THESIS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

Next-Generation Peptides:

AI-driven approaches for peptide therapeutics beyond the natural repertoire

GÖKÇE GEYLAN

CHALMERS

Department of Life Sciences
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2026

Next-Generation Peptides
AI-driven approaches for peptide therapeutics beyond the natural repertoire
GÖKÇE GEYLAN

ISBN 978-91-8103-367-0

Acknowledgements, dedications, and similar personal statements in this thesis, reflect the author's own views.

Doktorsavhandlingar vid Chalmers tekniska högskola
Ny serie nr 5824
ISSN 0346-718X
https://doi.org/10.63959/chalmers.dt/5824

Department of Life Sciences
Chalmers University of Technology
SE-412 96 Gothenburg
Sweden
Telephone + 46 (0)31-772 1000

Printed by Chalmers Digitaltryck
Gothenburg, Sweden 2026

Next-Generation Peptides:
AI-driven approaches for peptide therapeutics beyond the natural repertoire

Gökçe Geylan
Department of Life Sciences
Chalmers University of Technology

Abstract

Peptides are becoming an attractive modality in drug discovery as they sit in the intersection of small molecules and proteins. They combine advantages drawn from both modalities such as high specificity, low immunogenicity typically associated with protein therapeutics, and good efficacy, potential for membrane permeability more common to small molecules. Nevertheless, they demand an exhaustive search for an optimal amino acid sequence to meet a multi-objective profile including metabolic stability, solubility, and potency needed for becoming a standalone drug. Essential aspect of property optimization is incorporating non-natural amino acids (NNAAs) to peptides, generally to enhance their permeability and affinity. However, design-make-test-analyze (DMTA) cycles often rely on trial-and-error of positional mutations. This makes the identification of peptides meeting the design goals exhaustive and time-consuming. Drug discovery pipelines are accelerated by the recent surge of artificial intelligence (AI)-driven technologies for small molecules and proteins. This thesis presents *in silico* tools that facilitate drug design by extending AI-based methodologies to peptide therapeutics. The solutions presented here enable designs beyond the natural amino acids while allowing efficient exploration of the chemical space that is expanded by novel and diverse NNAAs. This includes developing AI-driven methodologies for design, evaluation, and optimization of next-generation therapeutic peptides. To design peptides, a chemistry-aware generative model was built to incorporate NNAAs into user-defined positions of a given starting peptide. This model is guided by reinforcement learning feedback to iteratively optimize designs for desired properties such as permeability and solubility. This design process is supplemented by a series of methodologies to evaluate the generated designs. First, peptide-specific predictive models that leverage model uncertainty were developed to efficiently predict permeability and steer design decisions toward reliable property space. Second, an NNAA synthesis assistance tool was proposed. This tool evaluates the chemical synthesizability of amino acids by considering protection strategies required for peptide synthesis and adapts predictive models for small molecule retrosynthesis and synthetic feasibility to peptide building blocks. Collectively, studies presented in this thesis develop cheminformatics and AI applications to design novel, synthesizable and pharmacologically relevant peptides while expanding the chemical space accessible to peptide drug discovery.

Keywords: drug discovery, peptide design, generative AI, predictive models, non-natural amino acids, reinforcement learning, uncertainty quantification, synthetic feasibility

Preface

This thesis serves as a partial fulfilment of the requirements for obtaining the degree of Doctor of Philosophy at the Department of Life Sciences at Chalmers University of Technology. The work is an industrial PhD, supported by the Swedish Foundation for Strategic Research (SSF) with Grant ID 20-0109, awarded to Associate Professor Florian David. Most of the work was carried out at the Molecular AI Department in AstraZeneca, in close collaboration with Division of Systems and Synthetic Biology. The PhD studies were carried out between September 2021 and February 2026 under the supervision of Adjunct Prof. Ola Engkvist and Assoc. Prof. Florian David and co-supervision of Research Prof. Verena Siewers, with Prof. Marija Cvijovic as examiner.

Gökçe Geylan
January 2026

List of Publications

This thesis is based on the work contained in the following papers, referred to by roman numerals in the text:

Paper I: **Geylan, G.**, Janet, J. P., Tibo, A., He, J., Patronov, A., Kabeshov, M., Czechtizky, W., David, F., Engkvist, O., & De Maria, L. (2025). PepINVENT: generative peptide design beyond natural amino acids. *Chemical Science*, *16*(20), 8682–8696. https://doi.org/10.1039/D4SC07642G

Paper II: **Geylan, G.**, De Maria, L., Engkvist, O., David, F., & Norinder, U. (2024). A methodology to correctly assess the applicability domain of cell membrane permeability predictors for cyclic peptides. *Digital Discovery*, 3, 1761-1775. https://doi.org/10.1039/D4DD00056K

Paper III: van Weesep, L., Chankeshwara, S., De Maria, L., David, F., Engkvist, O., **Geylan, G.** (2026). Conformal Prediction Enhances the Efficiency of Designing Permeable Peptides in Reinforcement Learning-Guided Optimization. *Manuscript.*

Paper IV: **Geylan, G.**, Kabeshov, M., Genheden, S., Kannas, C., Kogej, T., De Maria, L., David, F., & Engkvist, O. (2025). From concept to chemistry: integrating protection group strategy and reaction feasibility into non-natural amino acid synthesis planning. *Chemical Science*, 16, 17927-17938. https://doi.org/10.1039/D5SC04898B

Papers not included in this thesis:

- Kohl, F., Laufkötter, O., Firth, M., Krimpenfort, L., Mangla, P., Ansarizadeh, M., **Geylan, G.**, Eklund, L., De Maria, L., Jakobsson, L., & Wiseman, J. (2025). Identification of cell type-specific cell-penetrating peptides through *in vivo* phage display leveraged by next generation sequencing. *Biomedicine & Pharmacotherapy*, *182*, 117740. https://doi.org/10.1016/J.BIOPHA.2024.117740
- Mao, J., **Geylan, G.,** Scott, L. H., De Maria, L., Xia, Y., Ishchuk, O., Gutgsell, A., Firth, M., Taylor, A., Wiberg, F., Chankeshwara, S., Kwon, Y., Evander, M., Siewers, V., Engkvist, O., Davis, A. M., and David F. (2026). [Tool name redacted]: A holistic *in vivo* target-based drug discovery pipeline for cyclic peptide inhibitors demonstrated in a [Target name redacted] case study. *Manuscript.*
- Amirahmadi, A., **Geylan, G.**, De Maria, L., Etminani, F., Ohlsson, M., & Tibo, A. (2025). A decoupled alignment kernel for peptide membrane permeability predictions. *aRxiv*. https://arxiv.org/pdf/2511.21566
  Manuscript.

Contribution Summary

Paper I: I conceptualized the study with the other authors. I generated semi-synthetic data, trained the generative model including its training and evaluation, and wrote the manuscript. I edited the manuscript together with the other authors.

Paper II: I co-designed the study with O. E. and U. N. I carried out the experimental work with training and testing the models. I contributed equally to the methodology and formal analysis. I wrote the manuscript. I edited the manuscript together with the other authors.

Paper III: I co-designed the study with O. E. I supervised L.v.W. on carrying out the model building and uncertainty integration to reinforcement learning. With O. E. and L.v.W., I selected the peptides for wet-lab validation. L.v.W. wrote the manuscript and I edited the manuscript together with the other authors.

Paper IV: I contributed equally to the design of the study, developed the protection tool from heuristics created by M. K., built the end-to-end workflow with M. K. and S. G.. I wrote the manuscript. I edited the manuscript together with the other authors.

List of Abbreviations

ADME:	Absorption, Distribution, Metabolism, and Excretion
AI:	Artificial Intelligence
bRo5:	Beyond-the-rule-of-5
CP:	Conformal Prediction
CycPeptMPDB:	Cyclic Peptide Membrane Permeability Database
DF:	Diversity Filter
DL:	Deep Learning
DMTA:	Design-Make-Test-Analyze
ICP:	Inductive Conformal Prediction
LightGBM:	Light Gradient Boosting Machine
ML:	Machine Learning
MM-GBSA:	Molecular Mechanics Generalized Born Surface Area
MPO:	Multi-Objective Optimization
NLL:	Negative Log Likelihood
NNAA:	Non-natural Amino Acid
NP:	Non-permeable
P:	Permeable
PAMPA:	Parallel Artificial Membrane Permeability Assay
PCA:	Principal Component Analysis
PPI:	Protein-protein interaction
PTM:	Post-translational modification
RF:	Random Forest
RL:	Reinforcement Learning
RNN:	Recurrent Neural Network
SAR:	Structure-Activity Relationship
SMARTS:	SMILES Arbitrary Target Specification
SMILES:	Simplified Molecular Input Line Entry System
SPPS:	Solid-Phase Peptide Synthesis
SVM:	Support Vector Machine
t-SNE:	t-Distributed Stochastic Neighbor Embedding
USPTO:	United States Patent and Trademark Office
VS:	Virtual Screening
XGBoost:	Extreme Gradient Boosting

Table of Contents

Aim

The aim of this thesis is to advance AI-driven in silico design capabilities for peptides by addressing the key challenges in peptide therapeutics. In this thesis, I focus particularly on developing tools for peptide design, property prediction, and synthesis assessment.

- **Design.** Current computational peptide design methods are largely limited to 20 natural amino acids. This thesis aims to extend peptide design beyond natural building blocks by incorporating non-natural amino acids into peptides. By leveraging generative models, this work aims to access and navigate a broader and more diverse chemical space, and to propose peptides that better satisfy therapeutic objectives.

- **Property prediction.** Accurate prediction of therapeutically relevant properties presents a major bottleneck, limiting the utilization of predictive models in peptide drug discovery. The applicability of predictive models is often challenging when new peptide designs with novel chemistries are assessed. This thesis aims to establish methodologies to develop reliable property predictors capable of assessing chemically diverse peptides. The work focuses on permeability of cyclic peptides, a representative design case where new chemical modifications are commonly introduced to enhance permeability. Making reliable predictions in this context is critical for accurate and generalizable property predictors.

- **Synthesis.** In silico molecular design is generally accompanied by a fundamental problem related to the synthesis of the newly designed molecules. In peptide design, synthetic accessibility becomes a limiting factor for each new amino acid when navigating large chemical spaces. In this thesis, I focus on presenting an AI-based workflow that evaluates the synthetic feasibility of commercially unavailable NNAAs. Therefore, synthesis concerns can be addressed in early design stages and lead to reduced attrition between computational designs and experimental validation.

