A Biosafety-aware Framework for Generative Enzyme Design with Foundation Models

Xiaoyi Fu Tao Han Yuan Yao Song Guo
The Hong Kong University of Science and Technology
Clear Water Bay, Kowloon, Hong Kong, China
xfu854@connect.ust.hk
hantao10200@gmail.com
yuany@ust.hk
songguo@cse.ust.hk

Abstract

Generative enzyme design reduces wet lab costs by virtually screening high-reward variants of a wild-type enzyme from a vast, high-dimensional search space. This becomes particularly challenging when multiple substrates and reactions for the same enzyme yield complex reward functions, such as Enzyme Kinetic Parameters (EKP), compounded by increasing biosafety constraints from stakeholders. This paper presents an integrated framework with a Generative Flow Networks (GFlowNets) model tailored for enzyme design and a fine-tuned protein language model for predicting EKP. Different from existing related work, our framework handles the complex EKP landscape introduced by the hydrolysis reaction mixture with the enzymatic reaction. By preliminary experiments, our framework shows it can generate high-reward enzyme variants under biosafety constraints faster than alternative related methods.

1 Introduction

With the advancements of biotechnology and protein engineering, enzymes are increasingly utilized across various sectors, including food industry [Raveendran et al., 2018], medicine [Voller et al., 1976], agriculture [Bowles et al., 2014], environmental science [Bollag, 1992], etc. While wild enzymes exhibit efficient catalytic activity, their performance for specific substrates often requires calibration [Tiffany et al., 1972]. The technology of improving the substrate specificity of enzyme [Carter and Wells, 1987] has a broad impact on biological sciences, particularly in areas such as metabolic engineering [Bonk et al., 2018], and synthetic biology [Urlacher and Girhard, 2019]. Consequently, the directed evolution of enzymes [Schmidt-Dannert and Arnold, 1999] for more efficiency in a specific reaction becomes a cornerstone technology.

As an enabler to the computational directed evolution of enzymes, the Enzyme Kinetic Parameters (EKP) [Choi et al., 2017] are crucial for understanding the efficiency of a specific enzymatic reaction. Traditionally, the EKP's measurement is through experimental methods which require meticulous laboratory work [Eisenthal and Cornish-Bowden, 1974]. However, recent advancements in computational methods [Kiss et al., 2013] have provided new ways to predict and analyze these parameters. Some of these computational approaches utilize foundation models (FM) e.g. protein language models [Bepler and Berger, 2021]. Framework based on protein language models which are fine-tuned for the prediction of EKP, have also been developed to carry out these predictions. UniKP [Yu et al., 2023a], for instance, excels in predicting the kinetic parameters for enzymes in reactions with one specific substrate.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

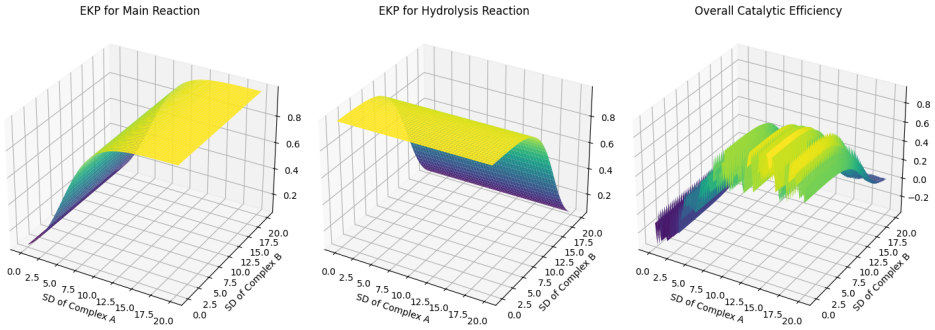


Figure 1: *SD* stands for structure divergence, the units are ignored in the caption for brevity. The bumpy curved surface in the rightmost reward landscape could be partially due to the penalty over the overall catalytic efficiency for toxicity.

The modes to the EKP function of different enzyme variants become even more complex when two or more substrates are involved in the reaction, e.g. when the enzymatic reaction is often accompanied by a Hydrolysis reaction [Qi and He, 2006] in which water (H_2O) acts as a receptor substrate, because of two reasons. First, although the divergence between the enzyme's structure is monotonic to the divergence between the enzyme's sequence by experiments [Chothia and Lesk, 1986], the binding energy is not. This can be better illustrated by the experiment of homing endonuclease I-AniI in [Thyme et al., 2009], where a small change in enzyme structure leads to significant differences in binding energy. Secondly, the simultaneous occurrence of two reactions, even when both exhibit monotonic functions for their kinetic parameters, can lead to complex patterns that introduce multiple modes to the reward function that represents the overall catalytic efficiency of reactions. We could imagine the curved spaces in Figure 1 for illustration propose of this phenomenon.

