

MFBind: a Multi-Fidelity Approach for Evaluating Drug Compounds in Practical Generative Modeling

Peter Eckmann¹ Dongxia Wu¹ Germano Heinzelmann² Michael K Gilson^{3,4} Rose Yu¹

Abstract

Current generative models for drug discovery primarily use molecular docking to evaluate the quality of generated compounds. However, such models are often not useful in practice because even compounds with high docking scores do not consistently show experimental activity. More accurate methods for activity prediction exist, such as molecular dynamics based binding free energy calculations, but they are too computationally expensive to use in a generative model. We propose a multi-fidelity approach, Multi-Fidelity Bind (MFBind), to achieve the optimal trade-off between accuracy and computational cost. MFBind integrates docking and binding free energy simulators to train a multi-fidelity deep surrogate model with active learning. Our deep surrogate model utilizes a pretraining technique and linear prediction heads to efficiently fit small amounts of high-fidelity data. We perform extensive experiments and show that MFBind (1) outperforms other state-of-the-art single and multi-fidelity baselines in surrogate modeling, and (2) boosts the performance of generative models with markedly higher quality compounds.

1. Introduction

Generative models for *de novo* drug design have gained significant interest in machine learning for their promised ability to quickly generate new compounds. However, generating compounds with real-world activity remains a fundamental challenge (Handa et al., 2023; Coley et al., 2020),

¹Department of Computer Science and Engineering, University of California San Diego, La Jolla, California, United States

²Departamento de Física, Universidade Federal de Santa Catarina, Florianópolis, Santa Catarina, Brazil ³Department of Chemistry and Biochemistry, University of California San Diego, La Jolla, California, United States ⁴Skaggs School of Pharmacy and Pharmaceutical Sciences, University of California San Diego, La Jolla, California, United States. Correspondence to: Michael Gilson <mgilson@health.ucsd.edu>, Rose Yu <roseyu@ucsd.edu>.

limiting the widespread adoption of generative models in practical drug discovery (Paul et al., 2021). One of the main difficulties is the computational evaluation of compound binding affinity. The generated compounds are often highly novel, so an activity predictor trained with existing experimental data is insufficient due to poor out-of-distribution generalization (Chatterjee et al., 2023; Ji et al., 2022). Instead, physics-based methods that model the 3D interaction between compound and target are commonly used.

Due to its speed, molecular docking is the prevalent physics-based method used to evaluate novel compounds by generative models (Eckmann et al., 2022; Jeon & Kim, 2020; Lee et al., 2023; Noh et al., 2022; Fu et al., 2022; Spiegel & Durrant, 2020; Peng et al., 2022; Guan et al., 2023a;b). However, docking is known to be a relatively poor predictor of activity (Pinzi & Rastelli, 2019; Handa et al., 2023; Coley et al., 2020; Feng et al., 2022), so it would be desirable to apply more accurate binding free energy calculation techniques (Pinzi & Rastelli, 2019; Feng et al., 2022). Such techniques, utilizing molecular dynamics simulations, are currently considered the most reliable approach to computational prediction of affinity (Moore et al., 2023; Cournia et al., 2021). However, they have not been used by generative models due to their high computational cost (Thomas et al., 2023), with a single compound-protein pair taking hours to days to simulate on a powerful computer (Wan et al., 2020). Thus, neither docking nor binding free energy techniques alone are sufficient for the practical application of generative models.

Multi-fidelity modeling (Fernández-Godino et al., 2016) is an approach to integrate data from simulators with varying accuracy and costs. Multi-fidelity modeling has been successfully applied in scientific areas such as climate modeling (Wu et al., 2022) and materials science (Fare et al., 2022), but their adoption in drug discovery has been limited. Hernandez-Garcia et al. (2023) study their use in peptide design, but they construct a proof-of-concept artificial set of fidelities and costs where even the highest fidelity is not expected to be very accurate.

In this paper, we address the difficulty of drug compound binding affinity evaluation by proposing a multi-fidelity modeling framework, Multi-Fidelity Bind (**MFBind**), to

achieve the optimal trade-off between accuracy and computational cost. Our framework (Figure 1) consists of a new multi-fidelity environment for binding affinity prediction, and a deep surrogate model that integrates data from each fidelity level to accurately and cheaply mimic the behavior of the binding free energy method. Our model learns a shared compound encoding across all fidelities, and then uses regularized linear heads to output predictions at each fidelity level. We also pretrain the surrogate model on the large quantity of lower fidelity data, and then fine-tune the model on all fidelities, using active learning. More specifically,

- we introduce a novel multi-fidelity modeling framework, **MFBind**, for evaluating the binding affinity of drug compounds in generative modeling that integrates data from existing experimental results, molecular docking, and binding free energy simulators
- we propose a deep surrogate model, which utilizes a pretraining technique on the data from the lower set of fidelities, and efficiently train it using a cost-aware multi-fidelity active learning approach.
- we perform extensive evaluations of ours and baseline models on two real-world problem settings, multi-fidelity surrogate modeling and generative modeling, showing the practicality of our framework.

2. Related Work

2.1. Molecular generative models

Generative models in drug discovery have gained much interest for their ability to quickly generate compounds with desired properties (Paul et al., 2021). Early works (Jin et al., 2018; Gómez-Bombarelli et al., 2018; You et al., 2018) focus on properties such as the octanol-water partition coefficient (logP) or quantitative estimate of drug-likeness (QED), which are of very limited practical utility (Coley et al., 2020; Xie et al., 2021). More recently, there has been an understanding that the binding affinity to a targeted protein is much more relevant for practical drug discovery (Xie et al., 2021; Eckmann et al., 2022; Fu et al., 2022).

One approach to guide generative models in optimizing compound binding affinity is to use a reward function for compound evaluation. This reward function can be applied to reinforcement learning (Jeon & Kim, 2020; Fu et al., 2022), VAEs (Eckmann et al., 2022; Noh et al., 2022), genetic algorithms (Spiegel & Durrant, 2020; Fu et al., 2022), or diffusion models (Lee et al., 2023). All of them use docking software, such as AutoDock (Morris et al., 2009), as the reward function, because it is the only reasonably fast option. However, docking is known to be inaccurate (Pinzi & Rastelli, 2019), and compounds with high docking scores

do not consistently show experimental activity (Handa et al., 2023; Coley et al., 2020; Feng et al., 2022).

More reliable molecular dynamics-based binding free energy calculations, which are much more accurate than docking (Moore et al., 2023; Cournia et al., 2021), have not yet been applied to *de novo* generative drug design due to their high computational cost (Thomas et al., 2023). While Ghanakota et al. (2020) use binding free energy calculations in combination with a molecular generative model, they focus on the optimization of an existing known lead compound. This allows them to rely on much cheaper relative binding free energy calculations, as opposed to the absolute binding free energy (ABFE) calculations needed for *de novo* design (Cournia et al., 2017).

Structure-based generative models are trained on 3D structures of protein-ligand pairs, and aim to predict a 3D ligand that fits in a given protein pocket with high binding affinity. Techniques include autoregressive generation (Peng et al., 2022) and diffusion modeling (Guan et al., 2023a;b). Despite not needing a reward function like docking during the generation process, the generated compounds are still evaluated with docking as a post-processing step. This means structure-based generative models do not avoid the issue of inaccurate binding affinity prediction.

2.2. Multi-fidelity modeling

Multi-fidelity modeling methods aim to fuse multiple data sources of variable accuracy and cost (Fernández-Godino et al., 2016), and are widely used in scientific fields for surrogate modeling and uncertainty quantification (Brevault et al., 2020). A popular choice of surrogate model is a Gaussian process (GP), which performs well in low data settings and produces well-calibrated uncertainty estimates (Brevault et al., 2020). One such technique to apply GPs to multi-fidelity modeling is described by Wu et al. (2020), where a downsampling kernel is used to output predictions at each fidelity level. While GPs cannot scale to large amounts of data, approaches like KISS-GP (Wilson & Nickisch, 2015) and DKL (Wilson et al., 2016) learn a deep neural kernel to scale GPs. Other surrogate modeling approaches utilize neural processes (Wang & Lin, 2020; Wu et al., 2022) and ordinary differential equations (Li et al., 2022).