Overall, these three components together aim to provide a design framework for peptide therapeutics and contribute to a more holistic in silico design of therapeutic peptides.

Introduction

Peptides have been used as drugs for more than a century, since insulin was discovered and became a central treatment for type I diabetes.[1] It is only in recent years, especially with the success of weight loss drugs, that peptide therapeutics attracted significant interest for researchers both in academia and industrial organizations.[1,2] Peptides also have been an important actor for targeted delivery for radiotherapy.[3] The interest transformed the field with novel peptide designs, advanced topologies, new assays and synthesis methods, going beyond traditional approaches and into next generation peptides.[4] But did the surge of new *in silico* drug discovery and development applications manage to keep up with the increased demand for next generation peptides?

Peptides in Drug Discovery

The major efforts in drug discovery have traditionally focused on small molecules. The drug-likeness of small molecules was clearly outlined by a set of physicochemical descriptors.[5] These descriptors are known as Lipinski's rule of five. The rule of five states that molecules with more than five hydrogen bond donors, ten hydrogen bond acceptors, molecular weight higher than 500 Dalton and lipophilicity measure above five, have poor absorption or permeability.[6] Biologics, or biological drugs did not fit this "ground truth", and have been explored as new modalities in drug discovery, necessitating a paradigm-shift from small molecules. These beyond-the-rule-of-5 (bRo5) molecules demanded a new set of rules and considerations. Meanwhile, understanding and navigating the design goals of new modalities have not ignored the learnings from small molecules.[7]

Peptides are made up of 2 to 50 building blocks, or amino acids and sit right in between the realms of small molecules and proteins that contain more than 50 amino acids (Figure 1).[8] As they are larger than small molecules and smaller than proteins, they offer significant advantages over both of these modalities.[9] Compared to small molecules, peptides generally have higher specificity, lower toxicity and immunogenicity as they are also acting as signaling molecules or hormones in various signaling cascades of many organisms.[9] With their large surface area and modular structure, they can target shallow or large binding pockets. They can target binding pockets located at the surface of targeted proteins or can be designed to modulate the interface of protein-protein interactions (PPIs) that are undruggable by small molecules.[5] Compared to proteins, peptides typically have lower production costs, requiring fewer building blocks and synthesis cycles.[9] While proteins can achieve intracellular bioactivity profiles only upon delivery agents transporting them inside cells, some peptides can cross cell membranes either through transporter-mediated

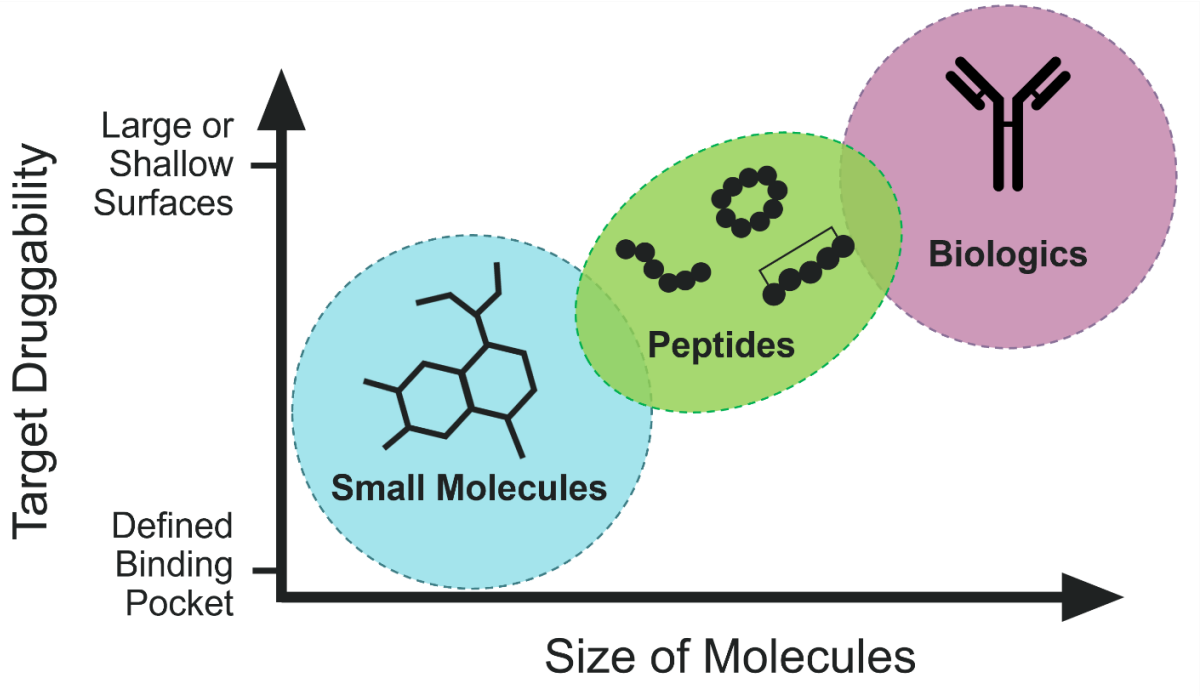


Figure 1. Peptide therapeutics in drug discovery

Peptides represent a modality in between small molecules and proteins. They have a larger size compared to small molecules, which allows them to target larger and more shallow binding pockets. Peptides are smaller than biologics such as some antibodies, oligonucleotides and proteins, facilitating targeting more defined pockets that cannot be targeted by biologics due to their bulkiness. The figure was inspired by Ji et al.[10]

uptake or by passive diffusion.[11] Additionally, small peptides are typically very flexible as opposed to proteins with defined conformational states, stabilized by intermolecular interactions between amino acids.[12] Dynamic conformational changes in peptides can enable passive diffusion, increasing cell permeability and potentially improving oral bioavailability compared to proteins.[11,13] Achieving membrane permeability gives access to intracellular targets comprising of approximately 60% of all drug targets.[14] Also, a subclass of permeable peptides, also known as cell-penetrating peptides, is leveraged as drug delivery agents to facilitate intracellular delivery of various cargoes such as small molecules, nucleic acids and proteins.[1] Another advantage over proteins is that it is generally easier to modify peptides. Minor changes in peptides can dramatically alter their properties. Automated chemical synthesis is more cost-efficient due to the lower number of building blocks in peptides.[1,13]

As a bRo5 modality, peptide drug candidates present design challenges to achieve the desired profile of stand-alone drugs. In this context, a stand-alone drug is a drug that reaches its target without delivery agents and achieves the desired efficacy. One of the most prominent challenges of peptide drugs is to make them stable.[13] Peptides have low metabolic stability as they tend to be targeted for proteolytic degradation, resulting in low plasma stability with much lower half-

lives.[5] When orally administered, peptides are generally digested by enzymes in gastrointestinal tract, i.e. pepsin in stomach degrading proteins and peptides by hydrolyzing them into fragments. Therefore, peptides are hardly orally bioavailable. This can also be deduced from the peptide drugs currently in market. Most of these drugs are administered through subcutaneous or intravenous injection instead of orally.[1] Finally, peptides can bring patentability considerations for weak patents. Patenting peptides made up of natural amino acids requires a set of conditions including a well-defined therapeutic activity, novelty or non-obviousness.[15,16] Patented peptides need to have sufficiently distinct sequences from naturally occurring peptides. This is because any sequence that is naturally occurring cannot be protected by patents as a novel molecule.[16] Therefore, designing a peptide drug also presents significant challenges.

Peptide chemical space and its expansion with NNAAs

Every design challenge is about evaluating and selecting molecules that satisfy the optimization objectives. The library of possible molecules considered for design is called the chemical space.[17] A defined set of options considered for candidate selection, such as enumeration of molecules from a database, form a predefined chemical space.[17] This set can be constrained depending on the properties or substructures relevant to the design objectives and experimental conditions.[17] The size of the chemical space of small molecules is estimated to be 10^{60}.[18] In the most general sense, the peptide chemical space encompasses all peptides that can be produced from combinatorial arrangements of amino acids. Therefore, a combinatorial library made up of n building blocks for peptide length of L, yields a chemical space with n^L peptides.[19] However, the theoretical chemical space would span all different lengths, topologies such as linear or cyclic peptides, and all possible amino acids.

Conventionally, chemical spaces considered for novel peptide designs are either restricted to a range of peptide lengths, a single topology and most notably, to only natural amino acids. Natural amino acids are the 20 proteinogenic building blocks encoded by the genetic code and incorporated into proteins.[20] Each amino acid has a backbone and a sidechain. The backbone contains an amino group, which serves as the N-terminus of a polypeptide chain, and a carboxyl group which becomes the C-terminus. In polypeptide chains, terminal modifications are used to enhance peptide stability.[9] Sidechains provide the chemical diversity to amino acids that enables functional diversification to peptides (Figure 2.A). Peptides made up of natural amino acids allow for lower cost in production via biological synthesis or fast reaction cycles in chemical synthesis by taking advantage of their modularity. Additionally, these amino acids constrain the peptide space to a vast sequence enumeration task with limited chemical diversity across building blocks when searching for new peptides. In these enumeration approaches, the size of the chemical space becomes 20^L with L being the length of the natural peptide.