The above phenomenon confines the applicable domain of existing biological sequence design methods, e.g. directly using the predicted utility as the reward function for a Reinforcement Learning agent Angermueller et al. [2019], for enzyme's design especially when the safety of synthetic biological sequence [Hoffmann et al., 2023] needs to be addressed. To this end, we propose a novel framework based on Generative Flow Networks (GFlowNet) [Bengio et al., 2021], a genre of generative model known for its capability to effectively sample the multiple modes to the utility functions of complex biological sequences [Jain et al., 2022]. Our core hypothesis is that by increasing the diversity of generated sequences, we can enhance our chance of identifying an enzyme design that is not only more efficient but also safer.

Our main contribution is a holistic framework that integrates generative models for enzyme design and the foundation models for the prediction of EKP. To the best of our knowledge, this work marks the first application of GFlowNet in enzyme design for substrates with a secondary Hydrolysis reaction. By preliminary experiments, our framework has shown its efficiency of generating *high-reward enzyme variants* within biosafety constraints faster than alternative related methods.

2 Preliminary

Our overall goal is to propose, for efficient wet lab verification, a batch of K most catalytic efficient enzyme variants $B_i = \{x_1^i, ..., x_k^i\}$ under the constraints of a biosafety measure, given the dataset $\mathcal{D}_i = \cup_{j=1}^{i-1}\{x^j\}$ of currently visited variants in each round $i \in \{1, ..., N\}$. We use Generative Flow Networks (GFlowNets) [Bengio et al., 2021] to generate a trajectory of mutations starting from a given amino acid sequence $\mathcal{D}_0 = \{\mathbf{SEQ}(E_0)\}$ of the wild enzyme E_0 to achieve this goal. In this section, we first introduce two key enablers to facilitate our design.

2.1 Prediction of Catalytic Efficiency

Assume an enzymatic reaction shown in Eqn. 1, and let E represent the enzyme, S represent the substrate, ES represent the enzyme-substrate (E–S) complex, and P represent the product:

$$E + S \underset{k_{-1}}{\overset{k_1}{\rightleftharpoons}} ES \xrightarrow{k_2} E + P \tag{1}$$

where k_1, k_{-1}, and k_2 represent their conversion rates to each other. $K_{cat} = k_2$ is the enzyme turnover number, $K_m = (k_{-1} + k_2)/k_1$ is Michaelis constant, and $K_{cat}/K_m)$ is enzyme catalytic efficiency. K_m is a parameter that describes the affinity between the enzyme and the substrate. It was proposed by German biochemist Leonor Michaelis and Canadian physician Mott Menton in the Michaelis-Menton kinetic equation in 1913.

The method of UniKP [Yu et al., 2023a] uses pre-training language models to capture the characteristics of enzyme sequences and substrate structures, thereby achieving accurate prediction of enzyme kinetic parameters. Concretely, UniKP uses a fine-tuned protein language model [Elnaggar et al., 2021] to predict enzyme kinetic parameters $\mathbf{EKP}(S, E)$ by encoding the amino acid sequences of the enzyme $\mathbf{SEQ}(E)$ in concatenation with the simplified molecular linear input specification (SMILES) string of the substrate $\mathbf{SMILES}(S)$ and other information such as environmental factors to perform the prediction of EKP.

2.2 Verification of Biosafety

The increasing accessibility to powerful and affordable synthetic biology technologies puts us at risk of inadvertent or deliberate creation and dissemination of pathogens [Hoffmann et al., 2023]. It is widely accepted that it is imperative and strategic for the regulatory bodies to provide safeguarding frameworks that evaluate potential risks associated with synthetic sequences, considering factors like pathogenicity, environmental impact, and potential for horizontal gene transfer. Traditionally, there are two key ways to measure biosafety:

- In Silico: use computation tools to predict the behavior of synthetic sequences in biological systems, assessing their stability, potential interactions, and unintended consequences.
- In Vitro: conducting a wet lab experiment to observe the effects of synthetic sequences, allowing for the assessment of toxicity, infectivity, and other biological responses.

Our framework integrates an improved version [Sharma et al., 2022] from a series of in-silico tools for assessing protein toxicity, of which the original version [Gupta et al., 2013] is a classifier trained over a dataset containing toxic peptides having 35 or fewer residues from various databases and non-toxic peptides randomly obtained from protein databases e.g. SwissProt [Boeckmann et al., 2003].

