Improving protein-ligand complex generation with force field guidance

Helen Lai
Molecular AI, Discovery Sciences, R&D
AstraZeneca, UK
helen.lai1@astrazeneca.com

Tingyu Wang
NVIDIA, Santa Clara, CA, USA
tingyuw@nvidia.com

Hassan Sirelkhatim
NVIDIA, London, UK
hsirelkhatim@nvidia.com

Joe Eaton
NVIDIA, Santa Clara, CA, USA
featon@nvidia.com

Howard Huang
NVIDIA, Santa Clara, CA, USA
howardhuangb@gmail.com

Brad Rees
NVIDIA, Santa Clara, CA, USA
brees@nvidia.com

Ola Engkvist
Molecular AI, Discovery Sciences, R&D, AstraZeneca, Sweden
Chalmers University of Technology, Sweden
ola.engkvist@astrazeneca.com

Jon Paul Janet
Molecular AI, Discovery Sciences, R&D
AstraZeneca, Sweden
jonpaul.janet@astrazeneca.com

Xiaoyun Wang
NVIDIA, Santa Clara, CA, USA
xiaoyunw@nvidia.com

Alessandro Tibo*
Molecular AI, Discovery Sciences, R&D
AstraZeneca, Sweden
alessandro.tibo@astrazeneca.com

Abstract

Generative models based on diffusion and flow matching have recently been applied to structure-based drug design, but their outputs often include unrealistic protein–ligand interactions that fail to obey the laws of physics. We present an energy guidance framework that incorporates a molecular mechanics force field (MMFF94) directly into the sampling process. The method steers molecular generation toward more physically plausible and energetically stable conformations without retraining the underlying model. We evaluate this approach using two state-of-the-art architectures, SemlaFlow, a flow matching model and EDM, a diffusion model, on the PDBBind dataset. Across both models, energy guidance improves binding affinity predictions, reduces strain energy by up to 75%, and generates over 1,000 ligands with better docking scores than native ligands. These results

*Corresponding author.

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

demonstrate that lightweight, physics-based guidance can significantly enhance generative drug design while preserving chemical validity and diversity.

1 Introduction

Structure-based drug design (SBDD) plays a central role in modern drug discovery [1], focusing on the design and optimization of ligand molecules that exhibit strong binding affinity to a specific protein receptor site informed by an experimentally observed or predicted 3D structure of the target. By leveraging the three-dimensional structural information of target proteins, SBDD enables the rational design of compounds that bind tightly to the target structure, for example via formation of specific interactions with amino acids in the target [2] (such as hydrogen bonds) or occupying hydrophobic pockets resulting in the displacement of energetically unfavorable waters [3]. SBDD is traditionally achieved using molecular docking [4], which uses physics-inspired approaches to position a given molecule in various conformations relative to a static target structure to find the most favorable binding position by predicting and subsequently minimizing the strength and energy strain of that interaction. These docking programs typically incorporate force fields, parameterized potential energy surfaces for atomic systems that are used to estimate both the strength of the protein-ligand interaction as well as the strain of the ligand.

Recent advances in "3D generation methods" leveraging diffusion or flow matching methods now allow machine learning models to create binding poses [5] or directly design potential binders conditioned on a provided target structure in a purely data-driven manner. This last category is particularly attractive as these models can potentially directly propose ligands that are complementary to the given target, eliminating the need for an additional ligand search strategies and profiling of potentially millions of ligands to find binders. However, many groups [6–9] have identified that the poses generated by these methods often fail rudimentary sanity checks relating to physically achievable bond distances and angles, fail to make meaningful interactions with the targets or are simply nonphysically strained geometries. While improvements over the initial generation of such models have been made [10], to produce physically reasonable geometries it remains standard practice to minimize the proposed geometries with respect to a classical force field after generation.

Our Contributions

We introduce a novel, *training-free force field guidance* framework that steers ligand generation using empirical molecular mechanics (e.g., MMFF94) during diffusion or flow-based sampling—without modifying or retraining the base generative model (e.g., EDM or Semflaflow by [11]). Our method operates as a plug-in during inference time, leveraging energy feedback to generate poses with lower strain and better interactions with the protein structure.

Our main contributions are as follows:

- **Energy-based guidance without retraining**: Unlike methods that require gradients from neural affinity predictors (e.g., BADGER [12]), our approach injects classical force field feedback (MMFF94) directly during the posterior sampling step.
- **Improved docking and strain metrics**: In benchmarks against unconditional EDM and Semflaflow, our guided inference yields consistently better AutoDock Vina scores and lower ligand strain energy, even after optimizing the final structures using the same force field.
- **Compatibility and flexibility**: Because the guidance module is external, it can be applied broadly to multiple generative backbones—without retraining or architecture modifications, and can be applied to arbitrary differentiable potential energy functions.
- **Theoretical guarantee of stability.** We prove in Appendix B that our energy guidance framework decreases the energy of the generated conformations at each step, assuming Lipschitz continuity of the energy gradient.

2 Related Work

Guided Diffusion and flow matching models. In recent years, diffusion and flow matching models have gained significant popularity and demonstrated strong generative capabilities across diverse

domains such as text-to-image generation [13, 14], natural language processing [15], and molecular design for drug discovery [16, 11]. The original formulation of the diffusion and flow matching models support only unconditional generation. However, recent developments have introduced mechanisms to guide the generation process towards desired outputs. Two foundational approaches in this direction are classifier-guided diffusion and classifier-free guided diffusion.

In classifier-guided diffusion, an external classifier is trained separately to predict the target category from a given sample [17]. During inference, at each diffusion step, the gradient of the classifier's output with respect to the current sample is computed and added to the predicted noise estimate. In contrast, classifier-free guidance does not require a separate classifier. Instead, the diffusion model is jointly trained on both conditional and unconditional data [18]. At inference time, guidance is applied by taking a weighted combination of the conditional and unconditional noise predictions. A scaling factor controls the strength of the conditioning, allowing for flexible adjustment of the generation process without relying on external models. Both methods discussed above are primarily designed for categorical conditioning, with classifier-free guidance also supporting text embeddings. However, these approaches are insufficient for our setting, where the conditioning variable is continuous in nature—for example, a molecular force field.

Another line of work extends the conditioning framework to the continuous regime by training the model to directly learn the gradient of the log-conditional density, $\nabla_{x_t} \log p(x_t \mid y)$ [19], where x_t denotes the generated sample at time t and y the desiderata label. An alternative approach adopts a reinforcement learning paradigm, reformulating the iterative denoising process of a diffusion model as a multi-step Markov Decision Process (MDP). In this framework, policy gradient methods are applied to optimize the sampling trajectory such that the generated samples maximize a task-specific reward, such as human feedback [20, 21].