Natural peptide enumeration is widely applied to find a peptide hit. With technologies such as DNA encoded libraries or display technologies, high-throughput screening of millions of peptide variants became achievable.[21] Still, previously mentioned design concerns such as metabolic stability, permeability, and oral bioavailability, have been commonly addressed by maturing the peptide hit to a lead by introducing non-natural amino acids (NNAAs).[22] Until 2024, over 100 peptide drugs containing NNAAs have been approved by the FDA, with 44% of these administered orally.[20] Modifying natural peptides is nothing new given the post-translational modifications (PTMs) to tune peptide stability, localization, and functional diversification in cellular processes.[23]

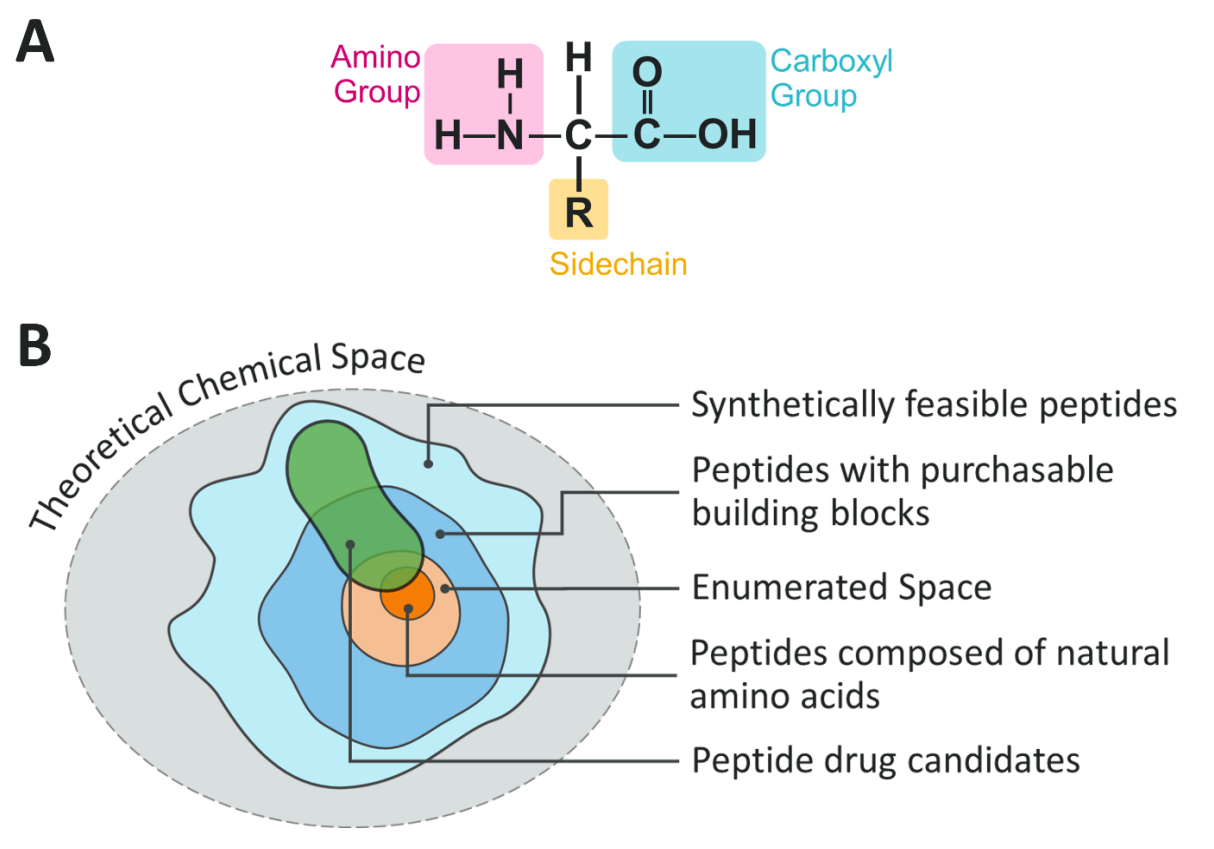


Figure 2. Peptide building block structure and chemical space

(A) Amino acids contain a backbone, with an amino group and a carboxylic group, and a sidechain that is the variable region. The chemical differences in sidechain and backbone give distinct physicochemical profiles to the building blocks. (B) The peptide chemical space is bounded by a theoretical unconstrained space that encompasses all possible peptides. This space contains nested subspaces ordered from larger to smaller as the synthetically accessible peptides, peptides with commercially available building blocks, enumerated space chosen for high-throughput screenings, and peptides containing natural residues. Therapeutic candidates, however, can be considered as a space that partially spans all the nested subspaces, with the design tendencies toward NNAA incorporation, new modifications, and so on.

Although many definitions exist, any amino acid that is not one of the 20 natural amino acids, including the non-proteinogenic ones, will be referred to as an NNAA throughout this thesis. In drug design, NNAAs have been incorporated into peptides to increase half-life by evading plasma proteins. For example, the only NNAA in Semaglutide, 2-aminoisobutyric acid or Aib, helps escape enzymatic degradation, improving its half-life.[24] Designing novel NNAAs or introducing modifications like backbone *N*-alkylation or stereoisomeric conversion to natural amino acids can enhance passive permeability.[25,26] Bulky, long sidechains of NNAAs can increase hydrophobicity, while *N*-alkylated backbones can decrease conformational flexibility, enabling conformational behaviors that favor permeability.[22] They can also improve a peptide's affinity to its target by tailoring a better fit to binding pockets among many other property alterations.[27] Additionally, NNAAs provide easier intellectual property protection since the modified peptides are generally not found in nature and novel NNAAs can be patented. Overall, peptides with NNAAs can better meet design objectives and overcome the disadvantages exhibited by their natural counterparts.

Including NNAAs in high-throughput screens dramatically expands the enumerated space, with the inclusion of diverse backbones, stereochemical inversions and arbitrary sidechain designs. This exponentially increases the size of the combinatorial library.[28] Theoretically, considering all possible NNAAs would transform the amino acid chemical space from 20 options to a space similar to that of small molecules, projecting the peptide chemical space to effectively unbounded size (Figure 2.B). The expanded design freedom unlocks a range of properties that once was unattainable with natural amino acids. This in turn enables targeting the undruggable targets.[5,29] Augmenting the peptide space with NNAAs during hit-to-lead and later, lead-optimization is the natural next step to design peptide drugs after identifying a sequence hit with enumeration. Although NNAAs enable customized peptides with drug-like profiles, navigating these ultra-large design spaces introduces substantial challenges with a major emphasis on multi-parameter optimization and synthesis complexity.

Synthesis challenges for peptides with NNAAs

NNAAs bring practical constraints across the drug development lifecycle, particularly in synthesizing the peptides that contain them. The set of building blocks in the designed sequence has to be available and later correctly assembled to yield the desired peptide. Peptides can be made biologically or chemically.

Biological synthesis utilizes "the central dogma" in which a DNA molecule encoding the peptide is transcribed into mRNA and translated into a peptide by ribosomes. This requires the corresponding DNA sequence encoding the peptide to be expressed in the cells to produce the desired peptide.[8] This synthesis process strictly yields natural peptides unless biological

engineering is applied to also enable integration of NNAAs, for both *in vitro* and *in vivo* systems.[1,8,30] In recent years, ribosomal synthesis of peptides with NNAAs was made possible with engineered tRNA carrying NNAAs to ribosomes or developing *in vitro* translation systems that apply PTMs.[22,31,32] Additionally, *in vivo* methods such as genetic code expansion or tailoring enzymatic reactions to mediate NNAA coupling to tRNAs, have enabled synthesizing peptides with NNAAs.[30,33] However, biological systems are costly, time consuming and demand

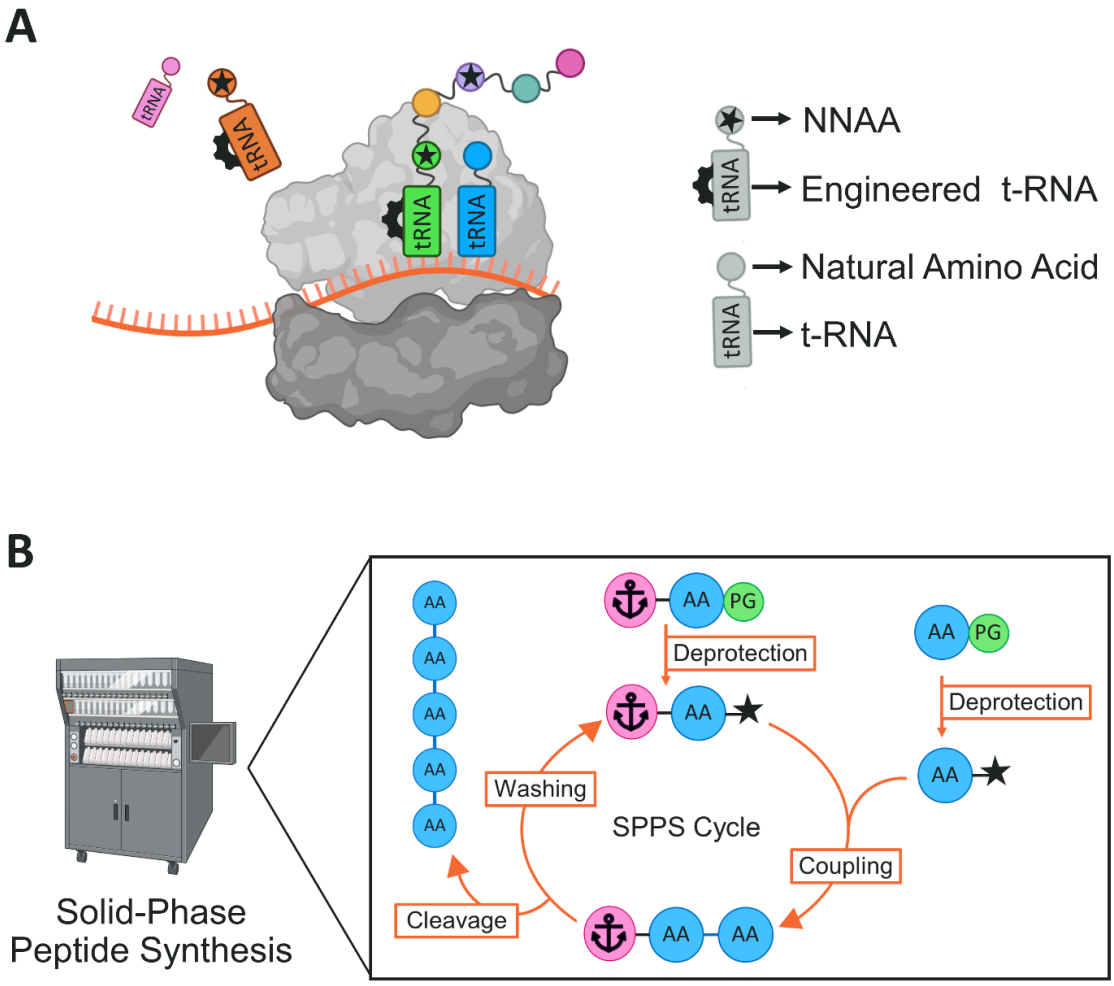


Figure 3. Biological and chemical synthesis for peptides

(A) Biological synthesis relies on the ribosomal translation with peptides encoded by the genetic code. This mechanism can be reprogrammed to include NNAAs by engineering new tRNAs to carry the novel NNAAs.[32] (B) SPPS is the most common chemical synthesis approach for peptides. The starting amino acid in the target sequence is anchored to a resin, followed by deprotecting and coupling both the anchored and the next amino acid in the sequence. Protecting groups removed by orthogonal deprotection conditions enable controlled exposure of the substructures, thus controlled building block coupling. The peptide chain iteratively grows until the end of the target sequence is reached and is later cleaved from the resin as the final product.[34] Parts of this figure were created with Biorender.

