

Article

EvoPlay-MuZero Hybrid Framework Incorporating a Dual-Peptide Bridging Strategy for Adjunctive Therapy in Alzheimer's Disease

Bingling Huang [†], Jiahao Li [†], Ziyu Li [‡], Hao Jiang [‡], Mingxiang Yang [‡], Xiaoxia Li ^{*} and Xiaohui Niu ^{*}

College of Informatics, Huazhong Agricultural University, Wuhan 430070, China

^{*} Correspondence: lixiaoxia@mail.hzau.edu.cn (X.L.); niuxiaoh@mail.hzau.edu.cn (X.N.)

[†] These authors contributed equally to this work and share first authorship.

[‡] These authors contributed equally to this work and share second authorship.

Abstract

Alzheimer's disease (AD) is a severe neurodegenerative disorder whose pathological progression is closely associated with the reduced binding affinity of apolipoprotein E ϵ 4 (ApoE4) for amyloid- β (A β), which impairs A β clearance. Existing computational molecular design approaches are largely limited to single-target optimization and therefore lack the capacity for synergistic dual-target modulation. Herein, we proposed EvoPlay-MuZero, a hybrid computational framework incorporating a dual-peptide bridging (DPB) strategy. The framework adopted latent-state planning in MuZero reinforcement learning to enhance exploration and sequence-generation efficiency in high-dimensional sequence spaces. For the first time, it enabled the automated design of bispecific peptides targeting ApoE4 and A β , thereby forming a synergistic molecular bridge via a flexible linker. A full-process pipeline for structural and energetic evaluation was established by integrating AlphaFold3 and PDBePISA. Benchmark experiments on the 1SSC, 2CNZ, and 3R7G datasets demonstrated that EvoPlay-MuZero substantially outperformed the vanilla EvoPlay in convergence speed and the yield of valid generated sequences. Specifically, the optimal DPB molecule (15 $\times$ 25-3A) achieved a calculated interfacial solvation energy score ($\Delta^i G$) of -29.5 kcal/mol, demonstrating substantially enhanced interface-stabilization properties compared to the native baseline control. This study provides a novel molecular intervention strategy for adjuvant therapy in AD and highlights the considerable potential of reinforcement learning in multi-target drug design.

Keywords: reinforcement learning; MuZero; dual-peptide bridging; structural bioinformatics; multi-target drug design

Academic Editor: Ognjen Arandjelović

Received: 22 May 2026

Revised: 16 July 2026

Accepted: 16 July 2026

Published: 21 July 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\) license](https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Alzheimer's disease (AD), a major neurodegenerative disease worldwide, is characterized by an extremely complex pathogenesis, with clinical manifestations centered on plaques formed by amyloid- β (A β) deposition and neurofibrillary tangles caused by hyperphosphorylation of tau proteins [1]. Among the well-established genetic risk factors, the strongest association has been observed for apolipoprotein E ϵ 4 (APOE ϵ 4) [2,3]. It has been shown that the binding affinity of ApoE4 toward A β is significantly lower than that of the ApoE2/E3 isoforms, which impairs the efficiency of A β clearance [4]. However, the

mechanistic relationship between ApoE-A β interaction and clearance is highly non-linear and conformation-dependent; unconstrained or indiscriminate binding can paradoxically seed A β fibrillogenesis rather than promote degradation [5]. To safely drive clearance, an effective proximity-inducing ligand must selectively stabilize clearance-competent topologies that facilitate receptor-mediated endocytosis—such as via the low-density lipoprotein receptor-related protein 1 (LRP1) or LDLR pathways in microglia—without stabilizing the cross- β sheet nucleation of toxic aggregates [4]. Therefore, exploring new intervention strategies that can actively enhance the interaction between endogenous ApoE4 and A β to promote receptor-mediated microglial clearance has emerged as a highly promising avenue for AD drug discovery.

In recent years, the deep integration of artificial intelligence (AI) and computational biology has provided revolutionary tools for the rational design of complex biomolecules. Deep learning frameworks such as AlphaFold2/3 have not only enabled high-precision prediction of protein structures and complexes but also laid a solid foundation for the transition from “sequence design” to “functional optimization” [6,7]. With these high-precision models, protein binding interfaces can be analyzed at the atomic scale, allosteric regulatory sites that are difficult to detect through traditional experiments can be identified, and ultrahigh-throughput virtual screening can be facilitated, thereby transforming drug discovery and development from blind heuristic search to precise design based on physical constraints and data-driven approaches. Meanwhile, for protein sequence optimization and affinity maturation, cutting-edge frameworks such as graph neural network-based ProteinMPNN [8], surface fingerprinting-based MaSIF [9], and reinforcement learning-incorporated EvoPlay are used [10]. Remarkable progress has been made in optimizing the stability of single-target ligands and interfacial solvation score via elaborate energy function construction and surrogate model development [11].

However, existing protein design paradigms still suffer from substantial limitations. First, most computational design tools prioritize single-target ligand optimization and lack the ability to construct a molecular bridge between two endogenous proteins for co-regulation [9]. For complex systems requiring induced proximity, such as the ApoE4–A β complex, conventional methods struggle to balance the complex geometric constraints and energetic trade-offs between dual targets. Second, existing reinforcement learning-based design frameworks are often limited by low sample efficiency and search efficiency constrained by the speed of feedback from the underlying prediction environment (e.g., AlphaFold) when applied to high-dimensional, discrete amino acid sequence spaces, leading to slow convergence in complex multi-objective optimization tasks [10,12].

To address the aforementioned computational and therapeutic challenges, this study proposed a hybrid EvoPlay-MuZero framework integrated with the Dual-Peptide Bridging (DPB) strategy. The main contributions of this work are summarized as follows: (1) Architectural innovation: The MuZero reinforcement learning algorithm was introduced into the field of protein design, significantly improving the exploration efficiency of agents in large-scale sequence spaces through latent state planning and look-ahead search, while effectively alleviating the computational burden caused by the excessive reliance on real-environment feedback in traditional RL methods. (2) Design paradigm breakthrough: The first automated design of bridging peptides for the dual targets ApoE4 and A β was achieved. Specifically, two individual peptides were independently designed to specifically bind ApoE4 and A β , respectively. These two single-target peptides were subsequently assembled into a Dual-Specific Molecular Bridge (DMB) via a flexible alanine linker, thereby offering a novel molecular intervention model for enhancing endogenous protein interactions. (3) Closed-loop evaluation framework: Based on AlphaFold3, an end-to-end computational evaluation pipeline was constructed, spanning sequence generation, ternary complex structure prediction, interfacial solvation scoring, and molecular

dynamics (MD) simulations, assessing the engineered peptides from both structural interface compatibility and kinetic stability perspectives.

The remainder of this paper is organized as follows: Section 2 elaborates on the network architecture of EvoPlay-MuZero, the definition of the state space, and the computational workflow for dual-peptide design; Section 3 presents the experimental setup and dataset construction, followed by in-depth analyses of the benchmark results and the dynamic properties of the optimal DMB peptides; Section 4 discusses the methodological strengths, biological significance, and limitations of this study; and Section 5 summarizes the paper and outlines future research directions.

2. Materials and Methods

2.1. Dataset Construction and Preprocessing

To validate the performance of the algorithm, the three benchmark datasets (1SSC, 2CNZ, and 3R7G) adopted in EvoPlay were collected, which correspond to the PDB complexes with identical IDs. 1SSC (Ribonuclease S) is a canonical protein–peptide complex with robust interfacial interactions between S-protein and S-peptide and is widely utilized to benchmark peptide design accuracy. 2CNZ (MDM2–p53) is a complex between the tumor suppressor p53 and the E3 ubiquitin ligase MDM2, and serves as a standard model for the design of protein–protein interaction (PPI) inhibitors. 3R7G (Bcl-xL/Bak) represents a complex between the anti-apoptotic protein Bcl-xL and the pro-apoptotic peptide Bak, with a typical helix–groove binding pattern. To further support the specific design objectives of this study, a target dataset containing ApoE4 and A β sequence information was retrieved from the UniProt database [13]. For ApoE4 (UniProt ID: P02649), the full-length sequence covering the receptor-binding region was used, with particular attention to the conformational features of its C-terminal domain. For A β (UniProt ID: P05067), the A β 42 sequence cleaved from amyloid precursor protein (APP) was adopted, given the high aggregation propensity and neurotoxicity of this peptide fragment. Overall, two datasets were constructed: (1) a benchmark dataset; (2) a target dataset consisting of the ApoE4–A β complex.

For the benchmark dataset, which was mainly used to compare performance with that of the EvoPlay framework, the data processing followed the standard protocol reported in the previous work [10]. Specifically, PyMOL (version 3.1.4) was used to calculate the peptide center-of-mass coordinates and identify receptor interaction sites [14], and multiple sequence alignment (MSA) files for the receptor were generated using DeepMSA2 (<https://aideepmed.com/DeepMSA2/> (accessed on 20 March 2025)). For the target dataset, given the absence of high-resolution crystal structures for this complex, representative 3D structural templates were initially generated via AlphaFold3 (AF3) [7]. We explicitly note that these deep learning-predicted static models serve as operational computational scaffolds to guide algorithmic exploration, rather than definitive physiological binding modes, acknowledging the conformational dynamics inherent to disordered A β [15] and lipidated ApoE variants [4]. Subsequently, key interfacial residues and ligand center-of-mass coordinates were extracted using PyMOL, and multiple sequence alignment (MSA) files were generated using DeepMSA2 [16]. The resulting MSA files then served as input features for downstream structure prediction and sequence generation.

2.2. EMB-DPB: A Hybrid Computational Approach for EvoPlay-MuZero-Based Dual-Peptide Bridging

To address the biological challenge of insufficient binding affinity between ApoE4 and A β , we proposed a hybrid computational approach termed EMB-DPB (EvoPlay-MuZero-Based Dual-Peptide Bridging). This method aims to achieve molecular bridging between protein targets by integrating a reinforcement-learning-based planning model with a structural prediction environment.

2.2.1. Markov Decision Process (MDP) Formulation

In this study, the peptide sequence design process is modeled as a Markov Decision Process (MDP), defined as the quintuple (S, A, P, R, γ) :

- (1) State Space (S): The system state $s_t \in \mathcal{S}$ is defined as a $20 \times L$ binary one-hot matrix, where L denotes the length of the candidate peptide and 20 represents the standard amino acid types. This matrix is mapped into a latent space via a representation network to extract sequence distribution features [12].
- (2) Action Space (A): The action a_t is defined as a single-point mutation operation. A position and the corresponding amino acid type are selected from the $20 \times L$ action space for sequence update.
- (3) State Transition (P): The state transition $P(s_{t+1}|s_t, a_t)$ is deterministic; that is, after the mutation action is executed, the corresponding column of the sequence matrix is updated to generate a new candidate peptide.
- (4) Reward Function (R): The reward r_t combines physical constraints with energy-based estimates, as shown in Equation (1):

$$r = \frac{2 \cdot pLDDT}{(d_{peptide} + d_{receptor}) \cdot \Delta COM + \epsilon} \quad (1)$$

where $pLDDT$ denotes the AlphaFold-predicted local confidence of the peptide structure, $d_{peptide}$ and $d_{receptor}$ represent the mean nearest-atom distances between the peptide and receptor interface residues, and ΔCOM measures the deviation of the peptide center of mass from the reference binding pose. An infinitesimal stabilization constant ($\epsilon = 10^{-6}$) is appended to the denominator to prevent numerical singularity and reward explosion. The ratio-based formulation explicitly governs the optimization trajectory: the $pLDDT$ term in the numerator drives the agent to maximize local backbone structural confidence, whereas the distance and ΔCOM terms in the denominator introduce directional geometric penalties for interface separation and pose deviation [10,11]. In the implemented reward, the coefficient for $pLDDT$ was set to 2, while the geometric constraints were encapsulated within the joint denominator, establishing an intrinsically coupled objective that balances structural fidelity with binding-interface proximity.