Although the two approaches above support continuous conditioning variables, they still require retraining the diffusion model for each new conditioning input. This limitation becomes particularly restrictive in structure-based drug design (SBDD), where different stages of a project—or entirely different projects—may involve varying conditioning inputs. The need to retrain the model for each new application is both time-consuming and computationally expensive. In the following sections, we describe the molecular generation task and detail the specific form of conditioning used in this work: differentiable molecular force field descriptors. We then introduce our proposed method—an adaptation of classifier-guided diffusion—that enables flexible integration of differentiable conditioning signals during sampling, without requiring retraining of the diffusion model nor the descriptors.

Molecular force fields. A molecular force field is a set of mathematical functions and parameters used to estimate the potential energy of a system of atoms based on their positions. Force fields are central to methods like molecular mechanics (MM) and molecular dynamics (MD) simulations. Over the years, various force fields have been developed for different applications—among the most prominent are AMBER [22], CHARMM [23], MMFF94 [24], and UFF [25]. AMBER and CHARMM are primarily tailored for large biomolecular systems such as proteins, peptides, and their interactions with ligands. While they offer high accuracy, their computational cost can be significant due to complex parameterization. This will become a major bottleneck in the current guidance framework where the force field evaluations need to be performed repeatedly during the denoising process. On the other hand, MMFF94 and UFF are designed for small, drug-like molecules and are much faster. However, UFF tends to be overly generic and is the least accurate among them, while MMFF94 shows a better balance between speed and accuracy, though it is traditionally limited to intra-ligand interactions [26]. In this work, we therefore rely on MMFF94, which we extend by conditioning it on the protein pocket. We provide a GPU implementation that enables fast and differentiable interaction modeling, making it suitable for integration into diffusion and flow matching sampling workflows.

3 Methods

In this section, we present a strategy for enhancing molecular sampling from flow matching and diffusion models guided by a chemo-physics score—specifically, the MMFF94 force field [24]. Importantly, our method does not require any fine-tuning of the pretrained diffusion model; instead, we act solely at inference time. This allows for a flexible integration of domain-specific knowledge without compromising the generality of the learned generative process.

We denote with $\mathcal{X}$ the molecular space, whose elements are molecules $X \in \mathcal{X}$, each represented as a graph $X = (V, E)$, where V is the set of nodes (atoms) and $E \subseteq V \times V$ is the set of edges (bonds). Each node $v \in V$ corresponds to an atom and is represented as a 3-tuple:

$$v = (x, a, c), \tag{1}$$

where $x \in \mathbb{R}^3$ denotes the 3D spatial coordinate of the atom, a is the atom type, and c is the formal charge. Note that both a and c are categorical variables. Each edge $e = (v_i, v_j) \in E$ corresponds to a bond between atoms v_i and v_j and is associated with a bond type attribute b_{ij}, which in our setting can be one of: single, double, triple, or aromatic. In addition to molecules, we model proteins (pockets) using a simpler representation. Unlike molecules, proteins are often provided as PDB files where explicit bond information is typically not included and must be inferred. Therefore, denoting the protein space as $\mathcal{Y}$, each protein $Y \in \mathcal{Y}$ is represented as a set of nodes $v \in V$, following the same semantic framework as in Equation 1. It is important to note that, throughout this paper, we do not distinguish between separate spaces for proteins and protein pockets, as the latter are regarded as a subset of the atoms comprising the original protein.

3.1 Conditional flow matching

Conditional flow matching [27–29] is a generative framework that directly models a continuous-time transport map between the noise distribution and the data distribution via an ordinary differential equation (ODE). A conditional flow matching defines a time-dependent conditional probability distribution $p_{t|1}(\cdot \mid z = (X_1, X_0))$, where $X_1 \in \mathcal{X}$ and $X_0 \sim p_{0|1}$ are a molecule and a sample drawn from a prior distribution $p_{0|1}$, respectively. A common choice for $p_{t|1}$, in the case of continuous variables [27], is a Gaussian distribution centered at the linear interpolation $X_t = tX_1 + (1-t)X_0$ with a constant standard deviation. From this conditional distribution, the conditional vector field $u(\cdot \mid t, z = (X_1, X_0))$ can be analytically derived as

$$u(\cdot \mid t, z) = X_1 - X_0. \tag{2}$$

We model the vector field in using a neural network parametrized by a set of weights θ, $u_\theta : [0,1] \times \mathcal{X} \to \mathcal{X}$, and train it to reconstruct the vector field defined in Equation 2. Instead of training the model to predict $X_1 - X_0$, we can train u_θ to reconstruct clean data X_1 from noisy inputs X_t [30, 31], and subsequently recover the underlying vector field. For instance, in the continuous setting, the following identity holds [11]:

$$X_1 - X_0 = \frac{1}{1-t}(X_1 - X_t). \tag{3}$$

To enable the generation of molecules that bind to protein targets, we extend the vector field u_θ to incorporate conditioning on a protein pocket $Y \in \mathcal{Y}$. We therefore redefine the neural network as

$$u_\theta : [0,1] \times \mathcal{X} \times \mathcal{Y} \to \mathcal{X}, \tag{4}$$

where $\mathcal{Y}$ denotes the space of protein pockets. To generate novel molecular samples, the vector field u_θ is integrated using a standard ODE solver. A basic Euler integration scheme is presented in Algorithm 1. As the backbone architecture, we adopt SemlaFlow [11], augmented to support conditioning on protein pockets. Details of this extension are provided in Section 3.2.

Algorithm 1 Conditional Flow Matching Sampling

Inputs: Number of steps T, learned vector field u_θ, prior $p_{t|1}$, protein Y.

Initialize $X_0 \sim p_{0|1}$, $\Delta T = \frac{1}{T}$
for $i = 0$ to $T - 1$ **do**
  $t = i \cdot \Delta T$
  $\hat{X}_1 = u_\theta(t, X_t, Y)$
  $v = \frac{1}{1-t}(\hat{X}_1 - X_t)$
  $X_{t+\Delta T} = X_t + \Delta T \cdot v$
end for
return X_1

3.2 Protein conditioning

For our diffusion and flow matching models, we adopt SemlaFlow [11] and the Equivariant Diffusion Model (EDM) [16], both extended to support protein conditioning. In this section, we describe the key architectural modifications introduced to enable conditioning on protein structures, particularly focusing on changes to the model layers.