extensive enzymatic engineering as well as a good purification method to extract the peptide product (Figure 3.A).[8,33]

A more common and straightforward peptide production is achieved with chemical synthesis.[8] Solid-Phase Peptide Synthesis (SPPS) is a chemical synthesis method that can synthesize natural or non-natural peptides by stepwise integration of the amino acids into the growing peptide chain.[35] SPPS requires orthogonal protection of the reactive residues in amino acids. This means that each protected functional group is capped with a protection group that can be removed under the conditions that do not affect the others.[36] During SPPS, an iterative cycle is conducted by deprotecting a specific reactive substructure of an amino acid and coupling it to the peptide chain.[35] Orthogonality ensures that at each step only the intended reactive substructure is deprotected for coupling. This prevents side reactions and enables precise assembly of peptides (Figure 3.B).[37] Furthermore, it enables synthesis of peptides with specific topologies or sequences as well as incorporation of NNAAs. The controlled deprotection reduces the byproducts and effectively yields the intended peptide molecule. When an NNAA is commercially available, it can be supplied to SPPS directly as ready-to-use protected building block. Opting for chemical synthesis allows incorporating greater diversity of NNAAs in peptides that are inaccessible to biological synthesis.

Peptide synthesis gets significantly more complex, when an NNAA is not commercially available whether due to a novel sidechain or additional reactive sites beyond the standard selection. Such cases present an additional level to synthesis: the synthesis of the protected NNAA itself. An NNAA of interest can be treated as a small molecule synthesis problem often requiring multi-step synthesis routes, route optimization with specialized reagents, and efficient orthogonal protection to make it SPPS-compatible. These prerequisites could lead to increased cost and time for the synthesis of novel NNAAs.

Cyclic versus Linear Peptides

Cyclic peptides have gained special attention in drug discovery. Cyclization can grant design advantages over linear peptides. Cyclic peptides can have their N- or C- termini, or both, contributing to the cyclization, shielding the terminal amino acids from proteolytic degradation. Exopeptidases hydrolyze peptides from their terminal residues and cyclization enhances the peptide's metabolic stability and thus, half-life.[10] Additionally, cell permeability has been associated with cyclic peptides with many studies exploring rational design of permeable sequences.[10,38] The mechanisms for passive permeability are not fully understood and also amino acid composition-dependent. Yet, cyclic peptides are known to present structural advantages over their linear counterparts.[10] Cyclization results in a conformationally constrained backbone, reducing the entropic penalty of structural reorganization and making adaptative conformational

shifts more favorable.[10,11] This adaptive behavior of peptides flexibly arranging their conformation, is called chameleonicity. Chameleonic behavior is hypothesized to correlate with permeability.[11,26,39] Cyclic peptides exhibiting this behavior, maintain a hydrophilic conformation in water-abundant extracellular environment. They extend their sidechains to expose polar groups and form interactions with the solvent.[26] When entering the nonpolar cell membrane, they reduce their solvent-accessible surface area by folding their sidechains inward. This increases the intramolecular hydrogen bonds as peptides conform to a nonpolar state.[26] Following the entry to the aqueous intracellular environment, they switch back to their hydrophilic conformation.[26] These dynamic conformational shifts enable passive membrane diffusion, allowing peptides to reach intracellular targets and potentially improving oral bioavailability by enhancing their gastrointestinal absorption.[11,26,40,41]

PPIs are fundamental for all cellular processes such as signaling cascades and metabolic pathways.[5] At the interface of PPIs, large and relatively flat contact surfaces are often complemented by loops from one partner extending into the binding pocket of the other.[26] These loops contribute significantly to PPI affinity with a drastic decrease in binding upon introducing mutations to any of its residues.[42,43] With their ring topology, cyclic peptides can mimic these hot spots, also known as hot loops, to modulate PPIs, either by physically blocking the interface or acting as anchoring motifs.[42,44] Cyclic peptides can mimic PPI's hot loops due to their structural resemblance. Therefore, these peptides are an interest in drug design, where incorporating NNAAs can enhance their therapeutic properties.

Artificial Intelligence and its use in developing peptide therapeutics

Artificial Intelligence (AI) has been gradually adopted by drug discovery in various forms. With the development of large-scale and high-throughput experiments such as display technologies, the large amounts of data generated have opened the door for AI applications. These applications are mostly represented by predictive models aiming to model activity data or absorption, distribution, metabolism, and excretion (ADME) properties.[1,45] The accumulation of data suitable for modelling together with advances in AI architectures has drifted the interest from traditional property prediction with classical methods, such as Support Vector Machines and Random Forest to more complex methods including Deep Learning (DL) techniques. Scientists started addressing challenges that were never considered before, such as design and ideation of novel molecules, synthetic feasibility and planning, homology modelling/structure prediction, cell imaging, protein expression level optimization and many others.[46] More specifically for peptides, sequence design, predicting structure, bioactivity, permeability as well as adverse effects such as toxicity, immunogenicity became heavily studied.[47] Here I will highlight the most relevant AI applications explored in this thesis:

Generative Models for Drug Design – The application of generative models for ideation of small molecules was introduced quite recently in drug discovery. However, it evolved rapidly and attracted significant attention since it became evident that it provides access to virtually unlimited chemical space of molecules to synthesize. While the application on small molecules evolved and matured, the generative capabilities for designing peptides lagged. The published models were rather task-specific and appeared to be limited to the 20 natural amino acids. Although a handful of NNAAs were added to the building block library, the models typically operated on a fixed space of enumerated building blocks for sequence-based design. They were often specific to a certain peptide topology or mixed with exhaustive structure-based methods.[48–51] This indicates the need for a generative model that enables design of peptides with NNAAs that is task agnostic and applicable to a range of topologies. In *Paper I*, we utilize generative modelling for peptide design.

In the realm of small molecules, there are many works addressing molecular design with generative models, but we can use REINVENT as a reference. REINVENT has been the industry standard for generative design and we have relied on its methodology in *Paper I*. REINVENT can be seen as composition of three main components: A generative model to capture the syntax of molecular representation, a set of multiple optimization objectives known as multiple parameter optimization (MPO) that is formulated in a scoring function and a Reinforcement Learning (RL) algorithm to train the model towards the scoring function.[52–54]

The generative model has been trained to learn a specific molecular format, Simplified Molecular Input Line Entry System (SMILES), which is essentially a string of characters that describe the chemical structure of molecules.[55] Various model architectures ranging from Recurrent Neural Networks (RNNs) to Transformers have been trained and established to capture successfully the SMILES syntax. REINVENT platform includes conditional and unconditional generative models, addressing different small molecule design tasks including scaffold hopping, linker design, library design, and lead optimization.[54,56] It is important to point out that the ability of RNNs to keep track of long-range dependencies drops with the increased length of the generated sequence.[57] This increases the risk of dropping the chemical validity for longer SMILES sequences. Since peptides are bRo5, implying longer SMILES sequences compared to small molecules, this shifts the choice of generative model towards the usage of Transformer architecture. Transformers are composed of encoder-decoder stacks of layers: an encoder transforms the input into a contextual vector representation, or to latent space, and a decoder generates the output autoregressively, predicting each token conditioned on previously generated tokens and the encoded input (Figure 4).[58] The robustness of Transformers comes from self-attention which helps the model assign importance to different parts of the input by weighing the relative importance of tokens to each other.[58] Combined with informing on the order of input tokens via positional encoding, self-attention enables capturing long-range dependencies and

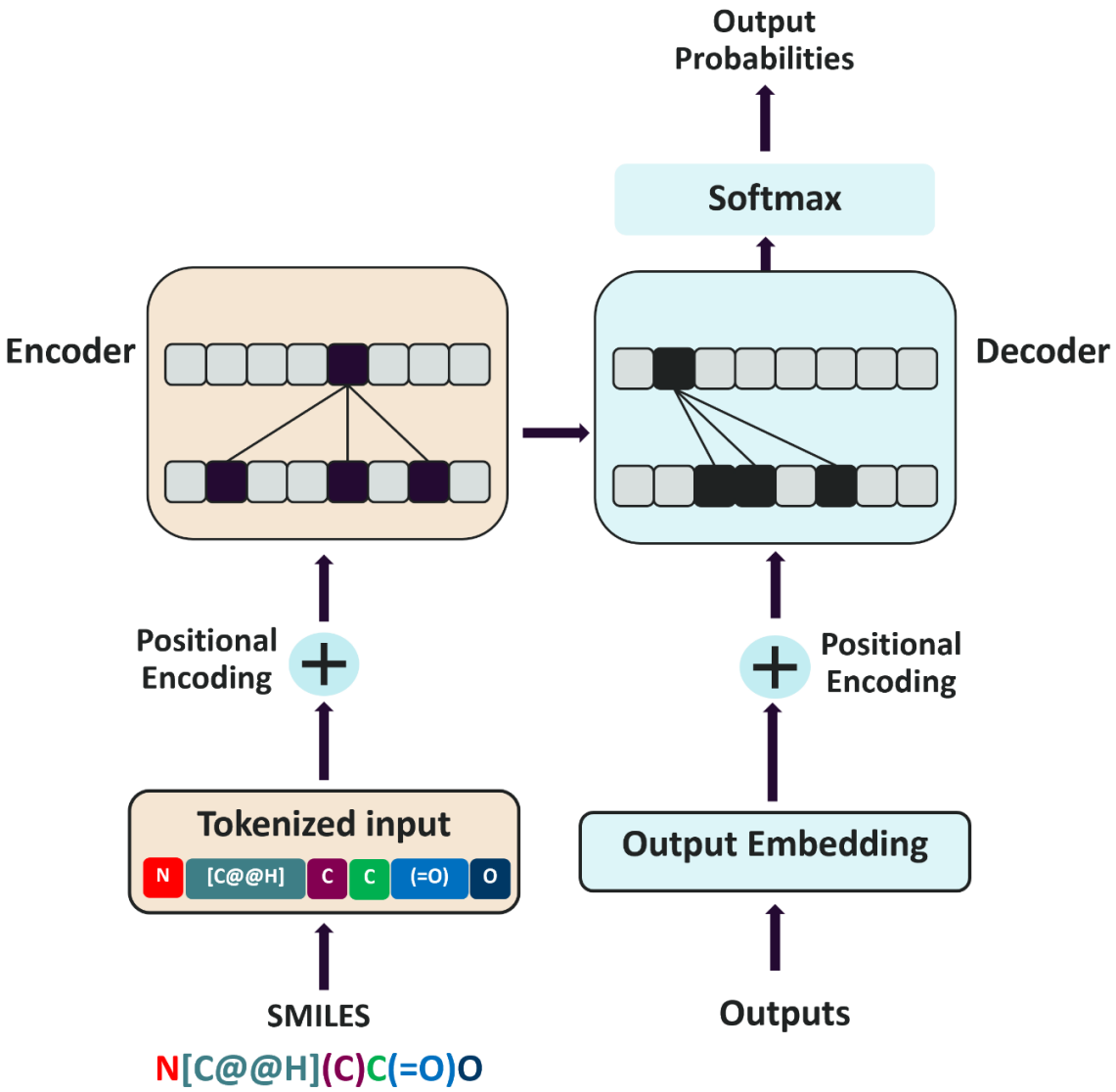


Figure 4. High-level Transformer architecture.[58]

