RAFA: REWARD-ALIGNED FLOW TUNING FOR ANTIBODY AND NANOBODY DESIGN

Anonymous authors
Paper under double-blind review

ABSTRACT

We introduce RAFA, an antigen-conditioned framework that jointly generates CDR sequences and the complete antibody structure from an antigen structure, epitope information, and an antibody framework sequence, without requiring initial antibody coordinates. Antibody and nanobody design requires coordinating sequence, structure, and the interface with a specified antigen. Generative models trained to reconstruct observed complexes do not necessarily optimize the quality of their generated interfaces. Building on La-Proteina's partially latent autoencoder, the model combines explicit C_α coordinates with per-residue latents encoding sequence and side-chain geometry. Joint flow matching is complemented by geometry supervision. We further introduce Reward-Gradient Transport (RGT), which approximately transports terminal reward gradients with respect to antibody C_α coordinates to recorded states along the same sampled trajectory and updates the shared generative velocity field. Relative to RAFA, RAFA (RL) improves mean DockQ by 22.0% for antibody design and 28.9% for nanobody design on a globally filtered, held-out benchmark. Additional structure-prediction, refolding-confidence, and interface analyses characterize both improvements and trade-offs. These results support coordinate reward-gradient post-training as an approach to improving computational antibody–antigen interface generation.

1 INTRODUCTION

Antibodies, also known as immunoglobulins, are Y-shaped glycoproteins secreted by B lymphocytes (Silverthorn, 2015; Akkaya et al., 2020), primarily found in blood and lymphatic fluid. They consist of two heavy chains and two light chains, each containing a variable domain (V_H or V_L) and a constant domain. The variable regions harbor three Complementarity-Determining Regions (CDRs) per chain, which form the antigen-binding paratope and determine binding specificity; the surrounding framework regions provide structural scaffolding with low sequence variability. Antibodies recognize and neutralize a broad spectrum of pathogens and have become indispensable tools in disease treatment, vaccine development, and personalized medicine (Schroeder Jr & Cavacini, 2010; Litman et al., 1993; Nelson et al., 2010; Weiner, 2015; Sliwkowski & Mellman, 2013). Traditional antibody discovery via animal immunization or phage display is time-consuming, costly, and requires extensive humanization to reduce immunogenicity (Georgiou et al., 2014). Computational antibody design, which involves autonomously generating CDR sequences and structures that bind a target antigen epitope, has therefore emerged as a critical capability for accelerating therapeutic discovery.

Early computational approaches relied on physics-based sampling and scoring (Adolf-Bryfogle et al., 2018; Pantazes & Maranas, 2010; Lapidoth et al., 2015), which are computationally expensive and sensitive to the quality of the energy function. Deep learning has substantially transformed this landscape. Sequence-based language models (Alley et al., 2019; Olsen et al., 2022; Ruffolo et al., 2021; Shin et al., 2021; Akbar et al., 2022) capture residue-level statistics from large antibody databases but lack structural awareness. Co-design methods that simultaneously generate protein sequences and structures (Anishchenko et al., 2021; Wang et al., 2022; Anand & Achim, 2022; Shi et al., 2023) demonstrated the feasibility of joint optimization, and this paradigm was subsequently applied to antibody design (Jin et al., 2021; Luo et al., 2022; Kong et al., 2023a;b; Wu & Li, 2024), enabling CDR backbone geometry to be modeled jointly with sequence conditioned on the antigen structure. More recently, diffusion models and consistency models have been adapted for full-complex generation (Watson et al., 2023; Wang et al., 2025), pushing toward end-to-end de novo antibody

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

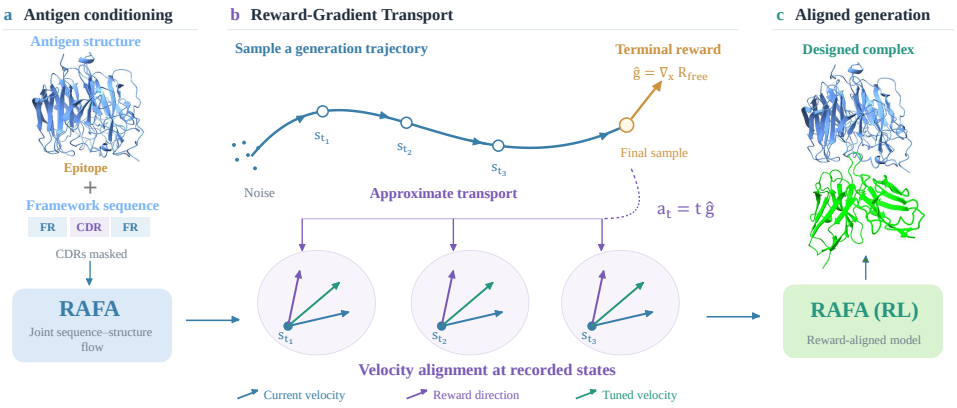

Figure 1: Overview of RAFA and Reward-Gradient Transport (RGT). (a) RAFA jointly generates sequences and structures conditioned on an antigen structure, an epitope, and a framework sequence. (b) RGT approximately transports the normalized terminal reward gradient $\hat{g}$ to recorded trajectory states. (c) The resulting RAFA (RL) model supports antibody and nanobody design.

design. Despite this progress, learning from observed complexes and generating a favorable interface remain distinct objectives. Sequence–structure co-design also requires a representation that supports coordinated sampling of residue identities and geometry. We address these needs by adapting an existing partial latent representation to antibody-conditioned generation and by post-training the generator with coordinate reward gradients.

We introduce RAFA, which generates CDR sequences and the complete antibody structure given an antigen structure, an epitope, and a framework sequence. Its local-latent representation and AE architecture are inherited from La-Proteina (Geffner et al., 2026). We use RAFA to denote the generator before reward-based post-training and RAFA (RL) to denote the model obtained by applying Reward-Gradient Transport (RGT). RGT reuses a terminal Cα reward gradient at sparse recorded states of the same stochastic rollout, replacing the intervening state sensitivities with an identity approximation. This yields local velocity updates without differentiating through the sampling trajectory. The method trades exact trajectory credit assignment for sparse gradient evaluations in the joint coordinate–latent generator.

In short, we summarize our contributions as follows:

- **Antigen-conditioned joint generation.** RAFA enables joint sequence–structure learning within a unified flow-matching formulation by adapting a partially latent flow framework to generate antibody and nanobody CDR sequences and complete structures.
- **Reward-Gradient Transport.** RGT couples sparse local velocity updates with same-trajectory coordinate-gradient reuse and an EMA reference penalty. We characterize the omitted trajectory sensitivity with a conditional error bound, and test trajectory pairing and time weighting through mechanism controls.
- **Computational evaluation across antibody formats.** We evaluate structure prediction and de novo design on antibody and nanobody complexes, complemented by sequence-refolding assessments, interface-quality analyses, and representative design examples. These experiments demonstrate improved interface quality of the generated designs.

2 Related Work

Protein Design. Protein design spans energy-based search (Adolf-Bryfogle et al., 2018; Pantazes & Maranas, 2010) and sequence generation with ProGen (Madani et al., 2023), ProtGPT2 (Ferruz et al., 2022), and EvoDiff (Alamdari et al., 2023). RFdiffusion (Watson et al., 2023) generates backbones for ProteinMPNN (Dauparas et al., 2022); Chroma (Ingraham et al., 2023) and AlphaProteo (Zambaldi et al., 2024) support programmable generation and binder design. Protpardelle (Chu et al., 2024; Lu et al., 2025) and APM (Chen et al., 2025b) generate all-atom structures; FlowDock (Morehead & Cheng, 2025) targets protein–ligand docking. BindCraft (Pacesa et al., 2025) and BoltzDesign (Cho

et al., 2025) optimize sequences through AlphaFold2 (Jumper et al., 2021) and Boltz-1 (Wohlwend et al., 2025); RL adds reward-driven optimization (Lutz et al., 2023).

Antibody Design. Methods span energy-based search (Pantazes & Maranas, 2010; Lapidoth et al., 2015; Adolf-Bryfogle et al., 2018; Warszawski et al., 2019; Ruffolo et al., 2021), sequence modeling (Alley et al., 2019; Saka et al., 2021; Shin et al., 2021; Akbar et al., 2022), joint CDR sequence–structure modeling (Jin et al., 2021; Kong et al., 2023a;b), diffusion (Luo et al., 2022; Peng et al., 2023; Zhu et al., 2024; Bennett et al., 2026), and flow-based design/refinement (Wu et al., 2025b; Wang et al., 2026b; Tan et al., 2025; Zhang et al., 2024; Chen et al., 2025a; Team & Tang, 2025). IgGM (Wang et al., 2025) and tFold (Wu et al., 2026) model antibody complexes; Germinal (Mille-Fragoso et al., 2025) targets framework-conditioned CDR design. Unlike fixed-backbone ESM-IF (Hsu et al., 2022) and PiFold (Gao et al., 2023), RAFA jointly generates CDR sequences and complete antibody structures.

Alignment for Generative Models. Diffusion (Song & Ermon, 2019; Ho et al., 2020; Austin et al., 2021), flow matching (Lipman et al., 2023; Albergo & Vanden-Eijnden, 2023; Liu et al., 2023), consistency models (Song et al., 2023; Song & Dhariwal, 2024), and protein latent representations (Geffner et al., 2025; 2026) underpin generation. Alignment spans DPO (Wallace et al., 2024; Liu et al., 2024; Wu et al., 2025c), GRPO (Shao et al., 2024; Xue et al., 2025; Wu et al., 2025a), and post-training methods including Diffusion-DPO (Wallace et al., 2024), DiffusionNFT (Zheng et al., 2025), Flow-GRPO (Liu et al., 2025a), TempFlow-GRPO (He et al., 2025), and Reinforce Adjoint Matching (RAM) (Bergmeister et al., 2026). DRaFT (Clark et al., 2024) differentiates through sampling; Adjoint Matching (Domingo-Enrich et al., 2025) and VGG-Flow (Liu et al., 2025b) use adjoint/value-gradient formulations. DRTune (Wu et al., 2024) detaches inputs; FlowBP-Sparse (Wang et al., 2026a) uses sparse, decoupled velocity gradients. RGT builds on this gradient structure to address interface alignment in joint antibody sequence–structure generation. Its contribution is a coordinate transport rule that combines each stochastic rollout's own endpoint gradient with its recorded states, unit-mean coordinate-time weights, endpoint-gradient normalization, and an EMA velocity constraint. These choices determine which geometric outputs receive supervision, how credit is allocated along the sampled path, and how local updates are stabilized. RGT uses differentiable coordinate rewards, whereas RAM uses scalar rewards. Protein-specific approaches include sequence fine-tuning (Stanton et al., 2022), backbone RL (Lutz et al., 2023), antibody preferences in AbDPO (Zhou et al., 2024), inference-time selection in BoltzGen (Stark et al., 2025) and Proteina-Complexa (Didi et al., 2026), and privileged-geometry on-policy distillation in ABOPD (Yang et al., 2026).

3 METHOD

3.1 PROBLEM FORMULATION

The condition $c = (\mathcal{A}, e, \mathbf{s}_{fw}, M_{\mathrm{CDR}})$ comprises the full antigen structure and sequence $\mathcal{A}$, an epitope mask e, the antibody framework sequence $\mathbf{s}_{fw}$, and a CDR-position mask M_{CDR}. Masked CDR identities are not provided. Chain lengths and CDR lengths are fixed by the input. Antigen Cα coordinates and sequence are held fixed; the sampling network uses explicit antigen Cα geometry.

For **de novo design**, we generate CDR sequences and the complete antibody structure, including framework coordinates, without an initial antibody structure. Framework sequence identities remain fixed. For **structure prediction**, the full antibody sequence is supplied and the structure is predicted. A single model supports both VH–VL antibodies and VHH nanobodies.

3.2 LATENT FLOW MATCHING FOR DE NOVO DESIGN

Partial Autoencoder. As illustrated in Figure 2 (bottom), we initialize the partially latent autoencoder (AE) from La-Proteina (Geffner et al., 2026) and fine-tune it on antibody data to support antigen-conditioned sequence–structure generation. The AE projects per-residue all-atom antibody coordinates into explicit C_α coordinates and a compact 8-dimensional latent space:

$$(\mathbf{x}_{ab}, \mathbf{z}) = \mathrm{Enc}(\mathbf{C}_{ab}, \mathbf{s}_{ab}) \in \mathbb{R}^{N_{ab}\times 3} \times \mathbb{R}^{N_{ab}\times 8} \tag{1}$$

The decoder reconstructs both full-atom coordinates and amino acid sequences from the tuple $(\mathbf{x}_{ab}, \mathbf{z})$, essentially encoding sequence identity and side-chain geometry jointly within the latent vector $\mathbf{z}$, while leaving the C_α coordinates $\mathbf{x}_{ab}$ explicit.

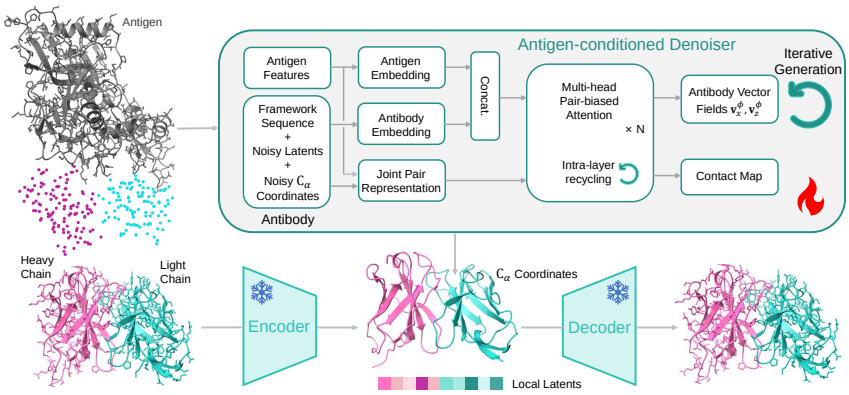

Figure 2: RAFA framework. **Top:** The antigen-conditioned denoiser processes antigen features and noisy antibody inputs (framework sequence, noisy latents, noisy C_α coordinates) through N layers of pair-biased attention with intra-layer recycling, outputting vector fields used by the stochastic sampler. A contact head provides auxiliary interface supervision. **Bottom:** The frozen partial autoencoder encodes full-atom antibody coordinates into C_α coordinates and 8-dim per-residue latents; the decoder reconstructs full-atom structures and CDR sequences at inference.

Conditional Flow Matching. We model the product state $\mathbf{s} = (\mathbf{x}, \mathbf{z})$, with modes $m \in \{x, z\}$ of dimensions $d_x = 3$ and $d_z = 8$ per residue. We denote coordinate and latent times by t_x and t_z, respectively, and write $\mathbf{t} = (t_x, t_z)$. Residue latents $\mathbf{z}_i$ are distinct from the pair representations $\mathbf{p}_{ij}$ used by the denoiser. For each mode, a clean state $\mathbf{s}_1^m$ and Gaussian noise $\mathbf{s}_0^m$ define

$$\mathbf{s}_{t_m}^m = (1 - t_m)\mathbf{s}_0^m + t_m \mathbf{s}_1^m, \qquad v^{*,m} = \mathbf{s}_1^m - \mathbf{s}_0^m. \tag{2}$$

The conditional network predicts $v_\theta^m(\mathbf{s_t}, \mathbf{t}; c)$ and the clean-state estimate $\widehat{\mathbf{s}}_1^m = \mathbf{s}_{t_m}^m + (1 - t_m)v_\theta^m$. For residue mask M and $N_M = \sum_i M_i$, the implemented per-mode loss is

$$\mathcal{L}_{\text{FM}}^m = \mathbb{E}\left[\frac{\sum_i M_i \|\widehat{\mathbf{s}}_{1,i}^m - \mathbf{s}_{1,i}^m\|_2^2}{N_M((1 - t_m)^2 + \epsilon_{\text{FM}})}\right], \quad \epsilon_{\text{FM}} = 10^{-5}. \tag{3}$$

3.3 Geometry-Aware Architecture

The denoiser employs a 14-layer transformer integrating 1D sequence tokens and 2D pair representations, which processes noisy coordinates, framework sequences, and relative geometry embeddings.

Differentiable Intra-layer Recycling. At each intermediate layer l, we project the antibody hidden states $\mathbf{h}_{ab}^{(l)}$ to predict a clean C_α structure:

$$\hat{\mathbf{x}}_1^{(l)} = \mathbf{x}_{t_x,\,ab} + (1 - t_x) \cdot \text{Linear}_3(\mathbf{h}_{ab}^{(l)}) \tag{4}$$

where $\mathbf{x}_{t_x,\,ab} \in \mathbb{R}^{N_{ab} \times 3}$ is the specific C_α coordinate component extracted from $\mathbf{x}_{t_x}$, and $\text{Linear}_3 : \mathbb{R}^d \to \mathbb{R}^3$ projects hidden states to 3D coordinates. Pairwise distances of $\hat{\mathbf{x}}_1^{(l)}$ are extracted, encoded via Gaussian RBFs, and injected back into the pair representation $\mathbf{p}_{ij}^{(l)}$ via t_x-conditioned AdaLN (Perez et al., 2018). This mechanism allows the network to self-correct.

Geometry supervision and teacher–student training. Auxiliary losses act on predicted clean Cα coordinates and contact-head outputs, with gate $\omega(t_x) = \mathbf{1}[t_x > 0.4]t_x$. They supervise virtual backbone geometry, interface distances, alignment, contact recovery, and steric separation. Base training also uses a training-only teacher branch with privileged target features and velocity distillation to a student branch using inference-available conditions. The objective is

$$\mathcal{L}_{\text{base}} = \sum_{b \in \{\text{student,teacher}\}} \left(\mathcal{L}_{\text{FM}}^b + \mathcal{L}_{\text{aux}}^b\right) + 0.5\mathcal{L}_{\text{distill}}. \tag{5}$$

3.4 Reward-Gradient Transport

We introduce **Reward-Gradient Transport (RGT)** to post-train RAFA using an interface reward (Figure 1). RGT turns a terminal interface gradient into sparse local velocity targets for the joint

coordinate–latent generator. Its transport rule preserves each recorded state's association with its own stochastic endpoint, balances trajectory contributions through gradient normalization, and allocates supervision across anchors with normalized coordinate-time weights. An EMA reference constrains the resulting velocity updates. Together, these choices adapt interface geometry while retaining the shared sequence–structure generator and avoiding differentiation through the rollout.

Interface-quality reward. We score each generated complex with a composite reward:

$$R_{\text{free}} = w_1 R_{\text{aff}} + w_2 R_{\text{fold}} + w_3 R_{\text{physical}} + w_4 R_{\text{dev}}, \tag{6}$$

where $w_i \geq 0$ and $\sum_{i=1}^{4} w_i = 1$. The four terms assess affinity-related interface geometry and sequence compatibility, predicted fold confidence, steric plausibility, and sequence developability. The fold-confidence surrogate is calibrated on Boltz-2-derived labels using the RAFA training split.