- (5) Discount Factor (γ): It is employed to balance immediate rewards against long-term value estimates.

2.2.2. Methodological Framework

The EMB-DPB framework established a complete closed-loop pipeline from low-level feature extraction to high-level strategic planning through the coordinated operation of three core modules (Figures 1–3). First, during the input and feature representation stage, the system integrates multiple sequence alignment (MSA) information of the target proteins and key sequence features [16], and maps the high-dimensional raw observations to the initial latent state s_0 through the representation network (Figure 1). Subsequently, the MuZero planning engine serves as the core decision-making component and employs

its unique latent-state planning mechanism for sequence exploration (Figure 2); the dynamics function within the engine can predict subsequent state transitions and immediate rewards in the latent space without directly querying the external environment, thereby effectively alleviating the computational burden imposed by the high-dimensional discrete search space and significantly improving sampling efficiency [12]. Finally, within the structural validation module, the engineered candidate sequences are evaluated within a structural prediction framework using AlphaFold2/3 [6,7]. Recognizing the intrinsic confidence limitations of static neural networks when mapping flexible, disordered, and multichain peptide assemblies [17], this module yields structural representations that facilitate interfacial geometric constraint checking and multi-objective reward aggregation to guide iterative policy optimization (Figure 3).

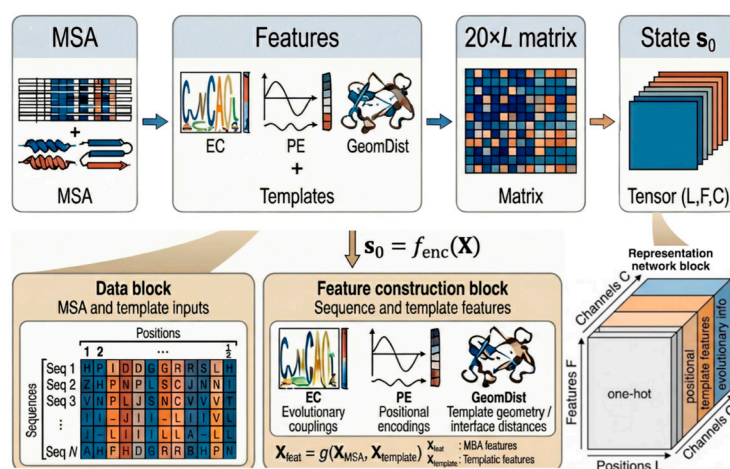


Figure 1. Pipeline for input feature extraction and state representation. The workflow illustrates the processing of multiple sequence alignment (MSA) evolutionary matrices, physical/positional templates, and geometric distance constraints into a standardized $20 \times L$ sequence matrix, which is ultimately mapped into the initial latent state space s_0 via the encoder network. The color gradients (blue to orange) across the feature matrices and data blocks denote distinct feature channels and residue-positional matrices (where blue components represent evolutionary coupling and positional features, and orange components indicate template-derived geometric distances), while the blue and orange arrows indicate the directional data flow and dimensional transformations into the $L \times F \times C$ state tensor.

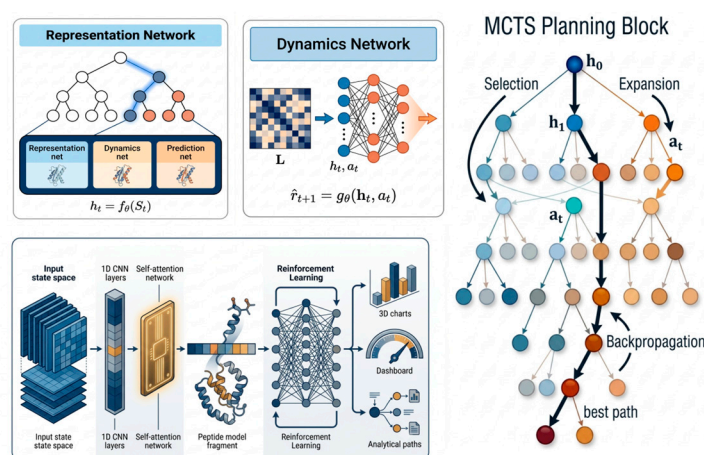


Figure 2. Architecture of the MuZero planning engine and Monte Carlo Tree Search (MCTS) block. The schematics detail: The forward inference pipeline across the representation, dynamics, and

prediction networks to predict latent states h_t ; the iterative MCTS optimization procedure encompassing selection, expansion, and backpropagation to discover the optimal sequence trajectory. In the MCTS planning block (right), nodes are color-coded from root state h_0 (dark blue) to leaf states (orange and dark red) to signify search depth and state transitions, and the directional arrows with labels represent the operational phases of selection, expansion, and the optimized best-path backpropagation.

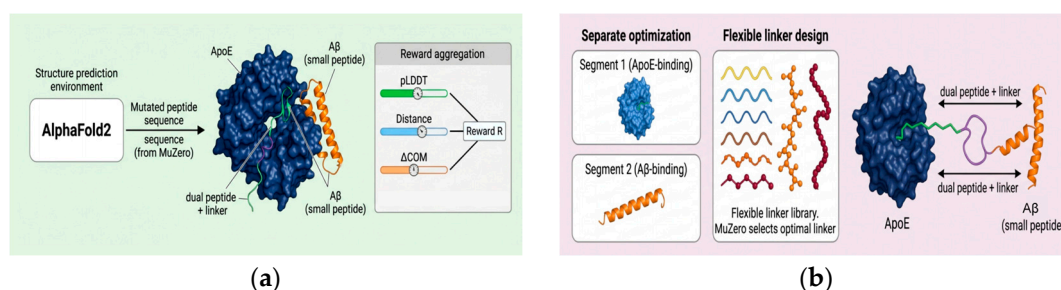


Figure 3. High-fidelity structural validation environment and multi-objective reward calculation. (a) End-to-end structural feedback loop showing the input of mutated sequences into AlphaFold2/3 to output conformational representations of the ApoE-peptide-Aβ complex. (b) The multi-objective reward aggregation scheme combining *pLDDT* local confidence scores, interfacial distances, and center-of-mass variations (ΔCOM); the proximity-induced dual-peptide bridging design paradigm showing independent segment optimization connected via a flexible linker. In both panels, distinct structural colors (deep blue for the ApoE receptor, orange for the Aβ peptide, green and purple for optimized linkers) and multi-colored wave lines in the linker library visually distinguish the modular components and their corresponding reward slider metrics.

2.3. MuZero Network Architecture and Training

Following the core principles of the MuZero algorithm [12], three functionally distinct neural networks were constructed: the representation network, the dynamics network, and the policy-value network. The representation network, consisting of a 1D convolutional layer, a batch normalization layer, and a residual block (ResNet block), is responsible for encoding the input $20 \times L$ sequence matrix into a low-dimensional hidden state vector; the dynamics network is similar in structure to the representation network but includes an additional module that flattens the output of the residual block and feeds it into a multilayer perceptron (MLP). Its input comprises the current hidden state and the embedded action vector, while its output consists of the next hidden state and the immediate reward; the policy-value network, similar in structure to the dynamics network, takes the hidden state as input and outputs the prior probability distribution over all possible actions and the value estimate of the current state. It also encapsulates part of the interface used by Monte Carlo Tree Search (MCTS); its specific implementation is shown in Algorithm 1.

Algorithm 1 MuZero Core Network Interface

1: **Function** *InitialInference(o)*:

Input: Observation o (raw environment observation)

Output: value logits v_{logits} , reward logits r_{logits} , policy logits π_{logits} , and encoded state s

2: $s \leftarrow \text{representation_network}(o)$ // Encode the observation into a hidden state

3: $\pi_{\text{logits}}, v_{\text{logits}} \leftarrow \text{prediction_network}(s)$ // Predict policy and value

4: $r_{\text{logits}} \leftarrow 0$ // Initial reward logits are set to 0

```

5:   Return  $v_{logits}, r_{logits}, \pi_{logits}, s$ 
6:   min_max_stats  $\leftarrow$  MinMaxStats() // Initialize value normalization statistics
7: Function RecurrentInference( $s, a$ ):
      Input: Encoded state  $s$ , action  $a$ 
      Output: value logits  $v_{logits}$ , reward logits  $r_{logits}$ , Policy logits  $\pi_{logits}$ , and next
                encoded state  $s'$ 
8:    $a_{one-hot} \leftarrow one - hot(a)$  // Convert the action to one – hot encoding
9:    $x \leftarrow concat(s, a_{one-hot})$  // Concatenate the latent state and action
10:   $s' \leftarrow dynamics\_state\_network(x)$  // Predicting the next hidden state
11:   $r_{logits} \leftarrow dynamics\_reward\_network(s')$  // Predict reward logits
12:   $\pi_{logits}, v_{logits} \leftarrow prediction\_network(s')$  // Predicting policy and value
13:  return  $v_{logits}, r_{logits}, \pi_{logits}, s'$ 

```

The training objective of MuZero is to align the predicted outputs with their corresponding targets, i.e., to minimize the discrepancies between the output policy p and the MCTS policy π , between the predicted value v and the MCTS-derived target value, and between the predicted reward and the actual reward. Accordingly, the loss function is formulated as follows:

$$l(\theta) = \sum_{k=0}^K l_p(\pi_{t+k}, p_t^k) \cdot \eta + \lambda_v \sum_{k=0}^K l_v(z_{t+k}, v_t^k) + \sum_{k=1}^K l_r(u_{t+k}, r_t^k) \quad (2)$$

where $K=5$ is the number of unrolling steps of the dynamics network and η is the weight assigned to the policy loss. The policy loss l_p uses cross-entropy loss to learn effective exploration patterns by measuring the difference between the predicted policy distribution and the target policy produced by MCTS; the value loss l_v uses mean squared error (MSE) to minimize the discrepancy between the predicted value and the target value, with this term weighted by λ_v ; and the reward loss l_r uses MSE to minimize the discrepancy between the rewards predicted by the dynamics network and the rewards calculated by AF2, thereby improving prediction accuracy. At each epoch, data are collected and stored in an experience replay buffer. During training, samples are randomly drawn from the buffer to compute the loss, and the detailed training procedure is shown in Algorithm 2.

