariant single representations that encode both sequence and structural information, which are subsequently used by the denoising networks.

Rigid Transformation Block. The Φ_{RT} is designed as a lightweight global network that predicts an incremental rigid transformation from a compact interface-level representation. At each layer i , Φ_{RT_i} uses a fixed-length global descriptor that summarizes the current alignment state and noised context, and predicts the residual rotation vector and translation vector.

Specifically, given the virtual interface point sets $\mathbf{Y}_{\mathcal{R}}$ and $\mathbf{Y}_{\tilde{\mathcal{T}}}$, we first compute their centroids:

$$\boldsymbol{\mu}_1 = \frac{1}{K} \sum_{k=1}^K \mathbf{y}_{\mathcal{R},k}, \quad \boldsymbol{\mu}_2 = \frac{1}{K} \sum_{k=1}^K \mathbf{y}_{\tilde{\mathcal{T}},k}. \quad (27)$$

We then construct a global rigid-state descriptor by concatenating:

$$H = \text{Concat}(\mathbf{R}^*, \mathbf{t}^*, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \mathbf{R}_t, \mathbf{t}_t, \mathbf{c}_t), \quad (28)$$

where $\mathbf{c}_t$ is a low-dimensional context vector obtained by projecting the timestep embedding. The descriptor h is processed by a shared multi-layer perceptron trunk

$$h = \text{MLP}_{\text{trunk}}(H), \quad (29)$$

which produces a latent global representation capturing the relative pose between the proteins and the timestep. Two linear heads predict rotation and translation updates:

$$\Delta\omega = \text{RotHead}(h), \quad (30)$$

$$\Delta\mathbf{t} = \text{TransHead}(h), \quad (31)$$

$$\mathbf{R}^* = \exp([\Delta\omega \times]) \mathbf{R}^*, \quad (32)$$

$$\mathbf{t}^* = \mathbf{t}^* + \Delta\mathbf{t}, \quad (33)$$

$$\mathbf{Y}_{\tilde{\mathcal{T}}} = \mathbf{R}^* \mathbf{Y}_{\tilde{\mathcal{T}}} + \mathbf{t}^*, \quad (34)$$

Notably, rotation is parameterized using the continuous 6D representation and converted to a valid rotation matrix via Gram–Schmidt orthogonalization. Finally, after T iterations, block Φ_{RT} outputs the prediction $\hat{\mathbf{R}}_1, \hat{\mathbf{t}}_1$.

Coordinate Block. The $\Phi_{\mathbf{X}}$ refines ligand atom positions that jointly model intra-ligand geometry and protein–ligand interactions. Given ligand coordinates $\mathbf{X}_t = \{\mathbf{x}_{i,t}\}_{i=1}^N$ at step t , the block predicts denoised coordinates through iterative message passing. Ligand atom features are first projected into a latent space with an IPA block and enriched with protein context using cross-attention. Specifically, ligand single features attend to the concatenated single features of the two proteins, allowing each atom to capture biochemical context:

$$\mathbf{h}_i = \text{CrossAttn}(\mathbf{h}_i, \mathbf{H}_{\mathcal{R}}, \mathbf{H}_{\tilde{\mathcal{T}}}). \quad (35)$$

The block performs L EGNN-style updates. At each layer, ligand features are updated using messages from three interaction types: ligand–ligand, ligand–receptor protein, ligand–target protein.

For any pair of nodes (i, j) , messages depend only on invariant features:

$$\mathbf{m}_{ij} = \phi(\mathbf{h}_i, \mathbf{h}_j, \|\mathbf{x}_{i,t} - \mathbf{x}_{j,t}\|^2, t),$$

and are aggregated via residual updates to produce refined atom embeddings.

After each feature update, coordinates are refined using a bi-level message passing with the messages:

$$\begin{aligned} \hat{\mathbf{x}}_i = \mathbf{x}_{i,t} &+ \underbrace{\sum_{j \in \mathcal{V}_I} (\mathbf{x}_{i,t} - \mathbf{x}_{j,t}) \phi_{\text{intra}}(\mathbf{h}_i, \mathbf{h}_j, \|\mathbf{x}_{i,t} - \mathbf{x}_{j,t}\|^2, t)}_{\text{Intrinsic ligand update}} \\ &+ \underbrace{\sum_{k \in \mathcal{V}_E} (\mathbf{x}_{i,t} - \mathbf{X}_k) \phi_{\text{extra}}(\mathbf{h}_i, \mathbf{H}_k, \|\mathbf{x}_{i,t} - \mathbf{X}_k\|^2, t)}_{\text{Protein-conditioned update}}, \end{aligned} \quad (36)$$

where $\phi(\cdot)$ is MLP, and only ligand coordinates are updated. Feature and coordinate updates are repeated for multiple layers, yielding a coarse-to-fine denoising process. Finally, the $\Phi_{\mathbf{X}}$ block outputs the predicted coordinate $\hat{\mathbf{x}}_1$.