Current-model rollouts and terminal gradients. For each condition c, the current model samples K stochastic trajectories, retaining their intermediate states and terminal states $\mathbf{s}_{1,k} = (\mathbf{x}_{1,k}, \mathbf{z}_{1,k})$. Here $\mathbf{x}$ denotes explicit antibody Cα coordinates and $\mathbf{z}$ denotes local latents. We evaluate the composite reward R_{free} at each terminal state. For antibody mask M_{Ab}, the terminal gradient is

$$g_k = M_{\text{Ab}} \odot \nabla_{\mathbf{x}_{1,k}} R_{\text{free}}, \qquad \widehat{g}_k = \frac{\rho\, g_k}{\max(\|g_k\|_2, \epsilon)}, \quad \rho = 0.5. \tag{7}$$

Same-trajectory transport. We select J recorded states from each rollout, rather than constructing new interpolation paths. The terminal direction is approximately transported to each state using

$$a_{k,j} = \eta \widetilde{w}_{k,j} \widehat{g}_k, \qquad w(t_x) = (1-d) + d t_x^{\nu}, \qquad w_{k,j} = \frac{w(t_{x,k,j})}{J^{-1} \sum_{\ell=1}^{J} w(t_{x,k,\ell})}. \tag{8}$$

Here $t_{x,k,j}$ is the coordinate time of recorded state j on trajectory k and ν is the time-weight exponent. We use $\eta = d = \nu = 1$, giving normalized linear weights $\widetilde{w}_{k,j} = t_{x,k,j} / (J^{-1} \sum_{\ell=1}^{J} t_{x,k,\ell})$ with unit mean across anchors. For a fixed rollout-noise realization, exact terminal credit at an intermediate state involves the transpose Jacobian of its remaining transition map. RGT substitutes the identity for this Jacobian and applies the normalized late-time weight.

Velocity-direction update. At each recorded state, we align the Cα velocity with the transported reward direction while regularizing its deviation from an exponential moving average (EMA) reference. Writing $v_{k,j} = v_\theta^x(\text{sg}(\mathbf{s}_{k,j}), \mathbf{t}_{k,j}; c)$ and $\bar{v}_{k,j} = v_{\bar{\theta}}^x(\text{sg}(\mathbf{s}_{k,j}), \mathbf{t}_{k,j}; c)$, we minimize

$$\mathcal{L}_{\text{RGT}} = \frac{1}{KJ} \sum_{k=1}^{K} \sum_{j=1}^{J} \left[- \langle v_{k,j}, \text{sg}(a_{k,j}) \rangle_{\text{Ab}} + \frac{\lambda}{2} \| v_{k,j} - \text{sg}(\bar{v}_{k,j}) \|_{\text{Ab}}^2 \right], \tag{9}$$

where λ controls the reference penalty, $\bar{\theta}$ denotes the EMA parameters, $\mathbf{t}_{k,j} = (t_{x,k,j}, t_{z,k,j})$, and sg denotes stop-gradient. We use $\langle u, v \rangle_{\text{Ab}} = (3N_{\text{Ab}})^{-1} \sum_i M_{\text{Ab},i} \sum_{q=1}^{3} u_{i,q} v_{i,q}$ and $\|u\|_{\text{Ab}}^2 = \langle u, u \rangle_{\text{Ab}}$. Gradients are accumulated over states and trajectories before a clipped optimizer update.

4 Experiments

4.1 Experimental Setup

Data and tasks. The antibody dataset contains 12,759 complexes from SAbDab (Dunbar et al., 2014): 12,364 training, 252 validation, and 143 test structures (93 VH–VL and 50 VHH). We combine a temporal split at 2025-06-30 with global sequence-alignment filtering at a normalized score threshold of 0.8. Appendix B describes the filtering score and the scope of generalization. CDRs use IMGT numbering (Lefranc et al., 2003; 2009); epitopes are specified by antigen contacts.

Baselines. Following IgGM (Wang et al., 2025), we compare both complex-structure prediction and antigen-conditioned CDR design, including structure-initialized design baselines. Prediction baselines include IgFold–HDock (Ruffolo et al., 2023; Yan et al., 2020), tFold-Ag (Wu et al., 2026), AlphaFold 3 (Abramson et al., 2024), and dyMEAN (Kong et al., 2023b); we extend this comparison with Boltz-2, Protenix-v2, and IgGM itself. For design, we include DiffAb (Luo et al., 2022), MEAN (Kong et al., 2023a), dyMEAN, and IgGM, together with RFantibody (Bennett et al., 2026),

Table 1: Results for complex structure prediction on the 93-antibody test set. Methods marked with † require the antigen structure as input, while $*$ denotes the use of antigen epitope information. ‡ indicates methods utilizing multiple sequence alignments (MSAs). CDR-H3 backbone RMSD (Å) is reported. For Boltz-2, Protenix-v2 and AlphaFold 3, per-chain unpaired MSAs are supplied.

Method	Antibody Structure			Docking			
	TM-Score↑	lDDT↑	RMSD↓	DockQ↑	iRMS↓	LRMS↓	SR↑
IgFold→ HDock†	$0.918^{0.119}$	$0.796^{0.047}$	$3.831^{1.094}$	$0.017^{0.031}$	$19.129^{9.861}$	$57.511^{23.135}$	0.000
tFold-Ag*‡	$0.929^{0.047}$	$0.813^{0.070}$	$2.061^{0.868}$	$0.061^{0.042}$	$12.658^{3.248}$	$37.851^{21.680}$	0.043
Boltz-2‡	$0.934^{0.061}$	$0.843^{0.072}$	$2.009^{1.139}$	$0.078^{0.119}$	$13.918^{7.232}$	$39.319^{22.507}$	0.075
Protenix-v2‡	$\mathbf{0.945}^{0.046}$	$\mathbf{0.850}^{0.078}$	$\underline{1.813}^{1.266}$	$\mathbf{0.219}^{0.300}$	$10.916^{8.566}$	$34.082^{29.078}$	0.269
AlphaFold 3‡	$\underline{0.944}^{0.047}$	$\underline{0.849}^{0.076}$	$\mathbf{1.801}^{1.371}$	$\underline{0.213}^{0.247}$	$11.032^{6.993}$	$32.466^{25.263}$	$\underline{0.280}$
dyMEAN†*	$0.916^{0.044}$	$0.801^{0.062}$	$2.153^{0.852}$	$0.107^{0.092}$	$9.359^{3.860}$	$24.325^{9.057}$	0.118
IgGM†*	$0.892^{0.125}$	$0.751^{0.047}$	$2.417^{1.025}$	$0.170^{0.164}$	$10.005^{5.732}$	$23.943^{16.170}$	0.258
RAFA†*	$0.932^{0.046}$	$0.820^{0.064}$	$2.268^{1.073}$	$0.165^{0.158}$	$\underline{9.087}^{4.881}$	$\underline{22.961}^{11.717}$	0.269
RAFA (RL)†*	$0.924^{0.052}$	$0.813^{0.064}$	$2.266^{1.057}$	$0.187^{0.163}$	$\mathbf{8.873}^{4.853}$	$\mathbf{22.408}^{11.608}$	**0.323**

Table 2: Results for de novo antibody design on the 93-antibody test set. RFantibody instead uses an IgFold-initialized framework structure. Other systems follow their documented design protocols.

Model		RFantibody (IgFold)	dyMEAN	IgGM	BoltzGen	Germinal	RAFA	RAFA (RL)
AAR↑	L1	$0.185^{0.162}$	$0.678^{0.237}$	$0.628^{0.245}$	$0.492^{0.217}$	$0.420^{0.156}$	$\underline{0.693}^{0.196}$	$\mathbf{0.698}^{0.191}$
	L2	$0.266^{0.233}$	$0.710^{0.332}$	$0.629^{0.343}$	$0.551^{0.252}$	$0.583^{0.211}$	$\underline{0.821}^{0.252}$	$\mathbf{0.822}^{0.256}$
	L3	$0.174^{0.131}$	$0.624^{0.174}$	$0.585^{0.188}$	$0.438^{0.175}$	$0.427^{0.141}$	$\mathbf{0.727}^{0.246}$	$\underline{0.718}^{0.246}$
	H1	$0.262^{0.119}$	$\mathbf{0.751}^{0.179}$	$0.644^{0.154}$	$0.601^{0.167}$	$0.532^{0.166}$	$0.696^{0.194}$	$\underline{0.703}^{0.179}$
	H2	$0.225^{0.155}$	$0.662^{0.207}$	$0.350^{0.179}$	$0.424^{0.159}$	$0.454^{0.183}$	$\underline{0.667}^{0.226}$	$\mathbf{0.672}^{0.225}$
	H3	$0.109^{0.077}$	$0.378^{0.126}$	$0.245^{0.133}$	$0.293^{0.108}$	$0.277^{0.113}$	$\underline{0.512}^{0.192}$	$\mathbf{0.514}^{0.190}$
RMSD↓	L1	$1.247^{1.057}$	$0.803^{0.603}$	$0.828^{0.796}$	$0.777^{0.811}$	$1.260^{0.827}$	$\mathbf{0.629}^{0.665}$	$\underline{0.653}^{0.671}$
	L2	$0.155^{0.160}$	$0.263^{0.303}$	$0.162^{0.147}$	$0.141^{0.169}$	$0.258^{0.434}$	$\underline{0.138}^{0.177}$	$\mathbf{0.137}^{0.186}$
	L3	$1.113^{0.781}$	$0.762^{0.729}$	$0.807^{0.788}$	$0.788^{0.800}$	$0.859^{0.588}$	$\mathbf{0.728}^{0.753}$	$\underline{0.740}^{0.749}$
	H1	$1.025^{0.694}$	$0.796^{0.522}$	$0.778^{0.540}$	$\mathbf{0.740}^{0.555}$	$0.774^{0.474}$	$0.772^{0.571}$	$\underline{0.757}^{0.546}$
	H2	$0.781^{0.670}$	$0.645^{0.397}$	$0.793^{0.488}$	$0.569^{0.524}$	$1.489^{0.717}$	$\mathbf{0.538}^{0.483}$	$\underline{0.563}^{0.515}$
	H3	$2.759^{1.009}$	$\mathbf{2.224}^{0.865}$	$2.572^{0.926}$	$2.533^{1.106}$	$2.607^{1.674}$	$\underline{2.335}^{1.035}$	$2.396^{1.085}$
Structure↑	TMscore	$0.907^{0.068}$	$\underline{0.928}^{0.043}$	$0.918^{0.045}$	$0.925^{0.053}$	$0.924^{0.100}$	$\mathbf{0.929}^{0.046}$	$0.925^{0.047}$
	lDDT	$0.721^{0.061}$	$0.777^{0.059}$	$0.635^{0.049}$	$0.705^{0.062}$	$0.653^{0.058}$	$\mathbf{0.783}^{0.066}$	$\underline{0.778}^{0.065}$
Side chain↓	Clash rate	$0.030^{0.027}$	$0.049^{0.020}$	$0.025^{0.007}$	$0.021^{0.002}$	$0.038^{0.020}$	$\mathbf{0.018}^{0.017}$	$\underline{0.019}^{0.017}$
	scRMSD	$17.667^{7.909}$	$17.779^{6.808}$	$15.055^{7.340}$	$19.999^{7.929}$	$20.398^{5.904}$	$\underline{14.227}^{7.308}$	$\mathbf{12.210}^{6.000}$
Interface	Polar contact↑	$0.004^{0.018}$	$0.013^{0.049}$	$0.011^{0.064}$	$\underline{0.016}^{0.048}$	$0.000^{0.000}$	$0.014^{0.038}$	$\mathbf{0.049}^{0.106}$
	$\lvert\Delta\text{BSA}\rvert\downarrow$	$976.0^{487.0}$	$1135.0^{1077.2}$	$819.9^{449.3}$	$672.1^{411.3}$	$698.8^{548.5}$	$\underline{248.9}^{151.3}$	$\mathbf{121.5}^{90.8}$
	Median $\Delta G\downarrow$	-41.8	7.4	$\underline{-42.3}$	-41.6	-15.0	-32.2	$\mathbf{-49.3}$
Docking	DockQ↑	$0.069^{0.073}$	$0.077^{0.065}$	$0.139^{0.160}$	$0.124^{0.114}$	$0.122^{0.121}$	$\underline{0.141}^{0.173}$	$\mathbf{0.172}^{0.165}$
	iRMS↓	$10.337^{5.688}$	$9.563^{3.936}$	$\underline{8.563}^{5.405}$	$8.707^{4.957}$	$8.849^{4.087}$	$8.993^{5.383}$	$\mathbf{8.371}^{5.194}$
	LRMS↓	$28.854^{16.073}$	$27.510^{11.337}$	$\underline{21.695}^{14.729}$	$21.697^{11.920}$	$26.712^{12.053}$	$23.825^{15.166}$	$\mathbf{21.062}^{12.356}$
	SR↑	0.043	0.032	0.194	0.172	0.172	$\underline{0.204}$	**0.280**

BoltzGen (Stark et al., 2025), and Germinal (Mille-Fragoso et al., 2025). DiffAb and MEAN use IgFold- or AF3-predicted initial structures, while RFantibody uses an IgFold-initialized framework.

Training and sampling. The RAFA architecture is a 14-layer transformer with token width 768, pair width 256, and 12 heads. We first train on general proteins for two days using 16 H20 GPUs, then continue training on the antibody/nanobody dataset for another two days on the same number of GPUs. The second stage uses AdamW with learning rate 10^{-4}, bf16, one sample per rank, and antibody:nanobody sampling slots of 3:1. RGT then post-trains this model on VH–VL conditions for 2,000 parameter updates using 8 H20 GPUs and learning rate 5×10^{-7}. Each condition uses eight fresh trajectories, with 50 simulation steps per trajectory and four recorded states.

Evaluation and comparison boundaries. DockQ, iRMS, LRMS, and SR are computed after generation against reference complexes. SR denotes the fraction of evaluated designs with DockQ > 0.23. All methods use the same test sets, metric definitions, and postprocessing protocol. Every post-training method uses the same composite reward R_{free} and identical component weights from Appendix C.3; the baselines retain their original optimization objectives and update procedures.

4.2 Complex Structure Prediction of Antibodies

Structure prediction separates interface placement from sequence generation because the antibody sequence is fixed. Table 1 shows that RGT improves docking quality and success over RAFA, with RAFA (RL) achieving the lowest iRMS and LRMS among the compared methods.

Table 3: Protenix-v2 sequence-refolding confidence and reference-based docking accuracy for existing designs on the 93-antibody test set. DockQ compares each refolded complex with the corresponding native reference complex.

Method	Global ipTM↑	pLDDT↑	pTM↑	ipSAE↑	pDockQ↑	Ranking score↑	DockQ↑
DiffAb (IgFold)	$0.516^{0.135}$	$\underline{83.211}^{5.100}$	$0.665^{0.097}$	$0.025^{0.073}$	$0.119^{0.062}$	$0.546^{0.125}$	$0.046^{0.042}$
MEAN (IgFold)	$0.496^{0.140}$	$81.988^{5.138}$	$0.660^{0.092}$	$0.023^{0.061}$	$0.125^{0.071}$	$0.529^{0.128}$	$0.047^{0.043}$
DiffAb (AF3)	$0.507^{0.142}$	$81.680^{5.919}$	$0.659^{0.104}$	$0.030^{0.079}$	$0.119^{0.054}$	$0.538^{0.132}$	$0.047^{0.043}$
MEAN (AF3)	$0.494^{0.136}$	$81.759^{4.918}$	$0.661^{0.090}$	$0.014^{0.034}$	$0.125^{0.064}$	$0.528^{0.123}$	$0.047^{0.042}$
RFantibody (IgFold)	$0.491^{0.133}$	$82.508^{5.534}$	$0.657^{0.085}$	$0.017^{0.060}$	$0.133^{0.074}$	$0.524^{0.120}$	$0.048^{0.054}$
dyMEAN	$0.504^{0.135}$	$82.978^{6.023}$	$0.661^{0.096}$	$0.019^{0.043}$	$0.110^{0.049}$	$0.535^{0.125}$	$0.058^{0.062}$
IgGM	$0.505^{0.133}$	$82.829^{5.797}$	$0.657^{0.097}$	$0.020^{0.056}$	$0.115^{0.058}$	$0.535^{0.124}$	$0.063^{0.063}$
BoltzGen	$0.511^{0.147}$	$82.670^{5.526}$	$0.666^{0.100}$	$0.031^{0.104}$	$\underline{0.134}^{0.078}$	$0.542^{0.135}$	$0.052^{0.054}$
Germinal	$0.487^{0.131}$	$80.362^{5.650}$	$0.656^{0.088}$	$0.022^{0.071}$	$0.120^{0.051}$	$0.521^{0.119}$	$0.068^{0.069}$
RAFA	$\underline{0.521}^{0.139}$	$82.272^{5.853}$	$\underline{0.672}^{0.096}$	$\underline{0.035}^{0.076}$	$0.125^{0.056}$	$\underline{0.551}^{0.128}$	$\underline{0.088}^{0.090}$
RAFA (RL)	$\mathbf{0.522}^{0.154}$	$\mathbf{83.307}^{7.177}$	$\mathbf{0.705}^{0.103}$	$\mathbf{0.065}^{0.119}$	$\mathbf{0.188}^{0.076}$	$\mathbf{0.553}^{0.139}$	$\mathbf{0.158}^{0.131}$

Table 4: Results for structure prediction and de novo design on the 50-nanobody test set. In the design section, (AF3) indicates that DiffAb uses AlphaFold 3 predictions as the initial framework.

Structure prediction of Nanobody								
Method	TM-Score↑	lDDT↑	RMSD↓		DockQ↑	iRMS↓	LRMS↓	SR↑
tFold-Ag*‡	$0.855^{0.204}$	$0.740^{0.218}$	$2.297^{1.666}$		$0.052^{0.072}$	$13.782^{5.010}$	$41.721^{18.821}$	0.040
Boltz-2‡	$0.926^{0.054}$	$0.829^{0.076}$	$1.926^{1.372}$		$0.142^{0.201}$	$11.012^{6.132}$	$32.516^{20.307}$	0.220
Protenix-v2‡	$\mathbf{0.938}^{0.055}$	$\mathbf{0.848}^{0.084}$	$\mathbf{1.481}^{1.306}$		$\mathbf{0.258}^{0.312}$	$9.420^{7.017}$	$28.073^{23.016}$	**0.380**
AlphaFold 3‡	$\underline{0.935}^{0.055}$	$\underline{0.844}^{0.081}$	$\underline{1.665}^{1.359}$		$\underline{0.237}^{0.308}$	$10.456^{7.923}$	$32.563^{25.322}$	$\underline{0.340}$
IgGM†*	$0.899^{0.058}$	$0.733^{0.012}$	$2.598^{1.204}$		$0.112^{0.094}$	$8.482^{3.233}$	$19.599^{8.776}$	0.100
RAFA†*	$0.913^{0.047}$	$0.794^{0.060}$	$2.392^{1.277}$		$0.181^{0.136}$	$\underline{7.859}^{3.488}$	$\underline{17.532}^{6.859}$	0.240
RAFA (RL)†*	$0.911^{0.048}$	$0.792^{0.061}$	$2.390^{1.324}$		$0.195^{0.158}$	$\mathbf{7.650}^{3.537}$	$\mathbf{16.977}^{7.182}$	0.280
Design of Nanobody								
Method	CDR1↑	CDR2↑	CDR3↑	RMSD↓	DockQ↑	iRMS↓	LRMS↓	SR↑
DiffAb (AF3)	$0.419^{0.207}$	$0.216^{0.160}$	$0.228^{0.177}$	$2.773^{1.963}$	$0.053^{0.031}$	$16.874^{6.567}$	$43.632^{23.630}$	0.020
IgGM	$0.243^{0.159}$	$0.180^{0.125}$	$0.107^{0.096}$	$2.806^{1.301}$	$0.114^{0.106}$	$\mathbf{7.656}^{3.587}$	$\underline{20.308}^{11.662}$	$\underline{0.120}$
BoltzGen	$0.362^{0.200}$	$0.404^{0.158}$	$0.204^{0.091}$	$\mathbf{2.268}^{1.037}$	$\underline{0.127}^{0.127}$	$8.050^{4.560}$	$21.555^{15.806}$	$\underline{0.120}$
Germinal	$0.347^{0.163}$	$0.256^{0.167}$	$0.240^{0.147}$	$2.639^{1.068}$	$0.112^{0.102}$	$8.656^{2.772}$	$22.373^{10.256}$	0.100
RAFA	$\underline{0.478}^{0.227}$	$\underline{0.553}^{0.235}$	$\underline{0.458}^{0.242}$	$\underline{2.473}^{1.253}$	$0.114^{0.112}$	$8.417^{3.324}$	$20.565^{8.292}$	$\underline{0.120}$
RAFA (RL)	$\mathbf{0.483}^{0.237}$	$\mathbf{0.559}^{0.238}$	$\mathbf{0.459}^{0.244}$	$2.526^{1.276}$	$\mathbf{0.147}^{0.134}$	$\underline{8.041}^{3.342}$	$\mathbf{19.916}^{8.331}$	**0.220**

4.3 De Novo Antibody Design

Interface improvement with retained sequence recovery. De novo design requires the model to generate CDR identities as well as the complete antibody structure. RAFA (RL) achieves the highest DockQ and design success in the comparison (Table 2), while retaining strong CDR recovery. Relative to RAFA, the coordinated improvement in DockQ, iRMS, and LRMS suggests better antibody–antigen placement rather than a gain confined to a single docking measure. Our models condition on the framework sequence without requiring initial antibody coordinates; comparisons with structure-initialized DiffAb and MEAN are presented in Appendix E.8.