Let us consider two tensors, $x \in \mathbb{R}^{m \times d}$ and $y \in \mathbb{R}^{n \times d}$, with dimensions $m \times d$ and $n \times d$, respectively. Here, x and y represent the feature vectors at any layer associated with ligand and protein atoms, respectively. The conditioning is applied within the attention layer, and its core idea can be summarized as follows:

$$x_i = x_i + \sum_{i \neq j} \frac{x_i - x_j}{\|x_i - x_j\|} \phi_{inv} + \sum_k \frac{x_i - y_k}{\|x_i - y_k\|} \psi_{inv}, \tag{5}$$

where the right-hand summation (highlighted in blue) in Equation 5 represents the protein conditioning we introduced. The functions, ϕ_{inv} and ψ_{inv} are learnable mappings applied to invariant features, such as atom and bond types.

3.3 Energy guidance

In the context of molecular generation, both diffusion models and flow matching have been leveraged to learn distributions over molecular graphs or 3D structures. However, these models are often trained solely on data likelihood objectives, potentially ignoring important physical or chemical properties that govern molecular stability. Our method addresses this gap by incorporating a chemo-physics score (MMFF94) into the inference process, biasing the generation towards physically plausible and energetically favorable molecules. Furthermore, we extended the two non-bonded interaction terms of MMFF94—van der Waals and electrostatic interactions—to account for the protein used to condition the generations. Formally, we denote the extended MMFF94 with a function mapping a protein and molecule into a real value number, $E : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$, defined as

$$E(X, Y) = \text{MMFF94}(X) + E_{vdW}(X, Y) + E_Q(X, Y), \tag{6}$$

where the terms $E_{vdW}(X, Y)$ and $E_Q(X, Y)$ model the Van der Waals and electrostatic interactions between a ligand X and a protein Y. The implementation is available at `https://github.com/MolecularAI/TorchMMFF94`. E is then used to steer the generation of the molecules towards regions with lower energies. To this end, we modified Algorithm 1 by incorporating an additional gradient-based term, scaled by a hyperparameter $\lambda > 0$ which governs the contribution of the gradient to the overall objective (see Algorithm 2). As shown in Appendix B, when λ is chosen below $2/L$ (with L denoting the local Lipschitz constant of $\nabla(E \circ f)$), each gradient-guided step guarantees non-increasing total energy, thereby maintaining numerical stability during sampling.

Algorithm 2 Conditional Flow Matching Sampling with Energy Guidance

Inputs: Number of steps T, learned vector field u_θ (as Equation 4), prior $p_{t|1}$, protein Y, energy function E, $\lambda > 0$.

1: Initialize $X_0 \sim p_{0|1}$, $\Delta T = \frac{1}{T}$
2: **for** $i = 0$ to $T - 1$ **do**
3: $\quad t = i \cdot \Delta T$
4: $\quad \hat{X}_1 = u_\theta(t, X_t, Y)$
5: $\quad v = \frac{1}{1-t}(\hat{X}_1 - X_t)$
6: $\quad X_{t+\Delta T} = X_t + \Delta T \cdot v - \lambda \nabla_{X_t} E(\hat{X}_1, Y)$
7: **end for**
8: **return** X_1

4 Experiments

4.1 Dataset

We use PDBBind [32] as our benchmark to demonstrate the quality of the generated ligands binding to proteins. PDBBind contains 19,443 protein–ligand complexes. From this dataset, we exclude 144

complexes to form our test set. These test pairs have no receptor overlap with the training set and are selected identically to those used in DiffDock [5] (i.e., `timesplit_test_no_rec_overlap` in the DiffDock repository). While there has been some disagreement regarding the test set chosen by the authors of DiffDock (see, e.g., [33]), our objective here is not docking but rather the generation of novel molecules within protein pockets. Therefore, this test set is suitable for our purposes.

We apply a series of preprocessing steps to the dataset using the Schrödinger suite [34]. First, we identify the protein associated with each ligand based on the distance between ligand and protein atoms. Next, we use Schrödinger's PrepWizard to prepare both the protein and the ligand, correcting geometries and assigning appropriate protonation states. After preparation, we recompute the docking score using Glide. If the complex successfully passes through the entire pipeline and receives a negative Glide score, we retain the pair; otherwise, it is discarded. The final training set consists of 18,990 protein–ligand pairs, and the test set includes 140 pairs. Although Schrödinger Glide [35] is a commercial software package, we release the code necessary to reproduce our preprocessing pipeline. However, running it will require a valid Schrödinger license.

Following [36], for each protein we extract a binding pocket by selecting all residues that have at least one atom within 3.5Å of the native ligand and contain more than 10 atoms in total.

4.2 Experimental setup

We evaluated our energy guidance method using two state-of-the-art generative models for molecular design: **SemlaFlow** [11], a flow matching-based generative model, and **EDM** [16] (in Appendix A), an equivariant diffusion model. Both models were initially pretrained on the **GeomDrugs** dataset [37], which contains approximately 37 million molecular conformations across more than 450,000 unique small molecules. Notably, this dataset does not include any protein structures. We adopted the default hyperparameters as reported in the respective original publications. Following pretraining, we fine-tuned each model on protein-ligand complexes from the **PDBBind** dataset, using the protein-ligand pair representation described in Section 4.1. For each protein pocket in the test set (140 targets in total), we generated 128 candidate ligands per model. To comprehensively assess the quality of the generated ligands, we employed a suite of evaluation metrics encompassing binding affinity, chemical validity, drug-likeness, intermolecular interactions, and conformational strain:
Glide Score: Estimated binding affinities computed using Schrödinger's Glide, a widely used commercial docking software. Glide employs a physics-based scoring function that combines molecular mechanics with empirical terms, accounting for van der Waals interactions, electrostatics, ligand strain, hydrophobic enclosure, hydrogen bonding, desolvation effects, and other force-field-derived contributions. Its scoring function is optimized to balance computational efficiency with predictive accuracy, making it suitable for high-throughput virtual screening and lead optimization [35].
Vina Score: An alternative open-source binding affinities estimation method. Vina employs an empirical scoring function that estimates ligand-protein binding based on steric complementarity, hydrogen bonding, hydrophobic interactions, and torsional flexibility penalties [38]. While its energy model is less detailed than Glide's physics-based scoring, Vina score remains a standard benchmark in molecular docking studies.
QED (Quantitative Estimate of Drug-likeness): A scalar score ranging from 0 to 1 that quantifies how drug-like a molecule is, with higher values indicating more favorable properties. QED integrates multiple physicochemical descriptors commonly associated with approved oral drugs, including molecular weight, lipophilicity (logP), number of hydrogen bond donors and acceptors, polar surface area, number of rotatable bonds, presence of structural alerts, and the number of aromatic rings [39].
PoseBuster Ratio (PBR): The proportion of generated ligands that pass all PoseBuster quality checks, serving as a proxy for structural and chemical plausibility [6].
Better-Than-Native Count (BNC): The number of generated ligands achieving a better (i.e., lower) docking score than the corresponding native ligand.
Validity: The percentage of generated molecules that are both syntactically correct (i.e., can be parsed into molecular graphs) and chemically interpretable. Validity is assessed using cheminformatics tools such as RDKit, which ensures that molecules can be successfully parsed from SMILES representations and can initialize a force field object (e.g., MMFF94).
Number of Interactions: The number of hydrogen bonds formed between the generated ligands and protein, computed using the `prolif` library [40].
Strain Energy: Defined as the difference in energy between the generated ligand conformation and its MMFF94-optimized geometry, normalized by the number of heavy atoms. Lower strain energy