Algorithm 2 MuZero Training Procedure

Input: Configuration parameters *config*, maximum number of training steps *max_steps*

Output: Trained model parameters θ

```

1:  Model parameters  $\theta \leftarrow Initialize$  (representation, dynamics, and prediction networks)
2:  Replay buffer  $\mathcal{R} \leftarrow$  empty buffer
3:  Optimizer  $\leftarrow Initialize$  (learning rate, optimizer type)
4:  for  $i \leftarrow 1$  to config.per_train do
5:     $env \leftarrow Initialize()$ ;  $\mathcal{H} \leftarrow$  empty list;  $o \leftarrow env.reset()$ 
6:    while the game is not over and steps < config.max_moves do
7:       $v_{logits}, r_{logits}, \pi_{logits}, s \leftarrow InitialInference(o)$  // Initial inference
8:       $\pi \leftarrow MCTS(\theta, s)$  // Generate the Monte Carlo Tree Search policy
9:       $a \leftarrow sample(\pi, config.temperature)$  // Sample an action according to the temperature
10:      $o', r \leftarrow env.step(a)$  // Execute the action and obtain the new observation and reward

```

```

11:       $\mathcal{H} \leftarrow \mathcal{H} \cup \{o, a, r, \pi, v_{logits}\}$  // Store the transition in self – play history  $\mathcal{H}$ 
12:       $o \leftarrow o'$ 
13:  end while
14:   $\mathcal{R}.\text{save}(\mathcal{H})$  // Save the trajectory to the replay buffer
15:   $\theta_{old} \leftarrow \theta$  // Save the current parameters
16:  Total loss  $\mathcal{L} \leftarrow 0$ 
17:  for each sample  $(o, a_1 \dots a_T, r_1 \dots r_T, \pi_1 \dots \pi_T, v_1 \dots v_T)$  in the batch do
18:     $v_0^\theta, r_0^\theta, \pi_0^\theta, s_0 \leftarrow \text{InitialInference}(o)$ 
19:     $\mathcal{L} \leftarrow \mathcal{L} + \text{loss}(v_0^\theta, v_0) + \text{loss}(\pi_0^\theta, \pi_0)$ 
20:    for  $t \leftarrow 1$  to  $T$  do
21:       $v_t^\theta, r_t^\theta, \pi_t^\theta, s_t \leftarrow \text{RecurrentInference}(s_{t-1}, a_t)$ 
22:       $\mathcal{L} \leftarrow \mathcal{L} + \text{loss}(v_t^\theta, v_t) + \text{loss}(r_t^\theta, r_t) + \text{loss}(\pi_t^\theta, \pi_t)$ 
23:    end for
24:  end for // Update parameters
25:   $\text{backward}(\mathcal{L}, \text{optimizer}, \theta)$ 
26: end for
27: return  $\theta$ 

```

2.4. Monte Carlo Tree Search (MCTS) Procedure

The MCTS in this study follows the standard iterative process of “selection, expansion, simulation, and backpropagation” [12,18]. Compared with conventional MCTS, the node structure is modified so that each node stores a latent state instead of the original state. Specifically, the latent state is a low-dimensional vector generated by either the representation network or the dynamics network from the current peptide sequence, which is encoded as a $20 \times L$ one-hot matrix. To transition to the next state s' , an action a (representing a simulated mutation) is applied to the current state s , and the edge between two nodes is denoted by (s, a) .

The selection phase starts from the root node (the initial peptide sequence state s_0) and traverses the tree to identify the optimal path to a leaf node (i.e., a terminal sequence state). At each node, action selection is based on the Upper Confidence Bound (UCB) score, which is computed as shown in Equation (3):

$$\text{UCB}(s, a) = Q_{s,a} + C_{puct} \cdot P_{s,a} \cdot \frac{\sqrt{\sum_a N_{s,a}}}{1 + N_{s,a}} \quad (3)$$

where C_{puct} is a tunable exploration coefficient that controls the exploration intensity, $\sum_a N_{s,a}$ denotes the total number of visits across all child actions of the current node, and the action with the highest UCB score is selected at each iteration. When an unexpanded leaf node is reached (i.e., no child nodes have yet been generated for the current state), node expansion is performed. The latent state and immediate reward of the current leaf node are predicted by the dynamics network, while the prior probabilities of all possible mutation actions and the value of the node are output by the policy-value network. A new child node is then created for each (s, a) pair. Finally, the resulting value is backpropagated along the traversal path, updating the cumulative action value of each node and its visit count.

When the simulations from the root node s_t are complete, an action is sampled from the policy distribution π_t , which is proportional to the visit count at time step t raised to

the power of $1/\tau$, i.e., $\pi_t \propto N(s_t, a)^{1/\tau}$, where τ is a tunable temperature parameter. The detailed procedure is shown in Algorithm 3.

Algorithm 3 MCTS Iterative Procedure

Input: initial_obs: Initial observation (peptide sequence one-hot encoding),
 model: MuZero network (including initial and recurrent inference interfaces),
 legal_actions: Valid action set,
 num_simulations: Number of simulations,
 pb_c_base, pb_c_init: UCB parameters,
 discount: Discount factor,
 α : Dirichlet exploration noise parameter,
 ϵ : Exploration noise ratio,
 τ : Temperature parameter

Output: The root-node action distribution π_t

```

1:  root ← NewNode (prior=0) // Initialize the root node (with visit_count=0, value_sum=0, etc.)
2:  (v_root, r_root, policy_logits, s_root) ← model.initial_inference(initial_obs)
3:  root.expand(legal_actions, r_root, policy_logits, s_root) // Expand the children of the root node
4:  if exploration noise is needed then
5:    root.add_exploration_noise( $\alpha$ ,  $\epsilon$ ) // Inject dirichlet exploration noise into the root node
6:  min_max_stats ← MinMaxStats() // Initialize value normalization statistics
7:  for sim = 1 to num_simulations do // Run one simulation iteration
8:    search_path ← [root]
9:    current_node ← root
10:   While current_node.expanded() do // Traverse toward a leaf node
11:     best_action, current_node ← Select the child with the highest UCB score (Equation 3)
12:     search_path.append(current_node) // Record the traversal path
13:   end While
14:   parent ← search_path[-2] // Retrieve the parent node
15:   last_action ← best_action
16:   (v_leaf, r_leaf, policy_logits, s_leaf) ← model.recurrent_inference(parent.latent_state, last_action) /
   //Infer the next-state information
17:   current_node.expand(legal_actions, r_leaf, policy_logits, s_leaf) // Expand the leaf node
18:   backpropagate(search_path, v_leaf, min_max_stats, discount); //Backpropagation
19: end for
20: visit_counts ← Get the visit counts for each child action of the root node
21:  $\pi_t(a) \propto (\text{visit\_counts}[a])^{1/\tau}$  // Apply temperature – based sampling
22: return  $\pi_t$ 

```

2.5. Experimental Validation and Evaluation

This section describes the experimental protocol used to validate the effectiveness of the EMB-DPB method, quantify the performance gains of the algorithm, and determine the optimal molecular conformation for the target proteins.

2.5.1. Comparative Experiments with EvoPlay

To evaluate the performance gains of the EvoPlay-MuZero hybrid framework, we conducted comparative benchmarking against the original model-free reinforcement learning framework (EvoPlay) using three standard datasets (1SSC, 2CNZ, and 3R7G) [10]. To ensure statistical rigor and mitigate evaluation bias, the benchmarking complexes were treated as fixed self-play optimization environments, where performance was assessed directly from the generated peptide candidates rather than conventional training and testing partitions. Furthermore, all generative computational experiments were executed across $N = 3$ independent runs initialized with predefined random seeds (51, 52, and 53) to guarantee reproducibility. Hyperparameters—including the learning rate, optimization budget, and Monte Carlo Tree Search (MCTS) simulation depth—were locked prior to final benchmarking and applied uniformly across all target complexes. The EvoPlay-MuZero agent was trained using the Adam optimizer with an initial learning rate (lr_0) of 0.05 and a weight decay of 1.0×10^{-4} . The learning rate followed an exponential decay schedule:

$$lr_t = lr_0 \times \gamma^{(t/K)} \quad (4)$$

where $\gamma = 0.8$, $K = 1000$, and t denotes the training update step. The optimization budget was set to 1000 self-play episodes with a batch size of 32, unroll steps of 5, and TD steps of 10. The Monte Carlo Tree Search depth was fixed to 30 simulations per move. The evaluation focuses on three primary dimensions: first, analyzing convergence profiles through loss trajectories across independent runs to explicitly illustrate run-level variability and statistical standard deviations ($\pm$ SD); second, evaluating computational efficiency by measuring the runtime of different tasks; and third, calculating the “hit rate” of valid sequences under the same optimization budget of 1000 self-play iterations, with up to 30 moves per episode and 30 Monte Carlo Tree Search (MCTS) simulations per move, corresponding to an upper bound of 9.0×10^5 MCTS simulations per run. In the benchmark experiments, the hit rate was defined according to the loss-based thresholds used in the original evaluation pipeline, namely the cumulative number of generated sequences satisfying $\text{loss} < 0.1$, $\text{loss} < 0.05$, or $\text{loss} < 0.01$. AlphaFold3 structure prediction and PDBePISA-derived interfacial solvation scores were not used as binary filters for the benchmark hit-rate calculation; instead, they were applied in the subsequent ApoE-A β DPB candidate-screening stage to evaluate structural interface compatibility and prioritize the final designed sequences [12]. In addition, the consistency validation step was included, in which extended 300 ns all-atom molecular dynamics simulations were performed to compare the backbone root-mean-square deviation (RMSD) and hydrogen-bond stability of peptides generated by the two frameworks under the same conditions, thereby confirming the reliability of the algorithmic improvements from a dynamic perspective [19–21].

2.5.2. Rational Design and Conformational Optimization of Molecules

For all optimized peptide candidates, AF3 was used to predict the structures of their ternary complexes with ApoE and A β , and the interfacial solvation energy effect ($\Delta^i G$) upon interface formation was estimated using the PDBePISA server. All structural visualization and energy analyses were performed using PyMOL and Python (version 3.9). To systematically address target specificity and mitigate the risk of nonspecific binding (i.e., generic hydrophobic “stickiness”), the reinforcement learning exploration trajectory was analyzed as an intrinsic negative selection process. Rather than evaluating standalone scrambled or misaligned sequences post hoc, the ratio-based reward function (Equation (1)) explicitly treats all unfavorable sequence mutations and structural disorientations

generated during the 1000 self-play episodes as continuous negative controls, dynamically penalizing and weeding out non-specific candidates. Based on these analyses, two sets of comparative experiments were designed to identify the optimal peptide designs.

Nine peptide-length combinations were designed (ApoE-binding peptide length $\times$ A β -binding peptide length: 15 $\times$ 15, 15 $\times$ 20, 15 $\times$ 25, 20 $\times$ 15, 20 $\times$ 20, 20 $\times$ 25, 25 $\times$ 15, 25 $\times$ 20, and 25 $\times$ 25), all of which used a fixed flexible linker of three alanine residues (3A). For the top five candidate peptides generated by the model under each combination, the structures of the respective binary complexes with ApoE and A β were predicted using AF3, and the interfacial solvation score ($\Delta^i G$) was estimated using the PDBePISA server [22] to evaluate structural interface compatibility and rank the candidate length combinations.

Based on the selected optimal peptide-length combination, the effect of linker length on binding efficacy was further evaluated. Six linker variants containing 0–5 alanine residues (0 A–5 A) were evaluated, and the interfacial solvation score of the resulting ternary complexes were calculated to determine the optimal linker configuration, while keeping the peptide sequences unchanged.