Multi-fidelity modeling is frequently used in an active learning context, where one uses an estimate of a model's uncertainty to most efficiently acquire more datapoints (Ren et al., 2021). In the multi-fidelity setting, this means iteratively querying across both the sampling space of molecules and each different fidelity level (Li et al., 2020; Hernandez-Garcia et al., 2023). This approach has been applied in climate modeling (Wu et al., 2022), fluid dynamics (Li et al., 2020; Wang et al., 2021), and materials science (Fare et al., 2022). For drug discovery, Hernandez-Garcia et al. (2023)

Multi-Fidelity Bind (MFBind)

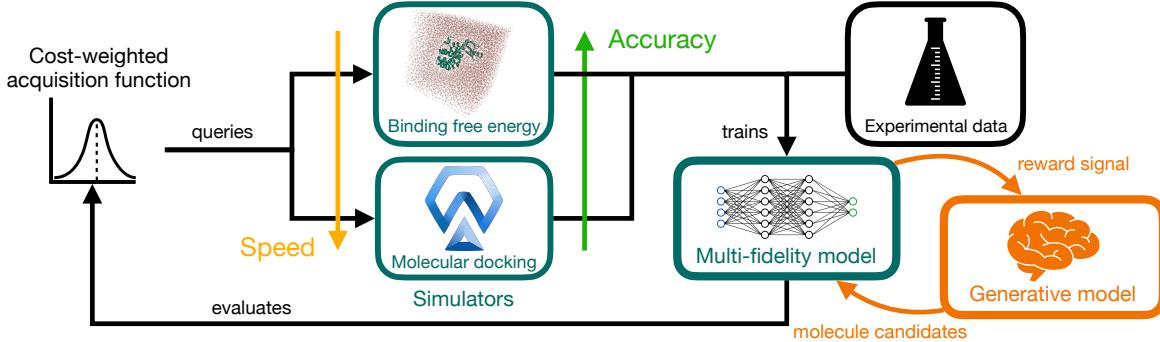


Figure 1. Overview of MFBind. We train a multi-fidelity surrogate model to predict the outputs from all fidelity simulators. Then, we use the model to evaluate the acquisition function, and then pick the next molecule and fidelity level to query the simulators. The result is then added to the training dataset, and the process is repeated. A generative model uses the trained multi-fidelity surrogate model to evaluate its candidate compounds.

seek to design peptides with anti-microbial activity using multi-fidelity active learning, but they construct an artificial hierarchy of fidelities and associated costs by training the same machine learning model on different data subsets. This means even the highest fidelity is likely to not be very accurate, as machine learning predictors are typically not as accurate as physics-based methods (Moore et al., 2023; Cournia et al., 2021; Chatterjee et al., 2023) and suffer from out-of-distribution generalization issues (Ji et al., 2022).

3. MFBind

We introduce **MFBind**, a novel multi-fidelity modeling framework for evaluating compound binding affinities. Our framework consists of a multi-fidelity environment, a deep surrogate model, and an active learning algorithm.

3.1. Multi-fidelity binding affinity environment

A multi-fidelity environment consists of a set of simulators $\{f_1, \dots, f_K\}$, each of which output an increasingly accurate estimate of the value of interest. Define $c > 0$ as the computational cost for a given simulator, such that $c_1 < c_2 < \dots < c_K$. The goal of multi-fidelity modeling is to learn a surrogate model \hat{f}_K that can accurately approximate f_K using a limited amount of high-fidelity data by incorporating data from multiple simulators.

We introduce a new multi-fidelity environment for binding affinity which uses three simulators, each of which takes a molecule as input and outputs an estimate of its binding affinity to a targeted protein with increasing accuracy:

1. **AutoDock4** ($f_1; c_1 = 30s$) (Morris et al., 2009). Uses geometric and charge information from the protein and compound to estimate the binding energy. It outputs a total binding energy prediction in kcal/mol, as well as a set of 15 other outputs, such as energy components

for each type of interaction and the number of protein-compound hydrogen bonds (see Appendix A for a full list), some of which are computed in a post-processing step by BINANA (Young et al., 2022).

2. **Experimental data** ($f_2; c_2 = N/A$) (Liu et al., 2007). Binding values from laboratory experimental studies, obtained from BindingDB. Because it is infeasible to “query” a laboratory for new compounds, we restrict this simulator to only evaluate compounds with known activity values.
3. **Absolute binding free energy** (ABFE) ($f_3; c_3 = 37,521s = 10.4hrs$) (Heinzelmann & Gilson, 2021). A binding free energy method applicable to *de novo* discovery that uses molecular dynamics simulations to accurately predict the binding energy in kcal/mol.

Note that AutoDock4 produces a total of 16 different outputs related to the protein-compound interaction, each of which can be modeled and may aid in the prediction of ABFE scores. The other two fidelities only output a single value. See Appendix A for more details about the environment.

AutoDock4 and ABFE can calculate activities for any compound, whereas the experimental data is only present for a limited set of compounds. While our goal is ultimately to discover compounds with strong experimental binding, we do not make experimental data the highest fidelity simulator because it cannot be queried for arbitrary molecular inputs. Instead, we treat ABFE as the highest fidelity simulator, and use the limited experimental data as a way to improve the surrogate modeling of ABFE scores.

To prove that the higher cost simulators are more accurate, Figure 2 shows the classification performance of the AutoDock4 and ABFE simulators on the test set for the BRD4(2) target (see Sec. 4). The test set consists of half

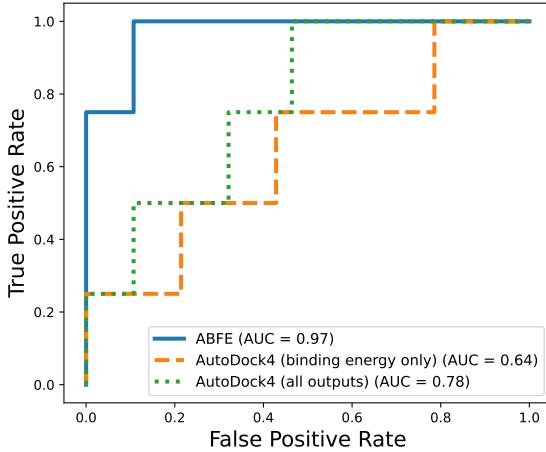


Figure 2. ROC curve of each simulator on the BRD4(2) test set. Two curves are shown for AutoDock4, one that uses the total binding energy output only, and one that uses a linear surrogate that takes all 16 outputs from AutoDock4 and outputs a prediction of the ABFE score.

experimentally confirmed active compounds and half presumed inactives. As expected, the ABFE simulator is the most accurate, while AutoDock4 is still moderately predictive. Additionally, a linear surrogate that uses all 16 AutoDock4 outputs outperforms using only AutoDock4’s total binding energy prediction. See Appendix C.1 for more information about these results and data from more targets.

3.2. Multi-fidelity deep surrogate model

Many previous multi-fidelity surrogate models are ill-suited to our proposed environment because the simulators at each fidelity have variable output dimensions, and collecting large amounts (> 100 samples) of high-fidelity data is very costly. Specifically, GP-based methods can only fit a single scalar output at each fidelity level (Wu et al., 2020) and/or do not support a variable number of output dimensions at each fidelity (Wang et al., 2021). Deep learning models (Li et al., 2020; Wu et al., 2023) also struggle to generalize from a limited amount of high-fidelity training data, likely due to over-parameterization of individual layers.

We propose a new deep surrogate model to address these limitations. Our model consists of a neural encoder, shared across all fidelities, to generate a representation of the input molecule, and linear fidelity-specific prediction heads that transform that representation into a prediction of the fidelity output. The prediction heads are able to output varying dimensionalities for each fidelity.

Mathematically, let h be a feedforward neural network that encodes a molecule x , represented as a 2048-dimensional Morgan fingerprint, into an n -dimensional real-valued vector. Then, define the fidelity-specific heads as a vector of

parameters $\mathbf{w}_1 \in \mathbb{R}^{n \times 16}$ and $\mathbf{w}_2, \mathbf{w}_3 \in \mathbb{R}^{n \times 1}$, and a set of biases, $b_1 \in \mathbb{R}^{16}, b_2, b_3 \in \mathbb{R}$, one for each fidelity level. Note that the weights and biases for the first fidelity level, AutoDock4, are 16-dimensional so that we can model all outputs from that simulator, while the other two fidelities only have scalar output. See Appendix B.1 for more details and a diagram of our surrogate model.

The surrogate model is trained on a dataset \mathcal{D} comprising tuples of the form $(x, k, f_k(x))$, where x is a molecule and k is a fidelity level. The model is trained to minimize the following loss function:

$$\begin{aligned} \frac{1}{|\mathcal{D}|} \sum_{x, k, f_k(x) \in \mathcal{D}} \lambda_k \sum_{i=1}^{|f_k(x)|} & \left((h(x)\mathbf{w}_k^{(i)} + b_k^{(i)}) - f_k(x)^{(i)} \right)^2 \\ & + \lambda_{reg} \sum_{k=1}^3 \|\mathbf{w}_k\|_2^2 \end{aligned} \quad (1)$$

where $f_k(x)^{(i)}$ is the i th output of the simulator at fidelity level k (only relevant for AutoDock4, which has 16 outputs), $\|\cdot\|_2^2$ denotes L2 regularization, λ_k is a weighting parameter for fidelity k , and λ_{reg} is the regularization strength. The regularization of the linear heads ensures that the model does not overfit on a small amount of high-fidelity data.