indicates more realistic and energetically favorable molecular conformations.

Importantly, we evaluate performance under two settings: (1) using the raw, unrefined ligand conformations directly output by the generative models, and (2) using conformations that have been post-processed via conditional MMFF94 minimization. In contrast to prior approaches that perform full re-docking or extensive pose refinement, our strategy is intentionally lightweight—designed to preserve the original generative intent while allowing for minor energy-based adjustments.

4.3 Results

We report the results for SemlaFlow in Tables 1 and 2. Note that the metrics were computed directly on the generated ligands, without any re-docking. Table 1 reports the Vina and Glide negative ratios—that is, the proportion of generated ligands with negative scores ($\mathrm{VR} < 0$ and $\mathrm{GR} < 0$)—as well as the average Vina and Glide scores (VS and GS), both expressed in kcal/mol. We also report results after applying protein-conditioned post-optimization to the generated ligands (denoted by + OPT in the Tables). The results highlight the benefits of incorporating force-field guidance during inference. Using MMFF94 guidance alone yields substantial improvements: VR increases from 47.00% to 64.25%, while GR improves even more dramatically, from 19.41% to 56.61%. Correspondingly, VS shifts from an unfavorable 3.04 kcal/mol to a much more favorable -4.20 kcal/mol, indicating better ligand poses. Applying post-optimization alone to the baseline model without guidance also leads to significant gains, with VR reaching 64.98% and VS improving to -4.23 kcal/mol. However, the best performance is achieved by combining MMFF94 guidance with post-optimization, which yields the highest negative ratios for both metrics: 65.59% for VR and 59.06% for GR. Most notably, this combined strategy produces the strongest VS of -5.21 kcal/mol, representing an improvement of over 8 kcal/mol compared to the baseline model without guidance (3.04 kcal/mol).

Table 1: Vina and Glide docking score evaluation for SemlaFlow. $\mathrm{VR} < 0$ denotes negative vina score ratio, $\mathrm{GR} < 0$ denotes negative docking score ratio, VS denotes the average Vina score in kcal/mol, and GS denotes the average Glide score in kcal/mol.

METHOD	$\mathrm{VR} < 0$	$\mathrm{GR} < 0$	VS	GS
NO GUIDANCE	47.00%	19.41%	3.04	-4.45
GUIDANCE	64.25%	56.61%	-4.20	-4.81
NO GUIDANCE + OPT	64.98%	53.85%	-4.23	-5.21
GUIDANCE + OPT	65.59%	59.06%	-5.21	-5.03

Table 2 reports complementary metrics to docking scores, namely QED, PoseBuster ratio (PBR), better-than-native ligand count (BNC), validity (VALID), number of interactions (# INTERACTIONS), and strain energy (kcal/mol). Incorporating guidance increases BNC from 296 to 696 (a 135% improvement), while substantially reducing strain energy from 6.58 kcal/mol to 1.54 kcal/mol. This reduction is particularly important, as strain energy is directly influenced by guidance during inference, in contrast to docking scores, which improve more indirectly. The post-optimization strategy without guidance also provides considerable benefits, achieving a BNC of 731 and lowering strain energy to 1.04 kcal/mol. The combination of guidance with post-optimization delivers the strongest results, with the highest BNC of 1152 (nearly a 4-fold improvement over baseline) and the lowest strain energy of 0.78 kcal/mol (an 8-fold reduction). Overall, this combined strategy reduces strain energy from 6.58 kcal/mol to 0.78 kcal/mol while maintaining robust performance across other metrics, including 34.83% PBR, 67.50% validity, and high molecular diversity (97.16%).

We plot the distributions of Glide Score, Vina Score, and strain energy for molecules generated with and without guidance using the SemlaFlow model. Glide Score distributions (Figure 1 (a)) show that molecules generated with guidance exhibit a tighter distribution centered around lower (more favorable) scores compared to those without guidance. Vina Score distributions (Figure 1 (b)) follow a similar trend, where the guided molecules cluster more tightly around favorable scores (e.g., $<$ -5 kcal/mol), whereas the no-guidance set includes a wider spread and a greater number of high (less favorable) outliers. This again supports the conclusion that guidance improves the consistency and quality of the generated molecular poses. Strain Energy distributions (Figure 1 (c)) reveal that the guided group tends to produce molecules with significantly lower strain energy. The distribution is

Table 2: Quality metrics for SemlaFlow. QED reports the average quantitative estimate of drug-likeness. PBR denotes the PoseBuster pass ratio. BNC gives the number of compounds whose docking scores are better than those of the native ligands. VALID indicates the proportion of valid molecules among the generated set. # INTERACTIONS specifies the average number of hydrogen bonds formed between generated ligands and their target proteins. STRAIN ENERGY reports the average conformational strain energy. The overall molecular diversity is 97.16%.

METHOD	QED	PBR	BNC	VALID	# INTERACTIONS	STRAIN ENERGY
NO GUIDANCE	0.66	16.28%	296	69.55%	0.79	6.58
GUIDANCE	0.66	39.33%	696	67.72%	0.77	1.54
NO GUIDANCE + OPT	0.66	37.48%	731	69.55%	0.91	1.04
GUIDANCE + OPT	0.65	34.83%	1152	67.50%	0.86	0.78

sharply peaked around 2–3 kcal/mol for guided molecules, while the unguided set shows a broader distribution with a heavier tail, suggesting a higher incidence of conformational strain in the absence of guidance. Finally, Figure 2 depicts some examples showing the impact of the energy guidance.