The additional interface measurements provide a finer-grained view of these gains. RAFA (RL) improves interface side-chain agreement, polar-contact recovery, and buried-surface-area agreement over RAFA. Figure 3 shows higher DockQ for RAFA (RL) than for RAFA in all four selected antibody and nanobody designs, with the magnitude of improvement varying by target.

Sequence-refolding assessment. To assess the designs beyond their directly generated coordinates, we refold their sequences with Protenix-v2 (Table 3). RAFA (RL) has the highest reported mean confidence scores and refolded DockQ among the compared methods. DockQ is computed between each refolded complex and the corresponding native reference complex. Relative to RAFA, mean DockQ after refolding increases from 0.088 to 0.158, supporting improved reference-based docking accuracy for the designed sequences after RGT.

4.4 Nanobody Prediction and Design

For prediction (Table 4), RGT improves docking and success relative to RAFA. RAFA (RL) achieves the lowest interface and ligand RMS errors. The VHH improvements suggest cross-format transfer

Table 5: Additional interface-quality results for de novo nanobody design on the 50-nanobody test set

Method	Clash rate (%)↓	Interface scRMSD (Å)↓	Polar-contact recovery (%)↑	Median ΔG (REU)↓	Mean $\lvert\Delta\text{BSA}\rvert$ (Å^2)↓
DiffAb (AF3)	$\mathbf{0.390}^{0.810}$	$\underline{15.694}^{7.186}$	$0.000^{0.000}$	-18.915	$303.274^{275.900}$
IgGM	$1.280^{1.560}$	$16.699^{6.952}$	$0.120^{0.820}$	-12.957	$212.271^{154.013}$
BoltzGen	$1.720^{1.400}$	$15.722^{6.952}$	$\underline{1.290}^{4.220}$	-7.561	$200.270^{146.672}$
Germinal	$1.900^{1.800}$	$18.539^{3.722}$	$0.100^{0.700}$	-17.464	$292.042^{237.260}$
RAFA	$1.920^{2.030}$	$16.001^{8.208}$	$0.920^{2.660}$	$\underline{-21.193}$	$\mathbf{130.714}^{104.794}$
RAFA (RL)	$\underline{0.912}^{1.832}$	$\mathbf{15.345}^{6.933}$	$\mathbf{2.294}^{5.787}$	$\mathbf{-27.975}$	$\underline{142.018}^{93.667}$

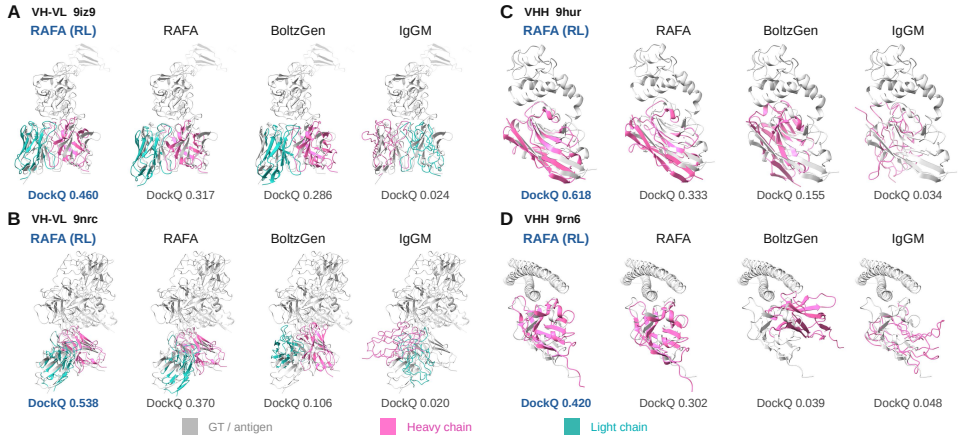

Figure 3: Selected antibody and nanobody designs superimposed on reference complexes. Heavy chain: pink; light chain: teal. Compared with RAFA, RAFA (RL) improves docking on `9iz9`, `9nrc`, `9hur`, and `9rn6`. BoltzGen and IgGM provide an additional comparison.

of interface-oriented adaptation from VH–VL RGT post-training. For de novo design, RAFA (RL) combines the strongest reported CDR recovery with the highest DockQ and success rate. This again distinguishes interface improvement from more accurate native-loop reconstruction. IgGM retains the lowest iRMS, showing that the relative strengths of the methods depend on the aspect of interface geometry being measured.

Table 5 examines the nanobody interfaces after the specified Rosetta postprocessing. Compared with RAFA, RAFA (RL) yields fewer clashes, better interface side-chain agreement, and greater polar-contact recovery, together with a more favorable median interface-energy score. These complementary improvements suggest that the docking gains are accompanied by better contact organization. Buried-surface-area agreement, however, is slightly worse, indicating that improved local contacts do not necessarily imply closer recovery of the native interface size. The energy scores characterize computational interface favorability rather than experimental binding affinity.

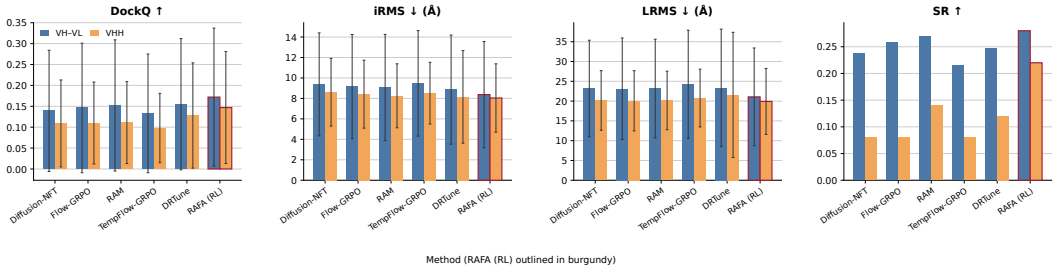

Figure 4: Comparison of RAFA (RL) with post-training baselines after 2,000 parameter updates using the same composite reward and component weights on antibody and nanobody design.

4.5 Comparison with Post-Training Methods

Figure 4 compares RAFA (RL) with Diffusion-NFT, Flow-GRPO, RAM, TempFlow-GRPO, and a DRTune control (Wu et al., 2024) after 2,000 parameter updates from the same base checkpoint. All

Table 6: Ablation study for de novo antibody design on the 93-antibody test set. Panel A reports base-training ablations, and panel B reports selected RGT mechanism controls. AAR and self-aligned RMSD are reported for CDR-H3. Additional controls are in Table 8.

Model	AAR H3↑	RMSD H3↓	TM-Score↑	lDDT↑	DockQ↑	iRMS↓	LRMS↓	SR↑
A. Base-training ablations								
RAFA	$\underline{0.512}^{0.192}$	$\mathbf{2.335}^{1.035}$	$\mathbf{0.929}^{0.046}$	$\mathbf{0.783}^{0.066}$	$0.141^{0.173}$	$8.993^{5.383}$	$23.825^{15.166}$	0.204
w/o Docking loss	$0.496^{0.194}$	$2.831^{1.066}$	$0.743^{0.252}$	$0.613^{0.041}$	$0.114^{0.115}$	$9.606^{4.637}$	$24.012^{15.243}$	0.129
w/o Structure loss	$0.491^{0.203}$	$2.584^{1.093}$	$0.865^{0.184}$	$0.762^{0.068}$	$0.134^{0.117}$	$\underline{8.510}^{4.580}$	$\underline{21.860}^{10.540}$	0.194
w/o Protein data	$0.343^{0.128}$	$2.556^{0.954}$	$0.854^{0.039}$	$0.558^{0.033}$	$0.122^{0.123}$	$9.270^{4.889}$	$23.300^{14.743}$	0.172
B. Selected RGT mechanism controls								
Random direction	$0.510^{0.194}$	$\underline{2.361}^{1.048}$	$\underline{0.927}^{0.048}$	$\underline{0.781}^{0.067}$	$0.150^{0.171}$	$9.040^{5.420}$	$23.960^{15.350}$	0.215
Cross-trajectory, $w(t_x) = t_x$	$0.509^{0.195}$	$2.401^{1.069}$	$0.926^{0.050}$	$0.779^{0.068}$	$0.154^{0.168}$	$8.870^{5.331}$	$23.380^{14.916}$	0.247
Same trajectory, $w(t_x) = 1$	$0.509^{0.194}$	$2.397^{1.061}$	$0.925^{0.049}$	$0.779^{0.067}$	$\underline{0.164}^{0.166}$	$8.570^{5.244}$	$22.210^{13.934}$	$\underline{0.269}$
RAFA (RL)	$\mathbf{0.514}^{0.190}$	$2.396^{1.085}$	$0.925^{0.047}$	$0.778^{0.065}$	$\mathbf{0.172}^{0.165}$	$\mathbf{8.371}^{5.194}$	$\mathbf{21.062}^{12.356}$	**0.280**

methods use the same reward function and weights, including the same frozen reward predictors. The DRTune control additionally matches the generator, gradient scale, anchor count, and regularization to RGT; the other baselines retain their respective optimization objectives and update procedures. RGT has the best reported means across all four docking measures in both antibody formats under this common-update protocol. Appendix F details the shared reward and update settings alongside the measured RGT and BPTT costs.

4.6 Ablation and Training Dynamics

The training variants in Table 6 reveal complementary roles for geometry supervision and protein data. Removing docking loss produces the largest decline in DockQ, together with poorer global reconstruction. Removing protein data particularly weakens sequence recovery, suggesting that broader protein training benefits the antibody generator beyond interface placement. The structure-loss variant has lower interface placement errors than RAFA but worse DockQ and success. On top of the complete base model, RGT further improves docking while largely preserving sequence recovery. These comparisons distinguish the contribution of base-model training from the additional benefit of reward-based post-training. Panel B of Table 6 examines selected mechanism controls.

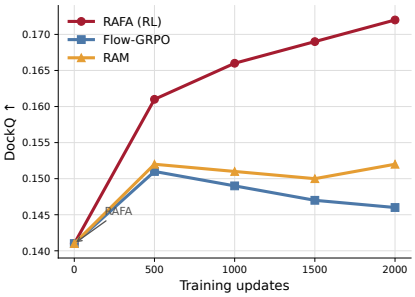

Figure 5: Evaluation DockQ during post-training from the same RAFA base.

The training dynamics in Figure 5 further distinguish the methods. RGT improves DockQ progressively across the evaluated checkpoints. Flow-GRPO improves initially and then declines after 500 updates, while RAM remains near its 500-update level. Both baselines finish above the initial RAFA DockQ but below RGT. This sustained trend suggests that the final advantage develops throughout post-training rather than appearing only at the final checkpoint. Because DockQ is evaluated separately from the training reward, the curve also shows that optimizing the proxy reward can translate into improved docking quality.

5 Conclusions

RAFA combines an inherited partial latent representation with antigen-conditioned antibody and nanobody generation. RGT produces RAFA (RL) by using terminal coordinate reward gradients to guide the velocity field at recorded states along the same sampled trajectory. In the reported comparisons, RAFA (RL) improves mean DockQ and design SR.

A key limitation of this work is the absence of wet-lab validation: our results are based entirely on computational evaluations and do not establish experimental binding. As a next step, we plan to experimentally characterize selected antibody and nanobody designs to assess their binding affinity and specificity for the intended antigens.

AI USE STATEMENT

We used generative AI tools to assist with writing and language polishing, literature retrieval and discovery, research ideation and execution, and drafting sections of the manuscript. These tools supported the organization of the paper, refinement of research questions and experimental analyses, identification of potentially relevant prior work, and preparation of code, tables, and figures.

The authors reviewed and revised AI-assisted text, checked references against the original sources, and examined AI-assisted analyses against experimental results. AI-generated suggestions were critically assessed before inclusion, and AI-assisted code and artifacts were checked for correctness and consistency with the underlying data. The authors made the final methodological and interpretive decisions and take responsibility for the final content of this work, including all text, claims, code, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work develops computational methods for antigen-conditioned antibody and nanobody design, with potential applications in therapeutic discovery. Our experiments use publicly available protein structures and sequences and do not involve human participants or animal studies. The evaluations are computational; predicted structures, interface scores, and design success rates do not establish experimental binding, therapeutic efficacy, or safety. Any downstream biomedical application requires experimental validation and appropriate safety assessment.

We recognize that protein-design methods have dual-use potential, including the possibility of generating molecules with harmful biological effects. Responsible use should include assessment of target selection and intended applications, particularly when working with toxins or pathogens. Public structural datasets also reflect uneven coverage of antibody families and antigens, which may limit generalization to underrepresented targets. We therefore report the evaluation conditions and limitations to support appropriate interpretation and use of the results.

REPRODUCIBILITY STATEMENT

We provide anonymized source code to facilitate reproduction of our results, with model-weight download links available in the code repository's README (Appendix A). The model architecture and Reward-Gradient Transport procedure are described in the Methods section, with detailed training objectives in Appendix C.5. Section 4 specifies the training and sampling configurations and baseline comparisons. Data preprocessing and split construction are documented in Appendix B, and evaluation metrics are defined in Appendix E.1. Table captions specify task-dependent evaluation conditions and the interpretation of reported standard deviations. Additional experiments, sampling-step comparisons, and computational costs are provided in Appendix E.

REFERENCES

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*, pp. 1–3, 2024.

Jared Adolf-Bryfogle, Oleks Kalyuzhniy, Michael Kubitz, Brian D Weitzner, Xiaozhen Hu, Yumiko Adachi, William R Schief, and Roland L Dunbrack Jr. Rosettaantibodydesign (rabd): A general framework for computational antibody design. *PLoS computational biology*, 14(4):e1006112, 2018.

Rahmad Akbar, Philippe A. Robert, Cédric R. Weber, Michael Widrich, Robert Frank, Milena Pavlović, Lonneke Scheffer, Maria Chernigovskaya, Igor Snapkov, Andrei Slabodkin, Brij Bhushan Mehta, Enkelejda Miho, Fridtjof Lund-Johansen, Jan Terje Andersen, Sepp Hochreiter, Ingrid Hobæk Haff, Günter Klambauer, Geir Kjetil Sandve, and Victor Greiff. In silico proof of principle of machine learning-based antibody design at unconstrained scale. *mAbs*, 14(1): 2031482, 2022.

Munir Akkaya, Kihyuck Kwak, and Susan K Pierce. B cell memory: building two walls of protection against pathogens. *Nature Reviews Immunology*, 20(4):229–238, 2020.

Sarah Alamdari, Nitya Thakkar, Rianne van den Berg, Alex Xijie Lu, Nicolo Fusi, Ava Pardis Amini, and Kevin K Yang. Protein generation with evolutionary diffusion: sequence is all you need. *bioRxiv*, 2023.

Michael Samuel Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. In *The Eleventh International Conference on Learning Representations*, 2023.

Ethan C. Alley, Grigory Khimulya, Surojit Biswas, Mohammed AlQuraishi, and George M. Church. Unified rational protein engineering with sequence-based deep representation learning. *Nature Methods*, 16(12):1315–1322, 2019.

Namrata Anand and Tudor Achim. Protein structure and sequence generation with equivariant denoising diffusion probabilistic models. In *arXiv preprint arXiv:2205.15019*, 2022.

Ivan Anishchenko, Samuel J Pellock, Tamuka M Chidyausiku, Theresa A Ramelot, Sergey Ovchinnikov, Jingzhou Hao, Khushboo Bafna, Christoffer Norn, Alex Kang, Asim K Bera, et al. De novo protein design by deep network hallucination. *Nature*, 600:547–552, 2021.

Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. In *Advances in Neural Information Processing Systems*, volume 34, pp. 17981–17993, 2021.

Nathaniel R. Bennett, Joseph L. Watson, Robert J. Ragotte, et al. Atomically accurate de novo design of antibodies with rfdiffusion. *Nature*, 649:183–193, 2026. doi: 10.1038/s41586-025-09721-5.

Andreas Bergmeister, Stefanie Jegelka, Nikolas Nüsken, Carles Domingo-Enrich, and Jakiw Pidstrigach. Reinforce adjoint matching: Scaling rl post-training of diffusion and flow-matching models, 2026.

Can Chen, Karla-Luise Herpoldt, Chenchao Zhao, Zichen Wang, Marcus D. Collins, Shang Shang, and Ron Benson. Affinityflow: Guided flows for antibody affinity maturation. In *International Conference on Machine Learning*, 2025a.

Ruizhe Chen, Dongyu Xue, Xiangxin Zhou, Zaixiang Zheng, Xiangxiang Zeng, and Quanquan Gu. An all-atom generative model for designing protein complexes. In *International Conference on Machine Learning*, 2025b.

Yehlin Cho, Martin Pacesa, Zhidian Zhang, Bruno E Correia, and Sergey Ovchinnikov. BoltzDesign1: Inverting all-atom structure prediction model for generalized biomolecular binder design. *bioRxiv*, 2025.

Alexander E Chu, Jinho Kim, Lucy Cheng, Gina El Nesr, Minkai Xu, Richard W Shuai, and Po-Ssu Huang. An all-atom protein generative model. *Proceedings of the National Academy of Sciences*, 121(27):e2311500121, 2024.

Kevin Clark, Paul Vicol, Kevin Swersky, and David J. Fleet. Directly fine-tuning diffusion models on differentiable rewards. In *The Twelfth International Conference on Learning Representations*, 2024.

Justas Dauparas, Ivan Anishchenko, Nathaniel Bennett, Hua Bai, Robert J Ragotte, Lukas F Milles, Basile IM Wicky, Alexis Courbet, Rob J de Haas, Neville Bethel, et al. Robust deep learning–based protein sequence design using proteinmpnn. *Science*, 378(6615):49–56, 2022.

Kieran Didi, Zuobai Zhang, Guoqing Zhou, Danny Reidenbach, Zhonglin Cao, Sooyoung Cha, Tomas Geffner, Christian Dallago, Jian Tang, Michael M. Bronstein, Martin Steinegger, Emine Kucukbenli, Arash Vahdat, and Karsten Kreis. Scaling atomistic protein binder design with generative pretraining and test-time compute. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=qmCpJtFZra.