Atom Type Block. To predict the atom type, we first form a joint protein context from single representations:

$$\mathbf{H}_{\mathcal{P}} = \text{Concat}(\mathbf{H}_{\mathcal{R}}, \mathbf{H}_{\tilde{\mathcal{T}}}). \quad (37)$$

Then, we update the ligand hidden state via cross-attention:

$$\mathbf{h}_{\tilde{\mathcal{M}}}^* = \text{CrossAttn}(\mathbf{h}_{\tilde{\mathcal{M}}}, \mathbf{H}_{\mathcal{P}}). \quad (38)$$

Finally, the atom-type distribution is predicted with timestep embedding $\mathbf{c}_t$:

$$\hat{\mathbf{a}}_1 = \Phi_{\mathbf{a}}(\mathbf{h}_{\tilde{\mathcal{M}}}^*, \mathbf{c}_t). \quad (39)$$

We sample the atom type from the distribution to decode the ligand.

D Flow Matching

Conditional Flow Matching. The conditional flow matching (CFM) framework [34] offers an approach to learn a probability time-dependent flow $\psi_t : [0, 1] \times \mathbb{R}^d \rightarrow \mathbb{R}^d$ that transforms samples from a simple base distribution p_0 into a target data distribution $p_1 \approx q$. The flow and its corresponding vector field $u_t : [0, 1] \times \mathbb{R}^d \rightarrow \mathbb{R}^d$ are governed by the ordinary differential equation (ODE): $\frac{d}{dt}x_t = u_t(x_t)$, where $x_t = \psi_t(x_0)$. However, it is challenging to model this vector field directly due to the complexity of its closed-form solution. Instead, CFM circumvents this limitation by conditioning the flow and vector field on a specific data pair: a sample $x_0 \sim p_0$ and a target $x_1 \sim p_1$. This leads to a tractable formulation of the interpolated trajectory $x_t = \psi_t(x_0 | x_1)$ and the conditional vector field $u_t(x_t | x_1) = \frac{d}{dt}x_t$. Hence, the standard CFM training loss is formulated as:

$$\mathcal{L}_{\text{CFM}}(\theta) = \mathbb{E}_{t \sim \mathcal{U}(0,1)} \left\| v_t(\psi_t(x_0 | x_1)) - \frac{d}{dt}x_t \right\|_2^2, \quad (40)$$

where $\mathcal{U}(0, 1)$ is a uniform distribution and v_t is a neural network parameterized by θ .

Based on the CFM, our Ternary Flow Network involves three types of flow:

Euclidean CFM for Coordinate. The Euclidean conditional flow interpolates between Gaussian noise $\mathbf{x}_{i,0} \sim \mathcal{N}(\mathbf{0}, I_3)$ and the data distribution $\mathbf{x}_{i,1} \sim p(\mathbf{x}_i | \star)$:

$$\psi_t(\mathbf{x}_{i,0} | \mathbf{x}_{i,1}) = t\mathbf{x}_{i,1} + (1-t)\mathbf{x}_{i,0} \quad (41)$$

The target vector field is:

$$u_t = \frac{d}{dt}\psi_t = \mathbf{x}_{i,1} - \mathbf{x}_{i,0}. \quad (42)$$

The CFM loss is:

$$\mathcal{L}_{\text{coord}} = \mathbb{E}_{t \sim \mathcal{U}(0,1)} \|v_\theta(\mathbf{x}_{i,t}, t, \star) - (\mathbf{x}_{i,1} - \mathbf{x}_{i,0})\|_2^2. \quad (43)$$

Simplex CFM for Atom Type. Following PepFlow [23], we map the discrete atom type $\mathbf{a}_i$ to a continuous logit space $\mathbf{s}_i \in \mathbb{R}^{|\mathcal{A}|}$ via soft one-hot encoding with a scale constant $K > 0$:

$$\mathbf{s}_i[j] = \begin{cases} K, & j = \mathbf{a}_i, \\ -K, & \text{otherwise.} \end{cases} \quad (44)$$

This yields a sharp probability distribution $\text{softmax}(\mathbf{s}_i)$ concentrated on the true type, corresponding to a point on the probability simplex $\Delta^{|\mathcal{A}|-1}$. We define the conditional flow in the logit space as a linear interpolation between the prior $\mathbf{s}_{i,0} \sim \mathcal{N}(\mathbf{0}, K^2 I_{|\mathcal{A}|})$ and the target $\mathbf{s}_{i,1}$ derived from $\mathbf{a}_i$:

$$\psi_t(\mathbf{s}_{i,0} | \mathbf{s}_{i,1}) = t\mathbf{s}_{i,1} + (1-t)\mathbf{s}_{i,0}. \quad (45)$$