The figure depicts the transformer architecture, used in Paper I. The input SMILES sequences are tokenized and converted into vector representation, also known as embeddings.[58] *Transformers lack understanding of token order since they process data in parallel. Positional encoding informs the model on the position of each token within a sequence. The architecture contains an encoder and a decoder that processes the same embeddings. Each encoder and decoder layer has self-attention and feed-forward layers. Self-attention mechanism enables the transformers to determine the relationship among the tokens within the generated sequence*[58]*. The architecture uses multiple attention heads in parallel. Each head captures distinct patterns and relationships in the data. The encoder transforms the input sequence into a vector, and the decoder converts this vector back into a sequence. Finally, the decoder output is passed through a linear layer after which a Softmax function converts the output to probabilities. These probabilities serve to choose the next output token.*[58]

thereby, the overall context of the input (Figure 4).[58] Training a Transformer model requires paired inputs: a source sequence to be encoded and a target sequence to be decoded.[58]

The learning objectives, or scoring components, considered for MPO are combined in a composite scoring function as a weighted sum or weighted product described, respectively, as[53]:

$$S(x) = \frac{\sum_i w_i \times p_i(x)}{\sum_i w_i} \qquad S(x) = [\prod_i p_i(x)^{w_i}]^{1/\sum_i w_i}$$

where $S(x)$ represents the score, or reward, of the MPO for the generated sequence, w_i is the user-configured weight, defining the importance of the learning objective i and $p_i(x)$ is the score assigned by the scoring component.

The RL setup consists of an actor and environment where the actor is the generative model. The actor takes a set of actions referred to as policy. The actions are essentially individual token selections which are eventually translated into a SMILES string. Actions are chosen by using multinomial distribution in REINVENT.[53] Only after completing the policy, the actor receives a reward by the environment, which in this RL scenario is referred to as a policy iteration. The environment includes the scoring function and a static copy of the actor, called prior. The prior is used for estimating the negative log-likelihood (NLL) for each newly generated sequence by the agent, thus serving as a regularization factor for instances where the actor drifts off too far from the initial distribution the model has been trained on.[53] This helps prevent the loss of syntax validity of the generated SMILES strings (Figure 5).

Predictive models – The most common application of AI in drug discovery is to build predictive models. Development of target and off-target activity predictive models along with ADME models are typical applications in a drug discovery project. While the amount of publicly available data for small molecules is much larger than for peptides, diverse methods, representations and models have been developed for small molecules. The limited data dominated by natural peptides has constrained the AI efforts to sequence-based predictors. This makes many published models for cell penetrating, antimicrobial, or signal peptides hardly applicable to peptides with NNAAs.[50,59] Furthermore, it is worth highlighting that due to the nature of the predictive models, they are primarily useful only within their domain of applicability and do not generalize well to modified peptides.[60] In *Paper II*, *III*, and *IV*, we train predictive models to predict cell permeability for peptides and employ small molecule predictive models for synthetic feasibility of amino acids.

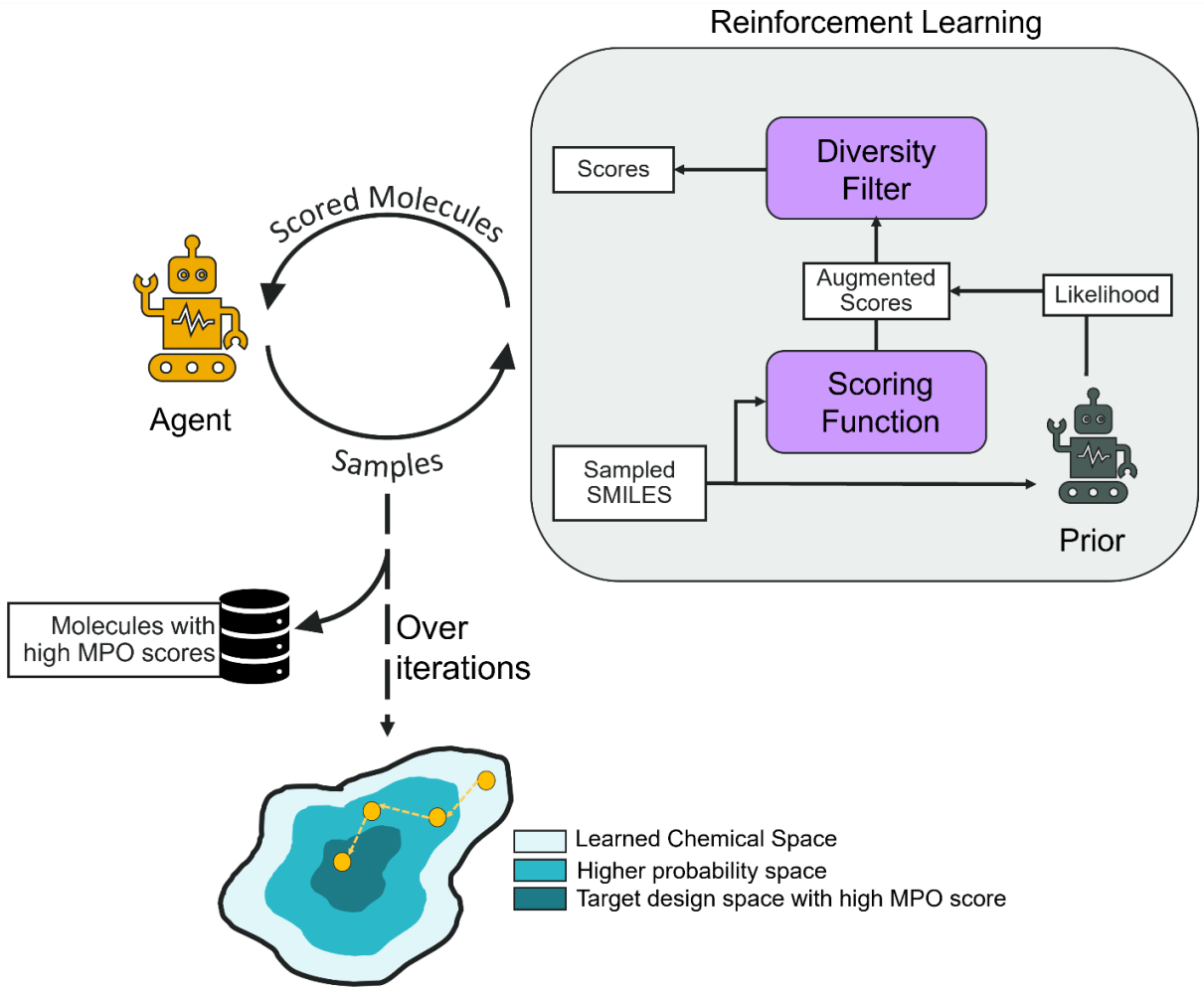


Figure 5. Reinforcement Learning process for REINVENT

Agent trained to generate SMILES strings for small molecules generates a batch of SMILES. This batch is scored by the scoring function, aggregating scores for various learning objectives in the MPO scenario. Prior evaluates the likelihood of the generated SMILES to penalize strings that drift off in the chemical space and likely represent invalid SMILES. Molecules are also filtered by the diversity filter that filters out highly similar molecules and ensures diverse chemistry over the learning steps. The resulting scores are fed back to the agent, iteratively fine-tuning it and navigating the learned chemical space to regions with high MPO scores with optimal designs.[56]

Applicability Domain and Uncertainty Quantification – Predictive models are known to be much more reliable at making interpolations rather than extrapolations. The applicability domain is essentially the chemical space within which a predictive model can reliably make accurate predictions.[17] Knowing whether query compounds are within or outside the model's applicability domain directly influences the confidence in the predictions. This implies the need to detect when the model is used inside or outside of its applicability domain. In other words, we need to quantify the certainty of the resulting predictions.