3 Methods

As the core generative model of the introduced framework, GFlowNets are designed to approximate an edge flow function defined over a graph G. The objective is to ensure that the terminal flow matches the reward function $R(x)$ while maintaining flow consistency. This is accomplished by establishing a loss function whose global minimum satisfies the consistency requirement. The initial formulation [Bengio et al., 2021] of this approach utilizes a learning objective akin to temporal difference methods, also referred to as flow-matching:

$$\mathcal{L}_{FM}(s;\theta) = \left(\log \frac{\sum_{s' \in \text{Parent}(s)} F_\theta(s' \to s)}{\sum_{s'' \in \text{Child}(s)} F_\theta(s \to s'')} \right)^2 \tag{2}$$

In [Bengio et al., 2021] proves that when trajectories are sampled from an exploratory training policy with full support, the edge flow that maintains consistency is obtained by minimizing Eqn. 2. At the same time, the probability of arriving at the 'final state' by this flow, which samples mutated enzyme variants x with a probability $P_{F_\theta} = \frac{F_\theta(s \to s')}{\sum_{s'' \in child} F_\theta(s \to s'')}$, is proportional to their reward $R(x)$.

In our design, we use the overall catalytic efficiency measured by the above-introduced UniKP framework as the reward function $R(x)$ in GFlowNets. The more efficient enzyme variants will receive a higher reward. Concretely, the predicted parameter K_{cat}/K_m of the enzyme is used to quantify the catalytic efficiency for a single reaction, and when two reactions are together the following formula is used to compute the overall catalytic efficiency.

$$R(x) = \mathbf{EKP_B}^{\alpha}(\mathbf{EKP_A} - \beta * \mathbf{EKP_C}) \tag{3}$$

- $\mathbf{EKP_B}$: The predicted catalytic efficiency K_{cat}/K_m of enzyme E for donor substrate S_B.
- $\mathbf{EKP_A}$: The predicted catalytic efficiency K_{cat}/K_m of enzyme E for acceptor S_A.
- $\mathbf{EKP_C}$: The predicted catalytic efficiency K_{cat}/K_m of enzyme E for water.
- α and β: weight parameters used to adjust the reward function.

The prediction of $\mathbf{EKP}(\mathbf{SEQ}(E), \mathbf{SMILES}(S))$ is enabled by the inference of a pre-trained UniKP model over different combinations of one substrate S with the target enzyme E.

3.1 Generating Mutations

The process of GFlowNet to generate new enzyme variants is through a series of actions, each resulting in a new mutation to a wild-type enzyme (one that exists in nature). Details are as follows:

We start from an initial state, which is the initial amino acid sequence of wild-type enzyme $\mathbf{SEQ}(E_0)$.

First, select an action. At each step, GFlowNet randomizes an action a_t with probability proportional to GFlownet's current policy, parameterized as $\pi(s)$ through the neural network. The action determines how to modify the current state s_t. Concretely, the s_t is a candidate mutation to $\mathbf{SEQ}(E_0)$. The action continuously modifies the current mutation by one of the following three choices:

- 1) *substitute the target position with another amino acid*,
- 2) *increment the target position of mutation by one (with a maximum of $StepSize$)*,
- 3) *increment the target position of mutation by $StepSize$ (with a maximum of $StepNums$).*

Update the current state, then repeat. The current state s_t is updated to a new state s_{t+1} based on the selection. The process of selecting actions and updating states is repeated until a final state s_n is reached, which represents an enzyme variant x.

Stop at final states. The final state can be reached when either of the two conditions are met: 1) the action a_t generated by GFlownet represents a 'stop' action, or 2) all available states are visited exhaustively. The validity of a final state also has to be verified by a biosafety assurance step which will be described later in this section.

3.2 Learning GFlowNets

Calculate Reward Once a final state is arrived, the reward $R(x)$ of the generated enzyme variant is calculated using Eqn. 3. More specifically, it involves three steps as follows:

- Convert the amino acid sequence of candidate enzyme variants and substrate SMILES structure into a vector representation.
- Use the pre-trained UniKP model to predict the comprehensive vector and calculate each of the enzyme's K_{cat}/K_m parameters, namely EKP_A, EKP_B, and EKP_C in Eqn. 3.
- Set the hyper-parameters α and β then feed them with the predicted K_{cat}/K_m parameters in the above step into the $R(x)$ formula for calculation, and record each generated enzyme variant and its corresponding reward value. The reward value is subject to be penalized by multiplying a factor γ range between 0 and 1 for sequences that are classified as toxic.