The associated vector field is:

$$u_t(\mathbf{s}_{i,t} | \mathbf{s}_{i,1}, \mathbf{s}_{i,0}) = \frac{d}{dt}\psi_t = \mathbf{s}_{i,1} - \mathbf{s}_{i,0}. \quad (46)$$

Similar to the Euclidean CFM, a neural network v_θ is trained to predict this vector field, and the CFM loss is:

$$\mathcal{L}_{\text{type}} = \mathbb{E}_{t \sim \mathcal{U}(0,1)} \left[\|v_\theta(\mathbf{s}_{i,t}, t, \star) - (\mathbf{s}_{i,1} - \mathbf{s}_{i,0})\|_2^2 \right]. \quad (47)$$

Riemannian CFM for Rigid Transformation. Following [38], we predict $(R, \mathbf{t})$ within the flow matching framework, where the interpolation of orientations is performed along geodesics on $\text{SO}(3)$, and translations are handled in Euclidean space.

For rotation, we employ geodesic interpolation on $\text{SO}(3)$, which serves as the manifold counterpart of linear interpolation in Euclidean space and guarantees shortest-path evolution. Formally, given a random prior $\mathbf{R}_0 \sim U(\text{SO}(3))$ and a target rotation $\mathbf{R}_1 \sim p(\mathbf{R} | \star)$, the conditional flow is:

$$\psi_t(\mathbf{R}_0 | \mathbf{R}_1) = \exp_{\mathbf{R}_0}(t \cdot \log_{\mathbf{R}_0}(\mathbf{R}_1)), \quad (48)$$

where $\exp$ and $\log$ denote the Riemannian exponential and logarithm maps on $\text{SO}(3)$, computable via Rodrigues' formula. The associated vector field is defined as:

$$u_t(\mathbf{R}_t | \mathbf{R}_0, \mathbf{R}_1) = \frac{d}{dt}\psi_t = \frac{\log_{\mathbf{R}_t}(\mathbf{R}_1)}{1-t}. \quad (49)$$

Then the CFM objective is formulated as:

$$\mathcal{L} = \mathbb{E}_{t \sim \mathcal{U}(0,1)} \left[\left\| v_{\theta}(\mathbf{R}_t, t, \star) - \frac{\log_{\mathbf{R}_t}(\mathbf{R}_1)}{1-t} \right\|_2^2 \right]. \quad (50)$$

For translations, a Euclidean conditional flow is defined between $\mathbf{t}_0 \sim \mathcal{N}(\mathbf{0}, \sigma^2 I)$ and $\mathbf{t}_1 \sim p(\mathbf{t} \mid \star)$, and the corresponding CFM objective $\mathcal{L}$ has the same form with Eq. (43).

Reparameterized CFM Objectives. Following the reparameterization strategy [23], instead of predicting the ground-truth conditional vector field, we directly predict the target data and compute the vector field implicitly from the reconstructed target. For the Euclidean flow, let $\hat{\mathbf{x}}_{i,1} = v_{\theta}(\mathbf{x}_{i,t}, t, \star)$ denote the reconstructed ligand coordinate, we reparameterized the objective in Eq. (43) as:

$$\mathcal{L}_{\text{coord}} = \mathbb{E}_{t \sim \mathcal{U}(0,1)} \|u_t(\mathbf{x}_{i,t} \mid \hat{\mathbf{x}}_{i,1}, \mathbf{x}_{i,0}) - u_t(\mathbf{x}_{i,t} \mid \mathbf{x}_{i,1}, \mathbf{x}_{i,0})\|_2^2 \quad (51)$$

$$= \mathbb{E}_{t \sim \mathcal{U}(0,1)} \|(\hat{\mathbf{x}}_{i,1} - \mathbf{x}_{i,0}) - (\mathbf{x}_{i,1} - \mathbf{x}_{i,0})\|_2^2 \quad (52)$$

$$= \mathbb{E}_{t \sim \mathcal{U}(0,1)} \|\hat{\mathbf{x}}_{i,1} - \mathbf{x}_{i,1}\|_2^2. \quad (53)$$

Similarly, the other CFM objectives are reformulated as follows:

$$\mathcal{L}_{\text{type}} = \mathbb{E}_{t \sim \mathcal{U}(0,1)} \|\hat{\mathbf{s}}_{i,1} - \mathbf{s}_{i,1}\|_2^2. \quad (54)$$

$$\mathcal{L}_{\text{rot}} = \mathbb{E}_{t \sim \mathcal{U}(0,1)} \left\| \left(\log_{\mathbf{R}_t}(\hat{\mathbf{R}}_1) - \log_{\mathbf{R}_t}(\mathbf{R}_1) \right) / (1-t) \right\|_2^2. \quad (55)$$

$$\mathcal{L}_{\text{trans}} = \mathbb{E}_{t \sim \mathcal{U}(0,1)} \|\hat{\mathbf{t}}_1 - \mathbf{t}_1\|_2^2. \quad (56)$$

In this work, we utilize the reparameterized CFM objectives to train our model.