In two of the papers, we heavily relied on conformal prediction (CP) as the uncertainty quantification method. CP is a mathematical framework that calibrates a trained model with a calibration set that is different from the training set.[61] The calibrated model then yields its predictions from the comparison of new instances to its calibration examples.[61] As an uncertainty quantification method, CP enables decision-making through any confidence level demanded by the user.[62] Asking for 80% confidence would be setting the significance level of 0.2 or 20% error rate.[62] Based on the confidence, a point prediction is projected into a prediction set or interval for classification and regression tasks, respectively.[62] We used binary conformal classifiers in *Papers II* and *III.* Therefore, this section focuses on diving into the methodology for assigning prediction sets for binary tasks. In practice, we used Inductive Conformal Prediction (ICP) where a training set gets separated into a proper training set used for model training and a calibration set used for calibrating the trained model.[63] The prediction region is constructed through a nonconformity score, a proxy for how dissimilar a new instance is to the calibration set examples.[64] Dissimilarity, in *Paper II* and *III*, is the predicted probability of belonging to either class. In inference, the nonconformity score of the test instance is computed and compared to the calibration set example.[64] The fraction of calibration nonconformity scores that are greater than the test instance's nonconformity score, relative to the size of the calibration set, defines the conformal p-value.[63] The p-values are calculated separately for the binary classes, yielding independent p-values for the individual classes, also known as Mondrian ICP.[64] The two p-values are then used to decide the output prediction set for the test instance:

$$Prediction\ set = \begin{cases} (p_0 \geq \alpha)\ \wedge (p_1 \geq \alpha) & \{0,1\} \\ (p_0 \geq \alpha) \wedge (p_1 < \alpha) & \{0\} \\ (p_0 < \alpha)\ \wedge (p_1 \geq \alpha) & \{1\} \\ (p_0 < \alpha) \wedge (p_1 < \alpha) & \emptyset \end{cases}$$

where p_0 and p_1 represent the p-values of the binary classes of 0 and 1, respectively, and α is the significance level.[65] When the output is an empty set, ∅, the model cannot make a reliable prediction under the user-specified confidence level. Predicting{0,1} , "Both" classes, implies that the model cannot differentiate between the classes under the confidence level. Finally, {0} or {1} can be predicted, where the model assigns a single-label prediction to the test instance.[65] While having the true label of the test instance, either a correct single-label or "Both" class would indicate a valid prediction, a single-label is an efficient prediction.[66] Validity and efficiency for the binary classes can be calculated independently through:

$$Validity_{class=x} = \frac{N_{Correct\ single\ label\ predictions\ of\ class=X}\ +\ N_{\text{"Both" }predictions\ of\ class=X}}{N_{True\ class=X\ samples}}$$

$$Efficiency_{class=x} = \frac{N_{Single\ label\ predictions\ of\ class=X}}{N_{True\ class=X\ samples}}$$

where class=X is one of the binary classes.[66] CP offers advantages over other uncertainty quantification methods for being model-agnostic and distribution-free, meaning its applicability does not depend on the choice of the underlying model or on assuming a specific probability distribution of the data.[67] The only assumption CP necessitates is the exchangeability in which the data is independent and identically distributed.[62] Under exchangeability, the predictions are directly influenced by the user-defined confidence, the error rate is guaranteed.[62] This facilitates retrieving valid and efficient predictions from the model. Higher confidence levels push the algorithm to assign "Both" labels more often to satisfy the allowed error rate, reducing the efficiency of predictions. Thus, balancing the trade-off between efficiency and validity is essential to maximize the confidence demanded while preserving the model's practical utility.[63] In *Paper II*, we use CP to calibrate the permeability predictors to extend the applicability of models to never-before-seen peptides. In *Paper III*, efficiency was used as the learning objective to improve the reliability of the generated designs.

Generative AI for Peptide Design

After finding a peptide hit through screening combinatorial libraries, being inspired by natural products or mimicking hot loops in PPIs, the identified hit is optimized into a lead compound. This process iteratively modifies the peptide to improve its potency along with pharmacokinetic properties such as stability, safety, solubility, and permeability.[68] In line with the complexity of the process, lead optimization is long and exhaustive.[1] Traditionally, after finding a hit, amino acid scanning experiments substitute individual positions to most commonly alanine. This helps identify residues fundamental for the peptide hit's activity, or the pharmacophore.[69] These critical residues could form a pattern, or a motif, that must be present in the peptide sequence to provide the key interactions of target binding. They can also represent the minimal sequence to retain interaction with the target and exhibit biological activity.[70,71] The pharmacophore residues can be developed to improve target interactions and thus, the affinity. The remaining residues are then diversified to broader amino acid libraries during lead optimization, to optimize properties relevant to the design goals.[1,4] While peptide-based screening and delivery platforms are advancing to consider NNAAs, computational tools offer faster and cheaper exploration of positional mutagenesis by virtual screening (VS) for large sets of NNAAs. VS applies computational scoring of predicted bioactivity and drug-like properties across a generated library to find promising peptides as potential drug candidates.[72] Recently, a set of building blocks considered for virtual peptide libraries was extended by 10,000 α-amino acids.[73] Compiled from eMolecules[74] as readily synthesizable residues, these amino acids were enumerated from available precursors of reactions commonly used in amino acid synthesis.[73]

Chemical space expands exponentially when considering multiple positions for mutation, a large number of building blocks or *de novo* sequence design for peptide optimization. Generative models offer a way to learn and explore such large chemical spaces while RL helps navigate it to design subspaces.[75] This has been established for small molecules and proteins with many publications including REINVENT.[53,76] Similarly, many models were proposed to navigate the peptide space. These models generally used a fixed vocabulary of natural amino acids or an extension with a few NNAAs since modified peptide data is limited for effective data-driven approaches.[60,77] These models treat amino acids as discrete tokens of the peptide language, disregarding the chemical similarity between building blocks and context at the atomic level.[77] At the time of this study, there was no solution that tackled understanding peptide chemistry rather than sequence patterns that truly extended design beyond a fixed repertoire. In *Paper I*, I will describe PepINVENT, a generative model that can flexibly navigate the chemical space of peptides.[78]

PepINVENT: Generative Modelling Beyond Natural Amino Acids (*Paper I*)

In *Paper I*, the aim was to build an idea generation tool to find tailor-made peptides that meet design goals in lead optimization. We explored how generative models can be used to learn the peptide chemical space expanded by NNAAs and how RL navigates this vast space. To learn peptide chemistry, understanding the building blocks was the starting point. In this work, our aim was to explore the natural amino acids, NNAAs readily available from vendors, and novel NNAAs that are not commercially available in lead optimization context. PepINVENT was established on two ideas: i) perceiving peptides as "beads-on-a-string" with the beads being small molecules, and ii) extending REINVENT's methodology to peptides. By treating amino acids as small molecules, peptide design could be transformed from sequence engineering to ligand design. Using RL enabled ligand design by traversing the learned space and proposing peptide candidates that satisfy multiple design objectives.

Training a chemistry-aware generative model

As mentioned in the *Introduction*, the residues critical for the activity of the peptide hit is almost always known. For lead optimization, this presents two main opportunities for modification:

1. Changing the critical residues to improve affinity
2. Modifying residues that are not critical to increase affinity or other drug-like properties

Therefore, late-stage peptide design consists of a subset of residues in the sequence we want to modify and the remaining ones to be fixed. PepINVENT was trained to operate under this design paradigm. We formulated peptide design as a text-infilling task in which amino acids are generated to fill in the user-specified positions in a query peptide. This requires the same number of amino acids as the selected positions to be generated. Additionally, a chemically meaningful peptide representation had to be achieved after substituting the user-defined positions with the corresponding generated amino acids (Figure 6).[79] This demanded the model to effectively capture the entire peptide context with topological awareness, chemical structure of amino acids and monomer connectivity rather than case-specific enumeration.

Since the model needed to generate amino acids based on the input peptide, a conditional generator had to be trained. For this purpose, PepINVENT leveraged Transformer architecture with proven track record on robustness across tasks from natural language processing and machine translation to drug discovery tasks such as *de novo* small molecule design and navigating the chemical space for identifying near-neighbors of molecules[80] Training a Transformer model requires paired inputs: a source to be encoded and a target to be decoded.[58] The source in our case was the query peptide

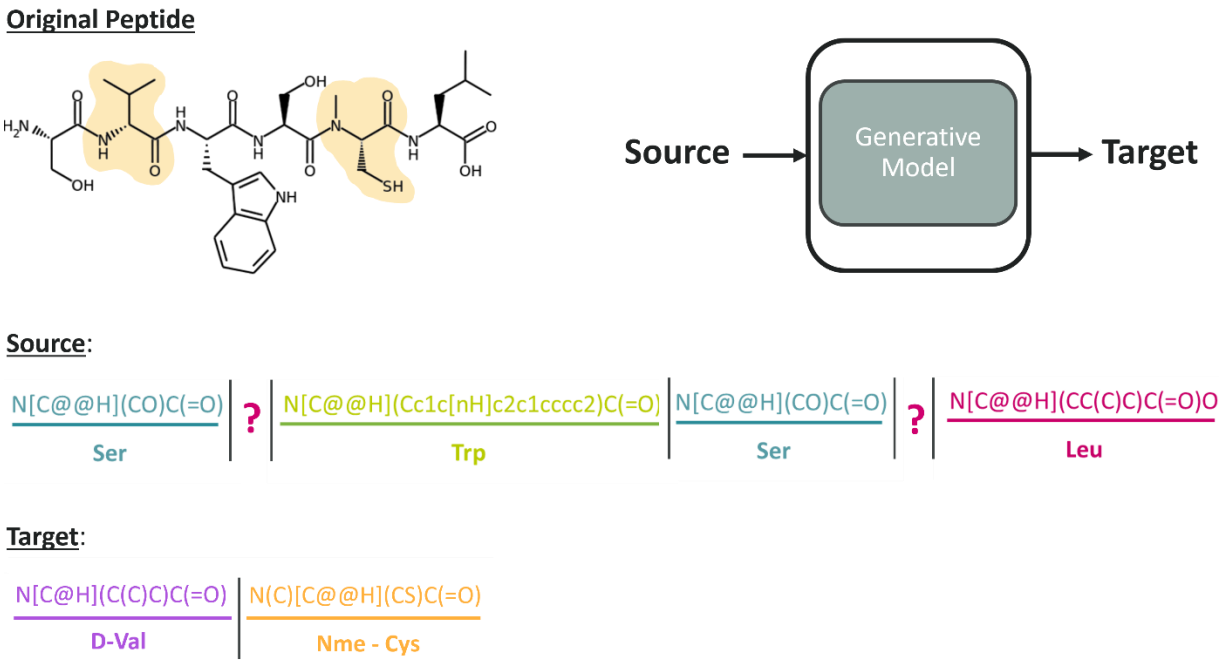


Figure 6. Training objective for the generative model

The generative model was trained on semi-synthetic peptides, represented by CHUCKLES strings. The source and target pairs required for training the Transformers were formed by extracting the CHUCKLES pattern of amino acids from one to multiple positions. The extracted amino acids became the targets, and the peptide with missing amino acids became the source. The figure shows a source-target pair representative of those used in the training set. The figure was extracted from Paper I [78] under a Creative Commons Attribution 3.0 Unported Licence[81].

with single or multiple positions marked for modification while the target was the amino acids to substitute the marked positions.