Set Reward Threshold Only when the reward value of the variant is higher than the threshold set by top K (e.g. K=16) highest rewarding enzyme variants, it can be considered a *high-reward enzyme variant* and should be recorded for reporting. The searching process can be early stopped when all top K is found. The threshold was pre-computed in our experiments, however, it is worth further investigation on how to set the threshold in practice.

Calculate Policy Gradient After arriving at a batch size K number of final states, the framework calculates the policy gradient based on the loss function given by Eqn. 2, and adjust the parameters of the GFlowNets to increase the probability of generating high-reward variants.

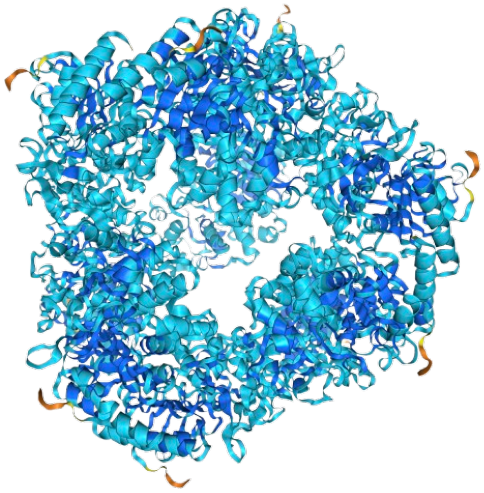
Generate New Variants Generate new enzyme variants using the updated GFlowNets' policy. The new strategy, updated by the back-propagation of policy gradients, should be more inclined to generate high-reward enzyme variants. Then according to the new strategy, we repeat the above steps until all safe *high-reward enzyme variants* are found or the number of training steps is reached.

Assessing Consistency To monitor the convergence of the policy generated by GFlowNets, we assess the empirical consistency of P_{F_θ} occasionally. Concretely, the density of each final state visited so far is calculated and compared with the true density given by the reward function $R(x)$.

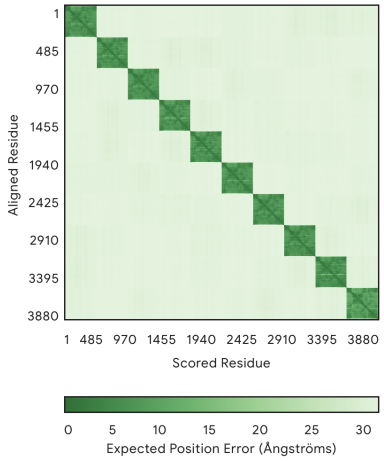
3.3 Biosafety Assurance

For every enzyme variant mutated by a final state, we use a pre-trained classifier [1] to measure its toxicity. Our framework first converts each final state into a mutated sequence, then relies on the predicted label (toxic/non-toxic) of the classifier over this sequence to decide whether to record a corresponding final state into a batch for output or repeat the generative process of GFlownets to sample more variants to meet the biosafety requirements. Compared with the reward function, this step is performed more frequently, because for every 'stop' action it has to be called the classifier. To this end, we choose a simpler (comparing to the protein foundation model used for $R(x)$) but effective (achieved testing accuracy around 90% according to [Rathore et al., 2024]) model.

4 Experiments



(a) 'Ribbon Diagram' for the 3D molecular structure of a enzyme variant for EstCE1 generated by our framework.



(b) The contact map of Expected Position Error (EPE) for aligned residues in the predicted structure of the enzyme variant in (a).

(c) What different colors represent in the 'Ribbon Diagram' in (a).

Figure 2: A generated variant of EstCE1, evaluated by AlphaFold3 server [Abramson et al., 2024].

4.1 Initial Dataset

We chose the EstCE1 as the enzyme E_0 in initial dataset $\mathcal{D}_0$ for our experiments because of follows:

[1] https://github.com/raghavagps/toxinpred2

- Efficient acyl transfer activity: EstCE1 is an efficient acyl transferase with significant catalytic activity for esterification, irreversible amidation of amines and carbamoylation. This enables it to exhibit excellent catalytic performance in acyl transfer reactions and is an ideal starting point for optimization studies.
- Unique structural properties: EstCE1 adopts a β-lactamase folding structure and has a catalytic triad (S65, K68 and Y171) located at the interface of the α/β subdomain and a more flexible helix. Its active site is covered by a significantly expanded Ω-loop, which can effectively block the central part of the R1 subsite facilitating acyl transfer activity.
- Conserved WGG motif: The conserved WGG motif near the catalytic triad has an important impact on the formation of the hydrophobic cavity at the substrate binding site, helping to improve the substrate affinity and catalytic efficiency of the enzyme. These structural features give EstCE1 strong catalytic ability in acyl transfer reactions.
- Rich research foundation: EstCE1 has been widely studied in existing studies and proven to have efficient acyl transfer-ability. Further optimization and improvements on EstCE1 can not only make full use of its known catalytic properties but also have a better chance of reaching a practical level of catalytic performance.