E Sample Algorithm

Algorithm 2 Sampling with Interface-Conditioned Ternary Flow Network

- 1: **Input:** receptor protein $\mathcal{G}_{\mathcal{R}}$, target protein $\mathcal{G}_{\mathcal{T}}$.
 - 2: $(\mathbf{Y}_{\mathcal{R}}, \mathbf{Y}_{\mathcal{T}}) = \text{InterfaceModel}(\mathcal{G}_{\mathcal{R}}, \mathcal{G}_{\mathcal{T}})$ ▷ Estimate global virtual interface
 - 3: $(\mathbf{R}^*, \mathbf{t}^*) = \text{DifferentiableKabschAlign}(\mathbf{Y}_{\mathcal{R}}, \mathbf{Y}_{\mathcal{T}})$ ▷ Compute first-order rigid transformation
 - 4: $\hat{\mathbf{X}}_{\mathcal{T}} = \mathbf{R}^* \mathbf{X}_{\mathcal{T}} + \mathbf{t}^*$ ▷ Interface correction
 - 5: Sample initial noisy state $\mathbf{R}_0, \mathbf{t}_0, \mathbf{x}_0, \mathbf{a}_0$
 - 6: **for** $t = 1$ to N **do**
 - 7: $\hat{\mathbf{R}}_{\frac{t}{N}}, \hat{\mathbf{t}}_{\frac{t}{N}}, \hat{\mathbf{x}}_{\frac{t}{N}}, \hat{\mathbf{a}}_{\frac{t}{N}} = \text{TernaryFlowNet}(\mathbf{R}_{\frac{t-1}{N}}, \mathbf{t}_{\frac{t-1}{N}}, \mathbf{x}_{\frac{t-1}{N}}, \mathbf{a}_{\frac{t-1}{N}}, \mathbf{Y}_{\mathcal{R}}, \mathbf{Y}_{\mathcal{T}})$
 - 8: $\mathbf{R}_{\frac{t}{N}} = \exp_{\mathbf{R}_{\frac{t-1}{N}}} \left(\Delta t \cdot \log_{\mathbf{R}_{\frac{t-1}{N}}}(\hat{\mathbf{R}}_{\frac{t}{N}}) \right)$ ▷ Update rotation
 - 9: $\mathbf{t}_{\frac{t}{N}} = \mathbf{t}_{\frac{t-1}{N}} + (\hat{\mathbf{t}}_{\frac{t}{N}} - \mathbf{t}_0) \Delta t$ ▷ Update translation
 - 10: $\mathbf{x}_{\frac{t}{N}} = \mathbf{x}_{\frac{t-1}{N}} + (\hat{\mathbf{x}}_{\frac{t}{N}} - \mathbf{x}_0) \Delta t$ ▷ Update coordinate
 - 11: $\mathbf{a}_{\frac{t}{N}} = \mathbf{a}_{\frac{t-1}{N}} + (\hat{\mathbf{a}}_{\frac{t}{N}} - \mathbf{a}_0) \Delta t$ ▷ Update atom type
 - 12: **end for**
 - 13: Decode complex $\mathcal{C}$ from $\{\hat{\mathbf{R}}_1, \hat{\mathbf{t}}_1, \hat{\mathbf{x}}_1, \hat{\mathbf{a}}_1\}$
 - 14: **return** $\mathcal{C}$
-

Algorithm 2 summarizes the sampling procedure of TriGlue. Given two unbound proteins, the interface model first predicts the global virtual interfaces and computes an initial rigid-body transformation through differentiable Kabsch alignment. The estimated transformation is then used to correct the target protein coordinates, providing an interface-consistent initialization for subsequent generation.

Starting from noisy rotations, translations, ligand coordinates, and atom types, the ternary flow network iteratively predicts the denoised states conditioned on the estimated interfaces. The rotational component is updated along the $\text{SO}(3)$ geodesic, while translations, ligand coordinates, and atom types are progressively refined using Euler steps. After N iterations, the final rigid transformation and ligand structure are decoded into the ternary complex.

F Proofs

SE(3) Equivariance of Virtual Interface and Ellipsoid Parameters.