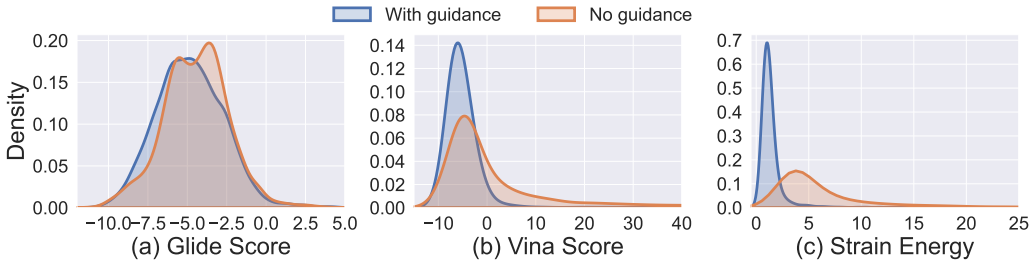

Figure 1: Distributions of (a) Glide Score, (b) Vina Score, and (c) Strain Energy for molecules generated with and without guidance using the SemlaFlow model. Guided molecules exhibit tighter distributions around more favorable scores and lower strain energy, whereas unguided molecules show broader distributions with higher variance and more unfavorable outliers.

5 Conclusion

This study introduces a novel energy-guided framework for protein-conditioned molecular generation that integrates physics-based MMFF94 force field constraints into diffusion and flow matching models. Our approach extends the MMFF94 force field to explicitly model protein-ligand interactions through van der Waals and electrostatic terms, enabling gradient-based steering during generation. It offers a lightweight yet effective alternative to existing methods that depend on computationally intensive re-docking or pose refinement. A comprehensive evaluation on the PDBBind dataset demonstrates substantial improvements across critical drug discovery metrics. Most notably, strain energy was reduced by 88% for SemlaFlow (6.58 to 0.78 kcal/mol/heavy atom) and 77% for EDM (3.73 to 0.87 kcal/mol/heavy atom), indicating significantly more energetically favorable conformations. Better-than-native counts increased by factors of 3.9× and 1.9×, respectively, while binding affinity predictions showed dramatic improvements, with negative Vina score ratios increasing from 47.00% to 65.59% for SemlaFlow. Compared to baseline models, where the loss function only encourages the model to match the data distribution in the training set, force-field guidance pushes samples toward physically plausible regions, leading to better docking scores out-of-the-box. The consistency of improvements across two distinct generative architectures establishes the broad applicability of our energy guidance approach. In addition to empirical improvements, our theoretical analysis offers a rigorous link between deep generative sampling and classical energy-descent theory. Importantly, these quality enhancements are achieved while maintaining high molecular diversity (>80%) and chemical validity, ensuring that improved binding characteristics do not restrict chemical space exploration.

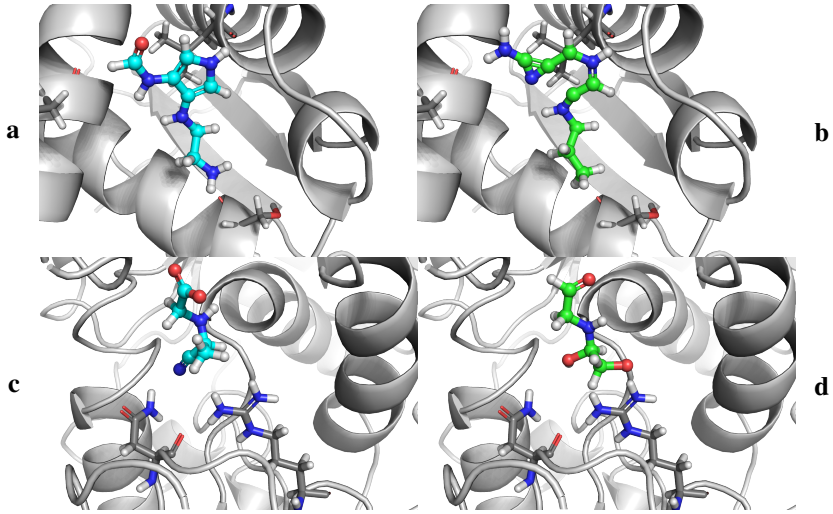

Figure 2: Some representative examples illustrating the impact of guidance terms on generated molecular composition from the EDM model. Results for generation with a fixed starting noise with guidance (a and c) and without guidance (b and d) for the Epstein-Barr Virus Nuclear Antigen-1 (top row, PBD:6NPM) and the Epstein-Barr Virus Nuclear Antigen-1 (top row, PBD:6NPM) and Aspartate semialdehyde dehydrogenase (bottom row, PBD:6C85). Proteins are shown as gray cartoons with some pocked residues shown with sticks, while generated ligands are shown with a ball-and-stick representations, with guidance samples colored with cyan carbons and default samples colored with green carbons.

References

[1] Amy C. Anderson. The process of structure-based drug design. *Chemistry & Biology*, 10(9): 787–797, Sep 2003. ISSN 1074-5521. doi: 10.1016/j.chembiol.2003.09.002.

[2] Xing Du, Yi Li, Yuan-Ling Xia, Shi-Meng Ai, Jing Liang, Peng Sang, Xing-Lai Ji, and Shu-Qun Liu. Insights into protein–ligand interactions: Mechanisms, models, and methods. *International Journal of Molecular Sciences*, 17(2), 2016. ISSN 1422-0067. doi: 10.3390/ijms17020144.

[3] Robert Abel, Tom Young, Ramy Farid, Bruce J. Berne, and Richard A. Friesner. Role of the active-site solvent in the thermodynamics of factor xa ligand binding. *Journal of the American Chemical Society*, 130(9):2817–2831, 2008. doi: 10.1021/ja0771033.

[4] Leonardo G. Ferreira, Ricardo N. Dos Santos, Glaucius Oliva, and Adriano D. Andricopulo. Molecular docking and structure-based drug design strategies. *Molecules*, 20(7):13384–13421, 2015. ISSN 1420-3049. doi: 10.3390/molecules200713384.

[5] Gabriele Corso, Hannes Stärk, Bowen Jing, Regina Barzilay, and Tommi S Jaakkola. Diffdock: Diffusion steps, twists, and turns for molecular docking. In *The Eleventh International Conference on Learning Representations*, 2023.

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

[7] Charles Harris, Kieran Didi, Arian R Jamasb, Chaitanya K Joshi, Simon V Mathis, Pietro Lio, and Tom Blundell. Benchmarking generated poses: How rational is structure-based drug design with generative models? *arXiv preprint arXiv:2308.07413*, 2023.