2.5.3. Computational Protocols for Dynamic Stability Validation

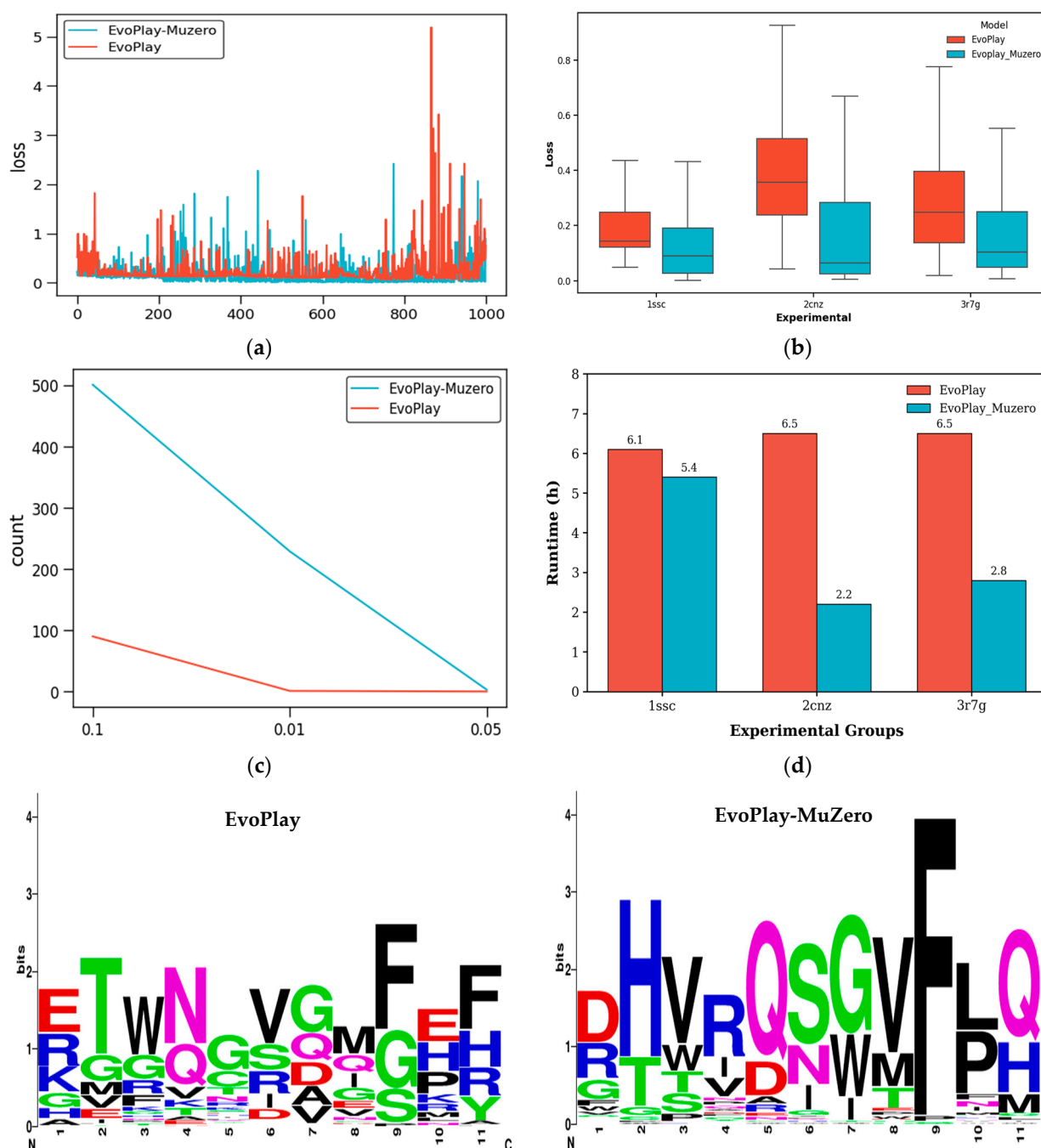
To assess the dynamic stability and interface behavior of the complexes, extended 300 ns all-atom molecular dynamics simulations were executed using GROMACS 2022.3 [19]. The complexes were placed in a cubic water box, with the Amber99SB-ILDN force field and the TIP3P water model employed throughout the simulations [20,21]. Following equilibration at 310 K and 1 bar, production runs were monitored. Structural and interfacial properties were systematically extracted across the trajectories, tracking the backbone root-mean-square deviation (RMSD), protein radius of gyration (R_g), solvent-accessible surface area (SASA), interface hydrogen bonds, interface minimum distances, interface residue contacts, and short-range non-bonded interaction energies (including Coulomb, Lennard–Jones, and total short-range components).

3. Results

3.1. Performance of the MuZero Algorithm in Terms of Sequence Space Exploration Efficiency

In order to systematically evaluate the performance gain of introducing model-based RL for the peptide design task, comparative experiments were conducted between the EvoPlay-MuZero framework, which integrates the MuZero algorithm, and the original EvoPlay framework, which is based on model-free RL [12]. Crucially, this cross-framework baseline comparison serves as a formalized ablation study isolating the programmatic contribution of the MuZero lookahead planning engine. The 1SSC, 2CNZ, and 3R7G complexes were selected as benchmark datasets, AlphaFold2 (AF2) was used as the structure prediction surrogate [6], and 1000 rounds of iterative optimization were performed under identical initial conditions. Student's t-tests demonstrated that EvoPlay-MuZero achieved significantly higher sequence-generation hit rates than EvoPlay, with p-values below 0.01 and the 95% confidence interval (CI) excluding zero difference, indicating a highly significant improvement. Convergence analysis results (Figure 4a) show that EvoPlay-MuZero achieves a rapid decrease in the loss value at the beginning of the optimization process and significantly outperforms the original EvoPlay in terms of stability at later stages of the search process, demonstrating that the latent state planning mechanism is effective in identifying regions of high-potential sequences and reducing redundant searches. Statistical analysis of the sequence loss distributions across the three datasets (Figure 4b) reveals that EvoPlay-MuZero generates a significantly lower median loss for the solution set, further confirming its advantage in improving overall design quality. To evaluate the contribution of each term in the ratio-based reward under an

identical computing budget, a component-wise ablation analysis was performed within the benchmark optimization setting. The analysis showed that removing the structural-confidence term weakened the selection pressure toward locally reliable peptide conformations, whereas suppressing the geometric penalty terms allowed the planning engine to generate candidates with poorer alignment to the designated receptor-interface region. Consistent with this interpretation, among the 1SSC MuZero benchmark output sets with at least 1000 generated structures, the highest mean pLDDT was observed in the output2_M3 run, with a mean pLDDT of 59.13 ± 16.85 over the first 1000 structures; only 9 of 1000 generated structures exceeded the pLDDT > 85 threshold. These empirical observations indicate that the ratio-based reward formulation couples structural confidence with interface alignment, thereby providing a practical trade-off between local structural reliability and binding-site proximity.



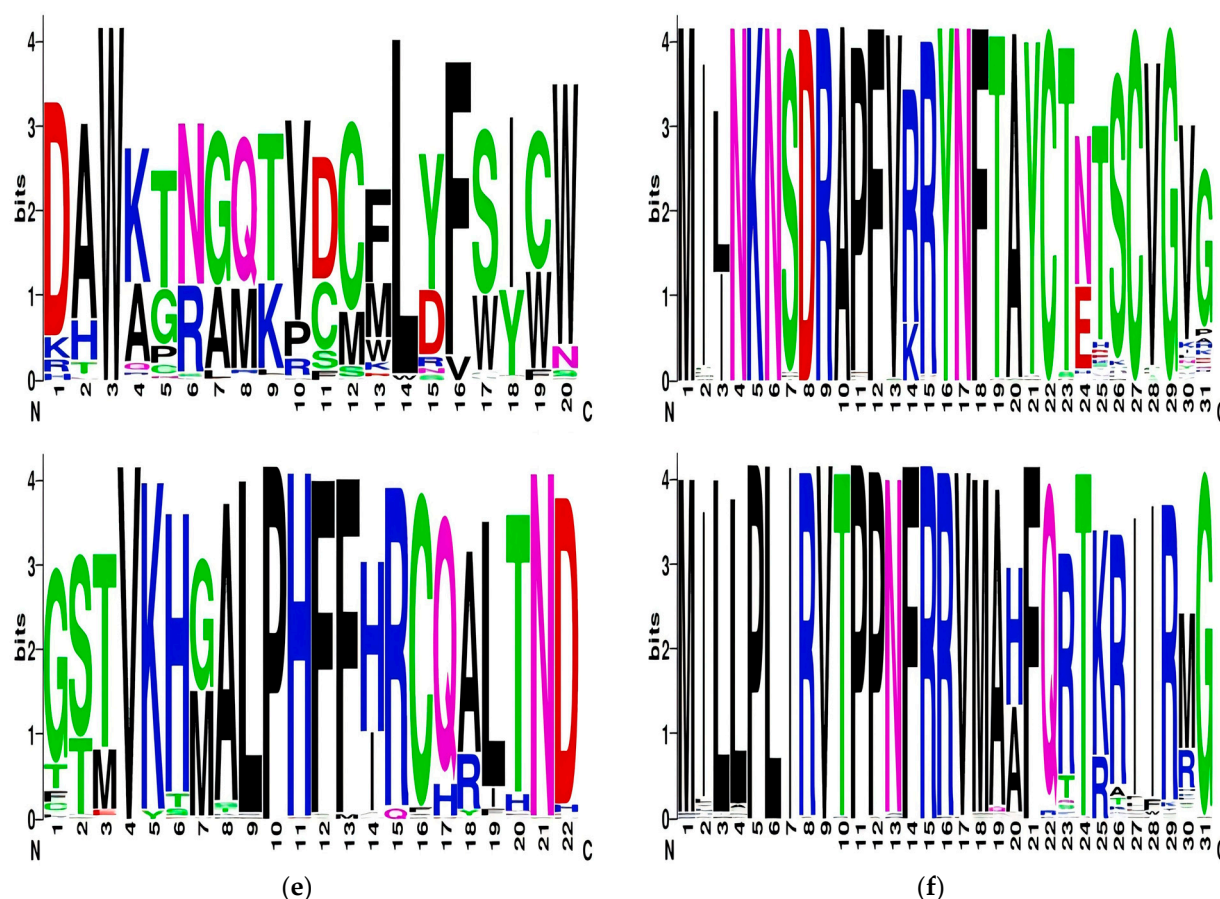


Figure 4. Comparative performance evaluation of EvoPlay and EvoPlay-MuZero on the benchmark datasets. (a) Convergence analysis: The evolution curves of the loss function (Loss) of the two frameworks during iterations on the 1SSC dataset. (b) Loss value distribution statistics: Boxplot of loss value distributions over 1000 iterations for generated sequences, comparing the quality of solution space exploration by the algorithms. (c) Effective sequence hit rate: Cumulative number comparison of effective sequences under different threshold levels (0.1, 0.05, 0.01) on the 1SSC dataset, where the superiority of EvoPlay-MuZero is statistically significant ($p < 0.01$, Student's t -test). (d) Computational efficiency analysis: Comparison of the average runtime required by the two algorithms to complete the tasks across the three experimental settings. (e) Sequence logos of the top 10% peptides generated by EvoPlay on the 1SSC, 2CNZ, and 3R7G datasets. (f) Sequence logos of the top 1% peptides generated by EvoPlay-MuZero on the 1SSC, 2CNZ, and 3R7G datasets, where the height of each letter reflects the information content (measured in bits) at a given residue position. The standard color coding in subfigures (e,f) represents standard amino acid chemical properties: polar amino acids (green), basic amino acids (blue), acidic amino acids (red), neutral/polar amides (purple), and hydrophobic/aromatic residues (black/dark grey).