4.2 Evaluation Protocol

In this section, we present experimental results comparing different algorithms for exploring the single-point mutation of EstCE1. Our experiments setting, following the protocol as [Bengio et al., 2021], aims to show the effectiveness of our GFlowNets-based method (can be labeled as "flownet" in the legend) against MCMC (Markov Chain Monte Carlo), PPO (Proximal Policy Optimization), and a random baseline. The evaluation protocol has several key varying parameters as follows:

- α: the exponential factor in reward function - This is varied across three different magnitudes $(0.5, 1, 1.5)$ to test the algorithm's robustness under different intensities of reactions.
- β: the hydrolysis factor in reward function - This is varied across two different magnitudes $(0.1, 0.5)$ to test the algorithm's robustness under the mixed hydrolysis reaction.
- Number of states visited: This is the primary x-axis in both plots, ranging from 0 to $8 * 10^3$, subject to early stopping. It represents the exploration budget or time given to each algorithm.

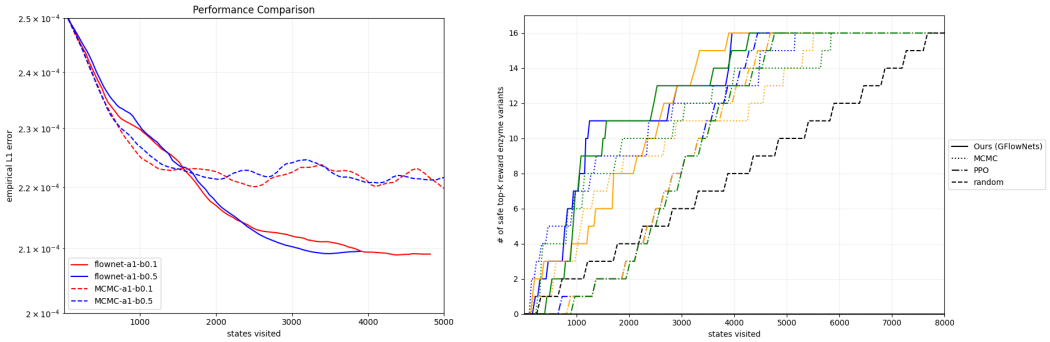
The evaluation metrics used are:

- *Empirical L1 error* measures the consistency of the estimated density compared to the true density for the distribution of the final states.
- Number of *safe high-reward enzyme variants* found measures the diversity of high-reward enzyme variants discovered, with the horizontal line delineating the maximum possible number in a single-point mutation for EstCE1.

By varying these parameters and comparing them with multiple baselines, we aim to provide a comprehensive review of our method's performance in designing catalytic efficient enzyme variants. The penalty factor γ introduced before is sensitive to different algorithms and tuning of other hyper-parameter settings. In our practice, introducing the penalty in the reward function does not always improve the overall performance and may sometimes lead to difficulties in convergence. Therefore, we did not include it here but left it for future investigation.

4.3 Results and Discussions

The Fig. 3a on the above left reveals several interesting trends. Our framework (solid lines) consistently outperforms the MCMC method (dashed lines) for both parameter settings, achieving lower error rates across most ranges. Notably, the GflowNets with the configuration of α = 1 and β = 0.5 (blue solid line) show the best performance overall, the fastest reaching the lowest error rate at the steps of early stopping. The difference in performance between the two parameter settings (β = 0.1 vs β = 0.5) is more pronounced for the MCMC method than for GFlowNets. This suggests that the GFlowNets approach may be more robust to parameter tuning, in addition to being more capable of achieving better results when optimally configured. The diminishing decreasing rate and fluctuation



(a) Comparison between GFlowNets (labeled as "flownet") and MCMC, each with two settings of β (e.g. labeled as 'b0.1') and a fixed α (= 1).

(b) We compare the number of top K (=16) *enzyme variants* found by each algorithm and setting. Blue, orange, and green lines each stand for $\alpha = 1.5, 1.0, 0.5$.

Figure 3: Our framework (w/ GFlowNets) consistently outperforms the alternative in both figures. We fix a $StepSize$ and $StepNums$ both of 20 and record the mean for 3 runs every 10 training steps.

in empirical error for MCMC at the beginning of the range indicates that GFlowNets benefits more significantly from increased sampling, with the potential for further investigation.