[8] Haoyang Liu, Yifei Qin, Zhangming Niu, Mingyuan Xu, Jiaqiang Wu, Xianglu Xiao, Jinping Lei, Ting Ran, and Hongming Chen. How good are current pocket-based 3d generative

models?: The benchmark set and evaluation of protein pocket-based 3d molecular generative models. *Journal of Chemical Information and Modeling*, 64(24):9260–9275, 2024. doi: 10.1021/acs.jcim.4c01598.

[9] Natasha Sanjrani, Damien E. Coupry, Peter Pogány, David S. Palmer, and Stephen D. Pickett. Benchmarking 3d structure-based molecule generators. *Journal of Chemical Information and Modeling*, Jul 2025. ISSN 1549-9596. doi: 10.1021/acs.jcim.5c01020.

[10] Julian Cremer, Ross Irwin, Alessandro Tibo, Jon Paul Janet, Simon Olsson, and Djork-Arné Clevert. Flowr: Flow matching for structure-aware de novo, interaction-and fragment-based ligand generation. *arXiv preprint arXiv:2504.10564*, 2025.

[11] Ross Irwin, Alessandro Tibo, Jon Paul Janet, and Simon Olsson. Semlaflow–efficient 3d molecular generation with latent attention and equivariant flow matching. In *The 28th International Conference on Artificial Intelligence and Statistics*, 2025.

[12] Yue Jian, Wu Curtis, and Danny Reidenbach. General binding affinity guidance for diffusion models in structure-based drug design. In *arxiv*, 2024.

[13] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents, 2022.

[14] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022.

[15] Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in neural information processing systems*, 34:17981–17993, 2021.

[16] Emiel Hoogeboom, Vıctor Garcia Satorras, Clément Vignac, and Max Welling. Equivariant diffusion for molecule generation in 3d. In *International conference on machine learning*, pages 8867–8887. PMLR, 2022.

[17] Shiyuan Zhang, Weitong Zhang, and Quanquan Gu. Energy-weighted flow matching for offline reinforcement learning. *arXiv preprint arXiv:2503.04975*, 2025.

[18] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

[19] Georgios Batzolis, Jan Stanczuk, Carola-Bibiane Schönlieb, and Christian Etmann. Conditional image generation with score-based diffusion models. *arXiv preprint arXiv:2111.13606*, 2021.

[20] Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. *arXiv preprint arXiv:2305.13301*, 2023.

[21] Yulai Zhao, Masatoshi Uehara, Gabriele Scalia, Sunyuan Kung, Tommaso Biancalani, Sergey Levine, and Ehsan Hajiramezanali. Adding conditional control to diffusion models with reinforcement learning. *arXiv preprint arXiv:2406.12120*, 2024.

[22] David A Case, Thomas E Cheatham III, Tom Darden, Holger Gohlke, Ray Luo, Kenneth M Merz Jr, Alexey Onufriev, Carlos Simmerling, Bing Wang, and Robert J Woods. The amber biomolecular simulation programs. *Journal of computational chemistry*, 26(16):1668–1688, 2005.

[23] Bernard R Brooks, Robert E Bruccoleri, Barry D Olafson, David J States, S a Swaminathan, and Martin Karplus. Charmm: a program for macromolecular energy, minimization, and dynamics calculations. *Journal of computational chemistry*, 4(2):187–217, 1983.

[24] Thomas A Halgren. Merck molecular force field. i. basis, form, scope, parameterization, and performance of mmff94. *Journal of computational chemistry*, 17(5-6):490–519, 1996.

[25] Anthony K Rappé, Carla J Casewit, KS Colwell, William A Goddard III, and W Mason Skiff. Uff, a full periodic table force field for molecular mechanics and molecular dynamics simulations. *Journal of the American chemical society*, 114(25):10024–10035, 1992.

[26] Toby Lewis-Atwell, Piers A Townsend, and Matthew N Grayson. Comparisons of different force fields in conformational analysis and searching of organic molecules: A review. *Tetrahedron*, 79:131865, 2021.

[27] 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.

[28] Xingchao Liu, Chengyue Gong, et al. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *The Eleventh International Conference on Learning Representations*, 2023.

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

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

[31] Andrew Campbell, Jason Yim, Regina Barzilay, Tom Rainforth, and Tommi Jaakkola. Generative flows on discrete state-spaces: Enabling multimodal flows with applications to protein co-design. In *International Conference on Machine Learning*, pages 5453–5512. PMLR, 2024.

[32] Renxiao Wang, Xueliang Fang, Yipin Lu, Chao-Yie Yang, and Shaomeng Wang. The pdbbind database: methodologies and updates. *Journal of medicinal chemistry*, 48(12):4111–4119, 2005.

[33] Ajay N Jain, Ann E Cleves, and W Patrick Walters. Deep-learning based docking methods: Fair comparisons to conventional docking workflows. *arXiv preprint arXiv:2412.02889*, 2024.

[34] Schrödinger, LLC. Schrödinger Release 2025-2: Maestro, 2025. Schrödinger, LLC, New York, NY.

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

[36] Zaixi Zhang, Wan Xiang Shen, Qi Liu, and Marinka Zitnik. Efficient generation of protein pockets with pocketgen. *Nature Machine Intelligence*, pages 1–14, 2024.

[37] Simon Axelrod and Rafael Gómez-Bombarelli. Geom, energy-annotated molecular conformations for property prediction and molecular generation. *Scientific Data*, 9(1):185, 2022. doi: 10.1038/s41597-022-01288-4. URL `https://doi.org/10.1038/s41597-022-01288-4`.

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

[39] G Richard Bickerton, Gaia V Paolini, Jérémy Besnard, Sorel Muresan, and Andrew L Hopkins. Quantifying the chemical beauty of drugs. *Nature chemistry*, 4(2):90–98, 2012.

[40] Cédric Bouysset and Sébastien Fiorucci. Prolif: a library to encode molecular interactions as fingerprints. *Journal of cheminformatics*, 13(1):72, 2021.

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

[42] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. pmlr, 2015.

[43] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International conference on machine learning*, pages 8162–8171. PMLR, 2021.