Carles Domingo-Enrich, Michal Drozdzal, Brian Karrer, and Ricky T. Q. Chen. Adjoint matching: Fine-tuning flow and diffusion generative models with memoryless stochastic optimal control. In *The Thirteenth International Conference on Learning Representations*, 2025.

James Dunbar, Konrad Krawczyk, Jinwoo Leem, Terry Baker, Angelika Fuchs, Guy Georges, Jiye Shi, and Charlotte M Deane. Sabdab: the structural antibody database. *Nucleic acids research*, 42 (D1):D1140–D1146, 2014.

Noelia Ferruz, Steffen Schmidt, and Birte Höcker. ProtGPT2 is a deep unsupervised language model for protein design. *Nature Communications*, 13(1):4348, 2022.

Zhangyang Gao, Cheng Tan, and Stan Z. Li. Pifold: Toward effective and efficient protein inverse folding. In *International Conference on Learning Representations*, 2023.

Tomas Geffner, Kieran Didi, Zuobai Zhang, Danny Reidenbach, Zhonglin Cao, Jason Yim, Mario Geiger, Christian Dallago, Emine Kucukbenli, Arash Vahdat, and Karsten Kreis. Proteina: Scaling flow-based protein structure generative models. In *International Conference on Learning Representations*, 2025.

Tomas Geffner, Kieran Didi, Zhonglin Cao, Danny Reidenbach, Zuobai Zhang, Christian Dallago, Emine Kucukbenli, Karsten Kreis, and Arash Vahdat. La-proteina: Atomistic protein generation via partially latent flow matching. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=RDerF20JYT.

George Georgiou, Gregory C Ippolito, John Beausang, Christian E Busse, Hedda Wardemann, and Stephen R Quake. The promise and challenge of high-throughput sequencing of the antibody repertoire. *Nature Biotechnology*, 32(2):158–168, 2014.

Xiaoxuan He, Siming Fu, Yuke Zhao, Wanli Li, Jian Yang, Dacheng Yin, Fengyun Rao, and Bo Zhang. Tempflow-grpo: When timing matters for grpo in flow models, 2025.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Chloe Hsu, Robert Verkuil, Jason Liu, Zeming Lin, Brian Hie, Tom Sercu, Adam Lerer, and Alexander Rives. Learning inverse folding from millions of predicted structures. In *International Conference on Machine Learning*, pp. 8946–8970. PMLR, 2022.

John B Ingraham, Max Baranov, Zak Costello, Karl W Barber, Wujie Wang, Ahmed Ismail, Vincent Frappier, Dana M Lord, Christopher Ng-Thow-Hing, Erik R Van Vlack, et al. Illuminating protein space with a programmable generative model. *Nature*, 623(7989):1070–1078, 2023.

Wengong Jin, Jeremy Wohlwend, Regina Barzilay, and Tommi Jaakkola. Iterative refinement graph neural network for antibody sequence-structure co-design. *arXiv preprint arXiv:2110.04624*, 2021.

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.

Xiangzhe Kong, Wenbing Huang, and Yang Liu. Conditional antibody design as 3d equivariant graph translation. In *The Eleventh International Conference on Learning Representations*, 2023a.

Xiangzhe Kong, Wenbing Huang, and Yang Liu. End-to-end full-atom antibody design. In *International Conference on Machine Learning*, pp. 17409–17429. PMLR, 2023b.

Gideon D Lapidoth, Dror Baran, Gabriele M Pszolla, Christoffer Norn, Assaf Alon, Michael D Tyka, and Sarel J Fleishman. AbDesign: An algorithm for combinatorial backbone design guided by natural conformations and sequences. *Proteins: Structure, Function, and Bioinformatics*, 83(8): 1385–1406, 2015.

Marie-Paule Lefranc, Christelle Pommié, Manuel Ruiz, Véronique Giudicelli, Elodie Foulquier, Lisa Truong, Valérie Thouvenin-Contet, and Gérard Lefranc. Imgt unique numbering for immunoglobulin and t cell receptor variable domains and ig superfamily v-like domains. *Developmental & Comparative Immunology*, 27(1):55–77, 2003.

Marie-Paule Lefranc, Veronique Giudicelli, Chantal Ginestoux, Joumana Jabado-Michaloud, Geraldine Folch, Fatena Bellahcene, Yan Wu, Elodie Gemrot, Xavier Brochet, Jeroˆme Lane, et al. Imgt®, the international immunogenetics information system®. *Nucleic acids research*, 37 (suppl_1):D1006–D1012, 2009.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023.

Gary W Litman, Jonathan P Rast, Michael J Shamblott, Robert N Haire, Michele Hulst, William Roess, Ronda T Litman, Kristin R Hinds-Frey, Anna Zilch, and Chris T Amemiya. Phylogenetic diversification of immunoglobulin genes and the antibody repertoire. *Molecular Biology and Evolution*, 10(1):60–72, 1993.

Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-grpo: Training flow matching models via online rl. *arXiv preprint arXiv:2505.05470*, 2025a.

Runtao Liu, Haoyu Wu, Zheng Ziqiang, Chen Wei, Yingqing He, Renjie Pi, and Qifeng Chen. VideoDPO: Omni-preference alignment for video diffusion generation. *arXiv preprint arXiv:2412.14167*, 2024.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations*, 2023.

Zhen Liu, Tim Z. Xiao, Carles Domingo-Enrich, Weiyang Liu, and Dinghuai Zhang. Value gradient guidance for flow matching alignment. In *Advances in Neural Information Processing Systems*, 2025b.

Tianyu Lu, Richard Shuai, Petr Kouba, Zhaoyang Li, Yilin Chen, Akio Shirali, Jinho Kim, and Po-Ssu Huang. Conditional protein structure generation with Protpardelle-1C. *bioRxiv*, 2025.

Shitong Luo, Yufeng Su, Xingang Peng, Sheng Wang, Jian Peng, and Jianzhu Ma. Antigen-specific antibody design and optimization with diffusion-based generative models for protein structures. *Advances in Neural Information Processing Systems*, 35:9754–9767, 2022.

Isaac D Lutz, Shunzhi Wang, Christoffer Norn, Alexis Courbet, Andrew J Borst, Yan Ting Zhao, Annie Dosey, Longxing Cao, Jinwei Xu, Elizabeth M Leaf, et al. Top-down design of protein architectures with reinforcement learning. *Science*, 380(6642):266–273, 2023.

Ali Madani, Ben Krause, Eric R Greene, Subu Subramanian, Benjamin P Mohr, James M Holton, Jose Luis Olmos, Caiming Xiong, Zachary Z Sun, Richard Socher, et al. Large language models generate functional protein sequences across diverse families. *Nature Biotechnology*, 41(8): 1099–1106, 2023.

Luis Santiago Mille-Fragoso, John N. Wang, Claudia L. Driscoll, Haoyu Dai, Talal M. Widatalla, Xiaowe Zhang, Brian L. Hie, and Xiaojing J. Gao. Efficient generation of epitope-targeted de novo antibodies with germinal. *bioRxiv*, 2025. URL https://www.biorxiv.org/content/10.1101/2025.09.19.677421v1.

Alex Morehead and Jianlin Cheng. Flowdock: Geometric flow matching for generative protein-ligand docking and affinity prediction. In *Intelligent Systems for Molecular Biology (ISMB)*, 2025.

Saul B Needleman and Christian D Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of molecular biology*, 48(3):443–453, 1970.

Aaron L Nelson, Eugen Dhimolea, and Janice M Reichert. Development trends for human monoclonal antibody therapeutics. *Nature Reviews Drug Discovery*, 9(10):767–774, 2010.

Tobias H Olsen, Iain H Moal, and Charlotte M Deane. Ablang: an antibody language model for completing antibody sequences. *Bioinformatics Advances*, 2(1):vbac046, 2022.

Martin Pacesa, Lennart Nickel, Christian Schellhaas, Joseph Schmidt, Ekaterina Pyatova, Lucas Kissling, Patrick Barendse, et al. One-shot design of functional protein binders with BindCraft. *Nature*, 2025.

Robert J. Pantazes and Costas D. Maranas. OptCDR: a general computational method for the design of antibody complementarity determining regions for targeted epitope binding. *Protein Engineering, Design and Selection*, 23(11):849–858, 2010.

Saro Passaro, Gabriele Corso, Jeremy Wohlwend, et al. Boltz-2: Towards accurate and efficient binding affinity prediction. *bioRxiv*, 2025.

Zhangzhi Peng, Chenchen Han, Xiaohan Wang, Dapeng Li, and Fajie Yuan. Generative diffusion models for antibody design, docking, and optimization. *bioRxiv*, 2023. URL https://www.biorxiv.org/content/early/2023/09/26/2023.09.25.559190.

Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. FiLM: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018.

Jeffrey A. Ruffolo, Jeffrey J. Gray, and Jeremias Sulam. Deciphering antibody affinity maturation with language models and weakly supervised learning. *arXiv preprint arXiv:2112.07782*, 2021.

Jeffrey A. Ruffolo, Lee-Shin Chu, Sai Pooja Mahajan, and Jeffrey J. Gray. Fast, accurate antibody structure prediction from deep learning on massive set of natural antibodies. *Nature Communications*, 14(1):2389, 2023.

Koichiro Saka, Taro Kakuzaki, Shoichi Metsugi, Daiki Kashiwagi, Kenji Yoshida, Manabu Wada, Hiroyuki Tsunoda, and Reiji Teramoto. Antibody design using lstm based deep generative model from phage display library for affinity maturation. *Scientific reports*, 11(1):5852, 2021.

Harry W Schroeder Jr and Lisa Cavacini. Structure and function of immunoglobulins. *Journal of Allergy and Clinical Immunology*, 125(2):S41–S52, 2010.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Mingchuan Zhang, Y.K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

Chence Shi, Chuanrui Wang, Jiarui Lu, Bozitao Zhong, and Jian Tang. Protein sequence and structure co-design with equivariant translation. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=pRCMXcfdihq.

Jung-Eun Shin, Adam J Riesselman, Aaron W Kollasch, Conor McMahon, Elana Simon, Chris Sander, Aashish Manglik, Andrew C Kruse, and Debora S Marks. Protein design and variant prediction using autoregressive generative models. *Nature Communications*, 12(1):2403, 2021.

Dee Unglaub Silverthorn. *Human Physiology: An Integrated Approach*. Jones & Bartlett Publishers, 2015.

Mark X Sliwkowski and Ira Mellman. Antibody therapeutics in cancer. *Science*, 341(6151):1192–1198, 2013.

Yang Song and Prafulla Dhariwal. Improved techniques for training consistency models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=WNzy9bRDvG.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *International Conference on Machine Learning*, pp. 32211–32252. PMLR, 2023.

Samuel Stanton, Wesley Maddox, Nate Gruver, Phillip Maffettone, Emily Delaney, Peyton Greenside, and Andrew Gordon Wilson. Accelerating bayesian optimization for biological sequence design with denoising autoencoders. *arXiv preprint arXiv:2203.12742*, 2022.

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 Hnisz, David M. Sabatini, William F. DeGrado, Jeremy Wohlwend, Gabriele Corso, Regina Barzilay, and Tommi Jaakkola. Boltzgen: Toward universal binder design. *bioRxiv*, 2025. doi: 10.1101/2025.11.20.689494.

Cheng Tan, Yijie Zhang, Zhangyang Gao, Yufei Huang, Haitao Lin, Lirong Wu, Fandi Wu, Mathieu Blanchette, and Stan Z Li. dyab: Flow matching for flexible antibody design with alphafold-driven pre-binding antigen. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025.

BioGeometry Team and Jian Tang. Rapid de novo antibody design with geoflow-v3. *bioRxiv*, 2025. URL https://www.biorxiv.org/content/early/2025/10/21/2025.10.20.682964.

Mihaly Varadi, Stephen Anyango, Mandar Deshpande, Sreenath Nair, Cindy Natassia, Galabina Yordanova, David Yuan, Oana Stroe, Gemma Wood, Agata Laydon, Augustin Žídek, Tim Green, Kathryn Tunyasuvunakool, Stig Petersen, John Jumper, Ellen Clancy, Richard Green, Ankur Vora, Mira Lutfi, Michael Figurnov, Andrew Cowie, Nicole Hobbs, Pushmeet Kohli, Gerard Kleywegt, Ewan Birney, Demis Hassabis, and Sameer Velankar. Alphafold protein structure database: massively expanding the structural coverage of protein-sequence space with high-accuracy models. *Nucleic Acids Research*, 50(D1):D439–D444, 2022.

Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. Diffusion model alignment using direct preference optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Jue Wang, Sidney Lisanza, David Juergens, Doug Tischer, Joseph L Watson, Karla M Castro, Robert Ragotte, Amijai Saragovi, Lukas F Milles, Minkyung Baek, et al. Scaffolding protein functional sites using deep learning. *Science*, 377(6604):387–394, 2022.

Rubo Wang, Fandi Wu, Xingyu Gao, Jiaxiang Wu, Peilin Zhao, and Jianhua Yao. IgGM: A generative model for functional antibody and nanobody design. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=zmmfsJpYcq.

Ruoyu Wang, Boye Niu, Xiangxin Zhou, Yushi Huang, Tongliang Liu, and Chi Zhang. Exploring the design space of reward backpropagation for flow matching. *arXiv preprint arXiv:2606.11075*, 2026a.

Wenda Wang, Yang Zhang, Zhewei Wei, and Wenbing Huang. Abflow: End-to-end paratope-centric antibody design by interaction enhanced flow matching. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2026b.

Shira Warszawski, Alon Borenstein Katz, Rosalie Lipsh, Lev Khmelnitsky, Gili Ben Nissan, Gali Javitt, Orly Dym, Tamar Unger, Orly Knop, Shira Albeck, et al. Optimizing antibody affinity and stability by the automated design of the variable light-heavy chain interfaces. *PLOS Computational Biology*, 15(8):e1007207, 2019.

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.

George J Weiner. Building better monoclonal antibody-based therapeutics. *Nature Reviews Cancer*, 15(6):361–370, 2015.

Jeremy Wohlwend, Gabriele Corso, Saro Passaro, Noah Getz, Mateo Reveiz, Ken Leidal, Wojtek Swiderski, et al. Boltz-1: Democratizing biomolecular interaction modeling. *bioRxiv*, 2025.

Fandi Wu, Yu Zhao, Jiaxiang Wu, et al. De novo design of epitope-specific antibodies via a structure-driven computational workflow. *Nature Communications*, 17:625, 2026. doi: 10.1038/s41467-025-67361-9.

Fang Wu and Stan Z Li. A hierarchical training paradigm for antibody structure-sequence co-design. *Advances in Neural Information Processing Systems*, 36, 2024.

Jialong Wu, Shaofeng Yin, Ningya Feng, and Mingsheng Long. RLVR-World: Training world models with reinforcement learning. In *Advances in Neural Information Processing Systems*, 2025a.

Jun Wu, Xiangzhe Kong, Ningguan Sun, Jing Wei, Sisi Shan, Fuli Feng, Feng Wu, Jian Peng, Linqi Zhang, Yang Liu, and Jianzhu Ma. Flowdesign: Improved design of antibody cdrs through flow matching and better prior distributions. *Cell Systems*, 16, 2025b.

Xiaoshi Wu, Yiming Hao, Manyuan Zhang, Keqiang Sun, Zhaoyang Huang, Guanglu Song, Yu Liu, and Hongsheng Li. Deep reward supervisions for tuning text-to-image diffusion models. In *European Conference on Computer Vision*, 2024.

Ziyi Wu, Anil Kag, Ivan Skorokhodov, Willi Menapace, Ashkan Mirzaei, Igor Gilitschenski, Sergey Tulyakov, and Aliaksandr Siarohin. DenseDPO: Fine-grained temporal preference optimization for video diffusion models. *NeurIPS*, 2025c.

Zeyue Xue, Jie Wu, Yu Gao, Fangyuan Kong, Lingting Zhu, Mengzhao Chen, Zhiheng Liu, Wei Liu, Qiushan Guo, Weilin Huang, and Ping Luo. DanceGRPO: Unleashing GRPO on visual generation. *arXiv preprint arXiv:2505.07818*, 2025.

Yumeng Yan, Huanyu Tao, Jiahua He, and Sheng-You Huang. The HDOCK server for integrated protein–protein docking. *Nature Protocols*, 15(5):1829–1852, 2020.

Zhuo Yang, Jiaying He, Jiaqing Xie, Daolang Wang, Xipeng Qiu, Yuxin Wang, Tianfan Fu, and Beilun Wang. Abopd: Antibody cdr design via on-policy distillation. In *arXiv preprint arXiv:2607.18835*, 2026.

Vinicius Zambaldi, David La, Alexander E Chu, Harshnira Patani, Amy E Danson, Tristan O C Kwan, Thomas Frerix, Rosalia G Schneider, David Saxton, Ashok Thillaisundaram, et al. De novo design of high-affinity protein binders with AlphaProteo. Google DeepMind Technical Report, 2024.

Jiying Zhang, Zijing Liu, Shengyuan Bai, He Cao, Yu Li, and Lei Zhang. Efficient antibody structure refinement using energy-guided SE(3) flow matching. In *2024 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, 2024.

Kaiwen Zheng, Huayu Chen, Haotian Ye, Haoxiang Wang, Qinsheng Zhang, Kai Jiang, Hang Su, Stefano Ermon, Jun Zhu, and Ming-Yu Liu. Diffusionnft: Online diffusion reinforcement with forward process. *arXiv preprint arXiv:2509.16117*, 2025.

Xiangxin Zhou, Dongyu Xue, Ruizhe Chen, Zaixiang Zheng, Liang Wang, and Quanquan Gu. Antigen-specific antibody design via direct energy-based preference optimization. In *Advances in Neural Information Processing Systems*, 2024.

Tian Zhu, Milong Ren, and Haicang Zhang. Antibody design using a score-based diffusion model guided by evolutionary, physical and geometric constraints. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=1YsQI04KaN.

A Code and Model Weights

The source code for RAFA, including model architecture, training scripts, and evaluation pipelines, is available at:

`https://anonymous.4open.science/r/RAFA-9C32`

Model weights are available through the download links in the code repository's README. We provide a single RGT post-trained checkpoint for both antibody (VH–VL) and nanobody (VHH) prediction and design. This is the shared checkpoint used for the main RAFA (RL) results; no separate task-specific post-trained weights are used for these four tasks.

B Dataset Curation and Filtering

Source and temporal split. All antibody-antigen complexes are sourced from SAbDab (Dunbar et al., 2014). Structures deposited on or before 2025-06-30 form the train/validation pool; structures deposited strictly after this date form the candidate test pool. The reported training and validation splits contain 12,364 and 252 complexes, respectively (12,616 combined). The final test set contains 143 complexes, giving 12,759 complexes in total, as reported in Section 4.

Upstream pretraining data. The AFDB data and upstream checkpoints used for autoencoder pretraining and general-protein training derive from data available before July 2022. This upstream data cutoff is separate from the 2025-06-30 temporal split of the antibody–antigen dataset. Subsequent antibody-specific training uses the antibody training split described above. The temporal cutoffs and the chain-level filtering rule below specify distinct restrictions on the evaluation data; neither alone establishes antigen-family disjointness.