Theorem 1. *The virtual interface points $\{\mathbf{y}_k\}_{k=1}^K$ and the ellipsoid parameters $(\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}})$ defined in Sec. 3.2 are SE(3)-equivariant.*

Proof. Consider a rigid-body transformation $(\mathbf{R}, \mathbf{t}) \in \text{SE}(3)$ applied to all residue coordinates:

$$\mathbf{z}'_i = \mathbf{R}\mathbf{z}_i + \mathbf{t}. \quad (57)$$

We first prove that the virtual interface points are equivariant. Since the attention weights are computed from EGNN invariant features, they remain unchanged under SE(3). Thus, we have:

$$\mathbf{y}'_k = \sum_i \alpha_i^k \mathbf{z}'_i = \sum_i \alpha_i^k (\mathbf{R}\mathbf{z}_i + \mathbf{t}) \quad (58)$$

$$= \mathbf{R} \sum_i \alpha_i^k \mathbf{z}_i + \mathbf{t} = \mathbf{R}\mathbf{y}_k + \mathbf{t}. \quad (59)$$

Hence $\mathbf{y}_k$ are SE(3)-equivariant. Then, it is not hard to obtain that:

$$\hat{\boldsymbol{\mu}}' = \frac{1}{K} \sum_k \mathbf{y}'_k = \frac{1}{K} \sum_k (\mathbf{R}\mathbf{y}_k + \mathbf{t}) = \mathbf{R}\hat{\boldsymbol{\mu}} + \mathbf{t}, \quad (60)$$

and the centered coordinates satisfy:

$$\mathbf{y}'_k - \hat{\boldsymbol{\mu}}' = \mathbf{R}(\mathbf{y}_k - \hat{\boldsymbol{\mu}}), \quad (61)$$

which yields:

$$\hat{\boldsymbol{\Sigma}}' = \frac{1}{K} \sum_k (\mathbf{y}'_k - \hat{\boldsymbol{\mu}}')(\mathbf{y}'_k - \hat{\boldsymbol{\mu}}')^\top \quad (62)$$

$$= \mathbf{R}\hat{\boldsymbol{\Sigma}}\mathbf{R}^\top. \quad (63)$$

Therefore, $(\mathbf{y}_k, \hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}})$ transform equivariantly under SE(3). $\square$

G Experimental Details

G.1 Dataset Details

We use the molecular-glue-induced ternary-complex split derived from TernaryDB¹ [17]. Each sample contains two protein components, including a target protein and an E3 ligase, and one small-molecule ligand extracted from an experimentally resolved ternary complex. Protein lengths are calculated based on the number of resolved C α residues in the curated protein PDB files. Ligand descriptors are computed from the per-complex SDF files using RDKit. QED denotes the quantitative estimate of drug-likeness. Detailed statistics are provided in Tab. 4. For numerical descriptors, we report the median, with the minimum and maximum values.

Table 4: Summary statistics of the training and held-out molecular-glue test splits. Numerical descriptors are reported as **Median (Minimum, Maximum)**.

Component	Statistic	Training set	Test set
<i>Complex</i>	Number of ternary complexes	21,481	94
	Sequence-pair overlap with training set	–	0/94
<i>Protein</i>	Length of E3 ligase	321 (3, 1765)	227 (107, 826)
	Length of target protein	125 (3, 1759)	76 (2, 512)
<i>Molecular Glue</i>	Number of heavy atoms	32 (2, 100)	25 (10, 69)
	Molecular weight (Da)	506.20 (26.04, 1464.07)	385.85 (136.15, 980.31)
	Number of rings	1 (0, 24)	3 (0, 13)
	Number of aromatic rings	0 (0, 7)	0 (0, 5)
	QED	0.24 (0.02, 0.93)	0.54 (0.12, 0.88)

¹<https://doi.org/10.5281/zenodo.15514874>

G.2 Baseline Details

Baselines. Our framework jointly performs *de novo* molecular glue generation and reconstructs the unbound target protein into the ternary-complex coordinate frame. However, no existing method evaluates both aspects simultaneously. Therefore, we select and compare representative baselines separately for each task. Overall, these baselines span a broad spectrum of representative methods:

- (i) *Molecular design method.* To assess the quality of generated molecular glues, three structure-based drug design methods are considered: **AR**² [26], **TargetDiff**³ [13], and **PocketXMol**⁴ [16], which perform *de novo* small-molecule generation conditioned on protein binding pockets.
- (ii) *Ternary complex structure prediction method.* **DeepTernary**⁵ [17] is the first model designed for ternary complex structure prediction. We adopt it to dock the unbound protein back into the ternary complex, thereby evaluating the ability to model the complex structure.
- (iii) *Protein-protein docking method.* To further evaluate the rigid-body docking capability, we include two representative protein-protein docking approaches: **EquiDock**⁶ [19] and **DiffDock-PP**⁷ [27], which predict the relative pose of interacting proteins from their unbound structures.