We employ a pretraining strategy to improve model performance and reduce overfitting issues, where we first train the model on only the two lowest fidelities (without ABFE), and then finetune on all fidelities, including ABFE. The number of epochs for both phases is determined via hyperparameter tuning. This approach is inspired by the pretraining method for multi-task learning (Kaplun et al., 2023). Intuitively, the pretraining phase helps the model learn an effective encoder on the two fidelities with a large amount of data, without overfitting the encoder on the small number of ABFE points. Then, after this “warm start” on the encoder, the finetuning phase allows the model to learn features specific for ABFE prediction.

We use Monte-Carlo (MC) dropout (Gal & Ghahramani, 2016) to estimate model uncertainty for active learning (explained in the subsequent section). For AutoDock4, which has a 16-dimensional output, we normalize each of the 16 elements to have a mean of zero and unit variance (across the training dataset), and then average the variance across all elements. We perform this normalization step so that one element with a greater magnitude does not dominate the uncertainty estimation.

Algorithm 1 Multi-fidelity active learning

Require: a multi-fidelity surrogate model g , a pre-populated multi-fidelity training dataset \mathcal{D} , a set of candidate points \mathcal{S} , the cumulative active learning cost $C \leftarrow 0$, a set of costs c_1, c_2, c_3 , and the computational budget B

while $C < B$ **do**

- $g \leftarrow \text{TrainSurrogateModel}(\mathcal{D})$
- $maxX, maxK, maxVar \leftarrow 0, 0, 0$
- for** x in \mathcal{S} **do**

 - for** k in $1 \dots 3$ **do**

 - $var \leftarrow a(x, k)$ (Sec. 3.3)
 - if** $var > maxVar$ **then**

 - $maxX, maxK, maxVar \leftarrow x, k, var$

 - end if**

 - end for**
 - end for**

$\mathcal{D} \leftarrow \mathcal{D} \cup \{(maxX, maxK, f_{maxK}(maxX))\}$

$C \leftarrow C + c_{maxK}$

end while

3.3. Active learning to train multi-fidelity surrogate

Learning a multi-fidelity surrogate model on our proposed environment requires significant computational resources, especially to gather data from ABFE. Instead of passively collecting training data, we propose an active learning approach to efficiently query the simulators.

Our active learning algorithm involves iterative querying of the simulators at the points where the model is most uncertain, weighted by the computational cost. The model is first trained on a limited prepopulated dataset \mathcal{D} , and then evaluates its uncertainty $a(x, k)$ on each of the candidate compounds x at each fidelity level k . After the compound and associated fidelity level, i.e. simulator, with the highest uncertainty is selected, that compound is run through the simulator and the new activity data is added to the training dataset \mathcal{D} . The model is updated with the new data, and the process is repeated until a computational budget B is reached. Algorithm 1 describes such a procedure.

To quantify model uncertainty we define an acquisition function $a(x, k)$ that outputs the expected utility of querying the simulator at fidelity k for a compound x . In this paper, we use the cost-weighted maximum variance, $a(x, k) = \frac{1}{c_k} \sigma^2(x, k)$. Here, $\sigma^2(x, k)$ is the model uncertainty (variance) at point x and fidelity k . We also tried an entropy-based acquisition function, but found empirically that the cost term dominated the entropy term, and thus the model always chose to query the lowest cost simulator.

Since querying ABFE and docking for a single compound is already parallelizable across multiple GPUs/cores (Heinzelmann & Gilson, 2021), we do not consider batch active learning algorithms (Kirsch et al., 2019). We chose to re-

train the surrogate model after every query because the computational cost of surrogate training is much lower than even the fastest simulator.

4. Experimental results

To evaluate the utility of our MFBind framework, we consider the following two experimental settings. First, we evaluate the predictive performance of our multi-fidelity surrogate model trained with active learning compared to baseline multi-fidelity techniques. Second, we integrate our deep surrogate model with existing generative models, and compare the generated compounds to those generated by the traditional single-fidelity methods. See Appendix B for experimental details.

We conduct experiments on two targeted proteins: BRD4(2) (PDB 5UF0) and c-MET (PDB 5EOB). BRD4(2), the second binding domain of bromodomain-containing protein 4, is a protein of interest for cancer treatment (French, 2008). c-MET is a receptor tyrosine kinase that also shows promise for treating cancer (Zhang et al., 2018). ABFE has previously been well-validated for both of these targets, where it shows strong correlation with experimental results (Heinzelmann & Gilson, 2021; Huggins, 2022).

4.1. Multi-fidelity surrogate modeling**4.1.1. SETUP**

We first evaluate the performance of our multi-fidelity surrogate model trained with active learning, using Algorithm 1 and a predefined candidate set of compounds, and compare the results with those of baseline methods.

The training dataset for this test is initialized with data across all fidelity levels, and then is supplemented with active learning data as the model queries the simulators. The initial dataset for BRD4(2) includes the AutoDock4 output for 100,000 compounds randomly sampled from the ZINC250k dataset (Irwin et al., 2012), 91 experimental activity values for the BRD4(2) target obtained from BindingDB (Liu et al., 2007), and one ABFE score of a compound randomly sampled from the same BindingDB dataset. The initial dataset for c-MET is the same, except the experimental data contains 102 experimental activity values from BindingDB’s c-MET target.

We initialize our dataset with only one datapoint from ABFE so that we can make maximal use of active learning. The candidate set, which is the pool of compounds available to active learning, is obtained from BindingDB under the “BRD4” target for BRD4(2) and “c-MET” for c-MET. Note that the BRD4 target is a larger superset of BRD4(2), which we chose because there were not enough compounds measured against only BRD4(2).

Our held-out test set consists of precomputed ABFE results for a curated set of 32 compounds for each target. Half of these compounds come from BindingDB’s BRD4(2) or c-MET dataset, with the constraint that the experimental activity be $< 1\mu M$, and the other half from BRD4(2) or c-MET decoys, which are presumed to be inactive, generated by DUD-E (Mysinger et al., 2012). To ensure diversity, we clustered BindingDB compounds in the test set so that no compound had a Tanimoto similarity higher than 0.4 with any other compound, and then generated decoy counterparts for each of these compounds using DUD-E. We also ensured the test compounds were dissimilar (Tanimoto similarity < 0.4) to the compounds in the initial training and candidate sets, removing them from the latter if any compounds were too similar. We constructed the test dataset with both actives and decoys so that we could measure the ability of each simulator to distinguish between them (see Appendix C.1).

4.1.2. BASELINES

We compare our model with the following baseline methods for multi-fidelity surrogate modeling:

- **Only ABFE (NN).** A simple feedforward neural network that is only trained on ABFE scores.
- **Direct-GP (DKL).** A DKL-based (Wilson et al., 2016) multi-fidelity GP model using a downsampling kernel for the fidelities (Wu et al., 2020). Uses the “Direct” output from AutoDock4, meaning the total binding energy prediction, and discards all other AutoDock4 outputs.
- **Surrogate-GP (DKL).** Same as above, except for f_2 we use a linear “Surrogate” model that takes all outputs from AutoDock4 as input and outputs a prediction of the ABFE score.
- **D-MFDAL (Wu et al., 2023).** A state-of-the-art neural process model and acquisition function for multi-fidelity modeling. Learns a global and local representation for each fidelity level, avoiding the propagation of errors from lower to higher fidelity levels.
- **DMFAL (Li et al., 2020).** A neural network-based approach for modeling multi-fidelity high-dimensional outputs by passing information from lower to higher fidelity levels.
- **Hadamard-MT (DKL) (Bonilla et al., 2007).** A DKL-based multi-task (MT) GP model where the kernel is the Hadamard product of an input and task kernel. Each fidelity is treated as its own task, except for AutoDock4 where we use all 16 outputs by treating each one as its own task.

See Appendix B.2 for further details about these baselines.

Table 1. MFBind ablations. We report the test set MSE, for each target, of each model modification when trained on the same set of data queried by the full MFBind model at the end of the active learning experiment.