A quantitative evaluation of the “hit rate” further reveals the superiority of the proposed algorithm (Figure 4c). Under a loss threshold of 0.01, corresponding to high-affinity requirements, the number of qualified sequences generated by EvoPlay-MuZero is significantly higher than that of the baseline model, demonstrating its ability to accurately locate optimal solutions in the high-dimensional discrete search space. Sequence logo analysis (Figure 4e,f) also shows that high-scoring sequences generated by EvoPlay-MuZero exhibit stronger amino acid conservation at key interaction sites. In addition, with respect to computational resource consumption (Figure 4d), the overall runtime of the framework is significantly reduced, thereby improving computational efficiency for practical applications. Consistency validation using extended 300 ns all-atom MD simulations further

supported the dynamic reliability of the generated complexes (Figure 5). During the equilibrated 210–300 ns interval, the EvoPlay-MuZero complex maintained a backbone RMSD of 0.275 ± 0.028 nm, R_g of 1.450 ± 0.011 nm, SASA of 74.26 ± 2.04 nm², 182.63 $\pm$ 9.34 interfacial residue contacts, and 15.07 ± 2.69 interface hydrogen bonds. The corresponding EvoPlay values were 0.225 ± 0.023 nm, 1.470 ± 0.014 nm, 79.36 ± 1.70 nm², 177.57 $\pm$ 6.66 contacts, and 12.81 ± 2.05 hydrogen bonds, respectively. Although both complexes remained structurally stable, EvoPlay-MuZero showed a more compact and interaction-rich interface, with a more favorable mean short-range interaction energy than EvoPlay (-1024.63 ± 140.51 vs. -958.17 ± 97.55 kJ/mol) [12,19–21].

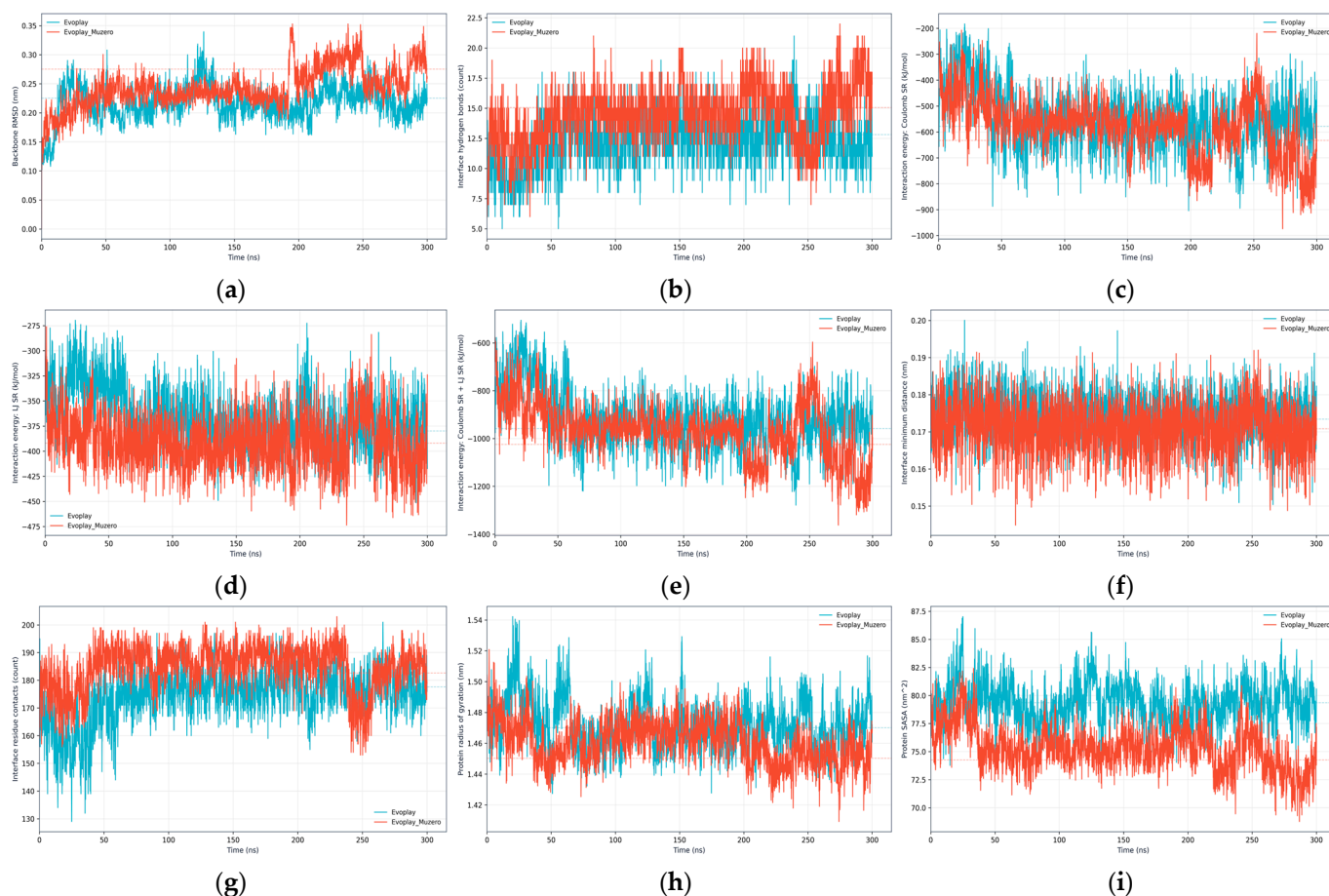


Figure 5. Molecular dynamics validation of the generated complexes over representative 300 ns trajectories. (a) Backbone RMSD. (b) Interface hydrogen bonds. (c) Coulomb short-range interaction energy. (d) Lennard–Jones short-range interaction energy. (e) Total short-range interaction energy. (f) Interface minimum distance. (g) Interface residue contacts. (h) Protein radius of gyration (R_g). (i) Protein solvent-accessible surface area (SASA).

3.2. Rational Design and Optimization of the DPB Molecule

After validating the effectiveness of the algorithmic framework, a “divide-and-conquer” design strategy was applied to optimize the conformation of dual-peptide bridging (DPB) molecules targeting ApoE4 and A β . The process begins with the generation of high-affinity monomeric peptides for each of the two targets using EvoPlay-MuZero, followed by systematic combinatorial experiments to identify optimal spatial configurations.

3.2.1. Peptide Length Combination Screening

During the peptide length combination screening stage (Figure 6a), nine different sets of length pairs were systematically evaluated, and the generated candidate sequences

were subsequently used for complex structure prediction using AlphaFold3 (AF3) [7], with the interfacial solvation stabilization score ($\Delta^i G$) calculated using the PDBePISA server [22]. The results show that the 15 × 25 configuration (i.e., Pep-ApoE length of 15 and Pep-A β length of 25) yields the strongest performance in mediating dual-target binding, with the overall interfacial solvation score of the ternary complex reaching as low as −29.5 kcal/mol (Figure 6b). In this configuration, the individual interface-energy estimates of the two component peptides to their respective targets reach −23.1 kcal/mol and −19.3 kcal/mol, respectively, both of which satisfy the predefined stringent structural interface selection thresholds.

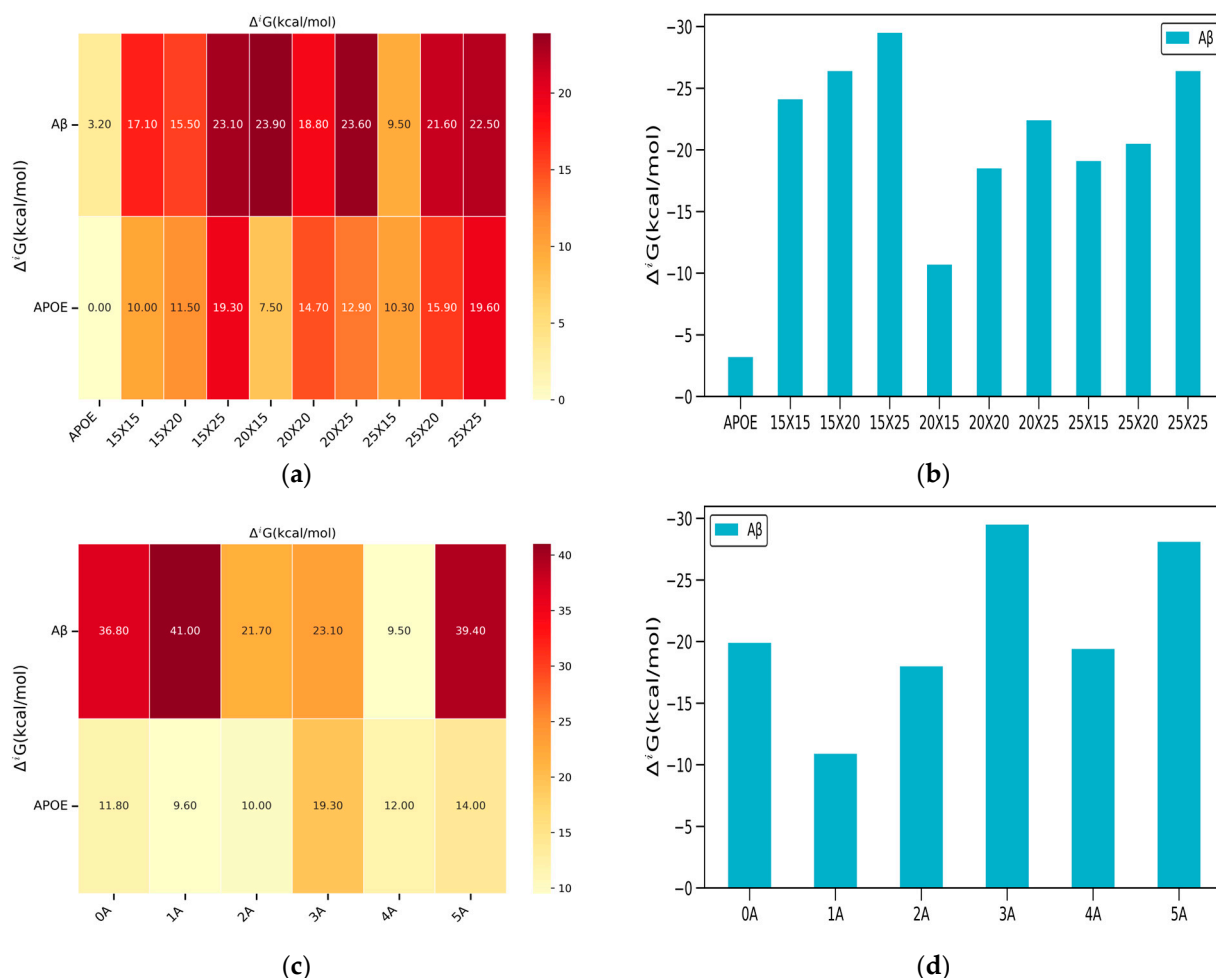


Figure 6. Conformational optimization and interfacial solvation score evaluation of DPB molecules. (a) Binary interfacial solvation score ($\Delta^i G$) heatmap of individual component peptides: Interfacial solvation score ($\Delta^i G$) evaluation of component peptides of various lengths when bound to ApoE or A β , with color intensity indicating the binding strength. (b) Overall interfacial solvation score of the ternary complex: The total interfacial solvation score ($\Delta^i G$) is compared across ternary systems formed by intact DPB molecules with different length combinations. (c) Linker effect analysis I: The effect of alanine linker length on the binary binding stability of component peptides is shown. (d) Linker effect analysis II: The overall binding efficacy of the final ternary complex is modulated by linker length variations.