All of the baselines are implemented with the official code.

G.3 Training Details

The proposed framework consists of two stages: interface estimation and interface-conditioned ternary flow generation. All experiments were conducted using the Adam optimizer with an initial learning rate of 5×10^{-4} .

Interface Estimation Module. The interface estimation model was trained for 1200 epochs with a batch size of 8 using distributed training on 8 GPUs. Node and edge embedding dimensions were set to 128 and 64, respectively. The SE(3)-equivariant encoder adopted a two-block IPA architecture with 8 attention heads, 8 query/key points, and 12 value points. A lightweight sequence transformer with 2 layers and 4 attention heads was further employed for sequence feature integration. The interface conditioner used a feature dimension of 128 and graph construction with 16 nearest neighbors.

Interface-Conditioned Ternary Flow Network. The ternary flow model was trained for 800 epochs with a batch size of 2 and gradient accumulation of 32, resulting in an effective batch size of approximately 64 samples per optimization step. The pretrained interface estimation module was frozen during this stage. The coordinate, atom-type, rotation, and translation objectives were jointly optimized with loss weights of 10.0, 1.0, 1.0, and 0.008, respectively. Gradient clipping with a maximum norm of 50 was applied to stabilize training.

For flow matching, translations were modeled with Euclidean conditional flow matching using Gaussian noise with $\sigma = 1.0$, while rotations were modeled on SO(3) using Riemannian flow matching. Atom types were represented on the probability simplex with simplex scale constant $K = 5.0$ and 25 atom categories, where atom identities were defined based on elemental type together with hybridization and aromaticity information. During sampling, the model performed 50 iterative denoising steps.

Dataset Split. The ternary complex dataset was divided into training and validation subsets using a 95%/5% split. Unless otherwise specified, all experiments used a random seed of 42.

Training the interface model takes approximately 2 days, training the ternary flow network requires about 5 days, and sampling a single complex takes around 40 seconds.

G.4 Evaluation Metric Details

We evaluate the models from the following two perspectives:

²<https://github.com/luost26/3D-Generative-SBDD>

³<https://github.com/guanjq/targetdiff>

⁴<https://github.com/pengxingang/PocketXMol>

⁵<https://github.com/youqingxiaozhua/DeepTernary>

⁶https://github.com/octavian-ganea/equidock_public

⁷<https://github.com/ketatom/DiffDock-PP>

(i) **Molecular generation.** We assess the generated ligands with respect to binding and physicochemical properties. Binding affinity is estimated using AutoDock Vina [29], including the **Vina score**, the minimized Vina score (**Vina min**) after local relaxation, and the redocking Vina score (**Vina dock**) that reflects the best achievable binding affinity. The high-affinity ratio (**Aff**) is defined as the proportion of generated molecules whose Vina dock surpasses that of the reference ligand for each target. To further examine chemical quality, we evaluate standard molecular property metrics, including quantitative estimation of drug-likeness (**QED**) and the synthetic accessibility (**SA**) score.

(ii) **Ternary-complex reconstruction.** We evaluate the structural accuracy of the generated complex from both ligand and protein perspectives. Ligand pose accuracy is measured using the root-mean-square deviation (**RMSD**) between predicted and reference ligand coordinates. Protein docking accuracy is assessed using the **DockQ** score [30], and we additionally report the success rate with DockQ greater than 0.23, which is commonly regarded as indicating acceptable docking quality.

Root Mean Square Deviation (RMSD). Ligand pose quality is assessed using the RMSD between the predicted ligand coordinates and the reference (ground truth) coordinates.

Let $\mathbf{x}_i \in \mathbb{R}^3$ denote the predicted coordinate of the i -th ligand atom and $\mathbf{x}_i^* \in \mathbb{R}^3$ denote the corresponding reference coordinate. Given N ligand atoms, RMSD is defined as:

$$\text{RMSD} = \sqrt{\frac{1}{N} \sum_{i=1}^N \|\mathbf{x}_i - \mathbf{x}_i^*\|_2^2}. \quad (64)$$

Lower RMSD values indicate higher agreement between the predicted ligand pose and the reference structure.

AutoDock Vina-based Binding Metrics. To evaluate the predicted protein-ligand binding quality of generated molecular glues, we report three AutoDock Vina-based scores: Vina score, Vina min, and Vina dock. Let P denote the protein binding environment, M denote a generated molecule, and $\hat{\mathbf{x}}$ denote its generated 3D conformation. We write $E_{\text{Vina}}(P, M, \mathbf{x})$ as the AutoDock Vina scoring function, which approximates the protein-ligand binding free energy in kcal/mol. More negative values indicate stronger predicted binding affinity.