ABLATION	BRD4(2)	c-MET
NONE (MFBIND)	18.1	36.0
w/o PRETRAINING	23.7	41.8
w/o REGULARIZATION	18.8	38.8
w/o AUTO DOCK4 DATA	20.4	37.5
w/o EXPERIMENTAL DATA	25.5	40.8

4.1.3. RESULTS

Surrogate model performance. Figure 3 shows the prediction error for both targets, measured in MSE between the actual and predicted ABFE results in kcal/mol for the 32 compounds in the held-out test set, as the computational budget allotted to active learning increases. The single-fidelity, only ABFE approach performs poorly for both targets, showing that MFBind aids in training models to predict ABFE scores at a lower computational cost than using only ABFE data. Among the multi-fidelity surrogate modeling baselines, ours performs the best across both targets, suggesting that it is the most efficient at using cheaper non-ABFE methods to enhance the prediction of ABFE results.

Ablation study. Table 1 (first set of rows) shows the performance of various ablations of the surrogate model, measured by MSE in kcal/mol on the test set. Each method was trained on the same data as collected by MFBind during active learning. “w/o Pretraining” means we did not perform any pretraining on the lower fidelity data, and instead trained the model on the full number of epochs using all fidelities. “w/o Regularization” means we removed the linear prediction head regularization term. These results show that both design choices, especially the pretraining step, significantly contribute to the performance of our model.

We also studied the impact of ablating each lower fidelity simulator (Table 1, second set of rows). “w/o AutoDock4 data” means we removed all data from the AutoDock4 fidelity level, and “w/o Experimental data” means we removed all experimental data. As demonstrated, both of the lower fidelity simulators are important for the model to learn to predict data at the highest fidelity level.

4.2. Compound generation with MFBind

4.2.1. SETUP

To show the utility of MFBind in assisting molecular generative modeling, we seek to generate compounds using the MFBind surrogate model as the reward function. We chose

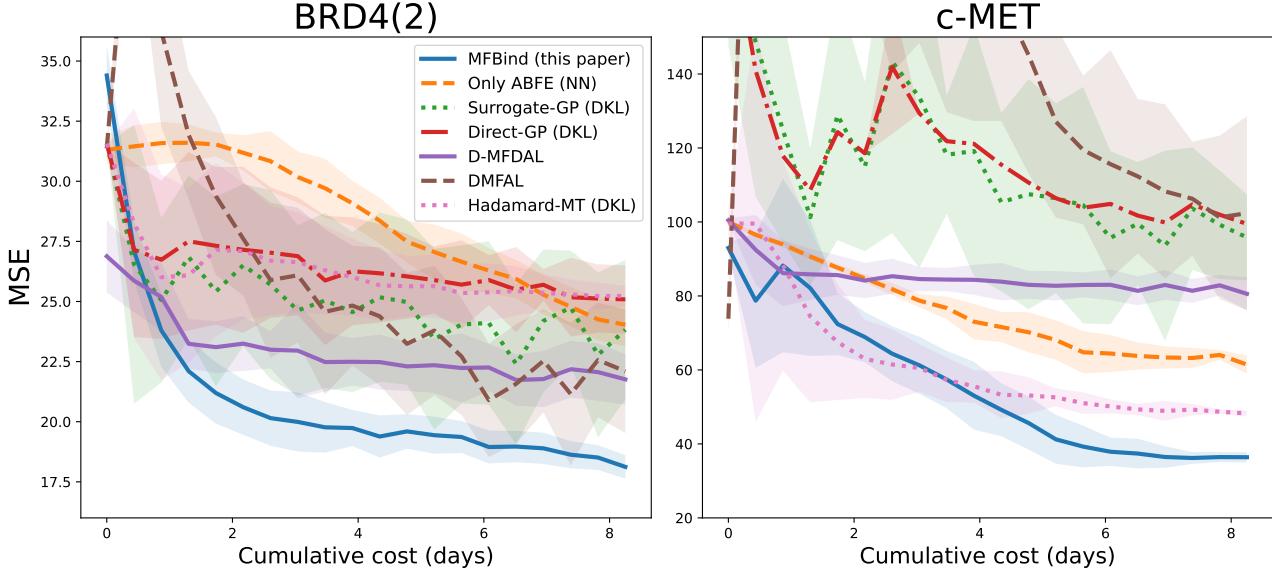


Figure 3. Regression of ABFE scores in an active learning setting. The y-axis shows the mean squared error (MSE), in kcal/mol, of each method on the held-out test set. The x-axis shows the cumulative active learning query cost in days (wall clock time on a 9 core, 8 GPU server). Each line represents an average over 20 runs with random seeds (using caching of ABFE results to reduce running times), with the shaded region indicating the standard deviation across runs.

LIMO (Eckmann et al., 2022), a variational autoencoder-based generative model, due to its strong performance on binding affinity optimization. We also explored MolDQN (Zhou et al., 2019), a reinforcement learning approach, as the generative model. However, we found MolDQN generated compounds that were not drug-like and in some cases chemically implausible, even when including a drug-likeness objective in the reward (see Appendix C.3).

We first trained ours and baseline surrogate models on the same initial training dataset as the previous active learning task, except now including the ABFE scores for all compounds in the candidate set. This ensured that the predictions from the reward function were as accurate as possible. Then, we froze the surrogate model and used it as a reward function to evaluate generated compounds. We experimented with periodically updating the surrogate model with active learning over the generated compounds, but found that it led to comparable or worse performance than not updating the model.

For each target and choice of surrogate model, we used LIMO to generate 10,000 compounds intended to bind the target, chose the top 20 unique compounds with the highest surrogate-computed reward, and used full ABFE calculations to estimate their binding affinities. We incorporated an additional drug-likeness (QED; Bickerton et al. (2012)) objective in the reward function, by adding 2x the QED of the compound to the ABFE score predicted by the surrogate model, to ensure the generated compounds were reasonably

drug-like. We also enforced a QED cutoff > 0.5 and maximum ring size < 7 for choosing the final 20 compounds.

4.2.2. BASELINES

We compared compounds generated with MFBind as the reward function against the following baseline reward functions:

- **Single fidelity (SF) ABFE.** A single-fidelity surrogate model that is trained only on ABFE data. This baseline is trained on the same number of ABFE datapoints as MFBind, but without the data from the other fidelities.
- **Single fidelity (SF) AutoDock4.** Same as above, except this baseline is only trained on the AutoDock4 total binding energy score without any other fidelity data, including ABFE. This baseline is the prevalent approach in molecular generative modeling, where the docking score is used as a reward function.

See Appendix B.2 for more details about each baseline. We did not include other multi-fidelity surrogate modeling baselines in this experiment because of computational constraints, and the high noise in the generative setting which makes a comparison between competing multi-fidelity surrogates difficult without a large number of samples.

Table 2. Evaluation of LIMO-generated compounds with different surrogate models. The mean and top 3 ABFE-computed energies, both in kcal/mol, are shown among 20 tested compounds from each method and for each target. “SF” refers to single-fidelity methods that only use one simulator, while our “MFBind” approach uses all simulators. All compounds have QED > 0.5.

METHOD	BRD4(2)				c-MET			
	MEAN	1ST	2ND	3RD	MEAN	1ST	2ND	3RD
SF ABFE	-2.94	-5.60	-5.26	-4.57	3.14	-5.19	-3.05	-2.79
SF AUTO DOCK4	-2.81	-4.46	-4.01	-3.40	2.81	-5.87	-3.30	-2.51
MFBIND	-4.16	-10.94	-10.04	-7.38	-3.69	-11.25	-11.06	-9.10

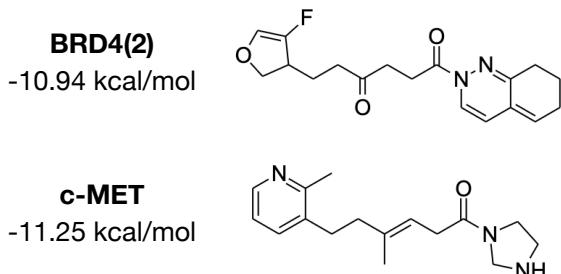


Figure 4. Selected generated compounds from LIMO + MFBIND. The top compound for both BRD4(2) and c-MET are shown. See Appendix C.2 for more compounds.

4.2.3. RESULTS

Table 2 shows the computed ABFE scores for the top 20 compounds with the highest predicted reward generated under the guidance of each surrogate model for each target. For both targets, among the sample of 20 compounds, the mean ABFE score of compounds from MFBIND is distinctly lower (indicating better binding affinity) than the same number of compounds generated using either the single-fidelity ABFE or single-fidelity AutoDock4 approaches. Additionally, the top three compounds for both targets from MFBIND have dramatically better ABFE results than the top three compounds from the single-fidelity methods. Note that MFBIND is the only approach that generated compounds in the nanomolar K_d range (< -8.2 kcal/mol), a widely used activity cutoff in early drug discovery to determine which compounds show promise (Hughes et al., 2011).