Reward-model data separation. The fold-confidence reward uses a frozen surrogate calibrated on Boltz-2-derived labels. Boltz-2's experimental PDB structure-training set contains structures released before 2023-06-01 (Passaro et al., 2025). Our test complexes were deposited strictly after 2025-06-30 and therefore could not occur in that experimental PDB training set. Surrogate calibration uses only the RAFA training split, with no validation or test samples used for fitting. These restrictions prevent direct test-structure leakage through Boltz-2's experimental PDB training data and held-out-sample leakage through surrogate fitting. The structural-data cutoff is distinct from the homology filtering described below. Sequence-refolding evaluation uses Protenix-v2, a different predictor from the Boltz-2 model supplying reward-calibration labels.

Preprocessing. Each complex undergoes a six-step preprocessing pipeline: (1) multi-antigen complexes are split into one sample per antigen chain; (2) FASTA files are generated for all chains; (3) merged H+L chains are detected and split using sequence anchoring; (4) PDB and FASTA sequences are reconciled (PDB is ground truth); (5) train/valid/test split indices are written; (6) all samples are validated for chain completeness and sequence consistency. Samples failing these preprocessing checks are excluded during dataset construction. Once the test sets are finalized, all methods are evaluated on the same complete target lists, with no targets excluded because of method failures.

Sequence-based redundancy filtering for the test set. To reduce sequence redundancy between training and test data, candidate test complexes are filtered against the full training split. For each candidate, both the antigen sequence(s) and the antibody sequence(s) are compared against every training sequence of the corresponding chain type using global pairwise alignment (Needleman & Wunsch, 1970) with match score $+2$, mismatch score -1, gap open -0.5, and gap extend -0.1. For sequences q and r, the dimensionless normalized global-alignment score is

$$S(q, r) = \frac{\text{alignment score}(q, r)}{2 \max(|q|, |r|)}. \tag{10}$$

Here $|q|$ and $|r|$ are sequence lengths and 2 is the per-residue match score. Let $\mathcal{Q}$ contain all antigen and antibody chains of the candidate complex, and let $\mathcal{T}(q)$ contain training sequences of the

corresponding chain type. A candidate is retained only if

$$S(q, r) \leq 0.8 \quad \text{for every } q \in \mathcal{Q} \text{ and every } r \in \mathcal{T}(q). \tag{11}$$

Thus, a score greater than 0.8 for any compared chain pair excludes the entire candidate complex; equality at the threshold is retained. The threshold is 0.8 on the dimensionless scale in Eq. equation 10, consistently with Section 4. Expressed as a percentage of the normalization denominator, the same score is 80%; it is not an 80% sequence-identity cutoff, because mismatches, gap penalties, and sequence lengths affect the alignment score. After filtering, the final test set contains 93 conventional antibodies and 50 nanobodies (143 total).

Sequence identity and scope of generalization. The identity strata in Appendix E.2 describe the retained test set using a separate MMseqs2 analysis. For each test antigen, the hit with the highest bitscore is selected, ties are broken by identity, and its alignment identity is reported without a coverage cutoff. These identity bins are distinct from the global-alignment score used to construct the test set. The reported evaluation therefore concerns temporally held-out complexes satisfying the stated chain-level score criterion, with performance additionally examined across antigen-identity strata. These criteria do not by themselves establish the absence of homologous antigens or an antigen-family-disjoint benchmark.

C Method Details

This appendix provides the partial autoencoder architecture, implementation-level definitions for the objectives in the main paper, and the full training objective.

C.1 Partial Autoencoder Architecture

We initialize the partially latent AE from La-Proteina (Geffner et al., 2026) and fine-tune it on the antibody training split. Its encoder receives sequence and full-atom geometry, while explicit Cα coordinates bypass the latent bottleneck. The stochastic encoder samples per-residue latents from a learned Gaussian distribution regularized by a KL term.

Encoder. The encoder is a 12-layer transformer with token dimension 768, pair representation dimension 256, and 12 attention heads ($\sim$130M parameters). Input features per residue include: chain-break indicator, amino acid type (one-hot), all-atom coordinates (37-atom representation, in nm), relative all-atom coordinates, backbone dihedral angles (ϕ, ψ, ω), and side-chain torsion angles. Pair features include relative sequence separation, backbone pairwise distances, and backbone pair orientations. The final hidden state is projected to an 8-dimensional mean and log-scale. The encoder samples $\mathbf{z}_i = \boldsymbol{\mu}_i + \exp(\boldsymbol{\ell}_i) \odot \boldsymbol{\epsilon}_i$, with $\boldsymbol{\epsilon}_i \sim \mathcal{N}(0, I)$. The C_α coordinates $\mathbf{x}^{ab}$ are extracted directly from the input and passed through unchanged; they are not encoded into $\mathbf{z}$.

Decoder. The decoder is a symmetric 12-layer transformer with the same hidden dimensions. Its inputs are the latent vectors $\mathbf{z}$ and the explicit C_α coordinates $\mathbf{x}^{ab}$. Pair features are computed from $\mathbf{x}^{ab}$ pairwise distances. The decoder outputs (i) per-residue all-atom coordinate offsets relative to the C_α frame, reconstructing full-atom geometry, and (ii) per-residue amino acid logits, reconstructing sequence identity.

Training. The AE is pre-trained on the AlphaFold Database (AFDB) (Jumper et al., 2021; Varadi et al., 2022), with upstream pretraining data dating to before July 2022 (Appendix B), and then fine-tuned on the antibody training split with a combined loss:

$$\mathcal{L}_{\text{AE}} = \mathcal{L}_{\text{struct}} + \mathcal{L}_{\text{seq}} + \beta\, \mathcal{L}_{\text{KL}} \tag{12}$$

where $\mathcal{L}_{\text{struct}}$ is an L2 loss on reconstructed all-atom coordinates, $\mathcal{L}_{\text{seq}}$ is cross-entropy on amino acid logits, and $\mathcal{L}_{\text{KL}}$ is a KL regularizer with weight $\beta = 10^{-4}$ (regularizing the stochastic latent distribution). The AE is frozen before any flow matching training begins.

C.2 Reward-Gradient Transport Details

RGT uses states recorded from the current model's stochastic rollouts. Terminal and intermediate states are detached from the sampling graph. Reward differentiation is performed with respect to terminal state leaves; parameter differentiation uses fresh velocity evaluations at selected recorded states.

Gradient normalization and time selection. Reward gradients are masked to antibody residues and normalized jointly across coordinate and latent modes. Unused gradients are replaced with zeros. Under the current reward, the direct local-latent gradient is zero, so the norm reduces to the coordinate gradient norm. Nonzero gradients are scaled to norm 0.5, with a numerical denominator floor. Four evenly spaced trajectory indices $[0, 16, 33, 49]$ are selected from each 50-step rollout. Their recorded coordinate times determine the normalized linear transport weights $\widetilde{w}_{k,j} = t_{x,k,j}/(J^{-1}\sum_{\ell=1}^{J} t_{x,k,\ell})$, as in Eq. equation 8.

Detached-target interpretation. For coordinate velocity v and detached transported direction a, define

$$\mathcal{L}_{\text{target}} = \tfrac{1}{2}\|v - \text{sg}(v + a)\|^2_{\text{Ab}}. \tag{13}$$

Its parameter gradient at the current evaluation equals that of $-\langle v, \text{sg}(a)\rangle_{\text{Ab}}$, providing a detached-target regression interpretation of the velocity-direction update. The training loss averages contributions from the selected states along each sampled trajectory.

Training configuration. Each update uses eight fresh trajectories, 50 simulation steps per trajectory, and four recorded states per trajectory. The learning rate is 5×10^{-7} and model gradient-norm clip is 1. The AE and pair-update modules are frozen. The final RAFA (RL) model uses 2,000 parameter updates. Final generation uses 400 sampling steps. RGT uses the identity approximation to future state sensitivity and differentiates fresh velocity evaluations at the recorded states.

C.3 Reward Components and Differentiability

Equation equation 6 defines the composite reward. For the composite reward in Eq. equation 6, we set $(w_1, w_2, w_3, w_4) = (0.60, 0.20, 0.15, 0.05)$ for affinity, fold confidence, physical plausibility, and developability, respectively. The affinity component combines the geometric proxy and frozen sequence predictor with weights 0.75 and 0.25; its geometric contact-coverage, burial, and compactness terms have weights 0.60, 0.25, and 0.15. The reward predictors remain frozen during RGT. Below, all geometric distances are in nm, σ denotes the sigmoid, and $[u]_+ = \max(u, 0)$. Native antibody coordinates, native contacts, DockQ, and SR are not inputs to these scores.

Affinity-related reward. Let D denote generated CDR residues and E the supplied antigen epitope. For their Cα distances d_{ij}, define nearest-neighbor distances $d_i = \min_{j\in E} d_{ij}$ and $e_j = \min_{i\in D} d_{ij}$. Soft contact indicators are $q_i = \sigma(12(0.8 - d_i))$ and $r_j = \sigma(12(0.8 - e_j))$. Contact coverage and the buried-interface proxy are

$$C = \frac{1}{2}\left(\frac{\sum_{i\in D} q_i}{|D|} + \frac{\sum_{j\in E} r_j}{|E|}\right), \qquad B = \tanh\left(\frac{\sum_i q_i + \sum_j r_j}{24}\right). \tag{14}$$

The burial term measures soft contacting-residue counts, not solvent-accessible surface area. With $Q = \sum_i q_i$, define the contact-weighted center $\bar{\mathbf{x}} = \sum_i q_i \mathbf{x}_i / \max(Q, 10^{-8})$ and spread $s = \sqrt{\sum_i q_i \|\mathbf{x}_i - \bar{\mathbf{x}}\|^2 / \max(Q, 10^{-8}) + 10^{-8}}$. Interface compactness is $I = \exp(-s/1.2)\tanh(Q/2)$, so a compact but distant antibody receives little interface reward.

For all antibody and antigen residues, let $\tilde{d}_i$ and $\tilde{e}_j$ be the corresponding nearest cross-interface distances. The soft clash fraction and steric gate are

$$f_{\text{clash}} = \frac{1}{2}\left(\text{mean}_i\, \sigma(12(0.35 - \tilde{d}_i)) + \text{mean}_j\, \sigma(12(0.35 - \tilde{e}_j))\right), \tag{15}$$

$$G = \exp\left(-12[f_{\text{clash}} - 0.05]_+\right). \tag{16}$$

An antibody radius-of-gyration prior is $P = \exp(-(r_g - 1.8)^2/(2 \cdot 0.6^2))$. The geometric score is $R_{\text{geom}} = (\alpha_C C + \alpha_B B + \alpha_I I)GP$. The affinity component combines this score with a frozen sequence predictor T_{seq}, evaluated on decoded CDR identities and the antigen sequence:

$$R_{\text{aff}} = (1 - \lambda)R_{\text{geom}} + \lambda T_{\text{seq}}. \tag{17}$$

If no CDR residues are available, the geometric score uses all antibody residues; if no epitope is available, it uses all antigen residues. The sequence predictor is calibrated on affinity measurements, but the combined score remains an affinity-related proxy rather than a measured binding affinity.

Fold-confidence reward. A frozen ridge-regression surrogate calibrated on Boltz-2-derived confidence labels (Passaro et al., 2025) maps generated-complex features $\boldsymbol{\phi}$ to

$$R_{\text{fold}} = \text{clip}_{[0,1]}\left[b + \boldsymbol{\beta}^\top \text{clip}_{[-5,5]}\left((\boldsymbol{\phi} - \boldsymbol{\mu})/\boldsymbol{\sigma}\right)\right]. \tag{18}$$

The surrogate is fitted exclusively on calibration samples from the RAFA training split; neither validation nor test samples are used for fitting. The checkpoint supplies the standardization statistics and fitted coefficients. Its 12 features comprise antibody and antigen contact coverage, their mean, contact count per antibody residue (implemented as antibody contact coverage), mean nearest-interface distance (the minimum of the two directional means), clash fraction, interface compactness, antibody radius of gyration, backbone continuity, bond-length deviation, and normalized antibody and antigen lengths. Contact and clash features here use hard distance thresholds; backbone continuity measures the fraction of consecutive Cα distances between 0.32 and 0.45, and bond deviation is measured relative to 0.38. Continuity and deviation are averaged over valid chain segments. This surrogate avoids running Boltz-2 during each reward evaluation. Boltz-2's experimental PDB structure-training data were released before 2023-06-01 (Passaro et al., 2025), preceding our test-set deposition cutoff of 2025-06-30. Thus, our test structures are excluded from that PDB training set, and held-out samples do not enter surrogate fitting (Appendix B).

Physical-plausibility reward. The physical term reuses the steric gate and soft clash fraction:

$$R_{\text{physical}} = \tfrac{1}{2}G + \tfrac{1}{2}\left(1 - \text{clip}_{[0,1]}(f_{\text{clash}})\right). \tag{19}$$

It penalizes antibody–antigen interpenetration at the Cα level rather than evaluating an all-atom energy.

Developability reward. For the decoded CDR sequence, let f_C be the cysteine fraction, f_{MW} the combined methionine/tryptophan fraction, and f_H the fraction in the implementation's hydrophobic set $\{A, C, I, L, M, F, W, Y\}$. Let $m_{NXS/T}$ count contiguous N–X–S/T triplets divided by the CDR residue count, with X unrestricted. Then

$$R_{\text{dev}} = \exp\left[-\left(2f_C + f_{MW} + 1.5m_{NXS/T} + [f_H - 0.55]_+\right)\right]. \tag{20}$$

Triplets spanning noncontiguous CDR positions are excluded. This bounded sequence-liability screen is a developability proxy, not a clinical or manufacturability prediction.

C.4 Differentiable Recycling: Derivation

Intra-layer clean sample prediction. At each transformer trunk layer l, the current hidden state $\mathbf{h}_{ab}^{(l)} \in \mathbb{R}^{N_{ab} \times d}$ is leveraged to predict a clean C_α structural state via:

$$\hat{\mathbf{x}}_1^{(l)} = \mathbf{x}_{t_x,\,ab} + (1 - t_x) \cdot \text{Linear}_3(\mathbf{h}_{ab}^{(l)}), \tag{21}$$

where $\mathbf{x}_{t_x,\,ab} \in \mathbb{R}^{N_{ab} \times 3}$ is the explicit C_α coordinate component of $\mathbf{x}_{t_x}$, and $\text{Linear}_3 : \mathbb{R}^d \to \mathbb{R}^3$ projects high-dimensional embeddings into 3D coordinate space. This formulation is a direct extension of the standard flow matching $\mathbf{x}_1$ prediction utilizing the velocity parameterization: given $\mathbf{v}_\theta \approx \mathbf{x}_1 - \mathbf{x}_0$ and $\mathbf{x}_{t_x} = (1 - t_x)\mathbf{x}_0 + t_x\mathbf{x}_1$, the identity $\hat{\mathbf{x}}_1 = \mathbf{x}_{t_x} + (1 - t_x)\mathbf{v}_\theta$ holds.

Geometric feedback. Pairwise distances from the intermediate prediction $\hat{\mathbf{x}}_1^{(l)}$ are encoded via Gaussian RBFs uniformly spaced across $D = 16$ centers:

$$\phi_{ij}^{(l)} = \exp\Big(-\frac{(d_{ij}^{(l)} - \mu_k)^2}{2\sigma^2}\Big)_{k=1}^{D} \in \mathbb{R}^D, \tag{22}$$

where $d_{ij}^{(l)} = \|\hat{\mathbf{x}}_{1,i}^{(l)} - \hat{\mathbf{x}}_{1,j}^{(l)}\|_2$. We denote the pair representation at layer l by $\mathbf{p}_{ij}^{(l)}$, reserving $\mathbf{z}_i$ for the per-residue latent. The RBF features are projected to the pair dimension and injected through a coordinate-time-conditioned AdaLN module:

$$\mathbf{p}_{ij}^{(l+1)} = \mathrm{AdaLN}\big(\mathbf{p}_{ij}^{(l)} + \mathrm{Linear}(\phi_{ij}^{(l)}),\ t_x\big), \tag{23}$$

where $\mathrm{AdaLN}(\mathbf{p}, t_x) = \gamma(t_x) \odot \mathrm{LN}(\mathbf{p}) + \beta(t_x)$, utilizing learned time-dependent scale and shift parameters. This explicitly allows each layer to self-correct its geometric trajectory dynamically based on the evolving structural prediction, entirely circumventing the need for a separate, computationally heavy recycling forward pass.

C.5 Full Training Objective

Training RAFA involves evaluating student and teacher branches with a shared trunk. The student uses masked CDR sequences and inference-available structural conditions. The teacher additionally receives training-only target features, including clean coordinates and latents. Both branches receive FM and auxiliary supervision:

$$\mathcal{L}_{\text{base}} = \mathcal{L}_{\text{FM}}^{S} + \mathcal{L}_{\text{aux}}^{S} + \mathcal{L}_{\text{FM}}^{T} + \mathcal{L}_{\text{aux}}^{T} + 0.5\mathcal{L}_{\text{distill}}. \tag{24}$$

The teacher prediction is detached only in the distillation target; teacher supervision still trains the shared parameters. The privileged teacher adapter is disabled at inference.

Velocity distillation. For each mode m, dimension d_m, and residue mask M, define

$$\mathcal{L}_{\text{distill}} = \mathbb{E}\left[\mathbf{1}[t_x \geq 0.4] \sum_{m \in \{x,z\}} \frac{\sum_i M_i \|v_i^{S,m} - \mathrm{sg}(v_i^{T,m})\|_2^2}{\max(N_M d_m, 1)}\right]. \tag{25}$$

The residue-feature normalization here differs from the FM term, which sums features and divides by residue count.

Auxiliary Cα geometry. For either branch, $\mathcal{L}_{\text{aux}} = \mathbb{E}[\omega(t_x) \sum_\ell \lambda_\ell \mathcal{L}_\ell]$, with $\omega(t_x) = \mathbf{1}[t_x > 0.4]t_x$. The losses use predicted clean Cα coordinates rather than decoded all-atom coordinates:

- Virtual consecutive Cα bond lengths (approximately 3.8 Å), Cα pseudo-bond angles, and pseudo-dihedrals.
- Paratope–epitope distance supervision, full-antibody aligned RMSD, and interface alignment.
- Soft native-contact recovery and an antibody–antigen Cα steric penalty.
- Contact-head BCE and log-distance regression over antibody–epitope pairs.

Native antibody geometry is valid supervision in base training; it is not supplied to the RGT reward.

Contact head and weights. With antibody–epitope pair mask P_{ij} and $Z = \max(\sum_{ij} P_{ij}, 1)$, the contact loss is

$$\mathcal{L}_c = \frac{1}{Z} \sum_{ij} P_{ij} \left[\mathrm{BCE}_{20}(\hat{c}_{ij}, \mathbf{1}[d_{ij}^* < 0.8\,\mathrm{nm}]) + \mathrm{SmoothL1}(\hat{\ell}_{ij}, \log \max(d_{ij}^*, 10^{-3}))\right]. \tag{26}$$

Here distances are in nm and the BCE positive weight is 20. Both terms receive the same contact weight and time gate. The training objective uses weights 1.0 (bond), 1.0 (angle), 0.5 (dihedral), 0.5 (interface distance), 1.0 (contact), 1.0 (full-antibody alignment), 1.0 (interface alignment), 1.0 (soft contact recovery), and 0.5 (steric clash). These training losses are distinct from the native-antibody-free RGT reward.