A Diffusion models

Alternatively to conditional flow matching models, diffusion models [41] are another class of generative models that learn to sample complex data distributions by learning to reverse a process that adds noise to the data. For clarity and consistency throughout the paper, we slightly depart from the standard notation commonly used in diffusion models by introducing a relabeling function, $\tau(t) = \lfloor T\,(1-t) \rceil$, where $\lfloor \cdot \rceil$ is the round to nearest integer operator, $t \in [0,1]$ denotes the normalized time and T is the total number of time steps. With this convention, we denote the clean sample as $X_{\tau(1)}$ and the noisy sample as $X_{\tau(0)}$. Note that this remains consistent with the typical diffusion model notation, where $X_{\tau(1)} = X_0$ (clean) and $X_{\tau(0)} = X_T$ (noisy). This choice aligns with the flow matching notation introduced in Section 3, enabling a unified presentation across both paradigms. Diffusion models consist of two steps: forward and reverse processes. In the forward process, a sample from the data distribution is progressively perturbed by adding noise, eventually mapping it to a simple known prior distribution $p_{\tau(0)}$ (e.g., Gaussian noise for continuous data). The reverse process is then learned via a neural network that gradually denoises the sample, reconstructing a data point from the noise.

Formally, given a data point $X_{\tau(1)}$, the forward process defines a Markov chain of T steps:

$$X_{\tau(1)} \to X_{\tau(1-\Delta T)} \to \cdots \to X_{\tau(0)}, \tag{7}$$

where $\Delta T = 1/T$ and $X_{\tau(0)} \sim p_{\tau(0)}$ for sufficiently large T. The reverse process is modeled by a neural network p_θ, parameterized by weights θ, and can be formulated in several equivalent ways: by directly estimating $X_{\tau(t+\Delta T)}$ from $X_{\tau(t)}$ [42], by predicting the noise added at each step [41], or by predicting the original clean sample $X_{\tau(1)}$ [43]. For convenience, we adopt the latter formulation, i.e., modeling

$$p_\theta(X_{\tau(1)} \mid X_{\tau(t)}), \tag{8}$$

as it enables direct computation of molecular energy at each step based on the current estimate of the original structure. Additionally, we extend our model to condition on proteins, resulting in modeling

$$p_\theta(X_{\tau(1)} \mid X_{\tau(t)}, Y). \tag{9}$$

As the backbone architecture, we adopt EDM [16], augmented to support conditioning on protein pockets (see Section 3.2 for the details). The sampling procedure is depicted in Algorithm 3. COMPUTEPOSTERIOR allows to samples $X_{\tau(t+\Delta T)}$ given $\hat{X}_{\tau(1)}$ and $X_{\tau(t)}$.

Algorithm 3 Diffusion Model Sampling

Inputs: Number of steps T, learned denoiser p_θ (as Equation 9), prior $p_{\tau(0)}$, protein Y

1: Initialize $X_{\tau(0)} \sim p_{\tau(0)}$
2: **for** $i = 0$ to $T-1$ **do**
3: $\quad t = i \cdot \Delta T$
4: $\quad \hat{X}_{\tau(1)} \sim p_\theta(X_{\tau(1)} \mid X_{\tau(t)}, Y)$
5: $\quad X_{\tau(t+\Delta T)}$ = COMPUTEPOSTERIOR$(\hat{X}_{\tau(1)}, X_{\tau(t)})$
6: **end for**
7: **return** $X_{\tau(1)}$

The COMPUTEPOSTERIOR function models the distribution $\mathcal{N}(X_{\tau(t+\Delta T)}; \mu_t(\hat{X}_{\tau(1)}, X_{\tau(t)}), \tilde{\beta} I)$, where

$$\mu_t(\hat{X}_1, X_{\tau(t)}) = \frac{\sqrt{\bar{\alpha}_{\tau(t+\Delta T)}}\,\beta_{\tau(t)}}{1-\bar{\alpha}_{\tau(t)}}\,\hat{X}_{\tau(1)} + \frac{\sqrt{\alpha_{\tau(t)}}\,(1-\bar{\alpha}_{\tau(t+\Delta T)})}{1-\bar{\alpha}_{\tau(t)}}\,X_{\tau(t)},$$

and

$$\tilde{\beta} = \frac{1-\bar{\alpha}_{\tau(t+\Delta T)}}{1-\bar{\alpha}_{\tau(t)}}\,\beta_{\tau(t)}, \quad \alpha_{\tau(t)} = 1-\beta_{\tau(t)}, \quad \bar{\alpha}_{\tau(t)} = \prod_{s=\tau(1-\Delta T)}^{\tau(t)} \alpha_s.$$

The values $\beta_{\tau(t)}$ are typically chosen deterministically, with a common choice being the cosine schedule [43]. Algorithm 4 shows the inference procedure guided by the energy function.

Algorithm 4 Diffusion Model Sampling with Energy Guidance

Inputs: Number of steps T, learned denoiser p_θ (as Equation 9), prior $p_{\tau(0)}$, protein Y, energy function E, $\lambda > 0$.

Initialize $X_{\tau(0)} \sim p_{\tau(0)}$
for $i = 0$ to $T - 1$ **do**
  $t = i \cdot \Delta T$
  $\hat{X}_{\tau(1)} \sim p_\theta(X_{\tau(1)} \mid X_{\tau(t)}, Y)$
  $\hat{X}_{\tau(t+\Delta T)}$ = COMPUTEPOSTERIOR$(\hat{X}_{\tau(1)}, X_{\tau(t)}) - \lambda \nabla_{X_{\tau(t)}} E(\hat{X}_{\tau(1)}, Y)$
end for
return $X_{\tau(1)}$

Similarly to SemlaFlow, Tables 3 and 4 report the same quality metrics for EDM. Consistent with SemlaFlow, the results here demonstrate a clear trend of improved performance across most metrics, with the combined guidance and post-optimization approach achieving superior outcomes in the majority of evaluation criteria. In Table 3, VR increases from 64.26% in the baseline to 74.45% with guidance and post-optimization, and VS improves from 1.01 kcal/mol to –4.05 kcal/mol. By contrast, GS remains relatively stable across all methods (–5.16 kcal/mol to –4.76 kcal/mol), while GR shows a marked increase from 20.84% in the baseline to 47.96% with guidance and post-optimization.

Table 3: Vina and Glide docking score evaluation for EDM. VR < 0 denotes negative vina score ratio, GR < 0 denotes negative docking score ratio, VS denotes the average Vina score in kcal/mol, and GS denotes the average Glide score in kcal/mol.