Figure 4 shows the top compound generated from LIMO + MFBIND for each target. The compounds appear relatively synthesizable and drug-like while having strong ABFE-predicted affinity. See Appendix C.2 for more examples.

These results, combined with the fact that ABFE results correlate well with experiment for both targets (Heinzelmann & Gilson, 2021; Huggins, 2022), indicate that the MFBIND surrogate model allows us to generate much more promising compounds than either of the single-fidelity approaches.

5. Discussion and Conclusion

We present a new multi-fidelity framework, MFBIND, for evaluating compound binding affinity in generative models. We introduce a new multi-fidelity environment for binding affinity that consists of docking (AutoDock4), experimental data (from BindingDB), and binding free energy (ABFE) simulators. Our framework also contains a deep surrogate model trained with an active learning algorithm to further reduce cost. Our surrogate model can fit small amounts of high-fidelity data using a pretraining method on the lower fidelity data, combined with regularized linear fidelity-specific prediction heads.

We perform extensive evaluation of ours and baseline approaches in multi-fidelity surrogate modeling, and find that our model is most capable of efficiently utilizing a limited computational budget to predict the ABFE score of unseen compounds. We also test our framework in a molecular generative modeling task, where we use the surrogate model as a reward function for generation. We find that MFBIND outperforms common approaches that use a single-fidelity reward function. It can generate compounds with markedly higher activity, as computed by the accurate binding free energy simulator, than competing methods. Therefore, MFBIND shows promise as a way to make generative models for drug discovery useful in practice.

We note that the greatest accuracy surrogate model would presumably be obtained by training purely against, e.g., thousands of ABFE results, without the other fidelities. However, generating these training data would be exceedingly costly, given that ABFE calculations are around 1250x slower than docking calculations. An important implication of the present results is that, although docking results are not as accurate as ABFE results, and although their correlation coefficients with ABFE results are modest ($r = 0.25$ (BRD4(2)) and $r = 0.23$ (c-MET), see Appendix C), the docking results can still be used to boost the predictivity of a model trained with a fixed number of ABFE results.

Limitations of our approach include a limited set of simulators and potentially a lack of synthesizability of the generated molecules. Since the only objectives we consider when generating compounds are the binding affinity and

QED, it is possible that the compounds could be difficult to synthesize. Additionally, our acquisition function for active learning is somewhat simple. Instead of averaging the uncertainty from all AutoDock4 outputs, further studies can be done on weighing them by importance.

Future work could include making the acquisition function more complex, and adding more fidelities, such as deep learning-based binding affinity predictors and ABFE with varying simulation times. Another next step would be to use a reaction-aware generative model that generates more synthesizable molecules, such as Horwood & Noutahi (2020).

Impact statement Similar to other works that apply machine learning to drug discovery, our work is subject to dual use (Urbina et al., 2022). There is potential for societal benefit, by helping develop new drug compounds to treat disease. However, there is also potential for harm, such as to generate new chemical weapons. Fortunately, the latter places a whole additional set of requirements on compounds (e.g. skin-absorbable or volatile and subject to inhalation), so this problematic direction does not appear to be imminent.

6. Acknowledgments

This work was supported in part by Army-ECASE award W911NF-07-R-0003-03, the U.S. Department Of Energy, Office of Science, IARPA HAYSTAC Program, CDC-RFA-FT-23-0069, NSF Grants #2205093, #2146343, and #2134274. MKG has an equity interest in and is a cofounder and scientific advisor of VeraChem LLC.

References

- Balandat, M., Karrer, B., Jiang, D. R., Daulton, S., Letham, B., Wilson, A. G., and Bakshy, E. BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization. In *Advances in Neural Information Processing Systems* 33, 2020.
- Bickerton, G. R., Paolini, G. V., Besnard, J., Muresan, S., and Hopkins, A. L. Quantifying the chemical beauty of drugs. *Nature chemistry*, 4(2):90–98, 2012.
- Bonilla, E. V., Chai, K., and Williams, C. Multi-task gaussian process prediction. *Advances in neural information processing systems*, 20, 2007.
- Brevault, L., Balesdent, M., and Hebbal, A. Overview of gaussian process based multi-fidelity techniques with variable relationship between fidelities. *arXiv preprint arXiv:2006.16728*, 2020.
- Chatterjee, A., Walters, R., Shafi, Z., Ahmed, O. S., Sebek, M., Gysi, D., Yu, R., Eliassi-Rad, T., Barabási, A.-L., and Menichetti, G. Improving the generalizability of protein-ligand binding predictions with ai-bind. *Nature Communications*, 14(1):1989, 2023.
- Coley, C. W., Eyke, N. S., and Jensen, K. F. Autonomous discovery in the chemical sciences part ii: outlook. *Angewandte Chemie International Edition*, 59(52):23414–23436, 2020.
- Cournia, Z., Allen, B., and Sherman, W. Relative binding free energy calculations in drug discovery: recent advances and practical considerations. *Journal of chemical information and modeling*, 57(12):2911–2937, 2017.
- Cournia, Z., Chipot, C., Roux, B., York, D. M., and Sherman, W. Free energy methods in drug discovery—introduction. In *Free energy methods in drug discovery: Current state and future directions*, pp. 1–38. ACS Publications, 2021.
- Eckmann, P., Sun, K., Zhao, B., Feng, M., Gilson, M. K., and Yu, R. Limo: Latent inceptionism for targeted molecule generation. *arXiv preprint arXiv:2206.09010*, 2022.
- Fare, C., Fenner, P., Benatan, M., Varsi, A., and Pyzer-Knapp, E. O. A multi-fidelity machine learning approach to high throughput materials screening. *npj Computational Materials*, 8(1):257, 2022.
- Feng, M., Heinzelmann, G., and Gilson, M. K. Absolute binding free energy calculations improve enrichment of actives in virtual compound screening. *Scientific Reports*, 12(1):13640, 2022.
- Fernández-Godino, M. G., Park, C., Kim, N.-H., and Haftka, R. T. Review of multi-fidelity models. *arXiv preprint arXiv:1609.07196*, 2016.
- French, C. A. Molecular pathology of nut midline carcinomas. *Journal of clinical pathology*, 2008.
- Fu, T., Gao, W., Coley, C., and Sun, J. Reinforced genetic algorithm for structure-based drug design. *Advances in Neural Information Processing Systems*, 35:12325–12338, 2022.
- Gal, Y. and Ghahramani, Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pp. 1050–1059. PMLR, 2016.
- Gardner, J. R., Pleiss, G., Bindel, D., Weinberger, K. Q., and Wilson, A. G. Gpytorch: Blackbox matrix-matrix gaussian process inference with gpu acceleration. In *Advances in Neural Information Processing Systems*, 2018.
- Ghanakota, P., Bos, P. H., Konze, K. D., Staker, J., Marques, G., Marshall, K., Leswing, K., Abel, R., and Bhat, S.