The **Vina score** is computed on the generated ligand pose without further pose optimization:

$$\text{VinaScore}(M) = E_{\text{Vina}}(P, M, \hat{\mathbf{x}}). \quad (65)$$

This metric evaluates whether the model directly generates a chemically plausible and energetically favourable binding pose.

Vina score further performs local energy minimization around the generated pose before scoring:

$$\text{VinaMin}(M) = \min_{\mathbf{x} \in \mathcal{N}(\hat{\mathbf{x}})} E_{\text{Vina}}(P, M, \mathbf{x}), \quad (66)$$

where $\mathcal{N}(\hat{\mathbf{x}})$ denotes the local conformational neighbourhood explored during minimization. Compared with the Vina score, Vina min is less sensitive to small local geometric errors in the generated conformation.

Vina dock evaluates the molecule after a full re-docking procedure:

$$\text{VinaDock}(M) = \min_{\mathbf{x} \in \Omega(P, M)} E_{\text{Vina}}(P, M, \mathbf{x}), \quad (67)$$

where $\Omega(P, M)$ denotes the docking search space of possible ligand poses in the binding region. This metric estimates the best binding score achievable by the generated molecule under the docking protocol, rather than only evaluating the initially generated pose. For all three Vina-based metrics, lower values are better.

Quantitative Estimate of Drug-likeness (QED). Drug-likeness is evaluated using the Quantitative Estimate of Drug-likeness (QED), which summarizes how similar a molecule is to known drug-like compounds. QED combines multiple molecular descriptors, including molecular weight, lipophilicity, topological polar surface area, hydrogen bond donors and acceptors, aromatic rings, rotatable bonds, and structural alerts. Given descriptor-specific desirability functions $d_k(M) \in [0, 1]$ and weights w_k , QED is computed as a weighted geometric mean:

$$\text{QED}(M) = \exp \left(\frac{\sum_{k=1}^K w_k \log d_k(M)}{\sum_{k=1}^K w_k} \right). \quad (68)$$

QED ranges from 0 to 1, where larger values indicate more favourable drug-like properties.

Synthetic Accessibility (SA). Synthetic accessibility measures how easy a generated molecule is expected to be synthesized. The commonly used raw synthetic accessibility score is based on fragment contributions and molecular complexity penalties:

$$\text{SAscore}_{\text{raw}}(M) = S_{\text{frag}}(M) + S_{\text{complexity}}(M) + S_{\text{correction}}(M). \quad (69)$$

Here, $S_{\text{frag}}(M)$ reflects whether the molecule contains fragments frequently observed in known synthesized compounds, while $S_{\text{complexity}}(M)$ penalizes structural features that make synthesis difficult, such as large rings, complex ring systems, stereochemical complexity, and large molecular size.

Since the raw SA score is usually defined such that lower values indicate easier synthesis, while our table reports SA with a larger-is-better direction, we use a normalized synthesizability score:

$$\text{SA}(M) = \text{clip} \left(\frac{10 - \text{SAscore}_{\text{raw}}(M)}{9}, 0, 1 \right). \quad (70)$$

Under this convention, larger SA values indicate molecules that are predicted to be easier to synthesize.

High Affinity Rate (Aff). Following common evaluation protocols in target-aware molecule generation, Aff measures the proportion of generated molecules whose predicted binding affinity is stronger than that of the reference ligand for the same target. Let $s_{t,j}^{\text{dock}}$ denote the Vina dock score of the j -th generated molecule for target t , and let s_t^{ref} denote the corresponding score of the reference ligand. Since lower Vina scores indicate stronger predicted binding, Aff is defined as:

$$\text{Aff} = \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} \frac{1}{K_t} \sum_{j=1}^{K_t} \mathbb{I}[s_{t,j}^{\text{dock}} < s_t^{\text{ref}}], \quad (71)$$

where $\mathcal{T}$ is the set of test targets, K_t is the number of generated molecules for target t , and $\mathbb{I}[\cdot]$ is the indicator function. Higher Aff values indicate that a larger fraction of generated molecules are predicted to bind more strongly than the reference ligand.