D RGT: Sensitivity Analysis and Local Objective

Pathwise reward sensitivity. The differentiable reward used by RGT acts on explicit antibody coordinates. In this analysis, decoded discrete identities are held fixed, as in the implemented gradient path. Let $y_n = (x_n, z_n)$ denote the flattened antibody state at sampler index n, with conditions fixed, and write a realized sampling step as

$$y_{n+1} = F_n(y_n; \theta, \xi_n), \qquad n = 0, \dots, S-1. \tag{27}$$

The noise draws ξ_n are held fixed for pathwise differentiation. Assume that the step maps and the selected reward branch are differentiable at the evaluated states; hard thresholds, argmax changes, and distance ties are excluded from this local analysis. Let $A_n = \partial F_n / \partial y_n$, $g = \nabla_{y_S} R = (g_x, 0)$, and

$$P_{n:S} = A_{S-1} \cdots A_n, \qquad p_n = P_{n:S}^\top g. \tag{28}$$

Here p_n is the exact pathwise terminal-reward sensitivity to an intermediate state. Coordinate–latent interactions during sampling can produce a nonzero latent component of p_n even when the terminal latent gradient is zero.

With parameter-independent initial noise, the exact pathwise parameter gradient is $\sum_{n=0}^{S-1} (\partial_\theta F_n|_{y_n})^\top p_{n+1}$. The partial derivative holds the input state fixed and includes the sampler's conversion from network outputs to a state increment. RGT differentiates coordinate velocity outputs at sparse recorded states, using normalized terminal gradients and coordinate-time weights. The bound below characterizes its approximation to future state sensitivity.

RGT approximates $P_{n:S}^\top g$ by g, then normalizes the terminal direction, applies a nonnegative time weight, and projects the update to coordinate velocity outputs. This identity approximation enables reuse of the terminal gradient along its sampled trajectory. The selected recorded states and normalized linear weights $\widetilde{w}_{k,j} = t_{x,k,j} / (J^{-1} \sum_{\ell=1}^{J} t_{x,k,\ell})$ allocate the local updates across coordinate times.

A conditional sensitivity bound. In any fixed Euclidean scaling of the joint state, suppose $\|A_\ell - I\|_2 \le \epsilon_\ell$ for the remaining steps. Define

$$\delta_n = \prod_{\ell=n}^{S-1} (1 + \epsilon_\ell) - 1. \tag{29}$$

Then

$$\|p_n - g\|_2 \le \delta_n \|g\|_2. \tag{30}$$

To see this, expand the ordered product $P_{n:S}$ into its identity term and all nonempty ordered products of $(A_\ell - I)$. Submultiplicativity bounds the sum of their norms by $\prod_\ell (1 + \epsilon_\ell) - 1$. Multiplication by g gives Eq. equation 30. Coordinate projection cannot increase this bound; moreover, the omitted latent sensitivity satisfies $\|p_n^z\|_2 \le \delta_n \|g\|_2$.

For an Euler step with state-independent additive noise,

$$F_\ell(y) = y + h_\ell b_\theta(y, t_\ell) + \sigma_\ell \xi_\ell, \qquad \|D_y b_\theta\|_2 \le L_\ell, \tag{31}$$

one can take $\epsilon_\ell = h_\ell L_\ell$, obtaining

$$\delta_n \le \exp\left(\sum_{\ell=n}^{S-1} h_\ell L_\ell \right) - 1. \tag{32}$$

The bound applies to the sampler drift b_θ, including the conversion from network velocity to sampling dynamics. For an ODE with uniformly bounded drift Jacobian L and scalar integration time t, the corresponding bound is $e^{L(1-t)} - 1$. For samplers with state-dependent noise or additional transformations, A_ℓ includes the derivatives of those operations.

Local direction and the role of late-time weighting. For $\|g\|_2 \ge \epsilon$ and $g \ne 0$, let $\hat{g} = \rho g / \|g\|_2$. A hypothetical coordinate-only perturbation of a recorded state in direction $w\hat{g}$ has directional derivative

$$\langle p_n, w\hat{g} \rangle \ge w\rho(1 - \delta_n)\|g\|_2, \qquad w \ge 0. \tag{33}$$

This follows by substituting $p_n = g + (p_n - g)$ and applying Cauchy–Schwarz. If $w > 0$ and $\delta_n < 1$, the direction is locally reward-increasing for this fixed path. This establishes a sufficient condition for local reward improvement under the identity approximation.

With nonnegative step bounds, the remaining product defining δ_n decreases as the anchor approaches the endpoint. This motivates assigning less weight to early anchors, where future sensitivity can be larger. RGT implements this late-time emphasis through unit-mean linear coordinate-time weights from Eq. equation 8, evaluated against constant, reversed, and quadratic weighting in the ablations. The local direction bound holds for each nonnegative weight, including the normalized choice. The weighted approximation error is at most $\widetilde{w}\rho\delta_n$ when compared with the similarly weighted exact sensitivity using the same terminal normalization.

Local proximal interpretation. At a fixed recorded state, with detached direction a and reference $\bar{v}$, completing the square gives, for $\lambda > 0$,

$$-\langle v, a\rangle_{\mathrm{Ab}} + \frac{\lambda}{2}\|v - \bar{v}\|^2_{\mathrm{Ab}} = \frac{\lambda}{2}\left\|v - \left(\bar{v} + \frac{a}{\lambda}\right)\right\|^2_{\mathrm{Ab}} + C, \tag{34}$$

where C is independent of v. RGT therefore fits a locally shifted EMA velocity target. Shared parameters couple the regression objectives across recorded states and both output modes. The optimizer aggregates these local objectives, and the EMA reference regularizes velocity changes as training progresses.

Distinction from existing methods. DRaFT and its truncated variants (Clark et al., 2024) propagate differentiable rewards through a contiguous terminal sampling segment. RGT performs detached rollout collection followed by isolated velocity evaluations at sparse anchors spanning that rollout. Adjoint Matching (Domingo-Enrich et al., 2025) provides an optimal-control construction with backward adjoint computation. RGT uses the identity sensitivity approximation to construct its local velocity targets. RAM (Bergmeister et al., 2026) uses scalar rewards and analytically re-noised endpoints, whereas RGT pairs each recorded state with the gradient of its own realized endpoint.

Mechanism controls. Random and sign-flipped directions test dependence on reward-gradient information. Cross-trajectory pairing tests the state–endpoint association; endpoint-only updates test the use of intermediate states. Table 9 adds normalized time-weight comparisons, reference-penalty and endpoint-normalization ablations, and BPTT comparisons with a shared 50-step rollout. Together, these controls evaluate the contributions of trajectory pairing, temporal allocation, gradient normalization, and reference regularization to design performance.

Contribution of the transport rule. RGT specifies a local credit-assignment rule for interface alignment in a joint coordinate–latent generator. Same-trajectory pairing retains the association between a sampled intermediate state and its realized endpoint; coordinate projection directs supervision to the explicit geometry while shared parameters couple it to sequence–structure generation. Endpoint-gradient normalization balances the contribution of different trajectories, and unit-mean time weighting allocates that contribution across anchors without changing its mean scalar weight. The EMA penalty turns these directions into constrained local velocity targets. This construction separates the choice of the supervised output mode, trajectory association, temporal allocation, and reference constraint. The mechanism controls and shared-rollout BPTT comparisons evaluate these choices in the antibody design setting, while the conditional bound characterizes the state-sensitivity approximation on which transport relies.

E Additional Experiments

E.1 Evaluation Metrics

We group the metrics into sequence recovery, reference-based structure and docking quality, sequence-refolding confidence, and interface quality. Arrows indicate whether higher (↑) or lower (↓) values are preferred.

Sequence recovery and autoencoder reconstruction. **AAR** (↑) is the fraction of evaluated residues whose generated amino-acid identity matches the reference. We report it separately for CDR-L1–L3

and CDR-H1–H3 in antibodies and CDR1–CDR3 in nanobodies; CDR and full-antibody AAR in the AE evaluation use the corresponding residue sets. **AE scRMSD** (↓, Å) measures reconstruction error over common side-chain heavy atoms after residue-local N–Cα–C alignment, aggregated over CDRs, H3, or the full antibody. **Atom coverage** (↑, %) reports the fraction of reference atoms covered by the reconstruction comparison. Coverage is reported alongside RMSD because errors on matched atoms alone do not capture missing atoms.

Reference-based structural quality. **TM-Score** (↑) measures global fold similarity using length-normalized, distance-weighted structural agreement after superposition. **lDDT** (↑) measures preservation of local interatomic distances without requiring a global superposition. Both are reported on the 0–1 scale. **CDR backbone RMSD** (↓, Å) measures deviations between corresponding backbone atoms in the designated loop. Self-aligned RMSD fits the loop itself and measures its conformation; antigen-aligned RMSD also includes antibody placement error. Each reported metric uses the same alignment protocol across methods on the complete test set.

Docking quality. **iRMS** (↓, Å) is the RMSD of corresponding interface backbone atoms after interface superposition. **LRMS** (↓, Å) measures ligand backbone displacement after aligning the receptor, assessing relative partner placement. **DockQ** (↑) combines native-contact recovery f_{nat}, iRMS, and LRMS into a score between 0 and 1:

$$\text{DockQ} = \frac{1}{3}\left[f_{\text{nat}} + \frac{1}{1+(\text{iRMS}/1.5)^2} + \frac{1}{1+(\text{LRMS}/8.5)^2}\right], \tag{35}$$

where RMSDs are in Å. Atom selections and chain assignments follow the docking evaluator.[1] **SR** (↑) is the fraction of evaluated targets with DockQ > 0.23. It is a structural success rate, not an experimental binding rate. These reference-based metrics are used for evaluation, whereas the training reward uses the proxies defined in Appendix C.3.

Sequence-refolding confidence and reference-based docking accuracy. Protenix-v2 refolds the generated sequences with the antigen; it does not redesign their identities. All confidence metrics below are higher-is-better and do not require a native reference structure. **pLDDT** (0–100) estimates local structural confidence. **pTM** (0–1) estimates global fold confidence, while **global ipTM** (0–1) estimates inter-chain placement confidence for the full complex. Global ipTM is distinct from an antibody–antigen chain-pair score. **ipSAE** (0–1) uses inter-chain predicted aligned errors with a PAE cutoff and an effective-residue normalization to assess interaction confidence. **pDockQ** (0–1) predicts docking quality from interface pLDDT and the number of interface contacts; unlike DockQ, it does not compare with native coordinates.[2] For VH–VL complexes, reported ipSAE and pDockQ are the maximum over the heavy-chain–antigen and light-chain–antigen pairs; VHH evaluation uses the heavy-chain–antigen pair. **Ranking score** is the composite confidence score returned by Protenix, combining ipTM and pTM with its implementation-specific disorder and clash adjustments. It is not a reference-based accuracy measure or a binding probability.[3]

DockQ after refolding (↑, 0–1), labeled DockQ in the refolding tables, compares the Protenix-v2-refolded complex of the designed sequence and antigen with the corresponding native reference complex. It uses the reference-based DockQ definition above and requires native coordinates. The refolded complex is the evaluated model; the original generated complex is not the reference. This measures docking accuracy after sequence refolding, separately from the confidence estimates reported in the same tables.

Interface geometry and contacts. **Clash rate** (↓) quantifies steric clashes under the interface evaluator's atom selections and clash criterion. **Interface scRMSD** (↓, Å) compares corresponding interface side-chain atoms after alignment on common antibody backbone atoms; unlike AE scRMSD, it retains inter-residue side-chain arrangement errors. **Polar-contact recovery** (↑) is the fraction of reference polar interface contacts recovered in a generated complex. Clash rate and polar-contact recovery are reported as fractions in the antibody table and percentages in the nanobody interface table. These discrete evaluation measures differ from the soft Cα contact and clash proxies used for RGT.

[1] DockQ implementation: `https://github.com/wallnerlab/DockQ`.

[2] ipSAE and chain-pair scoring implementation: `https://github.com/DunbrackLab/IPSAE`; pDockQ: `https://www.nature.com/articles/s41467-022-28865-w`.

[3] Protenix confidence implementation: `https://github.com/bytedance/Protenix/blob/main/protenix/model/sample_confidence.py`.

Table 7: De novo design performance stratified by test-antigen sequence identity to the training antigen library. For each query, identity is taken from the highest-bitscore MMseqs2 alignment, with identity breaking ties and no coverage cutoff.

Identity	n	Method	AAR H3↑	RMSD H3↓	TM-Score↑	lDDT↑	DockQ↑	iRMS↓	LRMS↓	SR↑
VH–VL antibodies ($N = 93$)										
$\leq 30\%$	19	RAFA	$0.545^{0.193}$	$2.321^{1.317}$	$0.919^{0.083}$	$0.789^{0.098}$	$0.162^{0.207}$	$9.552^{6.432}$	$22.712^{14.260}$	0.263
		RAFA (RL)	$0.548^{0.193}$	$2.359^{1.370}$	$0.912^{0.085}$	$0.783^{0.095}$	$0.187^{0.180}$	$9.065^{6.106}$	$20.283^{10.462}$	0.316
$(30, 40]\%$	17	RAFA	$0.440^{0.192}$	$2.212^{0.927}$	$0.936^{0.036}$	$0.791^{0.055}$	$0.127^{0.162}$	$8.365^{4.464}$	$28.361^{23.304}$	0.176
		RAFA (RL)	$0.416^{0.191}$	$2.269^{0.946}$	$0.933^{0.037}$	$0.789^{0.052}$	$0.150^{0.156}$	$7.793^{4.120}$	$26.451^{21.303}$	0.235
$(40, 60]\%$	7	RAFA	$0.524^{0.188}$	$2.273^{1.159}$	$0.941^{0.024}$	$0.799^{0.035}$	$0.043^{0.019}$	$11.236^{3.668}$	$33.112^{8.051}$	0.000
		RAFA (RL)	$0.533^{0.174}$	$2.344^{1.287}$	$0.940^{0.028}$	$0.792^{0.043}$	$0.066^{0.019}$	$10.424^{3.912}$	$29.726^{5.373}$	0.000
$(60, 80]\%$	50	RAFA	$0.523^{0.187}$	$2.391^{0.919}$	$0.929^{0.027}$	$0.775^{0.055}$	$0.152^{0.170}$	$8.681^{5.335}$	$21.406^{11.464}$	0.220
		RAFA (RL)	$0.531^{0.180}$	$2.461^{0.965}$	$0.926^{0.026}$	$0.770^{0.056}$	$0.188^{0.167}$	$8.017^{5.205}$	$18.312^{7.474}$	0.320
VHH nanobodies ($N = 50$)										
$\leq 30\%$	25	RAFA	$0.488^{0.239}$	$2.049^{0.750}$	$0.920^{0.021}$	$0.752^{0.048}$	$0.119^{0.110}$	$8.118^{3.219}$	$18.913^{7.172}$	0.120
		RAFA (RL)	$0.488^{0.240}$	$2.130^{0.769}$	$0.916^{0.022}$	$0.749^{0.048}$	$0.152^{0.121}$	$7.643^{3.239}$	$18.079^{6.898}$	0.200
$(30, 40]\%$	11	RAFA	$0.412^{0.272}$	$2.850^{1.941}$	$0.900^{0.070}$	$0.752^{0.091}$	$0.085^{0.065}$	$9.145^{2.834}$	$24.317^{9.724}$	0.091
		RAFA (RL)	$0.395^{0.274}$	$2.829^{1.931}$	$0.899^{0.068}$	$0.751^{0.089}$	$0.126^{0.134}$	$8.711^{3.033}$	$23.125^{9.652}$	0.091
$(40, 60]\%$	7	RAFA	$0.369^{0.082}$	$2.728^{0.578}$	$0.889^{0.087}$	$0.735^{0.100}$	$0.111^{0.085}$	$7.752^{3.061}$	$20.232^{4.924}$	0.143
		RAFA (RL)	$0.393^{0.093}$	$2.767^{0.670}$	$0.888^{0.083}$	$0.731^{0.097}$	$0.140^{0.087}$	$7.627^{2.929}$	$19.744^{4.390}$	0.286
$(60, 80]\%$	7	RAFA	$0.509^{0.273}$	$3.143^{1.253}$	$0.889^{0.032}$	$0.757^{0.041}$	$0.148^{0.173}$	$9.006^{4.242}$	$20.905^{10.125}$	0.143
		RAFA (RL)	$0.519^{0.277}$	$3.224^{1.398}$	$0.886^{0.032}$	$0.755^{0.040}$	$0.167^{0.197}$	$8.827^{4.150}$	$21.605^{11.360}$	0.429

Interface energy and buried surface area. **Median** ΔG (↓, REU) summarizes the Rosetta interface-energy score across targets following coordinate-constrained FastRelax and InterfaceAnalyzer evaluation. More negative values indicate more favorable scores under the Rosetta energy function, not measured binding free energies. **Mean** $|\Delta\mathrm{BSA}|$ (↓, Å^2) is the target-average absolute difference between generated and native interface buried surface areas, $|\mathrm{BSA}_{\mathrm{gen}} - \mathrm{BSA}_{\mathrm{native}}|$. Lower values indicate closer agreement with the reference interface size, rather than a larger interface. The same postprocessing protocol is applied to all methods for each reported comparison.

Aggregation and computational cost. Unless otherwise stated, metrics are averaged over the complete test sets of 93 VH–VL or 50 VHH targets, and superscripts denote population target-level standard deviations, $\sqrt{N^{-1}\sum_{i=1}^{N}(m_i - \bar{m})^2}$. All methods use the same target lists, with no exclusions due to method failures. SR is reported as a fraction, and interface ΔG is summarized by its median. Inference-seed and training-seed variability are identified explicitly in the corresponding tables. Inference cost is measured as end-to-end seconds per sample at the stated batch size; training cost is measured in H20-GPU-hours.

E.2 Design Performance by Antigen Sequence Similarity

Table 7 stratifies the 143-target design results by test-antigen similarity to the training antigen library. For each query, we select the MMseqs2 hit with the highest bitscore, breaking ties by identity, and bin its alignment identity without a coverage cutoff. The four intervals are $[0, 30]\%$, $(30, 40]\%$, $(40, 60]\%$, and $(60, 80]\%$, containing 19/17/7/50 VH–VL targets and 25/11/7/7 VHH targets, respectively. This alignment-based identity is distinct from the normalized global-alignment score used for dataset filtering.

RAFA (RL) improves mean DockQ and reduces mean iRMS in all four identity strata for both VH–VL and VHH targets. SR increases or remains unchanged across strata.

E.3 Extended RGT Mechanism Comparison

Table 8 extends the mechanism controls considered in Table 6. Random and shuffled directions increase mean DockQ from 0.141 to 0.150 and 0.152, respectively, with smaller gains than full RGT (0.172). Sign-flipped RGT has the same reported mean DockQ as the base model (0.141) and a slightly higher SR (0.215 versus 0.204), while its mean iRMS and LRMS are worse. Endpoint-only updates improve performance less than the full RGT configuration. RGT with normalized linear time

Table 8: Extended RGT mechanism comparison, including the controls reported in the main ablation. The RAFA (RL) row uses normalized linear time weights as in Eq. equation 8; explicitly listed $w(t_x)$ expressions specify the other weighting controls. AAR and self-aligned RMSD refer to CDR-H3. RMSD, iRMS, and LRMS are in Å. Exact BPTT uses a short rollout and is presented as a reference rather than a shared-rollout control.