- Combining cloud-based free-energy calculations, synthetically aware enumerations, and goal-directed generative machine learning for rapid large-scale chemical exploration and optimization. *Journal of Chemical Information and Modeling*, 60(9):4311–4325, 2020.
- Gómez-Bombarelli, R., Wei, J. N., Duvenaud, D., Hernández-Lobato, J. M., Sánchez-Lengeling, B., Sheberla, D., Aguilera-Iparraguirre, J., Hirzel, T. D., Adams, R. P., and Aspuru-Guzik, A. Automatic chemical design using a data-driven continuous representation of molecules. *ACS central science*, 4(2):268–276, 2018.
- Guan, J., Qian, W. W., Peng, X., Su, Y., Peng, J., and Ma, J. 3d equivariant diffusion for target-aware molecule generation and affinity prediction. *arXiv preprint arXiv:2303.03543*, 2023a.
- Guan, J., Zhou, X., Yang, Y., Bao, Y., Peng, J., Ma, J., Liu, Q., Wang, L., and Gu, Q. Decompdiff: Diffusion models with decomposed priors for structure-based drug design. *international conference on machine learning*, 2023b.
- Handa, K., Thomas, M. C., Kageyama, M., Iijima, T., and Bender, A. On the difficulty of validating molecular generative models realistically: a case study on public and proprietary data. *Journal of Cheminformatics*, 15(1):112, 2023.
- Heinzelmann, G. and Gilson, M. K. Automation of absolute protein-ligand binding free energy calculations for docking refinement and compound evaluation. *Scientific reports*, 11(1):1116, 2021.
- Hernandez-Garcia, A., Saxena, N., Jain, M., Liu, C.-H., and Bengio, Y. Multi-fidelity active learning with gflownets. *arXiv preprint arXiv:2306.11715*, 2023.
- Horwood, J. and Noutahi, E. Molecular design in synthetically accessible chemical space via deep reinforcement learning. *ACS omega*, 5(51):32984–32994, 2020.
- Huggins, D. J. Comparing the performance of different amber protein forcefields, partial charge assignments, and water models for absolute binding free energy calculations. *Journal of Chemical Theory and Computation*, 18(4):2616–2630, 2022.
- Hughes, J. P., Rees, S., Kalindjian, S. B., and Philpott, K. L. Principles of early drug discovery. *British journal of pharmacology*, 162(6):1239–1249, 2011.
- Irwin, J. J., Sterling, T., Mysinger, M. M., Bolstad, E. S., and Coleman, R. G. Zinc: a free tool to discover chemistry for biology. *Journal of chemical information and modeling*, 52(7):1757–1768, 2012.
- Jeon, W. and Kim, D. Autonomous molecule generation using reinforcement learning and docking to develop potential novel inhibitors. *Scientific reports*, 10(1):22104, 2020.
- Ji, Y., Zhang, L., Wu, J., Wu, B., Huang, L.-K., Xu, T., Rong, Y., Li, L., Ren, J., Xue, D., et al. Drugood: Out-of-distribution (ood) dataset curator and benchmark for ai-aided drug discovery—a focus on affinity prediction problems with noise annotations. *arXiv preprint arXiv:2201.09637*, 2022.
- Jin, W., Barzilay, R., and Jaakkola, T. Junction tree variational autoencoder for molecular graph generation. In *International conference on machine learning*, pp. 2323–2332. PMLR, 2018.
- Kaplun, G., Gurevich, A., Swisa, T., David, M., Shalev-Shwartz, S., and Malach, E. Subtuning: Efficient finetuning for multi-task learning. *arXiv preprint arXiv:2302.06354*, 2023.
- Kirsch, A., Van Amersfoort, J., and Gal, Y. Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning. *Advances in neural information processing systems*, 32, 2019.
- Lee, S., Jo, J., and Hwang, S. J. Exploring chemical space with score-based out-of-distribution generation. In *International Conference on Machine Learning*, pp. 18872–18892. PMLR, 2023.
- Li, S., Kirby, R. M., and Zhe, S. Deep multi-fidelity active learning of high-dimensional outputs. *arXiv preprint arXiv:2012.00901*, 2020.
- Li, S., Wang, Z., Kirby, R., and Zhe, S. Infinite-fidelity coregionalization for physical simulation. *Advances in Neural Information Processing Systems*, 35:25965–25978, 2022.
- Liu, T., Lin, Y., Wen, X., Jorissen, R. N., and Gilson, M. K. Bindingdb: a web-accessible database of experimentally determined protein–ligand binding affinities. *Nucleic acids research*, 35(suppl_1):D198–D201, 2007.
- Moore, J. H., Margreitter, C., Janet, J. P., Engkvist, O., de Groot, B. L., and Gapsys, V. Automated relative binding free energy calculations from smiles to $\delta\delta g$. *Communications Chemistry*, 6(1):82, 2023.
- Morris, G. M., Huey, R., Lindstrom, W., Sanner, M. F., Belew, R. K., Goodsell, D. S., and Olson, A. J. Autodock4 and autodocktools4: Automated docking with selective receptor flexibility. *Journal of computational chemistry*, 30(16):2785–2791, 2009.

- Mysinger, M. M., Carchia, M., Irwin, J. J., and Shoichet, B. K. Directory of useful decoys, enhanced (dud-e): better ligands and decoys for better benchmarking. *Journal of medicinal chemistry*, 55(14):6582–6594, 2012.
- Noh, J., Jeong, D.-W., Kim, K., Han, S., Lee, M., Lee, H., and Jung, Y. Path-aware and structure-preserving generation of synthetically accessible molecules. In *International Conference on Machine Learning*, pp. 16952–16968. PMLR, 2022.
- O’Boyle, N. M., Banck, M., James, C. A., Morley, C., Vandermeersch, T., and Hutchison, G. R. Open babel: An open chemical toolbox. *Journal of cheminformatics*, 3(1):1–14, 2011.
- Paul, D., Sanap, G., Shenoy, S., Kalyane, D., Kalia, K., and Tekade, R. K. Artificial intelligence in drug discovery and development. *Drug discovery today*, 26(1):80, 2021.
- Peng, X., Luo, S., Guan, J., Xie, Q., Peng, J., and Ma, J. Pocket2mol: Efficient molecular sampling based on 3d protein pockets. In *International Conference on Machine Learning*, pp. 17644–17655. PMLR, 2022.
- Pinzi, L. and Rastelli, G. Molecular docking: shifting paradigms in drug discovery. *International journal of molecular sciences*, 20(18):4331, 2019.
- Ren, P., Xiao, Y., Chang, X., Huang, P.-Y., Li, Z., Gupta, B. B., Chen, X., and Wang, X. A survey of deep active learning. *ACM computing surveys (CSUR)*, 54(9):1–40, 2021.
- Santos-Martins, D., Solis-Vasquez, L., Tillack, A. F., Sanner, M. F., Koch, A., and Forli, S. Accelerating autodock4 with gpus and gradient-based local search. *Journal of chemical theory and computation*, 17(2):1060–1073, 2021.
- Spiegel, J. O. and Durrant, J. D. Autogrow4: an open-source genetic algorithm for de novo drug design and lead optimization. *Journal of cheminformatics*, 12(1):1–16, 2020.
- Thomas, M., Bender, A., and de Graaf, C. Integrating structure-based approaches in generative molecular design. *Current Opinion in Structural Biology*, 79:102559, 2023.
- Urbina, F., Lentzos, F., Invernizzi, C., and Ekins, S. Dual use of artificial-intelligence-powered drug discovery. *Nature Machine Intelligence*, 4(3):189–191, 2022.
- Wan, S., Potterton, A., Husseini, F. S., Wright, D. W., Heifetz, A., Malawski, M., Townsend-Nicholson, A., and Coveney, P. V. Hit-to-lead and lead optimization binding free energy calculations for g protein-coupled receptors. *Interface Focus*, 10(6):20190128, 2020.
- Wang, Y. and Lin, G. Mfpc-net: Multi-fidelity physics-constrained neural process. *arXiv preprint arXiv:2010.01378*, 2020.
- Wang, Z., Xing, W., Kirby, R., and Zhe, S. Multi-fidelity high-order gaussian processes for physical simulation. In *International Conference on Artificial Intelligence and Statistics*, pp. 847–855. PMLR, 2021.
- Wilson, A. and Nickisch, H. Kernel interpolation for scalable structured gaussian processes (kiss-gp). In *International conference on machine learning*, pp. 1775–1784. PMLR, 2015.
- Wilson, A. G., Hu, Z., Salakhutdinov, R., and Xing, E. P. Deep kernel learning. In *Artificial intelligence and statistics*, pp. 370–378. PMLR, 2016.
- Wu, D., Chinazzi, M., Vespiagnani, A., Ma, Y.-A., and Yu, R. Multi-fidelity hierarchical neural processes. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 2029–2038, 2022.
- Wu, D., Niu, R., Chinazzi, M., Ma, Y., and Yu, R. Disentangled multi-fidelity deep bayesian active learning. *arXiv preprint arXiv:2305.04392*, 2023.
- Wu, J., Toscano-Palmerin, S., Frazier, P. I., and Wilson, A. G. Practical multi-fidelity bayesian optimization for hyperparameter tuning. In *Uncertainty in Artificial Intelligence*, pp. 788–798. PMLR, 2020.
- Xie, Y., Shi, C., Zhou, H., Yang, Y., Zhang, W., Yu, Y., and Li, L. Mars: Markov molecular sampling for multi-objective drug discovery. *arXiv preprint arXiv:2103.10432*, 2021.
- You, J., Liu, B., Ying, Z., Pande, V., and Leskovec, J. Graph convolutional policy network for goal-directed molecular graph generation. *Advances in neural information processing systems*, 31, 2018.
- Young, J., Garikipati, N., and Durrant, J. D. Binana 2: characterizing receptor/ligand interactions in python and javascript. *Journal of chemical information and modeling*, 62(4):753–760, 2022.
- Zhang, Y., Xia, M., Jin, K., Wang, S., Wei, H., Fan, C., Wu, Y., Li, X., Li, X., Li, G., et al. Function of the c-met receptor tyrosine kinase in carcinogenesis and associated therapeutic opportunities. *Molecular cancer*, 17(1):1–14, 2018.
- Zhou, Z., Kearnes, S., Li, L., Zare, R. N., and Riley, P. Optimization of molecules via deep reinforcement learning. *Scientific reports*, 9(1):10752, 2019.