DockQ. For protein–protein docking, model quality is evaluated using DockQ, a continuous score in the range $[0, 1]$ that summarizes interface accuracy by combining three complementary aspects. (i) The fraction of native contacts (F_{nat}) measures how many residue–residue contacts observed in the reference complex are correctly reproduced in the prediction, reflecting interface contact recovery. (ii) The interface RMSD (iRMSD) quantifies the structural deviation of backbone atoms for residues located at the binding interface after superposition of the complexes, capturing local interface accuracy. (iii) The ligand RMSD (LRMSD) measures the backbone deviation of the ligand protein after aligning the receptor, reflecting the global placement of the binding partner. These three components are smoothly combined into a single score:

$$\text{DockQ} = \frac{1}{3} \left(F_{\text{nat}} + \frac{1}{1 + \left(\frac{\text{iRMSD}}{1.5} \right)^2} + \frac{1}{1 + \left(\frac{\text{LRMSD}}{8.5} \right)^2} \right). \quad (72)$$

G.5 Ellipsoid Interface Prediction Metric Details

Each interface is represented as a Gaussian ellipsoid $\mathcal{N}(\mu, \Sigma)$, where $\mu \in \mathbb{R}^3$ denotes the ellipsoid center and $\Sigma \in \mathbb{R}^{3 \times 3}$ is a symmetric positive definite (SPD) matrix describing its orientation and spatial extent. Given a prediction (μ_p, Σ_p) and reference (μ_r, Σ_r) , we evaluate complementary geometric and distributional discrepancies using the following metrics.

Center-L2. This metric measures the Euclidean distance between predicted and reference ellipsoid centers:

$$d_{\text{center}} = \|\mu_p - \mu_r\|_2. \quad (73)$$

LogDet Error. Differences in ellipsoid scale are quantified by comparing the log-determinants of the covariance matrices:

$$d_{\log \det} = |\log \det(\Sigma_p) - \log \det(\Sigma_r)|. \quad (74)$$

Wasserstein-2 distance (W2 Dist). This geometry-aware distance jointly evaluates discrepancies in both the ellipsoid center and shape:

$$W_2^2 = \|\mu_p - \mu_r\|_2^2 + \text{Tr}\left(\Sigma_p + \Sigma_r - 2\left(\Sigma_r^{1/2}\Sigma_p\Sigma_r^{1/2}\right)^{1/2}\right), \quad (75)$$

and we report $W_2 = \sqrt{W_2^2}$.

Jensen-Shannon divergence (JSD). Distributional similarity between the predicted and reference Gaussian densities is assessed using JSD:

$$\text{JSD}(P\|Q) = \frac{1}{2}\text{KL}(P\|M) + \frac{1}{2}\text{KL}(Q\|M), \quad M = \frac{1}{2}(P + Q), \quad (76)$$

where $P = \mathcal{N}(\mu_r, \Sigma_r)$ and $Q = \mathcal{N}(\mu_p, \Sigma_p)$. Because M is a Gaussian mixture rather than a single Gaussian, JSD is estimated via Monte Carlo:

$$\widehat{\text{JSD}} = \frac{1}{2} \left[\frac{1}{N} \sum_{x_i \sim P} \log \frac{p(x_i)}{m(x_i)} + \frac{1}{N} \sum_{y_j \sim Q} \log \frac{q(y_j)}{m(y_j)} \right], \quad (77)$$

where $m(\cdot) = \frac{1}{2}(p(\cdot) + q(\cdot))$.

H Group Analysis

We evaluate the structural quality of the generated ternary complexes from a structure prediction perspective. For each group, molecular glues sampled by TriGlue on the test set are combined with the sequences of the two proteins and fed into AlphaFold3 [47] for complex structure prediction, after which pTM and ipTM confidence are computed. As shown in Fig. 7, AlphaFold3 predicts structurally reasonable ternary complexes for most groups, providing additional evidence for the quality of the sampled molecular glues. An exception is observed for the 6TZ8 family, which corresponds to the immunosuppressant drug FK506 [61], while TriGlue also achieves a relatively low DockQ score of 0.13 for this group.

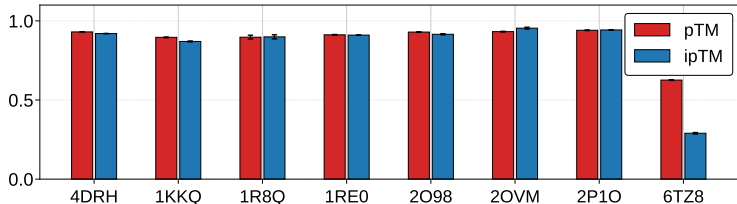


Figure 7: pTM and ipTM confidence by group.

I Broader Impacts

This work introduces a generative framework for molecular glue design, a modality with the potential to expand the druggable proteome and accelerate the discovery of therapies for diseases that remain difficult to treat using conventional small-molecule approaches. By enabling the computational exploration of ternary complex formation, our approach may reduce experimental trial-and-error, shorten early-stage drug discovery timelines, and lower research costs, which could ultimately improve access to novel therapeutics and benefit public health. Furthermore, the proposed biology-inspired generative modeling paradigm may stimulate interdisciplinary collaboration across machine learning, structural biology, and medicinal chemistry.