The transformer model can be described by a function, f_θ parametrized by the parameters, θ, that learns the chemical space, χ, through the probability distribution of the set of tokens forming vocabulary, V[78,80]:

$$f_\theta : \chi \times \chi \rightarrow [0,1]^{|V|}$$

The model was trained to minimize NLL of target tokens, y, with length T, conditioned on the source, x, in the space χ. At each step in autoregressive manner, the model computes the probability of t^{th} output token, y_t conditioned on the source x and previously generated tokens, $y_{0,\ldots,t-1}$.[78,80] NLL is described as:

$$NLL(x, y) = -\sum_{t=1}^{T} \log f_\theta\,(x, y_{0:t-1})[y_t]$$

CHUCKLES was used to describe peptides as a string of amino acids connected via peptide bonds.[55,82] It is a method that offers a standardized format for peptides with correct connectivity of amino acids.[82] At monomer-level, CHUCKLES yields SMILES for amino acids in N-to-C direction.[82,83] This enables direct concatenation of amino acids in the order of their occurrence in the sequence into a chemically valid peptide SMILES. Additionally, building up monomers into and breaking them down from peptides becomes straight-forward. Because CHUCKLES preserves the amino acid order and uses a structured atomic description rather than a scrambled line notation, it is easier to handle than SMILES.[78,82] PepINVENT was trained on source-target pairs encoded in this atomic representation. We also used a separator token, "|" between amino acids to emphasize monomer boundaries to the model. This token serves as a convenient handle for manipulating the string representation when adding or removing amino acids from peptides (Figure 6).

To learn the peptide chemical space effectively, the generative model must be trained on large and diverse training data. To overcome the limited availability of modified peptide data, we generated one million semi-synthetic peptides as our training data, setting aside 5% for validation and 5% for testing. We used natural amino acids and 10,000 NNAAs enumerated by Amarasinghe *et al*.[73] The training data spanned natural and non-natural amino acids, multiple topologies (linear, head-to-tail, sidechain-to-tail, or disulfide bridged), stereochemical variants, and backbone *N*-methylations. The paired inputs were formed by randomly extracting up to 30% of the amino acids from each training set peptide and replacing their CHUCKLES segment with a masking token, "?". The masked peptide and their extracted amino acids were supplied to the model as source-target pairs (Figure 6). The model was trained for 24 epochs.

Learning a peptide language at atomic resolution

After training the model, we extensively evaluated the learned chemical space. At inference, for each test set peptide, we sampled 1,000 amino acid sets and reconstructed 1,000 proposed peptides per query. 400 peptides, comprising the topologies included in the data generation equally, were used as the test set and sampled with:

1. Stochastic multinomial sampling[53,84]: To investigate how the model explores the learned chemical space through analyzing the diversity of generated amino acids and therefore the diversification of peptides.
2. Deterministic sampling with beam-search[80,85]: To examine accuracy of the learned task by looking at the high-likelihood outcomes.

Across these 400 masked test peptides, 1,000 proposed peptides were generated, making 400,000 samples. These proposed peptides were evaluated first for success at task completion. In nearly every case, the model generated exactly as many amino acids as the number of masked positions,

Table 1. Validity and uniqueness of generated peptides

The percentage of valid and unique peptides among the 1000 generated peptides, averaged over 400 test peptide queries. The scores are broken down on two sampling methods: beam-search and multinomial sampling (conducted in triplicates for reproducibility). The averages were further described according to the peptide topology. The topology completion task is tested under multinomial sampling to validate the contextual understanding of the peptides' CHUCKLES pattern in stochastic sampling. The table was adapted from Paper I[78] under a Creative Commons Attribution 3.0 Unported Licence[81].

Metric	Sampling Method	Total	Linear	Head-to-Tail	Disulfide Bridge	Sidechain-to-Tail
Peptide Validity (%)	Beam search	99	100	100	100	98
	Multinomial	98	98	98	99	97
Peptide Uniqueness (%)	Beam search	100	100	100	100	100
	Multinomial	98	100	99	94	99
Peptide Validity for Topology Completion (%)	Multinomial	98	100	96	98	99

with only one test case with 0.3% failure. Second, generation quality was investigated on chemical correctness, in other words validity as well as diversity, uniqueness and novelty. For both sampling methods, validity and uniqueness exceeded 98%. Thus, assembling the query peptide with the generated amino acids yielded chemically valid molecules and their CHUCKLES representations translated into unique canonical SMILES (Table 1). These tests showed similar performances across linear and various cyclic topologies, indicating robust performance independent of the peptide topology.

Validity, uniqueness, novelty, and diversity are evaluation metrics commonly used to assess generative models for small molecules.[86] For PepINVENT, we had to separate the generative capabilities into two levels: peptide-level and monomer-level. Generating unique peptides with various combinations of a handful of amino acids would indicate an overfitted chemical space that defeats our purpose. This is the same case for generating amino acids that reduce to the same canonical representation. At monomer-level, we included both sampling methods in the manuscript, but we were genuinely interested in the models' ability to navigate the chemical space when the exploration is "turned on" with multinomial sampling. PepINVENT generated ~1400 canonically different amino acids from ~1600 unique generated strings, on average per test peptide (Figure 7.A). This implies that the generator proposed at least one chemically distinct residue every time it sampled instead of reordering a set of amino acids. This massively expanded the traditional natural amino acid space and the combinatorial space when chaining these amino acids into peptides. We further characterized generated amino acids into either natural, training set NNAAs, or novel NNAAs. This analysis helped verify that chemically distinct monomers

were not solely from memorization of training residues. We found that the model can indeed come up with new designs with novel sidechains and without completely relying on simple stereochemical variations. Furthermore, novel NNAAs come from the learned monomer chemical space defined by the training set (Figure 7.B). It also reproduced both natural and non-natural amino acids from the training set.

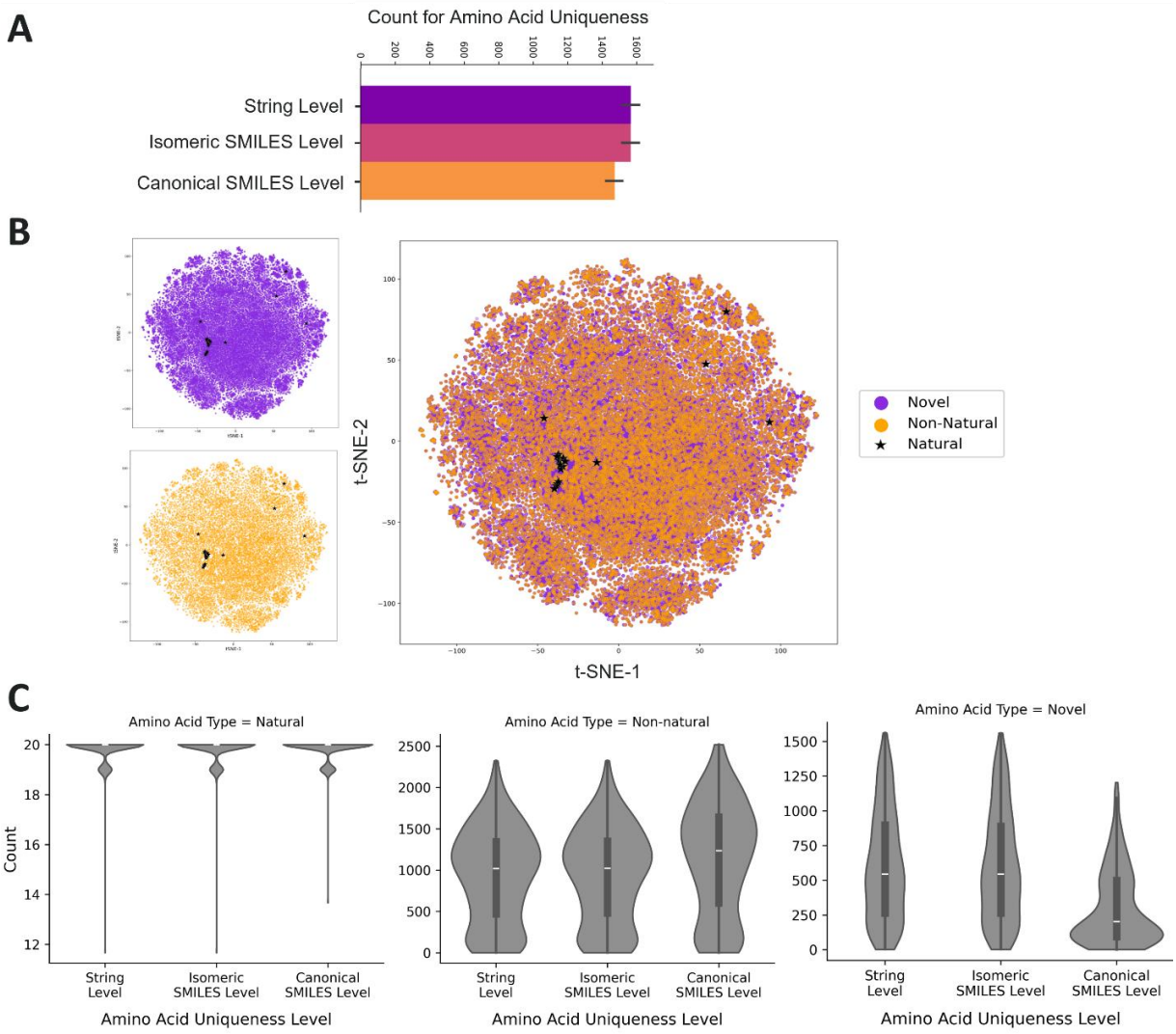


Figure 7. Generative capabilities of the generator

The following plots are created using the multinomial sampling that promotes randomness during generation to explore the generative capabilities of the models. (A) Average number of unique amino acids of the generated batch per test peptide with the standard deviation as error bars. (B) Chemical space visualization of natural amino acids, NNAAs from the training set (labeled as "Non-natural"), and novel NNAAs using T-distributed Stochastic Neighbor Embedding (t-SNE) conducted on Morgan fingerprints with radius=3, using chirality. The diversity of generated building blocks is from the learned chemical space of learned building blocks (C) Categorization of generated amino acids into natural, training NNAAs (labeled as "Non-natural"), and novel NNAAs is plotted with their corresponding distributions across generation for on test peptides. Plots (A) and (C) contain three levels of uniqueness:

i) String-level to evaluate generating unique SMILES strings, ii) Isomeric SMILES-level for unique molecules, including the stereochemical differences, and iii) canonical SMILES-level for unique molecules without the stereochemical differences. The figure was adapted from Paper I[78] under a Creative Commons Attribution 3.0 Unported Licence[81].