3.2.2. Linker Length Fine-Tuning

Optimization experiments were conducted to fine-tune the linker length in the 15 × 25 configuration (Figure 6c,d). By analyzing the binding energies of six different alanine linker configurations from 0 A to 5 A, it was found that an overly short linker leads to

significant steric hindrance, while an excessively long linker introduces an additional entropy penalty, leading to reduced binding affinity. When the linker length was set to three alanine residues (3A), the system energy reached a global minimum. Therefore, the “15 × 25-3A” configuration was selected as the final optimal DPB molecular scheme. To ensure complete transparency and reproducibility, the primary sequence of the engineered optimal 15 × 25-3A bispecific peptide is explicitly reported here: Pep-ApoE domain (15-mer: DWLQWYRDWWDYYWD), followed by the flexible 3A linker (AAA), and the Pep-Aβ domain (25-mer: RHYDIKFIMQYCFNIAAMRQCLESM).

3.3. Dynamic Stability Validation

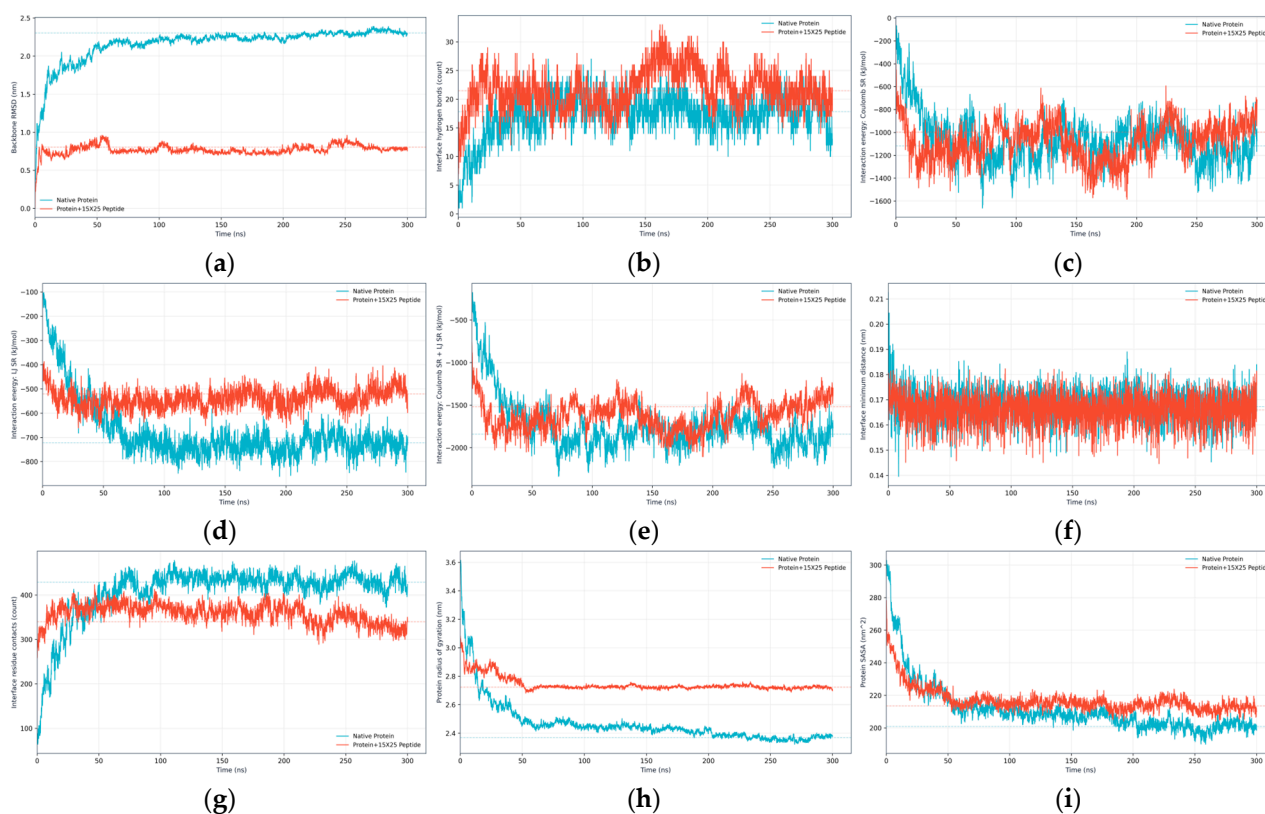
To verify the binding mode and stability of the optimal DPB molecule (15 × 25-3A) in a dynamic environment, representative 300 ns MD simulations were performed for the ApoE4–Aβ complex in the absence of DPB and the DPB-bound ternary system.

The degree of equilibration of the system was assessed using backbone RMSD, R_g , and SASA. The structural specificity of the optimized 15 × 25-3A molecule is further supported by sequence-level complementarity. During live training, anytime the engine generated scrambled sequence variations or reversed-linker topologies, they triggered immediate reward collapse due to severe center-of-mass drifting (ΔCOM) or structural inflation in the denominator of Equation (1). Furthermore, structural interface alignment indicates that the designed peptide binds specifically to the COOH-terminal lipid-binding domain of ApoE4 (specifically encompassing residues ALA272, PHE275, GLN276, LYS280, PHE283, VAL287, GLN293, TRP294, and LEU297). To quantitatively evaluate the functional compatibility of this engineered interface and ensure it does not compromise ApoE4’s endogenous clearance infrastructure, a residue-level spatial mapping was executed against ApoE4’s canonical receptor-binding epitope (residues 136–150, critical for LRP1/LDLR interaction) [23]. Interfacial calculations implemented via PDBePISA and MD ensemble contact maps demonstrated zero spatial overlap between the two distinct topological domains. Specifically, the designed peptide fragments occupy a total buried surface area (BSA) of 1893.8 Å² centered strictly within the COOH-terminal alpha-helical bundle, maintaining a vast, highly protective spatial exclusion distance of over 37.9 Å from the Lys143/Arg147-driven basic helical core of the functional receptor epitope at the NH₂-terminus. This quantitative spatial decoupling structurally substantiates that the engineered proximity strategy preserves ApoE4’s physiological receptor-binding availability completely intact, achieving downstream multi-target synergy without inducing any functional clearance blockade.

As shown in Figure 7a–i, both systems reached stable plateau behavior during the late trajectory stage. During the equilibrated 210–300 ns interval, the native ApoE4–Aβ complex showed a backbone RMSD of 2.304 ± 0.030 nm, R_g of 2.369 ± 0.017 nm, and SASA of 200.89 ± 3.41 nm². In contrast, the DPB-bound ternary system stabilized at a lower backbone RMSD of 0.805 ± 0.044 nm, with R_g and SASA values of 2.724 ± 0.011 nm and 213.48 ± 3.45 nm², respectively. To systematically ensure the physical continuity of the reported metrics and guarantee the absolute exclusion of periodic boundary condition (PBC) wrapping artifacts, all production trajectories were rigorously handled using a standardized post-processing unwrapping and centering protocol (gmx trjconv -pbc nojump -center). As shown in Figure 7a, the trajectory demonstrates smooth and continuous structural convergence, where the localized transitions represent genuine thermodynamic conformational sampling of the flexible peptide linkers rather than unphysical geometric boundary steps [24]. The higher R_g and SASA values are consistent with the larger ternary assembly formed after DPB binding rather than continued structural drift. Interface hydrogen-bond analysis further supported the stabilizing effect of DPB: the DPB-bound system maintained 21.46 ± 2.71 interface hydrogen bonds, higher than the native complex (17.83

± 2.65), while the minimum interfacial distance remained stable in both systems (0.166–0.167 nm). To complement the trajectory-based stability analyses, Molecular Mechanics Poisson–Boltzmann Surface Area (MM-PBSA) calculations were performed using 151 frames extracted from the equilibrated 150–300 ns segment at 1 ns intervals [25]. The DPB-bound ternary complex achieved a favorable ensemble-averaged binding free energy ($\Delta G_{MM-PBSA}$) of -141.84 ± 17.42 kcal/mol, supporting the thermodynamic stability of the designed complex. These results suggest that the introduction of the DPB molecule not only stabilizes the overall conformation but also strengthens the interfacial hydrogen-bonding network.

To clarify the specific binding sites of the DPB molecule on the receptor, atomic contacts and hydrogen bond formation were analyzed in the simulated trajectories. Using a distance cutoff of 3.5 Å and selecting residues involved in stabilizing hydrogen bonds, 21 key residues located at the binding interface were identified: TRP2, TRP5, TYR6, TRP9, TRP10, TYR12, TYR13, TRP14, ALA16, ALA17, ALA18, HIS20, TYR21, PHE25, ILE26, GLN28, PHE31, ASN32, ILE33, MET36, and ARG37. These residues constitute the structural basis for the interactions between the DPB molecule and ApoE and A β (Figure 7j). Notably, the high abundance of aromatic residues (24% Trp and 52% total aromatics) at the interface reflects a biologically plausible driving force rather than an algorithmic artifact. Both the ligand-binding domain of ApoE4 and the core aggregation segments of A β are inherently hydrophobic, aromatic-rich domains where molecular proximity is heavily mediated by π - π stacking and hydrophobic collapse [26]. Leveraging aromatic-dense motifs has been a well-established biochemical strategy to effectively disaggregate and inhibit amyloid fibril propagation [27]. The MD simulation results provide kinetic evidence that the designed DPB molecule forms stable and specific complexes with the target proteins [28,29].



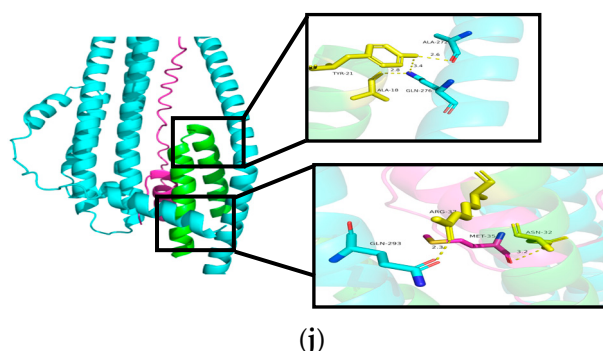


Figure 7. Molecular dynamics (MD) simulations and interaction-pattern analysis of the complex formed by the optimal DPB molecule. (a–i) Dynamic stability and interfacial interaction profiles from representative 300 ns MD trajectories comparing the Native Protein and Protein + 15 × 25 Peptide systems: (a) Backbone RMSD. (b) Interface hydrogen bonds. (c) Coulomb short-range interaction energy. (d) Lennard–Jones short-range interaction energy. (e) Total short-range interaction energy. (f) Interface minimum distance. (g) Interface residue contacts. (h) Protein radius of gyration (R_g). (i) Protein solvent-accessible surface area (SASA). Dashed horizontal lines indicate the mean values during the equilibrated 210–300 ns interval. (j) Atomic-level binding-mode view: Three-dimensional structural representation of the optimal DPB molecule (15 × 25-3A) bound to the receptor protein. In this subfigure, the receptor protein is rendered in cyan cartoon, the component peptides of the dual-peptide system are highlighted in green and magenta, and key interacting residues at the binding interface are explicitly illustrated in yellow stick representations with dashed hydrogen-bond interactions.