Fig. 3b compares several methodologies with a fixed β = 0.5. The GfFlowNets (flownets) algorithm used in our framework is compared using three distinct α values (1.5, 1, and 0.5) settings with three baselines: Markov Chain Monte Carlo (MCMC), Proximal Policy Optimization (PPO), and a random search baseline. The results demonstrate a clear performance hierarchy among these approaches. Generally speaking, algorithms with higher alpha values (e.g., 1.5) settings demonstrate enhanced efficiency, discovering a greater number of variants in fewer state visitations. This suggests that increasing alpha values may also facilitate a more effective exploration of the enzyme variant search space.

Both MCMC and PPO algorithms display comparable performance profiles, generally surpassing the random search baseline but falling short of the efficiency demonstrated by GFlowNets. As anticipated, the random search method exhibits the least efficient performance among Compared to all baselines, GFlowNets achieve the search goal significantly faster than other algorithms, indicating their superior efficiency in thoroughly exploring the variant space.

The above observations, putting them all together, partially validate our hypothesis that by increasing the diversity of sampled mutation trajectories, we can arrive at the *high-reward enzyme variants* under biosafety constraints with fewer budgets in terms of dry lab iterations.

5 Related Work

Alphafold, as noted by Abramson et al. [2024], exemplifies a significant advancement in deep learning (DL) applications for protein structure prediction, marking a pivotal achievement in the field of bioinformatics Sapoval et al. [2022]. One of the industrial applications of protein engineering is enzyme engineering, which can hence greatly benefit from the rapid advancements in DL techniques. These advancements have enabled the tailoring of enzymes for specific functions, as highlighted by [Singh et al., 2021] and [Yu et al., 2023b]. The consensus within the scientific community is that deep learning will revolutionize and enhance enzyme design [Goldman et al., 2022]. For example, MusiteDeep, a deep-learning-based web server, exemplifies the use of DL for predicting and visualizing protein post-translational modification sites [Wang et al., 2020]. Furthermore, the application of artificial intelligence in enzyme and pathway design for metabolic engineering has also shown promising results Jang et al. [2022].

Recent advances in generative AI open new avenues for the automated design of biological sequences, although the field of protein design has undergone significant transformations over the past four decades [Chronowska et al., 2024]. Despite these advancements, existing enzyme design databases remain relatively small compared to the vast diversity and abundance of potential enzyme variants. Innovative methods based on generative AI, such as machine learning-guided co-optimization of fitness

and diversity, have shown promise in accelerating the accumulation of enzyme design data [Ding et al., 2024]. A primary consideration in engineering new enzymes is enhancing their catalytic efficiency, which deep learning has successfully addressed by predicting genome-scale Michaelis constants from structural features [Kroll et al., 2021]. Recent efforts have leveraged foundational models in machine learning to facilitate few-shot and zero-shot predictions for unseen enzyme variants through transfer learning [Yu et al., 2023a]. In addition to catalytic efficiency, biosafety has become increasingly crucial in the context of synthetic biology's rapid industrialization. Machine learning models can streamline biosafety assessments by serving as in-silico proxies for evaluating the toxicity of new sequence designs [Gupta et al., 2013].

To integrate these predictors for protein's property, reinforcement learning-based agents can be employed to create novel enzyme designs [Angermueller et al., 2019]. However, utilizing predicted properties directly as reward functions in these models can lead to practical challenges. Recent developments in GFlowNets have demonstrated unique advantages for biological sequence design tasks, enabling the generation of more diverse candidates and mitigating the risk of converging on suboptimal solutions. The most salient point of difference between GFlowNets [Bengio et al., 2021] and Reinforcement Learning (RL) [Angermueller et al., 2019] for biological sequence design lies in the definition of the objective function. The flow function in GFlowNets estimates the proportion of the total reward of a state or transition, while the value function of RL estimates the expected amortized total rewards starting from a state or transition. The reward function of GFlowNets focuses on the distribution of the final state, which enables efficient sampling of high-reward final states.

6 Conclusions

This paper introduces a comprehensive system for enzyme engineering that combines a Generative Flow Network model with an EKP predictor based on a protein language model for designing enzyme variants. Unlike previous approaches, our system is equipped to handle the enzymatic reaction mixed with hydrolysis reaction and provides biosafety assurance. In the future, how to fine-tune the pre-trained protein language model to support the prediction of catalytic efficiency for multiple substrates in mixture reactions is worth further investigation.