METHOD	VR < 0	GR < 0	VS	GS
NO GUIDANCE	64.26%	20.84%	1.01	-5.16
GUIDANCE	68.45%	39.49%	-2.43	-5.19
NO GUIDANCE + OPT	72.81%	35.23%	-2.43	-4.92
GUIDANCE + OPT	74.45%	47.96%	-4.05	-4.76

Table 4 presents the molecule quality metrics for the EDM model. PBR increases notably from 25.23% in the baseline to 37.51% for the guidance case. BNC more than doubles from 540 to 1118, demonstrating that guidance enables the generation of significantly more molecules that outperform native binding conformations. The strain energy also shows considerable improvement, decreasing from 3.73 kcal/mol to 2.67 kcal/mol, reflecting more energetically favorable molecular conformations. When adopting post-optimization BNC increases from 540 to 801, while strain energy reduces from 3.73 kcal/mol to 1.25 kcal/mol. PBR also improves from 25.23% to 33.66%. With guidance and post-optimization, the strain energy decreases to 0.87 kcal/mol, representing a substantial 77% reduction compared to the baseline value of 3.73 kcal/mol. Note that post-optimization alone does not yield the best results in strain energy; such improvements are observed only when it is combined with guidance.

Table 4: Quality metrics for EDM. QED reports the average quantitative estimate of drug-likeness. PBR denotes the PoseBuster pass ratio. BNC gives the number of compounds whose docking scores are better than those of the native ligands. VALID indicates the proportion of valid molecules among the generated set. # INTERACTIONS specifies the average number of hydrogen bonds formed between generated ligands and their target proteins. STRAIN ENERGY reports the average conformational strain energy. The overall molecular diversity is 81.30%

METHOD	QED	PBR	BNC	VALID	# INTERACTIONS	STRAIN ENERGY
NO GUIDANCE	0.46	25.23%	540	81.26%	2.19	3.73
GUIDANCE	0.42	37.51%	1118	77.68%	2.13	2.67
NO GUIDANCE + OPT	0.46	33.66%	801	81.26%	1.83	1.25
GUIDANCE + OPT	0.42	33.85%	1052	77.68%	1.79	0.87

B Theoretical insights

Lemma 1 (Descent lemma) *Let $g : \mathbb{R}^n \to \mathbb{R}$ be continuously differentiable with L–Lipschitz gradient:*

$$\|\nabla g(u) - \nabla g(v)\| \le L\|u - v\| \quad \forall\, u, v \in \mathbb{R}^n.$$

Then for all $x, y \in \mathbb{R}^n$,

$$g(y) \;\le\; g(x) \;+\; \nabla g(x)^\top (y - x) \;+\; \tfrac{L}{2}\,\|y - x\|^2.$$

Fix $x, y \in \mathbb{R}^n$ and define the line segment $\gamma(t) = x + t(y - x)$ for $t \in [0, 1]$. By the Fundamental Theorem of Calculus applied to the scalar function $t \mapsto g(\gamma(t))$,

$$g(y) - g(x) \;=\; \int_0^1 \frac{d}{dt}\, g(\gamma(t))\, dt \;=\; \int_0^1 \nabla g(\gamma(t))^\top \gamma'(t)\, dt \;=\; \int_0^1 \nabla g(\gamma(t))^\top (y - x)\, dt.$$

Add and subtract $\nabla g(x)$ inside the integrand:

$$g(y) - g(x) = \nabla g(x)^\top (y - x) + \int_0^1 \big(\nabla g(\gamma(t)) - \nabla g(x)\big)^\top (y - x)\, dt.$$

By Cauchy–Schwarz and the Lipschitz property of ∇g,

$$\big|(\nabla g(\gamma(t)) - \nabla g(x))^\top (y - x)\big| \;\le\; \|\nabla g(\gamma(t)) - \nabla g(x)\|\, \|y - x\| \;\le\; L\,\|\gamma(t) - x\|\, \|y - x\| \;=\; L\,t\,\|y - x\|^2.$$

Therefore,

$$g(y) - g(x) \;\le\; \nabla g(x)^\top (y - x) \;+\; \int_0^1 L\,t\,\|y - x\|^2\, dt \;=\; \nabla g(x)^\top (y - x) \;+\; \tfrac{L}{2}\,\|y - x\|^2,$$

which is the claimed inequality.

Theorem 2 *Let $E : \mathbb{R}^m \to \mathbb{R}$ and $f : \mathbb{R}^n \to \mathbb{R}^m$ be C^1 functions, and define the composite*

$$g(x) := E(f(x)), \quad x \in \mathbb{R}^n.$$

Suppose g is L–smooth, i.e.,

$$\exists L > 0 \;\; \textit{such that} \;\; \|\nabla g(x) - \nabla g(y)\| \le L\|x - y\|, \quad \forall x, y \in \mathbb{R}^n.$$

Then it follows that

$$E(f(x - \lambda \nabla g(x))) \le E(f(x)), \quad 0 < \lambda < \tfrac{2}{L}, \tag{10}$$

with strict inequality whenever $\nabla g(x) \neq 0$.

Step 1 (Descent lemma). For any L–smooth function g, we have

$$g(y) \le g(x) + \nabla g(x)^\top (y - x) + \frac{L}{2}\|y - x\|^2. \tag{11}$$

—

Step 2 (Gradient step). Choose

$$y = x - \lambda \nabla g(x), \quad \lambda > 0.$$

Substituting into (11) gives

$$\begin{aligned} g(x - \lambda \nabla g(x)) &\le g(x) + \nabla g(x)^\top (-\lambda \nabla g(x)) + \frac{L}{2}\|-\lambda \nabla g(x)\|^2 \\ &= g(x) - \lambda \|\nabla g(x)\|^2 + \frac{L\lambda^2}{2}\|\nabla g(x)\|^2. \end{aligned}$$

—

Step 3 (Condition on step size). Rearranging yields

$$g(x - \lambda \nabla g(x)) \le g(x) - \Big(\lambda - \tfrac{L\lambda^2}{2}\Big)\|\nabla g(x)\|^2. \tag{12}$$

If $0 < \lambda < \frac{2}{L}$, then the coefficient $\lambda - \frac{L\lambda^2}{2}$ is strictly positive. Hence: - If $\nabla g(x) \neq 0$, inequality (12) is strict:

$$g(x - \lambda \nabla g(x)) < g(x).$$

- If $\nabla g(x) = 0$, then x is a stationary point and equality holds:

$$g(x - \lambda \nabla g(x)) = g(x).$$

—

Conclusion. Since $g(x) = E(f(x))$, we deduce that under the L–smoothness assumption on g, one gradient step yields

$$E(f(x - \lambda \nabla g(x))) \leq E(f(x)), \quad 0 < \lambda < \frac{2}{L},$$

with strict inequality whenever $\nabla g(x) \neq 0$.