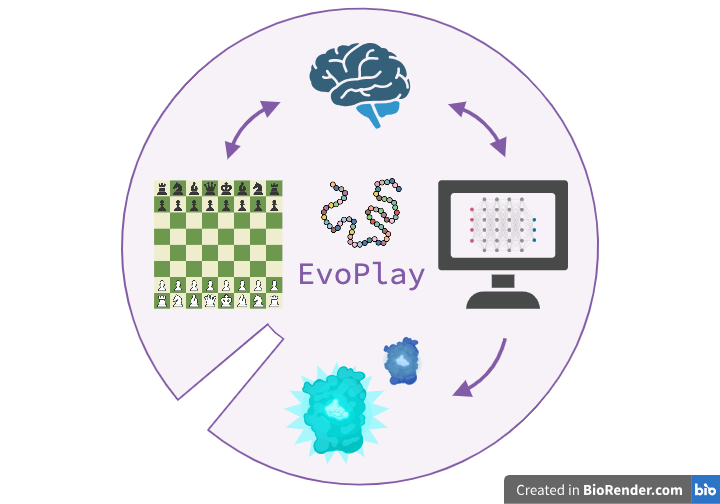
**EvoPlay: Self-play reinforcement expedites protein engineering1**

**Yi Wang, Hui Tang, Lichao Huang, Lulu Pan, Lixiang Yang, Huanming Yang, Feng Mu & Meng Yang**



Protein engineering is a useful method for tuning and enhancing properties of interest in natural organisms to exploit their proteins for industrial, biomedical, and environmental purposes. One main strategy, directed evolution (DE), iteratively mutates proteins and selects the most fit variant.2 Fitness is defined by the engineer and encompasses activity such as catalytic activity, bind affinity, or organismal growth rates. DE which is conducted in a lab suffers the vastness of protein sequence space. DE being revolutionized by improved sequencing techniques and machine learning (ML). ML has been shown to accelerate the protein design process and results in higher performance products.3 In general, implementations are trained on labeled sequence-fitness data and then predict the fitness of unseen variants. Recently, clustering-based and other unsupervised methods have been used to boost the performance of machine-learning-guided direction evolution (MLDE). An important consideration in both traditional DE and MLDE is the balance between exploration and exploitation. Exploration describes the determination of protein fitness over a diverse landscape; exploitation describes the iterative greedy-uphill walk towards a fitness optima. Many ML models have proclaimed to have the correct balance, although this is yet to be determined since the appropriate balance is landscape specific.

**Figure 1| EvoPlay is a self-play reinforcement learning framework for protein engineering based on AlphaZero**. Like a computer playing chess, EvoPlay navigates the protein sequence-fitness space by mutating a single residue to optimize a sequence. The model then computes mutant fitness and compares it with a look-ahead Monte Carlo tree search to guide the optimization agent in a manner akin to playing a piece in a game of chess. To demonstrate EvoPlay’s abilities, the authors engineered a fluorescent protein with 7.8-fold enhanced bioluminescence.

One newly proposed approach is reinforcement learning (RL). In RL, an intelligent agent learns how to perform actions by attempting to maximize a reward function. RL in combination with Monte Carlo tree search (MCTS) has been successful in winning against humans in chess and AlphaGO, notoriously strategic games.4,5 In the Nature Machine Intelligence paper, Wang et al. propose a single-player implementation of AlphaZero to optimize directed evolution. Their model, EvoPlay, performs *in silico* mutations and learns to propose new mutations based on the maximization of calculated fitness scores. They validate EvoPlay’s effectiveness with four tasks: (1) benchmarking of protein engineering in the full-length design space, (2) in silico DE of peptide binders using AlphaFold2 to incorporate structure-based reward, (3) implementation of MLDE for a four-site combinatorial library of GB1 and PhoQ and (4) engineering an enhanced fluorescent protein (Gaussia luciferase, GLuc).1

EvoPlay can be used to simulate random mutagenesis over the whole protein space or to simulate the more confined site-saturation mutagenesis. The authors describe EvoPlay’s optimization as a series of “sequential play episodes” in which a mutation is assumed, and the associated fitness simulated. If the simulated fitness is higher the in the previous round, the mutation is incorporated and the whole process repeated.

The EvoPlay framework can be used to improve upon any proposed starting sequence, which does not have to be a naturally occurring protein. In addition, it can accept multiple staring sequences for simultaneous optimization. The mutating positions can also be pre-selected or left to random choice as in error-prone PCR.

In each round of play, a single residue is mutated and the RL agent continues to simulate mutations within this amino acid context until a specified end condition is reached. Each simulation entails traversing the MCTS tree from the root node a leaf node where nodes represents variants. The method for calculation of fitness at each terminal node is determined by the user. Examples of methods are ML models, physical parameter models, or structure predictors.

In this paper, the authors demonstrated that EvoPlay’s neural network-guided MCTS can outperform existing tools for exploring the complex protein-fitness landscape. For Task 1, EvoPlay generated diverse novel protein sequences akin to green fluorescent protein (GFP) with higher maximal fitness than previously tested methods. Across all tasks, EvoPlay demonstrated robustness to multiple models for calculating fitness by preferring computational speed over nominally more accurate predictive models.1

To conclude their results, the authors several potential improvements to their model. First, they suggest that EvoPlay could be improved the inclusion of various encodings. They also suggest that EvoPlay may be improved by muting segments of residues simultaneously instead of individual residues.

This exciting advancement in protein engineering has the potential to expedite and expand the realm of catalytic capabilities.

1. Wang, Y., Tang, H., Huang, L. *et al.* Self-play reinforcement learning guides protein engineering. *Nat Mach Intell* **5**, 845–860 (2023).https://doi.org/10.1038/s42256-023-00691-9

2. Romero PA, Arnold FH. Exploring protein fitness landscapes by directed evolution. Nat Rev Mol Cell Biol. 2009 Dec;10(12):866-76. doi: 10.1038/nrm2805. PMID: 19935669; PMCID: PMC2997618.

3. Yang, K.K., Wu, Z. & Arnold, F.H. Machine-learning-guided directed evolution for protein engineering. *Nat Methods* **16**, 687–694 (2019). <https://doi.org/10.1038/s41592-019-0496-6>

4. Browne, C. B. et al. A survey of Monte Carlo tree search methods. *IEEE Trans. Comput. Intell. AI Games* **4**, 1–43 (2012).

5. Silver, D. et al. A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science* **362**, 1140–1144 (2018).