Method	AAR H3↑	RMSD H3↓	TM-Score↑	lDDT↑	DockQ↑	iRMS↓	LRMS↓	SR↑
RAFA	$\underline{0.512}^{0.192}$	$\mathbf{2.335}^{1.035}$	$\mathbf{0.929}^{0.046}$	$\mathbf{0.783}^{0.066}$	$0.141^{0.173}$	$8.993^{5.383}$	$23.825^{15.166}$	0.204
Random direction	$0.510^{0.194}$	$2.361^{1.048}$	$0.927^{0.048}$	$\underline{0.781}^{0.067}$	$0.150^{0.171}$	$9.040^{5.420}$	$23.960^{15.350}$	0.215
Shuffled gradient	$0.511^{0.191}$	$\underline{2.354}^{1.041}$	$\underline{0.928}^{0.047}$	$\underline{0.781}^{0.066}$	$0.152^{0.170}$	$8.960^{5.360}$	$23.750^{15.080}$	0.247
Sign-flipped RGT	$0.501^{0.199}$	$2.518^{1.124}$	$0.918^{0.058}$	$0.769^{0.071}$	$0.141^{0.161}$	$9.380^{5.550}$	$25.170^{16.020}$	0.215
Endpoint-only	$0.510^{0.193}$	$2.372^{1.052}$	$0.927^{0.048}$	$\underline{0.781}^{0.066}$	$0.158^{0.169}$	$8.780^{5.287}$	$23.120^{14.702}$	0.258
Cross-trajectory, $w(t_x) = t_x$	$0.509^{0.195}$	$2.401^{1.069}$	$0.926^{0.050}$	$0.779^{0.068}$	$0.154^{0.168}$	$8.870^{5.331}$	$23.380^{14.916}$	0.247
Same trajectory, $w(t_x) = 1 - t_x$	$0.508^{0.197}$	$2.415^{1.077}$	$0.924^{0.051}$	$0.778^{0.068}$	$0.158^{0.169}$	$8.740^{5.290}$	$22.970^{14.450}$	0.258
Same trajectory, $w(t_x) = 1$	$0.509^{0.194}$	$2.397^{1.061}$	$0.925^{0.049}$	$0.779^{0.067}$	$0.164^{0.166}$	$8.570^{5.244}$	$22.210^{13.934}$	0.269
Same trajectory, $w(t_x) = t_x^2$	$0.507^{0.198}$	$2.441^{1.086}$	$0.923^{0.052}$	$0.777^{0.069}$	$0.168^{0.165}$	$8.460^{5.211}$	$21.630^{13.201}$	0.269
RAFA (RL)	$\mathbf{0.514}^{0.190}$	$2.396^{1.085}$	$0.925^{0.047}$	$0.778^{0.065}$	$\mathbf{0.172}^{0.165}$	$\underline{8.371}^{5.194}$	$\underline{21.062}^{12.356}$	**0.280**
Exact BPTT, short rollout	$0.505^{0.201}$	$2.468^{1.106}$	$0.921^{0.054}$	$0.774^{0.070}$	$\underline{0.171}^{0.164}$	$\mathbf{8.210}^{5.100}$	$\mathbf{20.710}^{12.100}$	**0.280**

Table 9: Additional RGT configuration ablations and measured training cost. (A) Normalized time-weight comparison, with t denoting coordinate time t_x. (B) Reference-penalty and endpoint-gradient normalization ablations. (C) RGT and BPTT with the same 50-step rollout. (D) Measured cost on $8\times$ NVIDIA H20 GPUs with the same global batch size, 50-step rollouts, and 2,000 updates; memory is peak allocated memory per GPU.

A. Normalized time weights

Weight	DockQ↑	iRMS↓ (Å)	LRMS↓ (Å)
Constant 1	0.164 ± 0.166	8.570 ± 5.244	22.210 ± 13.934
Normalized t	$\mathbf{0.172} \pm 0.165$	$\mathbf{8.371} \pm 5.194$	$\mathbf{21.062} \pm 12.356$
Normalized t^2	0.169 ± 0.167	8.440 ± 5.230	21.580 ± 12.910
Normalized $1 - t$	0.158 ± 0.169	8.750 ± 5.310	22.960 ± 14.220

B. Reference penalty and endpoint-gradient normalization

Reference	Endpoint grad. normalization	DockQ↑	TM-Score↑	Training-seed STD (DockQ)↓
EMA	Yes	$\mathbf{0.172} \pm 0.165$	0.925 ± 0.047	**0.003**
None	Yes	0.164 ± 0.176	0.914 ± 0.065	0.008
EMA	No	0.166 ± 0.173	0.922 ± 0.053	0.007
None	No	0.154 ± 0.185	0.901 ± 0.089	0.014
Fixed pretrained	Yes	0.168 ± 0.163	$\mathbf{0.928} \pm 0.045$	0.004

C. Shared 50-step rollout

Method	DockQ↑	iRMS↓ (Å)	LRMS↓ (Å)
RGT, 4 anchors	$\mathbf{0.172} \pm 0.165$	8.371 ± 5.194	21.062 ± 12.356
Truncated BPTT, last 4 steps	0.165 ± 0.168	8.520 ± 5.260	21.950 ± 13.480
Truncated BPTT, last 16 steps	0.171 ± 0.166	8.383 ± 5.252	21.426 ± 12.420
Full BPTT, 50 steps	$\mathbf{0.172} \pm 0.165$	$\mathbf{8.275} \pm 5.163$	$\mathbf{20.945} \pm 12.231$

D. Measured training cost

Method	Time/update (s)↓	Wall time (h)↓	GPU-hours↓	Peak allocated (GiB/GPU)↓
RGT, 4 anchors	**43.2**	**24.0**	**192**	**24**
Full BPTT, 50 steps	99.0	55.0	440	72

weighting has the highest mean DockQ and success rate among the listed transport configurations. Short-rollout BPTT yields a similar mean DockQ and the same reported success rate as RGT, with lower mean iRMS and LRMS. The additional configuration ablations below compare time weights at matched mean scale and BPTT with a shared rollout length.

Additional configuration ablations. Table 9 examines three choices in the antibody joint-generation setting: time weighting, reference regularization and endpoint-gradient normalization, and the differentiated portion of the rollout. These experiments quantify how the RGT components affect design quality and training cost.

Table 10: DockQ across three inference random seeds for selected RGT controls. The final column gives the mean and sample standard deviation across the three seed-level means.

Method	Seed 7	Seed 8	Seed 9	Mean ± SD
RAFA	0.139	0.143	0.141	0.141 ± 0.002
Endpoint-only	0.156	0.160	0.158	0.158 ± 0.002
Constant transport	0.162	0.167	0.163	0.164 ± 0.003
Cross-trajectory	0.151	0.157	0.153	0.154 ± 0.003
RGT, normalized t_x	**0.169**	**0.175**	**0.171**	$\mathbf{0.172} \pm 0.003$

Normalized time weights. Panel A compares normalized time weights, using the main RGT configuration as the normalized-linear reference. For each weighting rule $w(t) \in \{1, t, t^2, 1-t\}$, we use the same actual sampled coordinate times $t_j = t_{x,j}$ and set

$$\widetilde{w}_j = \frac{w(t_j)}{J^{-1}\sum_{\ell=1}^{J} w(t_\ell)}, \qquad \frac{1}{J}\sum_{j=1}^{J} \widetilde{w}_j = 1. \tag{36}$$

Reward-gradient normalization, the EMA coefficient, learning rate, trajectory count, and number of parameter updates are held fixed. This matches the mean scalar weight on the reward term while varying its allocation across anchors. The normalized-linear row is the same RGT reference reported in the main results and Table 8; it is not a separate unnormalized-linear run. Linear weighting gives the highest mean DockQ (0.172) and the lowest mean iRMS and LRMS among these settings. Its DockQ exceeds constant, quadratic, and reversed weighting by 0.008, 0.003, and 0.014, respectively. These results support a moderate late-time emphasis in the evaluated configuration.

Reference penalty and endpoint-gradient normalization. Panel B separates target-level variation (the ± entries) from variability across independently trained runs (the last column). With endpoint-gradient normalization enabled, removing the EMA penalty reduces mean DockQ from 0.172 to 0.164 and increases the training-seed STD from 0.003 to 0.008. With EMA retained, removing normalization reduces DockQ to 0.166 and increases the training-seed STD to 0.007. Removing both gives the lowest DockQ (0.154) and largest training-seed STD (0.014). These results support retaining both components for docking quality and consistency across training seeds in this setting. A fixed pretrained reference gives slightly higher mean TM-Score than EMA (0.928 versus 0.925), but lower DockQ (0.168 versus 0.172), indicating a trade-off between global fold preservation and interface improvement.

BPTT with a shared rollout length. All methods in panel C use 50 rollout steps. RGT differentiates fresh velocity evaluations at four sparse anchors, whereas truncated BPTT differentiates through the last 4 or 16 consecutive steps and full BPTT differentiates through all 50 steps. RGT improves all three reported docking means over both truncated-BPTT variants. Compared with last-16-step BPTT, RGT has higher mean DockQ (0.172 versus 0.171), lower iRMS (8.371 versus 8.383), and lower LRMS (21.062 versus 21.426). Full BPTT matches the reported mean DockQ of RGT (0.172) and has lower iRMS (8.275) and LRMS (20.945). RGT obtains these results using four sparse local velocity evaluations per trajectory.

Measured training cost. Panel D compares RGT and full BPTT on 8× NVIDIA H20 GPUs with the same global batch size, 50-step rollouts, and 2,000 parameter updates. RGT requires 43.2 seconds per update and 24.0 hours (192 GPU-hours) overall, compared with 99.0 seconds and 55.0 hours (440 GPU-hours) for full BPTT. This is a measured 2.29× speedup, or a 56.4% reduction in wall time and GPU-hours. Peak allocated memory per GPU decreases from 72 to 24 GiB, a 66.7% reduction. Together with panel C, these measurements show a quality–cost trade-off: RGT has lower training time and allocated memory, with the same reported mean DockQ as full BPTT; full BPTT retains lower iRMS and LRMS.

Across-seed comparison. Table 10 evaluates robustness to inference randomness using three sampling seeds. Linear-weighted RGT exceeds the base model, endpoint-only updates, constant transport, and cross-trajectory pairing in each reported seed column. From the rounded entries, its paired seed-level gains over the base are 0.030, 0.032, and 0.030. These consistent gains demonstrate robustness of the observed improvement across the evaluated inference seeds.

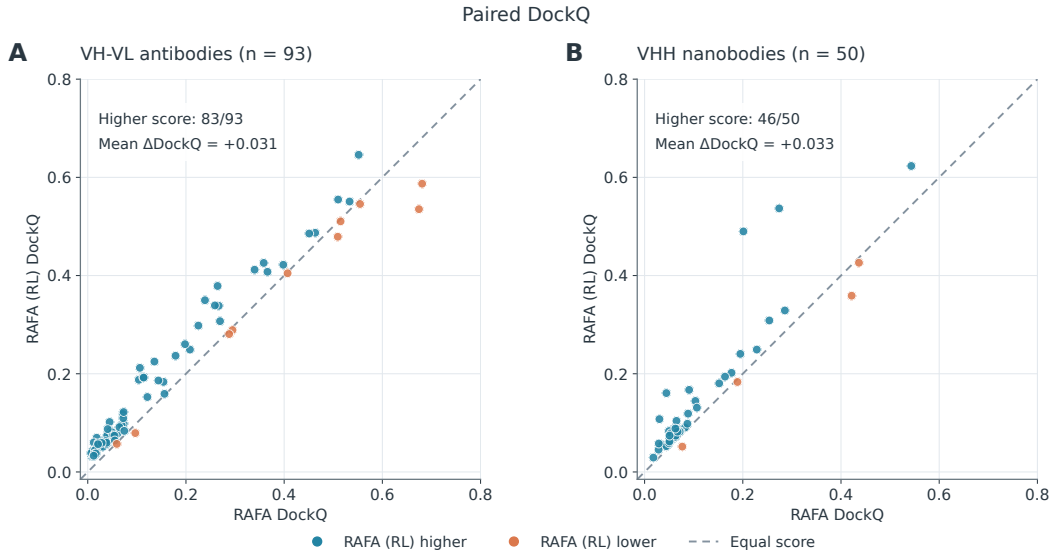

Figure 6: Paired DockQ visualization for RAFA (horizontal axis) and RAFA (RL) (vertical axis). Each point represents one target: (A) 93 VH–VL antibodies and (B) 50 VHH nanobodies.

Table 11: Autoencoder reconstruction quality on the identical held-out set of 143 complexes (93 conventional antibodies and 50 nanobodies). We compare the original released La-Proteina checkpoint pretrained on AFDB with our antibody-finetuned checkpoint.

AE checkpoint	CDR AAR↑	Full-Ab AAR↑	CDR scRMSD↓	H3 scRMSD↓	Ab scRMSD↓	Atom coverage↑
La-Proteina (AFDB pretrained)	$0.992^{0.039}$	$0.989^{0.044}$	$0.300^{0.125}$	$0.308^{0.213}$	$0.347^{0.094}$	99.960%
Antibody-finetuned	$\mathbf{1.000}^{0.000}$	$\mathbf{1.000}^{0.000}$	$\mathbf{0.187}^{0.064}$	$\mathbf{0.188}^{0.091}$	$\mathbf{0.180}^{0.045}$	**100.000%**

E.4 Paired DockQ Visualization

Figure 6 visualizes paired scores for 93 VH–VL and 50 VHH targets, with RAFA on the horizontal axis and RAFA (RL) on the vertical axis. The vertical-axis score exceeds the horizontal-axis score for 83/93 VH–VL targets and 46/50 VHH targets.

E.5 Autoencoder Reconstruction

We compare the released AFDB-pretrained La-Proteina AE and the antibody-finetuned AE on the same 143-complex held-out evaluation set. Fine-tuning increases CDR/full-antibody AAR from 0.992/0.989 to 1.000/1.000. CDR, H3, and full-antibody side-chain RMSD decrease from 0.300/0.308/0.347 Å to 0.187/0.188/0.180 Å, respectively. These side-chain errors are computed after residue-local N–Cα–C alignment to assess local reconstruction accuracy. Whole-interface scRMSD uses antibody alignment to assess the spatial arrangement of interface side chains. The AE passes explicit Cα coordinates through a bypass and reconstructs sequence identities and all-atom geometry using stochastic per-residue latents with KL regularization.

E.6 Sampling-Step Evaluation

Table 12 reports the inference-step comparison for RAFA (RL). The 400-step results are also reported for H3, global structure, and docking in Tables 2 and 14. Among the tested settings, 400 steps gives the highest H3 AAR and lowest self-aligned H3 RMSD and LRMS, whereas 1,000 steps gives the highest DockQ and lowest iRMS. SR is highest at both 200 and 1,000 steps. More sampling steps therefore do not uniformly improve every metric. These are inference sampling steps, not parameter updates.

Table 12: Effect of inference sampling steps on de novo antibody design by RAFA (RL) on the 93-antibody test set.

Steps	AAR H3↑	RMSD H3↓	TM-Score↑	lDDT↑	DockQ↑	iRMS↓	LRMS↓	SR↑
20	$0.504^{0.198}$	$6.752^{1.169}$	$0.267^{0.037}$	$0.151^{0.032}$	$0.033^{0.020}$	$14.304^{2.783}$	$36.063^{15.006}$	0.000
50	$0.504^{0.198}$	$3.842^{0.798}$	$0.521^{0.089}$	$0.303^{0.039}$	$0.078^{0.079}$	$11.298^{3.912}$	$28.337^{12.545}$	0.086
100	$0.502^{0.202}$	$2.569^{1.002}$	$0.851^{0.070}$	$0.568^{0.033}$	$0.131^{0.121}$	$8.965^{4.554}$	$24.116^{15.333}$	0.172
200	$0.508^{0.198}$	$2.422^{1.079}$	$0.923^{0.048}$	$\mathbf{0.778}^{0.065}$	$0.170^{0.163}$	$8.651^{5.276}$	$21.358^{12.043}$	**0.312**
400	$\mathbf{0.514}^{0.190}$	$\mathbf{2.396}^{1.085}$	$\mathbf{0.925}^{0.047}$	$\mathbf{0.778}^{0.065}$	$0.172^{0.165}$	$8.371^{5.194}$	$\mathbf{21.062}^{12.356}$	0.280
800	$0.505^{0.200}$	$2.472^{1.033}$	$0.923^{0.048}$	$\mathbf{0.778}^{0.066}$	$0.151^{0.158}$	$9.004^{5.089}$	$23.016^{14.411}$	0.269
1000	$0.508^{0.199}$	$2.431^{1.089}$	$\mathbf{0.925}^{0.047}$	$\mathbf{0.778}^{0.065}$	$\mathbf{0.175}^{0.159}$	$\mathbf{8.285}^{5.258}$	$21.512^{12.464}$	**0.312**

Table 13: End-to-end sampling cost on a single NVIDIA H20 GPU with batch size 4. Time is reported in seconds per sample.

ODE steps	20	50	100	200	400
Seconds/sample↓	1.963	3.813	6.531	11.346	21.546

E.7 Inference and Training Cost

We measure end-to-end sampling on a single NVIDIA H20 GPU with batch size 4 (Table 13).

Base training comprises two stages: general-protein training on 16 H20 GPUs for two days, followed by continued training on the antibody dataset on 16 H20 GPUs for another two days. Each stage requires approximately 768 H20-GPU-hours, for a combined base-training cost of approximately 1,536 H20-GPU-hours. RGT post-training uses 8 H20 GPUs for 2,000 parameter updates. The measured RGT run in panel D of Table 9 takes 24.0 hours (192 H20-GPU-hours), with 24 GiB peak allocated memory per GPU; the corresponding full-BPTT run takes 55.0 hours (440 H20-GPU-hours), with 72 GiB per GPU.

For hardware context, the H20 provides approximately 148 TFLOPS of dense FP16/BF16 Tensor Core throughput, compared with approximately 989 TFLOPS for the H200 SXM, a $6.7\times$ difference in theoretical dense compute. Memory bandwidth is 4.0 and 4.8 TB/s, respectively. The H200's advertised 1,979 TFLOPS uses sparsity and is not directly comparable to the dense H20 figure.[4] These peak specifications do not imply an equivalent application-level speedup.

E.8 Alternative Framework Initializations

Table 14 reports DiffAb and MEAN with IgFold- and AF3-initialized frameworks alongside RAFA and RAFA (RL). The reported DockQ scores are 0.011/0.014 for the IgFold variants and 0.019/0.019 for the AF3 variants; the corresponding RAFA/RAFA (RL) values are 0.141/0.172. The comparison evaluates these design pipelines on the same test targets using unified metrics and postprocessing. RAFA and RAFA (RL) generate antibody coordinates directly from the antigen and framework-sequence conditions.

E.9 Docking Accuracy After Sequence Refolding

Evaluating refolded complexes against native references. Tables 3 and 15 evaluate the generated sequences by refolding them with their antigens using Protenix-v2 on the complete 93-target VH–VL and 50-target VHH test sets. DockQ is computed between each refolded complex and the corresponding native reference complex. It therefore assesses reference-based docking accuracy of structures predicted from the designed sequences. The additional confidence metrics characterize the resulting refolded complexes without using native coordinates. Protenix-v2 differs from Boltz-2, which supplies the training-split labels for the frozen fold-confidence reward surrogate.

Docking improvements across both antibody formats. RGT improves mean DockQ after refolding by 0.070 in both formats: from 0.088 to 0.158 for VH–VL antibodies and from 0.094 to

[4]NVIDIA H200 specifications: `https://www.nvidia.com/en-us/data-center/h200/`.

Table 14: Additional results for framework-sequence-conditioned de novo antibody design on 93 VH–VL targets. DiffAb and MEAN are evaluated with IgFold- and AlphaFold 3 (AF3)-initialized framework structures; IgFold variants use HDOCKlite docking. RAFA and RAFA (RL) receive the framework sequence but no framework structure.