Acknowledgement

Y. Yao was in part supported by the HKRGC GRF-16308321 and the NSFC/RGC Joint Research Scheme Grant N_HKUST635/20. The authors would like to extend their gratitude to Edge Science Limited for providing computing resources and conducting partial experiments.

References

Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J Ballard, Joshua Bambrick, et al. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, pages 1–3, 2024.

Christof Angermueller, David Dohan, David Belanger, Ramya Deshpande, Kevin Murphy, and Lucy Colwell. Model-based reinforcement learning for biological sequence design. In *International conference on learning representations*, 2019.

Emmanuel Bengio, Moksh Jain, Maksym Korablyov, Doina Precup, and Yoshua Bengio. Flow network based generative models for non-iterative diverse candidate generation. *Advances in Neural Information Processing Systems*, 34:27381–27394, 2021.

Tristan Bepler and Bonnie Berger. Learning the protein language: Evolution, structure, and function. *Cell systems*, 12(6):654–669, 2021.

Brigitte Boeckmann, Amos Bairoch, Rolf Apweiler, Marie-Claude Blatter, Anne Estreicher, Elisabeth Gasteiger, Maria J Martin, Karine Michoud, Claire O'Donovan, Isabelle Phan, et al. The swiss-prot protein knowledgebase and its supplement trembl in 2003. *Nucleic acids research*, 31(1):365–370, 2003.

Jean Marc Bollag. Decontaminating soil with enzymes. *Environmental Science & Technology*, 26 (10):1876–1881, 1992.

Brian M Bonk, Yekaterina Tarasova, Michael A Hicks, Bruce Tidor, and Kristala LJ Prather. Rational design of thiolase substrate specificity for metabolic engineering applications. *Biotechnology and bioengineering*, 115(9):2167–2182, 2018.

Timothy M Bowles, Veronica Acosta-Martínez, Francisco Calderón, and Louise E Jackson. Soil enzyme activities, microbial communities, and carbon and nitrogen availability in organic agroecosystems across an intensively-managed agricultural landscape. *Soil Biology and Biochemistry*, 68:252–262, 2014.

Paul Carter and James A Wells. Engineering enzyme specificity by" substrate-assisted catalysis". *Science*, 237(4813):394–399, 1987.

Boseung Choi, Grzegorz A Rempala, and Jae Kyoung Kim. Beyond the michaelis-menten equation: Accurate and efficient estimation of enzyme kinetic parameters. *Scientific reports*, 7(1):17018, 2017.

Cyrus Chothia and Arthur M Lesk. The relation between the divergence of sequence and structure in proteins. *The EMBO journal*, 5(4):823–826, 1986.

Marta Chronowska, Michael James Stam, Derek N Woolfson, Luigi F Di Costanzo, and Christopher W Wood. The protein design archive (pda): insights from 40 years of protein design. *bioRxiv*, pages 2024–09, 2024.

Kerr Ding, Michael Chin, Yunlong Zhao, Wei Huang, Binh Khanh Mai, Huanan Wang, Peng Liu, Yang Yang, and Yunan Luo. Machine learning-guided co-optimization of fitness and diversity facilitates combinatorial library design in enzyme engineering. *Nature Communications*, 15(1): 6392, 2024.

Robert Eisenthal and Athel Cornish-Bowden. The direct linear plot. a new graphical procedure for estimating enzyme kinetic parameters. *Biochemical journal*, 139(3):715–720, 1974.

Ahmed Elnaggar, Michael Heinzinger, Christian Dallago, Ghalia Rehawi, Yu Wang, Llion Jones, Tom Gibbs, Tamas Feher, Christoph Angerer, Martin Steinegger, et al. Prottrans: Toward understanding the language of life through self-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 44(10):7112–7127, 2021.

Samuel Goldman, Ria Das, Kevin K Yang, and Connor W Coley. Machine learning modeling of family-wide enzyme-substrate specificity screens. *PLoS computational biology*, 18(2):e1009853, 2022.

Sudheer Gupta, Pallavi Kapoor, Kumardeep Chaudhary, Ankur Gautam, Rahul Kumar, Open Source Drug Discovery Consortium, and Gajendra PS Raghava. In silico approach for predicting toxicity of peptides and proteins. *PloS one*, 8(9):e73957, 2013.

Stefan A Hoffmann, James Diggans, Douglas Densmore, Junbiao Dai, Tom Knight, Emily Leproust, Jef D Boeke, Nicole Wheeler, and Yizhi Cai. Safety by design: Biosafety and biosecurity in the age of synthetic genomics. *Iscience*, 26(3), 2023.