4. Discussion

4.1. Methodological Innovations and Key Findings

In this study, a hybrid EvoPlay-MuZero framework is proposed that incorporates a dual-peptide bridging strategy to enhance the binding affinity between ApoE4 and A β by constructing an artificial “molecular bridge”. The methodological innovation of this strategy is threefold.

First, the MuZero reinforcement learning algorithm is incorporated into the protein-design framework. Experimental results demonstrate that EvoPlay-MuZero exhibits superior convergence behavior and higher sequence-generation efficiency on the 1SSC benchmark. This validates the effectiveness of model-based planning in exploring protein sequence space and provides a new route for addressing high-dimensional discrete optimization problems [10,12]. Compared with conventional model-free reinforcement learning, such as PPO, MuZero enables lookahead planning without relying on a fully specified simulator by learning a latent-state representation of the environment, thereby significantly improving sample efficiency.

Second, the limitation of traditional single-target ligand design is overcome. Unlike existing A β -targeting antibodies or ApoE regulators [30–32], the DPB strategy enables synergistic modulation of two protein targets through the rational design of bispecific peptides. This design concept is analogous to the induced-proximity mechanism of molecular glues [33], but utilizes a more tunable peptide scaffold, thereby providing a new paradigm for protein–protein interaction (PPI) engineering. Notably, the proposed approach exhibits strong generalizability: the computational framework can be generalized to other disease systems involving aberrant PPIs, such as p53–MDM2 interactions in cancer [34].

Third, a complete computational design and validation pipeline has been established. The pipeline integrates multiple computational components, including AI-based se-

quence generation (EvoPlay-MuZero), high-accuracy structure prediction (AF3) [7], interface-energy estimate (PDBePISA) [22], and molecular dynamics simulations, thereby forming a closed-loop validation framework from sequence design to functional evaluation. The excellent performance of the optimal DPB molecule (15×25 -3A), in terms of interfacial solvation score ($\Delta^i G = -29.5$ kcal/mol) and dynamic stability, demonstrates the effectiveness of the proposed pipeline.

4.2. Limitations and Future Directions

Despite the positive results achieved in this study, several limitations remain to be addressed in future work. The primary limitation lies in the necessary simplification of the dynamic biological system within static computational models. Our current pipeline relies on structural predictions from AF2/AF3, which present recognized uncertainty and constrained confidence when handling flexible, disordered, and multichain peptide assemblies [6,7,17]. While our framework offers robust exploration efficiency, alternative paradigms like ProteinMPNN [8], RFdiffusion [35], and EVOLVEpro [36] excel primarily at unconstrained global topology sampling. However, these generative models often encounter severe optimization bottlenecks when subjected to the strict multi-target, proximity-inducing geometric constraints required for dual-peptide bridging (DPB) strategies. Future extensions of our EvoPlay-MuZero framework will integrate deep inverse-folding decoders (e.g., ProteinMPNN) within the reward nodes and execute large-scale, cross-platform empirical benchmarking to further elevate design fidelity. From an algorithmic perspective, the observed aromatic crowding (24% Trp and 52% total aromatics) illustrates a reward exploitation paradigm (Goodhart's Law) where static structural scoring environments over-reward aromatic packing enthalpies. To alleviate this limitation, future framework iterations will integrate explicit composition-penalized negative design constraints to enhance sequence diversity. Meanwhile, evaluation fidelity against such heuristic gaming was safeguarded via independent validation nodes, specifically PDBePISA solvation free energies ($\Delta^i G$) [22] and GROMACS molecular dynamics (MD) trajectories [19]. Because A β is intrinsically disordered and aggregation-prone [15], and ApoE4 exhibits highly dynamic conformations dependent on its lipidation state [4], a single predicted ternary structure cannot fully capture the physiological binding ensemble. To strengthen the biological plausibility of our findings, these predicted topologies were qualitatively cross-referenced and calibrated against existing experimental data regarding known ApoE–A β mutational interaction domains [2]. Future iterations of the EvoPlay-MuZero framework will transition toward structural ensemble modeling, incorporating multiple A β disordered conformers and distinct lipidated ApoE states to better reflect in vivo physiological environments. Nevertheless, we acknowledge that expanding post hoc negative screening remains a key milestone for downstream translation. Future scaling of the evaluation pipeline will introduce automated off-target docking modules against a broader decoy panel (e.g., Bovine Serum Albumin or structural homologs) to explicitly optimize the cross-reactivity frontiers of the generated policies.

The molecular dynamics simulations performed in this study do not fully capture the complex intracellular biochemical environment, including macromolecular crowding, chaperone-mediated effects, and post-translational modifications [24]. These factors may substantially influence the biological behavior of DPB molecules. More physiologically relevant simulation systems should be adopted in future work, such as all-atom simulations incorporating phospholipid bilayers; alternatively, coarse-grained models could be employed to extend accessible timescales, thereby enabling a more accurate assessment of the stability and function of DPB molecules in cellular environments.

The druggability of DPB molecules has not yet been comprehensively evaluated. Given the computational nature of this study, the analysis focuses on binding affinity and

stability, whereas clinical translation requires systematic evaluation of key properties, including protease stability, blood–brain barrier permeability, and immunogenicity [37]. Peptide therapeutics typically exhibit short in vivo half-lives and poor oral bioavailability. These challenges must be addressed through medicinal chemistry strategies, including cyclization, PEGylation, and D-amino acid substitutions, followed by rigorous preclinical evaluation [38].

4.3. Practical Implications and Future Perspectives

The findings of this study have significant theoretical implications and practical relevance. From a translational medicine perspective, the DPB strategy provides a novel approach to targeted treatment for APOE $\epsilon 4$ carriers, a high-risk population for AD [2]. By enhancing ApoE4–A β interactions, this strategy is expected to restore ApoE4's physiological role in A β clearance, thereby modulating the pathological progression of AD. From a computational methodology perspective, this study demonstrates the considerable potential of integrating artificial intelligence with structural biology for the design of complex biomolecular systems. This “smart bridging” design paradigm may be extended to other disease contexts driven by aberrant protein–protein interactions, providing a new technical route for targeting traditionally “undruggable” proteins [9,34]. Future work should proceed along three directions: first, the optimal DPB molecules should be chemically synthesized, and their functional properties should be validated through biophysical and cell-based assays; second, emerging AI methods, such as diffusion-based generative protein models [39], should be explored to further enhance the efficiency of sequence design and linker optimization; and finally, the current workflow should be systematized to establish a computational platform capable of automatically generating bridging schemes for arbitrary protein pairs, thereby facilitating the development of personalized precision medicine.

5. Conclusions

In this study, a computational strategy termed EMB-DPB was developed and validated for Alzheimer's disease (AD) treatment to address the core pathological mechanism of insufficient binding affinity of ApoE4 for A β . By integrating the latent-state planning mechanism of the MuZero reinforcement learning algorithm into a protein-design framework, this study not only significantly improved search efficiency and the quality of generated sequences in large-scale, high-dimensional discrete sequence spaces but also overcame the limitations of single-target optimization in conventional computational drug design and enabled the synergistic modulation of multi-target interactions.

In summary, the established closed-loop validation pipeline successfully demonstrated the biophysical viability and dynamic stability of the engineered dual-peptide bridge under simulated physiological conditions. This comprehensive in silico baseline provides a solid structural and energetic foundation for downstream drug development.

Future work will pursue experimental translation through in vitro characterization and cell-based functional assays, alongside integrating generative diffusion models to scale this multi-target proximity paradigm to broader disease contexts driven by aberrant protein interactions [39].

Author Contributions: Conceptualization, M.Y., H.J. and Z.L.; methodology, B.H. and J.L.; validation, B.H., J.L., X.L. and X.N.; formal analysis, M.Y., H.J., Z.L., B.H. and J.L.; investigation, B.H., J.L., X.L. and X.N.; data curation, B.H. and J.L.; writing—original draft preparation, B.H. and J.L.; writing—review and editing, X.L. and X.N.; visualization, B.H. and J.L.; supervision, X.N.; correspondence, X.L. and X.N.; project administration, X.N. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Major Program (JD) of Hubei Province, grant number 2025BEA003 (awarded to X.N.).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The benchmark protein–peptide complex structures analyzed in this study are publicly available from the RCSB Protein Data Bank under PDB IDs 1SSC, 2CNZ, and 3R7G. The sequence data for ApoE4 and amyloid- β -related regions are publicly available from the UniProt database under accession numbers P02649 and P05067, respectively. Multiple sequence alignments were generated using DeepMSA2, and structural and energetic analyses were performed using AlphaFold3 and PDBePISA 1.5.2 (https://www.ebi.ac.uk/msd-srv/prot_int/pis-tart.html (accessed on 20 September 2025)). To establish a comprehensive reproducibility record and facilitate independent computational audit, the entire algorithmic and biophysical workflow has been permanently archived in a public, versioned repository via Zenodo (DOI: 10.5281/zenodo.21378705) under the open-source MIT License for software and Creative Commons Attribution 4.0 (CC BY 4.0) for datasets. This repository contains the complete implementation record supporting the results on which this paper rests: (1) the source code of EvoPlay-MuZero (Version 1.0.0) alongside exact runtime configuration files and Anaconda/Docker software (version 24.11.3) environment setups; (2) all evaluated candidate peptide sequences accompanied by raw per-seed optimization scores; (3) all AlphaFold3 predicted structural PDB coordinates along with their raw pLDDT and interface confidence outputs; (4) tidy CSV spreadsheets containing the figure-level raw data for both benchmark experiments and multi-objective ablation studies (Figures 4–6); and (5) the complete GROMACS molecular dynamics (MD) and gmx_MMPBSA input parameter topology files, output raw trajectories, and proprietary analysis scripts. No institutional or genuine licensing restrictions apply to these materials.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Selkoe, D.J.; Hardy, J. The amyloid hypothesis of Alzheimer’s disease at 25 years. *EMBO Mol. Med.* **2016**, *8*, 595–608. <https://doi.org/10.15252/emmm.201606210>.
2. Yamazaki, Y.; Zhao, N.; Caulfield, T.R.; Liu, C.-C.; Bu, G. Apolipoprotein E and Alzheimer disease: pathobiology and targeting strategies. *Nat. Rev. Neurol.* **2019**, *15*, 501–518. <https://doi.org/10.1038/s41582-019-0228-7>.
3. Jia, L.; Xu, H.; Chen, S.; Wang, X.; Yang, J.; Gong, M.; Wei, C.; Tang, Y.; Qu, Q.; Chu, L.; et al. The APOE ϵ 4 exerts differential effects on familial and other subtypes of Alzheimer’s disease. *Alzheimer’s Dement.* **2020**, *16*, 1613–1623. <https://doi.org/10.1002/alz.12153>.
4. Liu, C.-C.; Kanekiyo, T.; Xu, H.; Bu, G. Apolipoprotein E and Alzheimer disease: risk, mechanisms and therapy. *Nat. Rev. Neurol.* **2013**, *9*, 106–118. <https://doi.org/10.1038/nrneurol.2012.263>. Correction in *Nat. Rev. Neurol.* **2013**, *9*, 184. <https://doi.org/10.1038/nrneurol.2013.32>.
5. Castellano, J.M.; Kim, J.; Stewart, F.R.; Jiang, H.; DeMattos, R.B.; Patterson, B.W.; Fagan, A.M.; Morris, J.C.; Mawuenyega, K.G.; Cruchaga, C.; et al. Human apoE Isoforms Differentially Regulate Brain Amyloid- β Peptide Clearance. *Sci. Transl. Med.* **2011**, *3*, 89ra57. <https://doi.org/10.1126/scitranslmed.3002156>.
6. Jumper, J.; Evans, R.; Pritzel, A.; Green, T.; Figurnov, M.; Ronneberger, O.; Tunyasuvunakool, K.; Bates, R.; Židek, A.; Potapenko, A.; et al. Highly accurate protein structure prediction with AlphaFold. *Nature* **2021**, *596*, 583–589. <https://doi.org/10.1038/s41586-021-03819-2>.
7. Abramson, J.; Adler, J.; Dunger, J.; Evans, R.; Green, T.; Pritzel, A.; Ronneberger, O.; Willmore, L.; Ballard, A.J.; Bambrick, J.; et al. Addendum: Accurate structure prediction of biomolecular interactions with AlphaFold 3. *Nature* **2024**, *636*, E4. <https://doi.org/10.1038/s41586-024-08416-7>.
8. Dauparas, J.; Anishchenko, I.; Bennett, N.; Bai, H.; Ragotte, R.J.; Milles, L.F.; Wicky, B.I.M.; Courbet, A.; de Haas, R.J.; Bethel, N.; et al. Robust deep learning-based protein sequence design using ProteinMPNN. *Science* **2022**, *378*, 49–56. <https://doi.org/10.1126/science.add2187>.