Model		DiffAb (IgFold)	MEAN (IgFold)	DiffAb (AF3)	MEAN (AF3)	RAFA	RAFA (RL)
AAR↑	L1	$0.500^{0.247}$	–	$0.462^{0.249}$	–	$\underline{0.693}^{0.196}$	$\mathbf{0.698}^{0.191}$
	L2	$0.489^{0.301}$	–	$0.430^{0.323}$	–	$\underline{0.821}^{0.252}$	$\mathbf{0.822}^{0.256}$
	L3	$0.406^{0.152}$	–	$0.369^{0.177}$	–	$\mathbf{0.727}^{0.246}$	$\underline{0.718}^{0.246}$
	H1	$0.625^{0.168}$	–	$0.513^{0.177}$	–	$\underline{0.696}^{0.194}$	$\mathbf{0.703}^{0.179}$
	H2	$0.271^{0.142}$	–	$0.255^{0.136}$	–	$\underline{0.667}^{0.226}$	$\mathbf{0.672}^{0.225}$
	H3	$0.256^{0.107}$	$0.356^{0.123}$	$0.253^{0.111}$	$0.343^{0.128}$	$\underline{0.512}^{0.192}$	$\mathbf{0.514}^{0.190}$
RMSD↓	L1	$1.005^{0.922}$	–	$1.245^{1.032}$	–	$\mathbf{0.629}^{0.665}$	$\underline{0.653}^{0.671}$
	L2	$0.192^{0.179}$	–	$0.213^{0.180}$	–	$\underline{0.138}^{0.177}$	$\mathbf{0.137}^{0.186}$
	L3	$0.892^{0.873}$	–	$1.038^{0.842}$	–	$\mathbf{0.728}^{0.753}$	$\underline{0.740}^{0.749}$
	H1	$0.900^{0.749}$	–	$1.664^{2.724}$	–	$\underline{0.772}^{0.571}$	$\mathbf{0.757}^{0.546}$
	H2	$0.705^{0.604}$	–	$1.129^{1.798}$	–	$\mathbf{0.538}^{0.483}$	$\underline{0.563}^{0.515}$
	H3	$2.667^{1.245}$	$\mathbf{2.262}^{0.894}$	$2.951^{2.700}$	$2.657^{0.910}$	$\underline{2.335}^{1.035}$	$2.396^{1.085}$
Structure↑	TMscore	$0.910^{0.112}$	$0.914^{0.113}$	$0.867^{0.099}$	$0.851^{0.096}$	$\mathbf{0.929}^{0.046}$	$\underline{0.925}^{0.047}$
	lDDT	$0.337^{0.044}$	$0.342^{0.044}$	$0.641^{0.097}$	$0.637^{0.097}$	$\mathbf{0.783}^{0.066}$	$\underline{0.778}^{0.065}$
Side chain↓	Clash rate	$\underline{0.002}^{0.002}$	$\mathbf{0.002}^{0.002}$	$0.018^{0.020}$	$0.003^{0.007}$	$0.018^{0.017}$	$0.019^{0.017}$
	scRMSD	$\underline{13.320}^{8.284}$	$17.572^{9.235}$	$17.148^{9.724}$	$19.907^{7.626}$	$14.227^{7.308}$	$\mathbf{12.210}^{6.000}$
Interface	Polar contact↑	$0.000^{0.000}$	$0.000^{0.000}$	$0.002^{0.017}$	$0.000^{0.000}$	$\underline{0.014}^{0.038}$	$\mathbf{0.049}^{0.106}$
	$\lvert\Delta\text{BSA}\rvert\downarrow$	$645.1^{439.3}$	$639.5^{432.6}$	$705.2^{645.6}$	$762.5^{768.3}$	$\underline{248.9}^{151.3}$	$\mathbf{121.5}^{90.8}$
	Median $\Delta G\downarrow$	$\mathbf{-53.1}$	-39.1	-44.1	-43.8	-32.2	$\underline{-49.3}$
Docking	DockQ↑	$0.011^{0.008}$	$0.014^{0.014}$	$0.019^{0.017}$	$0.019^{0.017}$	$\underline{0.141}^{0.173}$	$\mathbf{0.172}^{0.165}$
	iRMS↓	$21.055^{9.076}$	$19.894^{9.811}$	$16.881^{5.240}$	$16.409^{5.238}$	$\underline{8.993}^{5.383}$	$\mathbf{8.371}^{5.194}$
	LRMS↓	$61.100^{20.423}$	$60.930^{22.769}$	$50.232^{19.456}$	$50.819^{19.624}$	$\underline{23.825}^{15.166}$	$\mathbf{21.062}^{12.356}$
	SR↑	0.000	0.000	0.000	0.000	$\underline{0.204}$	**0.280**

Table 15: Protenix-v2 sequence-refolding confidence and reference-based docking accuracy for existing designs on the 50-nanobody test set. Protenix-v2 refolds the supplied sequences with unpaired MSA. VHH pDockQ is read from the antibody–antigen H–A chain pair. DockQ compares each refolded complex with the corresponding native reference complex.

Method	Global ipTM↑	pLDDT↑	pTM↑	ipSAE↑	pDockQ↑	Ranking score↑	DockQ↑
DiffAb (AF3)	$0.349^{0.165}$	$79.179^{8.006}$	$0.697^{0.107}$	$0.054^{0.143}$	$\underline{0.194}^{0.101}$	$0.418^{0.147}$	$0.057^{0.038}$
IgGM	$0.353^{0.182}$	$80.052^{6.868}$	$0.689^{0.102}$	$\underline{0.061}^{0.151}$	$0.181^{0.099}$	$0.421^{0.160}$	$0.074^{0.058}$
BoltzGen	$0.347^{0.152}$	$\underline{82.795}^{6.891}$	$0.700^{0.096}$	$0.043^{0.121}$	$0.174^{0.083}$	$0.418^{0.134}$	$0.082^{0.063}$
Germinal	$0.330^{0.133}$	$78.349^{7.575}$	$0.691^{0.103}$	$0.029^{0.120}$	$0.145^{0.091}$	$0.402^{0.117}$	$0.078^{0.059}$
RAFA	$\underline{0.370}^{0.163}$	$\mathbf{82.865}^{6.870}$	$\underline{0.710}^{0.094}$	$0.057^{0.153}$	$0.189^{0.093}$	$\underline{0.438}^{0.143}$	$\underline{0.094}^{0.068}$
RAFA (RL)	$\mathbf{0.425}^{0.154}$	$82.307^{7.177}$	$\mathbf{0.713}^{0.102}$	$\mathbf{0.070}^{0.128}$	$\mathbf{0.198}^{0.078}$	$\mathbf{0.452}^{0.139}$	$\mathbf{0.164}^{0.102}$

0.164 for VHH nanobodies. These correspond to relative improvements of 79.5% and 74.5% over RAFA, respectively. The advantage also extends to all compared external methods. For VH–VL, the strongest non-RAFA baseline on refolded DockQ is Germinal at 0.068, compared with 0.158 for RAFA (RL) (a 0.090 absolute difference and a 2.32× score). For VHH, the strongest non-RAFA baseline is BoltzGen at 0.082, compared with 0.164 for RAFA (RL) (a 0.082 difference and a 2.00× score). Table 16 summarizes these comparisons. Thus, improved docking accuracy against native references is observed after sequence refolding relative to both the untuned generator and the other evaluated design systems.

Complementary evidence from refolded-interface confidence. For VH–VL, RAFA (RL) achieves the highest reported mean across all seven refolding measures. Alongside the improvement in reference-based DockQ, ipSAE increases from 0.035 to 0.065 and pDockQ from 0.125 to 0.188 relative to RAFA. Both exceed the strongest external values, 0.031 and 0.134, respectively, obtained by BoltzGen. For VHH, RAFA (RL) leads on six of the seven reported means, including DockQ, global ipTM, ipSAE, and pDockQ. Relative to RAFA, global ipTM increases from 0.370 to 0.425, ipSAE from 0.057 to 0.070, and pDockQ from 0.189 to 0.198. The improvements in DockQ are accompanied by higher interface-confidence estimates, although the magnitude varies across metrics. Local pLDDT is the exception for VHH, decreasing from 82.865 to 82.307, so the docking improvements do not imply uniform gains in local structural confidence.

Table 16: Mean DockQ after Protenix-v2 refolding, evaluated against native reference complexes, and derived improvements from Tables 3 and 15. The best external method excludes both RAFA variants and is selected by mean refolded DockQ. Percentages and ratios are computed from the reported rounded means; they describe DockQ scores, not design success rates.

Format	RAFA	Best external method	RAFA (RL)	Gain over RAFA	Ratio to best external method
VH–VL	0.088	0.068 (Germinal)	**0.158**	+79.5%	2.32×
VHH	0.094	0.082 (BoltzGen)	**0.164**	+74.5%	2.00×

Persistence of the comparative advantage after refolding. For directly generated complexes, the mean DockQ advantage of RAFA (RL) over RAFA is 0.031 for VH–VL (0.172 versus 0.141) and 0.033 for VHH (0.147 versus 0.114). After Protenix-v2 refolding, the advantage is 0.070 in both formats. Both evaluations use the corresponding native complexes as references. These results show that the aggregate docking advantage persists when structures are predicted from the designed sequences. They support improved reference-based docking accuracy after refolding, while the mean scores do not specify the fraction of initially successful designs that remain successful or measure experimental binding.

F Post-Training Cost and Evaluation Summary

F.1 Measured RGT and Full-BPTT Cost

Panel D of Table 9 reports measured training costs on eight H20 GPUs with the same global batch size, 50-step rollouts, and 2,000 updates. RGT takes 24.0 hours and 192 GPU-hours, versus 55.0 hours and 440 GPU-hours for full BPTT, giving a $55.0/24.0 \approx 2.29\times$ speedup in this configuration. Peak allocated memory is 24 versus 72 GiB per GPU. RGT and full BPTT both achieve a reported mean DockQ of 0.172, while RGT reduces training time and allocated memory. Full BPTT has lower mean iRMS (8.275 versus 8.371) and LRMS (20.945 versus 21.062).

F.2 Analytical Operation Counts

The following operation-count model characterizes the denoiser work underlying RGT. Consider one condition with K trajectories, S sampling steps, and J recorded states per trajectory. Assume one denoiser evaluation per sampling step, one fresh trainable evaluation and one EMA-reference evaluation per selected state, and one parameter backward pass per fresh evaluation. The reported $K = 8$, $S = 50$, $J = 4$ then imply

$$N_{\text{rollout}} = KS = 400, \qquad N_{\text{fresh}} = N_{\text{EMA}} = N_{\text{backward}} = KJ = 32. \tag{37}$$

There are eight terminal reward evaluations and eight terminal coordinate-gradient evaluations per condition. The latter differentiate the reward, not the denoising trajectory. These are sample-equivalent evaluations: batching can reduce the number of program invocations without changing these counts.

If a forward evaluation costs F and a local parameter backward pass costs βF, the denoiser-only model is

$$C_{\text{RGT}}/F = K\left[S + J(2+\beta)\right]. \tag{38}$$

Using $\beta = 2$ in this analytical model gives 528 forward-equivalent units per condition; $\beta = 1$ or 3 gives 496 or 560. The model assigns equal forward costs to rollout, fresh, and EMA evaluations and counts denoiser work. End-to-end measurements additionally capture decoding, reward computation, optimizer work, communication, and any activation recomputation.

The full-rollout BPTT scenario requires $1200/528 \approx 2.27$ times the modeled denoiser work. RGT stores detached trajectory states and differentiates fresh local velocity evaluations. When these evaluations are processed sequentially, denoiser activations cover one local evaluation at a time. Panel D of Table 9 reports the corresponding end-to-end timing and peak allocated-memory measurements.

Table 17: Analytical denoiser operation counts per condition for $K = 8$, $S = 50$, and $\beta = 2$. The BPTT scenarios count differentiation through a terminal segment of length L with no reference penalty or activation recomputation.

Scenario	Expression in units of F	Value
Detached endpoint update ($J = 1$), with EMA	$K[S + 2 + \beta]$	432
RGT ($J = 4$), with EMA	$K[S + 4(2 + \beta)]$	528
Terminal-segment BPTT ($L = 4$), no EMA	$K[S + 4\beta]$	464
Full-rollout BPTT ($L = 50$), no EMA	$K[S + 50\beta]$	1200

With B conditions per global update and $U = 2000$ updates, the analytical RGT count is $1{,}056{,}000B$ forward-equivalent units and $16{,}000B$ terminal reward evaluations. The measured RGT run takes 24.0 hours on eight H20 GPUs, totaling 192 GPU-hours.

F.3 Baseline Implementation and Shared Post-Training Protocol

All post-training methods in the main comparison—RGT, Flow-GRPO, TempFlow-GRPO, RAM, and DiffusionNFT—start from the same RAFA base checkpoint and undergo 2,000 parameter updates. Every method evaluates the same composite reward R_{free} in Eq. equation 6, using identical frozen reward predictors and weights: $(0.60, 0.20, 0.15, 0.05)$ for affinity, fold confidence, physical plausibility, and developability; $(0.75, 0.25)$ within the affinity term; and $(0.60, 0.25, 0.15)$ within the geometric term. The fold-confidence surrogate is calibrated on Boltz-2-derived labels using only the RAFA training split. The baselines preserve their original optimization objectives and update procedures when applied to the RAFA generator.

Specifically, Flow-GRPO follows the stochastic transition-density formulation and clipping and advantage conventions of the original method (Liu et al., 2025a). TempFlow-GRPO additionally follows its prescribed branching and temporal-weighting scheme (He et al., 2025). RAM uses the original analytic noising law and reward-corrected regression target (Bergmeister et al., 2026), while DiffusionNFT follows its reward normalization, forward-noising construction, and positive/negative update rule (Zheng et al., 2025). These method-specific transformations operate on the shared reward; its component definitions and weights remain unchanged. RGT obtains coordinate gradients from its differentiable terms, while scalar-reward methods use the composite reward values. Matching 2,000 updates controls optimization length; measured wall time, GPU-hours, and memory for RGT and full BPTT are reported separately above.

F.4 Relationship to Matched DRTune

Shared gradient structure and the remaining distinction. DRTune (Wu et al., 2024) and RGT both use differentiable reward information to train selected sampling steps while stopping gradients through the denoiser inputs. Sparse supervision, detached network inputs, and associating a sampled path with its reward are therefore shared ideas. A key distinction concerns the coefficients used to assign terminal reward credit to local network outputs.

To make this distinction explicit, consider the affine sampler form underlying the DRTune derivation, written in increasing sampling order as

$$x_{n+1} = a_n x_n + b_n v_\theta(\mathrm{sg}(y_n), t_n; c) + \sigma_n \xi_n, \tag{39}$$

where the scalar coefficients are independent of the state and parameters, y_n includes the network inputs, and the noise realization is fixed. For a reward evaluated at x_S, let $g = \nabla_{x_S} R$ and let $\mathcal{J}$ be the trained steps. With untrained network outputs detached, the reward-gradient contribution is

$$G_{\mathrm{DT}} = \sum_{j \in \mathcal{J}} \left(\partial_\theta v_\theta(\mathrm{sg}(y_j), t_j; c)\right)^\top \left[b_j \prod_{\ell=j+1}^{S-1} a_\ell \right] g. \tag{40}$$

Thus, stopping the network-input gradients removes inter-step denoiser Jacobians but retains the sampler's linear propagation factors. This expression illustrates the gradient construction for the

stated sampler and endpoint; intermediate clean-state prediction or early stopping, as considered by DRTune, changes the corresponding endpoint map and coefficients.

RGT instead evaluates fresh coordinate velocities at recorded states and uses the detached directions $\eta\widetilde{w}_j\widehat{g}$, with $\widetilde{w}_j = t_{x,j}/(J^{-1}\sum_\ell t_{x,\ell})$. Its local update substitutes an identity approximation to future state sensitivity and explicitly allocates credit by normalized coordinate time, rather than retaining the propagation coefficients in Eq. equation 40. Gradient normalization controls the per-trajectory direction scale, and the EMA penalty constrains local velocity changes. In the special case $a_n = 1$ and constant b_n, the detached sampler contribution has constant step weights; RGT's linear coordinate-time weighting remains a distinct allocation choice. For other samplers, the comparison depends on their actual coefficients. This places RGT within the family of sparse direct-gradient methods while identifying the transport rule evaluated here.

Faithful algorithms with aligned experimental settings. All post-training baselines are implemented according to their original papers: their core objectives, gradient estimators, and method-specific update rules are retained, while common experimental settings are aligned with RGT. In particular, the matched DRTune control in Figure 4 uses the same RAFA generator and starting checkpoint, training conditions, composite reward and frozen reward predictors, and evaluation protocol. The common sampling and optimization configuration uses eight trajectories per condition, a 50-step rollout budget, four trained states, 2,000 parameter updates, learning rate 5×10^{-7}, and parameter gradient-norm clipping at 1. The DRTune control additionally matches RGT's reward-gradient scale and reference regularization, as specified in the main comparison. Consequently, an advantage over this matched control cannot be attributed solely to adding gradient normalization, regularization, more trained states, or a different generator. Method-specific temporal coefficients and gradient construction follow the respective algorithms. Equal update counts define this comparison; the separately measured speedup over full BPTT is not a measured speedup over DRTune.

Quantified contribution of temporal allocation. The matched DRTune comparison evaluates the complete update rules. The normalized-weight controls in Table 9A more narrowly isolate temporal allocation within RGT: they keep the actual sampled coordinate times, mean scalar weight, endpoint-gradient normalization, EMA coefficient, learning rate, trajectory count, and update count fixed. Replacing constant weights with normalized linear weights increases mean antibody-design DockQ from 0.164 to 0.172, an absolute gain of 0.008 (4.9% relative), and reduces iRMS from 8.570 to 8.371 Å and LRMS from 22.210 to 21.062 Å. Linear weighting also exceeds normalized quadratic and reversed weighting by 0.003 and 0.014 DockQ, respectively. Across the three inference seeds in Table 10, the linear-minus-constant DockQ differences are 0.007, 0.008, and 0.008, computed from the rounded entries.

These results support a contribution from temporal credit allocation beyond a constant-weight detached update. The constant-weight control is an RGT ablation, not the separately implemented DRTune baseline; its 0.008 gain therefore quantifies that component comparison rather than the RGT–DRTune difference. Likewise, the reference and normalization ablations in Table 9B establish their utility within RGT but do not explain an advantage over a DRTune control that already matches them. These are conditional component comparisons, rather than additive or statistically independent contributions.

F.5 Design Performance and Inference Seeds

All methods are evaluated on the same complete test sets using unified metric definitions and postprocessing, with no targets excluded because of method failures. On the 93 VH–VL targets, RGT increases mean design DockQ from 0.141 to 0.172, a 22.0% relative improvement, and SR from 0.204 to 0.280. On the 50 VHH targets, mean design DockQ increases from 0.114 to 0.147, a 28.9% relative improvement, and SR from 0.120 to 0.220.

Table 10 evaluates robustness to inference randomness using three sampling seeds. RGT improves mean DockQ over the base model in each seed, with gains of 0.030, 0.032, and 0.030. Panel B of Table 9 additionally reports training-seed standard deviations for the reference-penalty and gradient-normalization ablations.

Shared checkpoint and cross-format transfer. The base model is trained on both VH–VL and VHH data. RGT then post-trains it using VH–VL conditions, and the resulting single checkpoint is used for both antibody and nanobody prediction and design. The VHH results demonstrate transfer of interface-oriented VH–VL post-training to the VHH setting.