A. Environment details

For all simulators, we estimated the cost using the average over 10 samples with random input compounds.

AutoDock4 We prepared the AutoDock4 grid files using AutoDockTools (Morris et al., 2009). Arbitrary ligands were prepared using obabel (O’Boyle et al., 2011) with pH 7.4 and gasteiger partial charges. We used AutoDock-GPU (Santos-Martins et al., 2021), a GPU-accelerated version of AutoDock4, for all computation. The full set of outputs we collected from AutoDock4 are as follows, with the last 9 collected in a post-processing step from BINANA (Young et al., 2022):

- Total binding energy (“Estimated Free Energy of Binding” in the AutoDock4 output), minimum over 20 random restarts
- Total binding energy, mean over 20 random restarts
- Intermolecular energy
- Internal energy
- Torsional energy
- Unbound system energy
- Number of ligand atoms
- Number of protein-ligand hydrogen bonds
- Number of protein-ligand pi-pi stacking bonds
- Number of protein-ligand salt bridges
- Number of protein-ligand T-stacking interactions
- Number of protein-ligand close contacts
- Backbone alpha flexibility
- Backbone other flexibility
- Sidechain alpha flexibility
- Sidechain other flexibility

Absolute binding free energy (ABFE) We use the Binding Affinity Tool (BAT.py) implementation (Heinzelmann & Gilson, 2021) for absolute binding free energy calculation, available at <https://github.com/GHeinzelmann/BAT.py>, which uses the simultaneous decoupling and recoupling (SDR) method. All molecular dynamics simulators are run with AMBER with GPU support. As BAT.py requires a starting pose for the ligand, we used the pose generated from AutoDock4. We found that we were able to reduce the simulation times up to 80% from the default times for each phase of the SDR computation without losing much accuracy. We additionally wrote custom scripts to parallelize molecular dynamics runs across all available GPUs.

B. Experimental details

All experiments were conducted on a server with 8 RTX 2080 Ti GPUs. For our model and each baseline, we performed a random hyperparameter search with 20 trials (across the hyperparameters listed below) and took the combination with the best test set MSE when trained on the initial dataset combined with ABFE datapoints for the entire candidate set. We conducted separate hyperparameter searches for each target, and used the same set of hyperparameters for both the surrogate modeling and generative experiments. All models were trained with the Adam optimizer.

Multi-Fidelity Bind (MFBind)

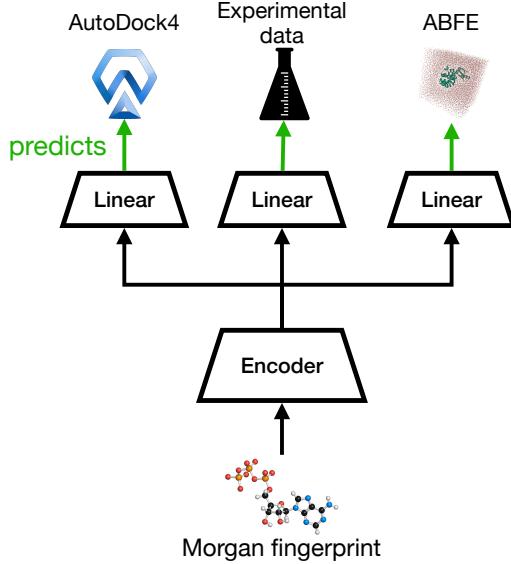


Figure 5. Diagram of the MFBind surrogate model. An input molecule, represented as a Morgan fingerprint, is fed through the deep encoder to produce a latent representation. That representation is then passed to linear fidelity-specific prediction heads to produce a prediction for each fidelity level.

B.1. MFBind surrogate model details

Figure B.1 is a diagram of the MFBind surrogate model. Our deep molecular encoder consisted of 4 linear layers with ReLU activations and dropout after each layer, except the final layer. For estimating the model uncertainty, we used the predictive variance over 50 samples using the same input with random dropout. Hyperparameters: encoding dim(n), hidden layer dim, lr, lr decay beta, λ_0 , λ_1 , $\lambda_2 = 1$, λ_{reg} , pretraining epochs, finetuning epochs, dropout p

B.2. Baseline details

B.2.1. MULTI-FIDELITY SURROGATE MODELING

Only ABFE (NN) Simple feedforward neural network using the same architecture of molecular encoder as MFBind, and a final linear layer to produce the ABFE prediction. Only trained on ABFE data. Used the same MC dropout technique as our model to estimate the uncertainty. Hyperparameters: encoding dim, hidden layer dim, lr, lr decay beta, num epochs, dropout p

Direct-GP (DKL) Exact GP model using a 3-layer deep kernel with ReLU activations to encode the input molecule, which is then passed to the GP. A downsampling kernel (Wu et al., 2020) is used to produce the output at each fidelity level. Since this model can only fit a scalar for each fidelity level, we used the total energy prediction from AutoDock4 only (instead of all 16 outputs). For this and all other GP-based baselines, we used the posterior variance to estimate model uncertainty. Implemented using the BoTorch (Balandat et al., 2020) library. Hyperparameters: encoding dim, DKL hidden layer dim, lr, num epochs

Surrogate-GP (DKL) Same as above, except instead of using the total energy prediction from AutoDock4, we used a linear surrogate. Specifically, we took the available ABFE datapoints and trained a linear regression model with L2 regularization to predict the ABFE score of a compound using all outputs from AutoDock4 as input. Then, this model was applied to all AutoDock4 datapoints to transform the multi-dimensional output into a scalar for use in the GP model. Hyperparameters: same as above

D-MFDAL Defines its own acquisition function, which we used. Used the code available at <https://github.com/Rose-STL-Lab/Multi-Fidelity-Deep-Active-Learning>. Hyperparameters: hidden dim, epoch num, lr

DMFAL Defines its own acquisition function, which we used. Used the code available at <https://github.com/shib0li/DMFAL>. Hyperparameters: lr, reg strength, max epoch, hidden layer dim, base dim

Hadamard-MT (DKL) Exact GP model using a 3-layer deep kernel with ReLU activations to encode the input molecule, which is then passed to the GP. This multi-task GP uses the Hadamard product of the input kernel and a task kernel (Bonilla et al., 2007). The task index is concatenated to the input (from the deep kernel). Since this is a multi-task model, we can model all outputs from AutoDock4 by treating each one as its own task. Therefore, we learned 18 total tasks: 1 for experimental data, 1 for ABFE, and 16 for AutoDock4. Implemented using GPyTorch (Gardner et al., 2018). Hyperparameters: encoding dim, DKL hidden layer dim, lr, num epochs

B.2.2. COMPOUND GENERATION WITH MFBIND

For these baselines, we used the same hyperparameters as those chosen for the Only ABFE (NN) baseline in the above surrogate modeling task.

Single fidelity (SF) ABFE Simple feedforward neural network using the same architecture of molecular encoder as MFBind, and a final linear layer to produce the prediction. Only trained on ABFE data.

Single fidelity (SF) AutoDock4 Same as above, except only trained on the total binding energy prediction from AutoDock4 (without the 15 other outputs).

C. Additional results

C.1. Analysis of MFBind environment

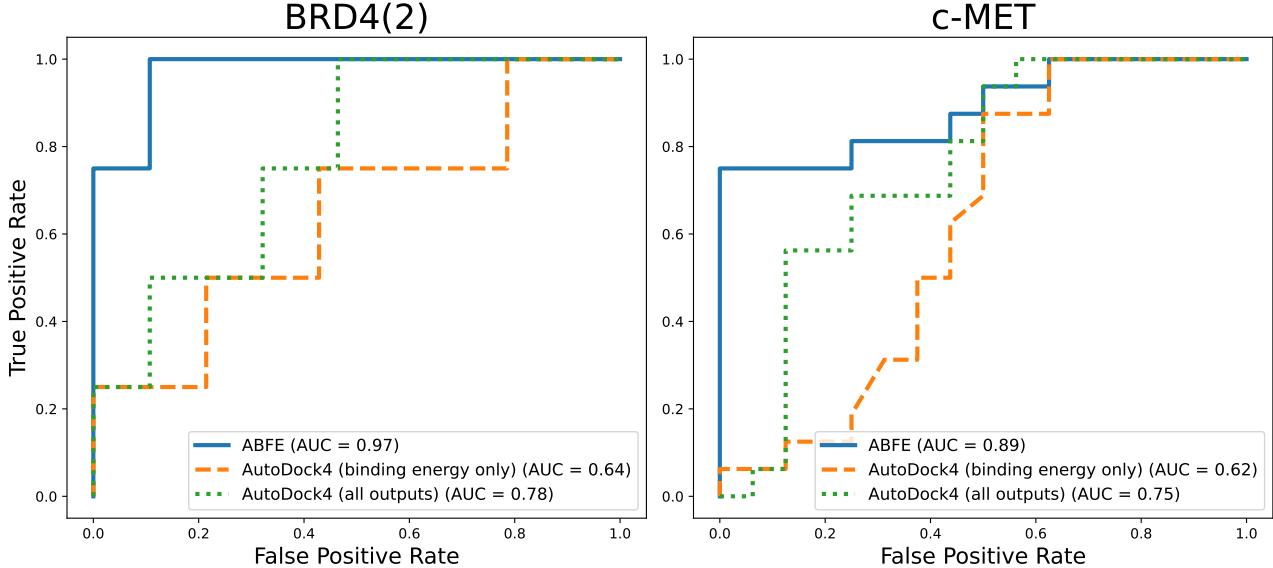


Figure 6. ROC curve of each simulator. Compounds from the BindingDB BRD4(2) and c-MET datasets are classified as active, and decoys are classified as inactive.