Moksh Jain, Emmanuel Bengio, Alex Hernandez-Garcia, Jarrid Rector-Brooks, Bonaventure FP Dossou, Chanakya Ajit Ekbote, Jie Fu, Tianyu Zhang, Michael Kilgour, Dinghuai Zhang, et al. Biological sequence design with gflownets. In *International Conference on Machine Learning*, pages 9786–9801. PMLR, 2022.

Woo Dae Jang, Gi Bae Kim, Yeji Kim, and Sang Yup Lee. Applications of artificial intelligence to enzyme and pathway design for metabolic engineering. *Current Opinion in Biotechnology*, 73: 101–107, 2022.

Gert Kiss, Nihan Çelebi-Ölçüm, Rocco Moretti, David Baker, and KN Houk. Computational enzyme design. *Angewandte Chemie International Edition*, 52(22):5700–5725, 2013.

Alexander Kroll, Martin KM Engqvist, David Heckmann, and Martin J Lercher. Deep learning allows genome-scale prediction of michaelis constants from structural features. *PLoS biology*, 19(10): e3001402, 2021.

Wei Qi and Zhimin He. Enzymatic hydrolysis of protein: Mechanism and kinetic model. *Frontiers of Chemistry in China*, 1:308–314, 2006.

Anand Singh Rathore, Shubham Choudhury, Akanksha Arora, Purva Tijare, and Gajendra PS Raghava. Toxinpred 3.0: An improved method for predicting the toxicity of peptides. *Computers in Biology and Medicine*, 179:108926, 2024.

Sindhu Raveendran, Binod Parameswaran, Sabeela Beevi Ummalyma, Amith Abraham, Anil Kuruvilla Mathew, Aravind Madhavan, Sharrel Rebello, and Ashok Pandey. Applications of microbial enzymes in food industry. *Food technology and biotechnology*, 56(1):16, 2018.

Nicolae Sapoval, Amirali Aghazadeh, Michael G Nute, Dinler A Antunes, Advait Balaji, Richard Baraniuk, CJ Barberan, Ruth Dannenfelser, Chen Dun, Mohammadamin Edrisi, et al. Current progress and open challenges for applying deep learning across the biosciences. *Nature Communications*, 13(1):1728, 2022.

Claudia Schmidt-Dannert and Frances H Arnold. Directed evolution of industrial enzymes. *Trends in biotechnology*, 17(4):135–136, 1999.

Neelam Sharma, Leimarembi Devi Naorem, Shipra Jain, and Gajendra PS Raghava. Toxinpred2: an improved method for predicting toxicity of proteins. *Briefings in bioinformatics*, 23(5):bbac174, 2022.

Nitu Singh, Sunny Malik, Anvita Gupta, and Kinshuk Raj Srivastava. Revolutionizing enzyme engineering through artificial intelligence and machine learning. *Emerging Topics in Life Sciences*, 5(1):113–125, 2021.

Summer B Thyme, Jordan Jarjour, Ryo Takeuchi, James J Havranek, Justin Ashworth, Andrew M Scharenberg, Barry L Stoddard, and David Baker. Exploitation of binding energy for catalysis and design. *Nature*, 461(7268):1300–1304, 2009.

TO Tiffany, JM Jansen, CA Burtis, JB Overton, and CD Scott. Enzymatic kinetic rate and end-point analyses of substrate, by use of a gemsaec fast analyzer. *Clinical Chemistry*, 18(8):829–840, 1972.

Vlada B Urlacher and Marco Girhard. Cytochrome p450 monooxygenases in biotechnology and synthetic biology. *Trends in biotechnology*, 37(8):882–897, 2019.

A_ Voller, DE Bidwell, and ANN Bartlett. Enzyme immunoassays in diagnostic medicine: theory and practice. *Bulletin of the World Health Organization*, 53(1):55, 1976.

Duolin Wang, Dongpeng Liu, Jiakang Yuchi, Fei He, Yuexu Jiang, Siteng Cai, Jingyi Li, and Dong Xu. Musitedeep: a deep-learning based webserver for protein post-translational modification site prediction and visualization. *Nucleic Acids Research*, 48(W1):W140–W146, 2020.

Han Yu, Huaxiang Deng, Jiahui He, Jay D Keasling, and Xiaozhou Luo. Unikp: a unified framework for the prediction of enzyme kinetic parameters. *Nature communications*, 14(1):8211, 2023a.

Tianhao Yu, Haiyang Cui, Jianan Canal Li, Yunan Luo, Guangde Jiang, and Huimin Zhao. Enzyme function prediction using contrastive learning. *Science*, 379(6639):1358–1363, 2023b.