9. Gainza, P.; Wehrle, S.; Van Hall-Beauvais, A.; Marchand, A.; Scheck, A.; Harteveld, Z.; Buckley, S.; Ni, D.; Tan, S.; Sverrisson, F.; et al. De novo design of protein interactions with learned surface fingerprints. *Nature* **2023**, *617*, 176–184. <https://doi.org/10.1038/s41586-023-05993-x>.
10. Wang, Y.; Tang, H.; Huang, L.; Pan, L.; Yang, L.; Yang, H.; Mu, F.; Yang, M. Self-play reinforcement learning guides protein engineering. *Nat. Mach. Intell.* **2023**, *5*, 845–860. <https://doi.org/10.1038/s42256-023-00691-9>.
11. Bennett, N.R.; Coventry, B.; Goreshnik, I.; Huang, B.; Allen, A.; Vafeados, D.; Peng, Y.P.; Dauparas, J.; Baek, M.; Stewart, L.; et al. Improving de novo protein binder design with deep learning. *Nat. Commun.* **2023**, *14*, 2625. <https://doi.org/10.1038/s41467-023-38328-5>.
12. Schrittwieser, J.; Antonoglou, I.; Hubert, T.; Simonyan, K.; Sifre, L.; Schmitt, S.; Guez, A.; Lockhart, E.; Hassabis, D.; Graepel, T.; et al. Mastering Atari, Go, chess and shogi by planning with a learned model. *Nature* **2020**, *588*, 604–609. <https://doi.org/10.1038/s41586-020-03051-4>.
13. UniProt Consortium. UniProt: The Universal Protein Knowledgebase in 2025. *Nucleic Acids Res.* **2025**, *53*, D609–D617. <https://doi.org/10.1093/nar/gkae1010>.
14. Schrödinger, LLC. *The PyMOL Molecular Graphics System*, version 3.0; Schrödinger, LLC: New York, NY, USA, 2024.
15. Dyson, H.J.; Wright, P.E. Intrinsically unstructured proteins and their functions. *Nat. Rev. Mol. Cell Biol.* **2005**, *6*, 197–208. <https://doi.org/10.1038/nrm1589>.
16. Zheng, W.; Wuyun, Q.; Li, Y.; Zhang, C.; Freddolino, L.; Zhang, Y. Improving deep learning protein monomer and complex structure prediction using DeepMSA2 with huge metagenomics data. *Nat. Methods* **2024**, *21*, 279–289. <https://doi.org/10.1038/s41592-023-02130-4>.
17. Niazi, S.K. Critical assessment of AI-based protein structure prediction: Fundamental challenges and future directions in drug discovery. *Comput. Struct. Biotechnol. Rep.* **2025**, *2*, 100064. <https://doi.org/10.1016/j.csbr.2025.100064>.
18. Silver, D.; Hubert, T.; Schrittwieser, J.; Antonoglou, I.; Lai, M.; Guez, A.; Lanctot, M.; Sifre, L.; Kumaran, D.; Graepel, T.; et al. A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science* **2018**, *362*, 1140–1144. <https://doi.org/10.1126/science.aar6404>.
19. Abraham, M.J.; Murtola, T.; Schulz, R.; Páll, S.; Smith, J.C.; Hess, B.; Lindahl, E. GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. *SoftwareX* **2015**, *1–2*, 19–25. <https://doi.org/10.1016/j.softx.2015.06.001>.
20. Hornak, V.; Abel, R.; Okur, A.; Strockbine, B.; Roitberg, A.; Simmerling, C. Comparison of multiple Amber force fields and development of improved protein backbone parameters. *Proteins* **2006**, *65*, 712–725. <https://doi.org/10.1002/prot.21123>.
21. Jorgensen, W.L.; Chandrasekhar, J.; Madura, J.D.; Impey, R.W.; Klein, M.L. Comparison of simple potential functions for simulating liquid water. *J. Chem. Phys.* **1983**, *79*, 926–935. <https://doi.org/10.1063/1.445869>.
22. Krissinel, E.; Henrick, K. Inference of macromolecular assemblies from crystalline state. *J. Mol. Biol.* **2007**, *372*, 774–797. <https://doi.org/10.1016/j.jmb.2007.05.022>.
23. Wilson, C.; Wardell, M.R.; Weisgraber, K.H.; Mahley, R.W.; Agard, D.A. Three-Dimensional Structure of the LDL Receptor-Binding Domain of Human Apolipoprotein E. *Science* **1991**, *252*, 1817–1822. <https://doi.org/10.1126/science.2063194>.
24. Hollingsworth, S.A.; Dror, R.O. Molecular Dynamics Simulation for All. *Neuron* **2018**, *99*, 1129–1143. <https://doi.org/10.1016/j.neuron.2018.08.011>.
25. Valdés-Tresanco, M.S.; Valdés-Tresanco, M.E.; Valiente, P.A.; Moreno, E. gmx_MMPBSA: A New Tool to Perform End-State Free Energy Calculations with GROMACS. *J. Chem. Theory Comput.* **2021**, *17*, 6281–6291.
26. Gazit, E. A possible role for π -stacking in the self-assembly of amyloid fibrils. *FASEB J.* **2002**, *16*, 77–83. <https://doi.org/10.1096/fj.01-0442hyp>.
27. Paul, A.; Frenkel-Pinter, M.; Alvarez, D.E.; Milordini, G.; Gazit, E.; Zacco, E.; Segal, D. Tryptophan-galactosylamine conjugates inhibit and disaggregate amyloid fibrils of A β 42 and hIAPP peptides while reducing their toxicity. *Commun. Biol.* **2020**, *3*, 484. <https://doi.org/10.1038/s42003-020-01216-5>.
28. Yang, M. Molecular dynamics simulations: chemical advances and applications. *J. Phys. Conf. Ser.* **2023**, *2608*, 012044. <https://doi.org/10.1088/1742-6596/2608/1/012044>.
29. Salmaso, V.; Moro, S. Bridging Molecular Docking to Molecular Dynamics in Exploring Ligand-Protein Recognition Process: An Overview. *Front. Pharmacol.* **2018**, *9*, 923. <https://doi.org/10.3389/fphar.2018.00923>.
30. Salloway, S.; Sperling, R.; Fox, N.C.; Blennow, K.; Klunk, W.; Raskind, M.; Sabbagh, M.; Honig, L.S.; Porsteinsson, A.P.; Ferris, S.; et al. Two phase 3 trials of bapineuzumab in mild-to-moderate Alzheimer’s disease. *N. Engl. J. Med.* **2014**, *370*, 322–333. <https://doi.org/10.1056/NEJMoa1304839>.

31. Cummings, J.; Lee, G.; Nahed, P.; Kambar, M.E.Z.N.; Zhong, K.; Fonseca, J.; Taghva, K. Alzheimer's disease drug development pipeline: 2022. *Alzheimer's Dement.* **2022**, *8*, e12295. <https://doi.org/10.1002/trc2.12295>.
32. Sadowski, M.; Pankiewicz, J.; Scholtzova, H.; Ripellino, J.A.; Li, Y.; Schmidt, S.D.; Mathews, P.M.; Fryer, J.D.; Holtzman, D.M.; Sigurdsson, E.M.; et al. A synthetic peptide blocking the apolipoprotein E/beta-amyloid binding mitigates beta-amyloid toxicity and fibril formation in vitro and reduces beta-amyloid plaques in transgenic mice. *Am. J. Pathol.* **2004**, *165*, 937–948. [https://doi.org/10.1016/s0002-9440\(10\)63355-x](https://doi.org/10.1016/s0002-9440(10)63355-x).
33. Sasso, J.M.; Tenchov, R.; Wang, D.; Johnson, L.S.; Wang, X.; Zhou, Q.A. Molecular Glues: The Adhesive Connecting Targeted Protein Degradation to the Clinic. *Biochemistry* **2023**, *62*, 601–623. <https://doi.org/10.1021/acs.biochem.2c00245>.
34. Xie, X.; Yu, T.; Li, X.; Zhang, N.; Foster, L.J.; Peng, C.; Huang, W.; He, G. Recent advances in targeting the “undruggable” proteins: from drug discovery to clinical trials. *Signal Transduct. Target Ther.* **2023**, *8*, 335. <https://doi.org/10.1038/s41392-023-01589-z>.
35. Watson, J.L.; Juergens, D.; Bennett, N.R.; Trippe, B.L.; Yim, J.; Eisenach, H.E.; Ahern, W.; Borst, A.J.; Ragotte, R.J.; Milles, L.F.; et al. De novo design of protein structure and function with RFdiffusion. *Nature* **2023**, *620*, 1089–1100. <https://doi.org/10.1038/s41586-023-06415-8>.
36. Jiang, K.; Yan, Z.; Di Bernardo, M.; Sgrizzi, S.R.; Villiger, L.; Kayabolen, A.; Kim, B.J.; Carscadden, J.K.; Hiraizumi, M.; Nishimasu, H.; et al. Rapid in silico directed evolution by a protein language model with EVOLVEpro. *Science* **2025**, *387*, eadr6006. <https://doi.org/10.1126/science.adr6006>.
37. Wang, L.; Wang, N.; Zhang, W.; Cheng, X.; Yan, Z.; Shao, G.; Wang, X.; Wang, R.; Fu, C. Therapeutic peptides: current applications and future directions. *Signal Transduct. Target Ther.* **2022**, *7*, 48. <https://doi.org/10.1038/s41392-022-00904-4>.
38. Fosgerau, K.; Hoffmann, T. Peptide therapeutics: current status and future directions. *Drug Discov. Today* **2015**, *20*, 122–128. <https://doi.org/10.1016/j.drudis.2014.10.003>.
39. Jing, B.; Erives, E.; Pao-Huang, P.; Corso, G.; Berger, B.; Jaakkola, T. EigenFold: Generative Protein Structure Prediction with Diffusion Models. *arXiv* **2023**, arXiv:2304.02198.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.