We aim to show that each simulator in the MFBind environment meaningfully correlates to real-world experimental data. Figure 6 shows a ROC curve for the ABFE and AutoDock4 simulators on our BRD4(2) and c-MET test datasets. The curve

for experimental data is not shown, even though it is its own “simulator”, because there is no experimental data for half of the test set (the decoy compounds). We classified all compounds from the BindingDB BRD4(2) and c-MET datasets, which had activity $< 1\mu M$, as “Active”, and all other decoy compounds as “Inactive.” “AutoDock4 (binding energy only)” uses the total binding energy prediction from AutoDock4, while “AutoDock4 (all outputs)” uses a linear surrogate model that takes all outputs from AutoDock4 as input and outputs a prediction of the ABFE score.

As shown, ABFE is the most predictive data source for experimental data, with a ROC-AUC of 0.97 for BRD4(2) and 0.89 for c-MET. As expected, the total binding energy from AutoDock4, the computationally cheaper data source, is a worse predictor, but is still moderately predictive (ROC-AUC of 0.64 and 0.62). AutoDock4 (all outputs), which uses all outputs from AutoDock4 to make predictions, is more predictive (ROC-AUC of 0.78 and 0.75) than AutoDock4 total binding energy, but still worse than ABFE. We also measured the correlation between the binding energy predictions from AutoDock4 and ABFE, finding a correlation of $r = 0.25$ for BRD4(2) and $r = 0.23$ for c-MET. This helps explain why the multi-fidelity approach works, because the cheaper simulator is correlated with the more expensive simulator.

These results show that the MFBind environment has the desirable property that the more expensive simulators make more accurate predictions, meaning that it holds promise as an approach to making high-quality ABFE predictions without incurring an infeasibly high computational cost. They also help motivate the multi-output approach with AutoDock4, because it appears that considering all outputs from AutoDock4 is more useful than just the total energy prediction.

C.2. Compounds from LIMO generation

Figure 7 shows the top 3 compounds for each target generated from the LIMO generative model (Eckmann et al., 2022) with MFBind as the reward function. As shown, the generated compounds are relatively drug-like while showing favorable binding.

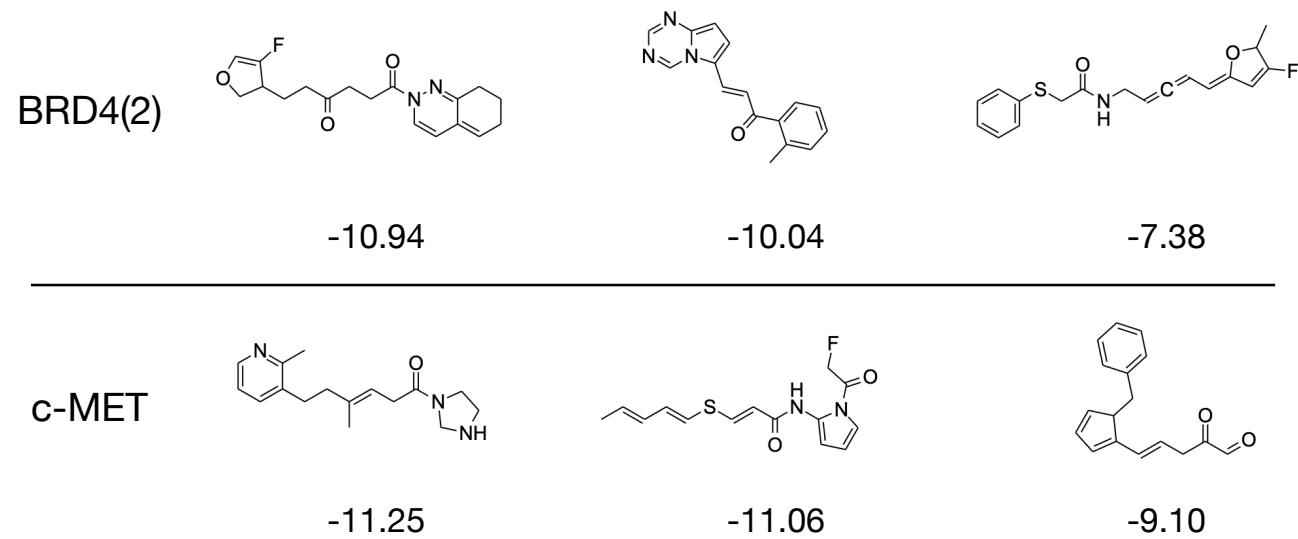


Figure 7. Generated compounds from LIMO + MFBind. The ABFE score of each compound is shown at the bottom. The top row shows the top 3 compounds generated for BRD4(2), and the bottom row for c-MET.

C.3. Generation with MolDQN

Table 3 shows the results from the same experimental procedure as from Section 4.2, except using MolDQN as the generative model. We only tested the BRD4(2) target. While we decided not to include these results in the body text due to the generated compounds not being very drug-like, these results show that MFBind is superior to the single-fidelity approaches on another generative model. The lack of drug-likeness is likely due to the nature of the generative model, and cannot be attributed to the surrogate model used as the reward.

Figure 8 shows the top 3 generated compounds from MolDQN with MFBind as the reward function. The compounds are generally non drug-like and have implausible structures such as the triple bond in a ring on the rightmost compound.

Table 3. Evaluation of MolDQN-generated compounds for BRD4(2). The mean and top 3 ABFE-computed energies are shown among 20 tested compounds from each method. “SF” refers to single-fidelity methods that only use one simulator, while our “MFBIND” approach uses all simulators. All compounds have QED > 0.5.

METHOD	MEAN	1ST	2ND	3RD
SF ABFE	1.78	-6.52	-4.47	-3.89
SF AUTO DOCK4	-1.14	-6.66	-6.64	-5.41
MFBIND	-3.41	-14.06	-8.14	-7.06

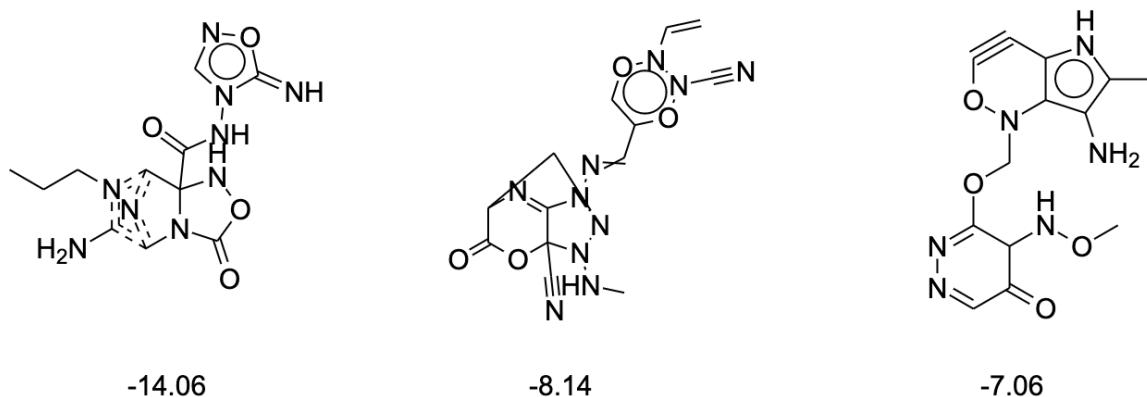


Figure 8. Generated compounds from MolDQN + MFBBind. The ABFE score of each compound is shown at the bottom. The compounds are generally non drug-like, and some are chemically implausible (e.g. the rightmost compound with a triple bond in a ring).