This indicated that the model also revisited known monomers, providing an opportunity to tune down the explored space when desired (Figure 7.C).

An additional 40 test peptides were used to assess whether the model understood the overall peptide context. When a single cyclization anchor was present, the model completed the ring structures correctly by generating an amino acid that carried the ring-closure. This showed that the model not only generated the correct number of amino acids but captured the long-range dependencies of the peptide without compromising validity (Table 1).

These results show that we achieved our aim to establish a generator that learns the peptide space at atomic resolution, recognizes the overall topology, and explores novelty and diversity at the amino acid level. Once we confirmed that the generator generalizes to the intended peptide language and space, the next step was to use it to design peptides with optimized properties.

Guiding the generator for peptide optimization

The generative model provides the base with defining the chemical space we want to delve into. RL fine-tunes the exploration to design goals targeted for a query peptide. Before going into the related work, a short introduction on how the scoring works would help understand the system. RL uses one or multiple scoring components to score the batch of peptides generated in each learning step. These scoring components can be simple physicochemical descriptors such as molecular weight, number of hydrogen bond donors, or a binary score for whether a substructure exists or not. It can also employ more complex scoring components like predictive models or structure-based workflows such as docking.[54] Scores from these components are transformed into [0, 1] range, if not already in this range. This transformation describes the objective of the score. Transformed scores are then aggregated into a final score, or reward, through a scoring function such as weighted arithmetic or geometric mean. In RL-guided generation, the model learns to propose molecules that would be rewarded with higher scores. Scoring components, their transformations, and the scoring function are configurable by the user. In *Paper I*, RL was applied following REINVENT's framework.[53]

Before applying RL to a true MPO task in peptide design, it is important to check whether RL can steer the generation toward a desired design space using simple optimization objectives. This should be accomplished without compromising the underlying performance of the generative model. In addition, investigating the flexibility of the generative process provides insight into the learning process and convergence behavior. To evaluate flexibility, the idea was to showcase

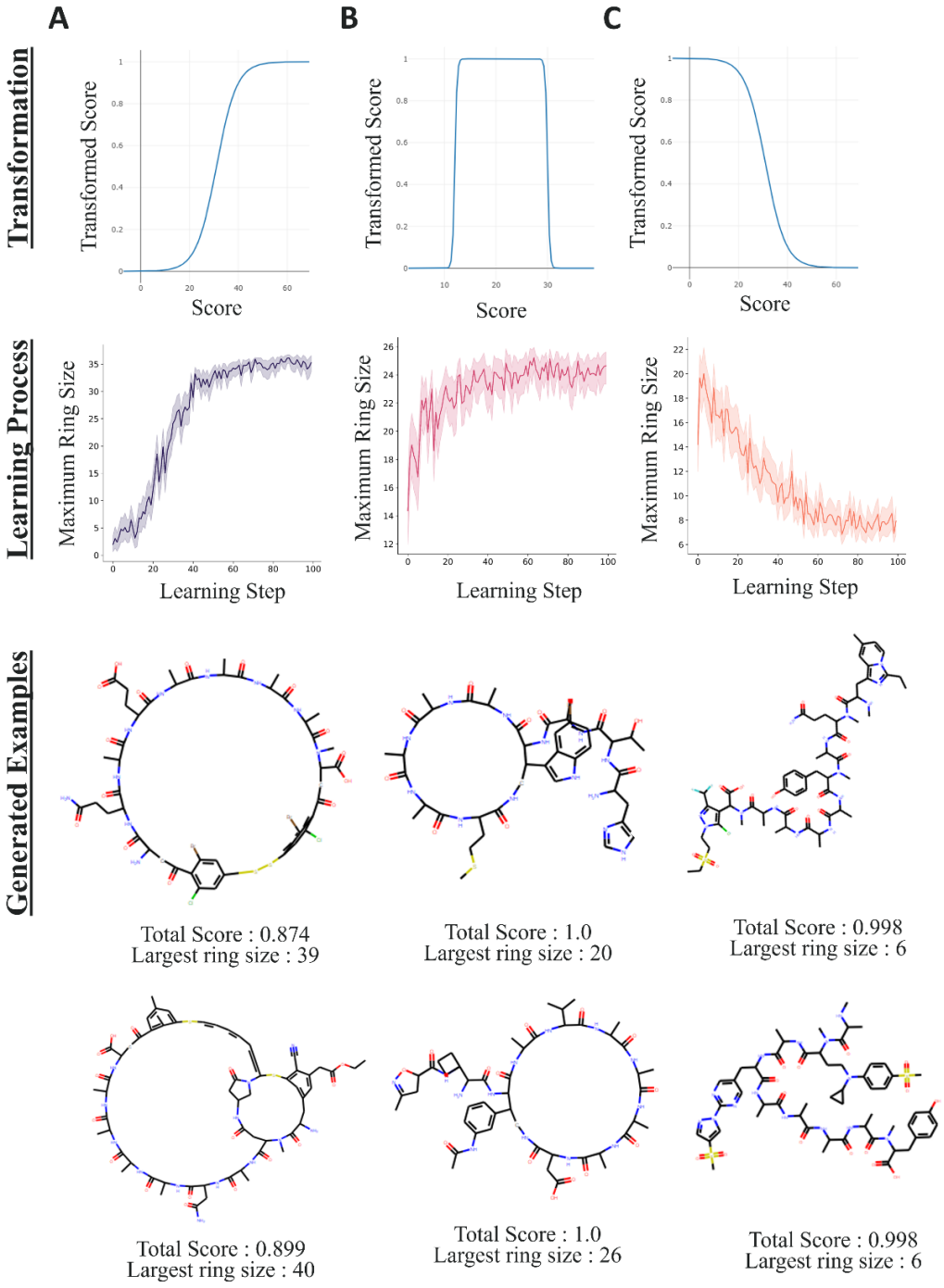


Figure 8. Constraining generation to a specific topology with RL

RL-steered generation to generate peptides with a specific topology was shown over three objectives: (A) maximizing the size of the largest ring by transforming the scores, or the computed ring size, with a sigmoid function set between [0, 60 atoms] to generate disulfide bridged peptides, (B) limiting the size of the largest ring to a range that favors head-to-tail or sidechain-to-tail cyclized molecules. This is achieved by transforming the scores with a double sigmoid function set between [0, 30 atoms], (C) minimizing the size of the largest ring by transforming the scores with a reverse sigmoid function set

between [0, 60 atoms] to generate linear peptides. In each case, the transformation that describes the objective, the learning process when these objectives are used in generative design, and randomly selected peptides with high MPO scores are provided to illustrate RL-guided optimization. The example peptides are labeled with their MPO score, labeled as "Total score" and with the largest ring size. The figure was adapted from Paper I[78] under a Creative Commons Attribution 3.0 Unported Licence[81].

how the generation can be constrained to a specific peptide topology, using a straightforward RDKit-computed property.[87] Motivated by the convention that macrocycles contain rings with at least 12 atoms, the number of atoms in the largest ring was used as our scoring component to influence molecular topology. Next, we defined three objectives (Figure 8):

1. Maximizing the maximum ring size: To generate disulfide bridged peptides where the cycle encompasses all backbone and the sidechain atoms of the amino acids with cyclization anchors (Figure 8.A).
2. Keeping the maximum ring size between 12 and 30: To generate head-to-tail or sidechain-to-tail cyclized peptides. 12 is the minimum for macrocyclic molecules, and 30 is the maximum, rounded up from 27 atoms needed to make a head-to-tail cyclized 9-mer peptide representing an arbitrary case chosen for this experiment (Figure 8.B).
3. Minimizing the maximum ring size: To generate linear peptides (Figure 8.C).

We ran separate RL-guided generative runs with these objectives to fill four positions of a 9-mer linear peptide as our query. As expected, PepINVENT started generating disulfide bridged peptides to maximize our scoring component and linear peptides when minimizing it. Additionally, the model rapidly converged to the desired topological space and started exploiting designs with the expected topology in under 50 learning steps. During these runs, the chemical validity was preserved, above 90% for exploration, and above 95% for exploitation.[78]

Finally, a full MPO task was set up to design a soluble and permeable cyclic peptide. The application case included a linear peptide identified as the pharmacophore from an antibody interacting with a target protein. This protein plays a key role in the viral replication pathway of human immunodeficiency virus and therefore has been targeted as antiviral therapy. Previous works have identified four amino acids that can be modified to improve the permeability of the linear peptide through a series of modifications while preserving its solubility.[26,88,89] In our work, we pursued the same goal, generating modified versions of this peptide to make it soluble, permeable, and cyclic. We implemented six scoring components or design filters into PepINVENT's framework for this purpose:

1. Maximum ring size: Using the learnings from the flexible topology generation, the generation was constrained to a head-to-tail or sidechain-to-tail cyclized peptide for the MPO task.
2. CAMSOL-PTM[90]: Implemented as the solubility calculator. This rule-based scoring model can digest any NNAA to predict the solubility of peptides.

3. Permeability predictor: Integrated as a baseline ML model to predict the probability of being a permeable cyclic peptide. This predictive model also served as the baseline model for XGBoost algorithm in *Paper II*.
4. Custom Alerts[53]: Used to penalize the generation of substructures that are not desired due to toxicity or high reactivity in drug discovery.
5. Diversity Filter (DF)[53]: Used to penalize repeated generation of similar scaffolds during sampling. DF computes molecular similarity to penalize highly similar peptides, therefore promoting diversity throughout the learning process. Also, it prevents mode collapse during the exploitation phase where the generator outputs the same high-reward sequence constantly.

Finding a soluble and permeable peptide is one of the most essential challenges in peptide therapeutics. Permeable peptides tend to be very hydrophobic even though there is a sweet spot known to exist that satisfied both qualities.[91] In this application case, PepINVENT demonstrated that it can explore the large chemical space and identify designs in that sweet spot. Permeability and solubility scores increased during exploration and plateaued in exploitation among other MPO components (Figure 9.A-F). Although the average predicted permeability across the batch fluctuated around the probability of 40-45%, individual designs exhibited significantly higher probabilities of being permeable (Figure 9.C). PepINVENT balanced these components along with cyclic topology and diverse designs that do not contain unwanted substructures (Figure 9.A, G). These results confirmed PepINVENT's robust performance on MPO tasks while maintaining high validity (Figure 9.E).

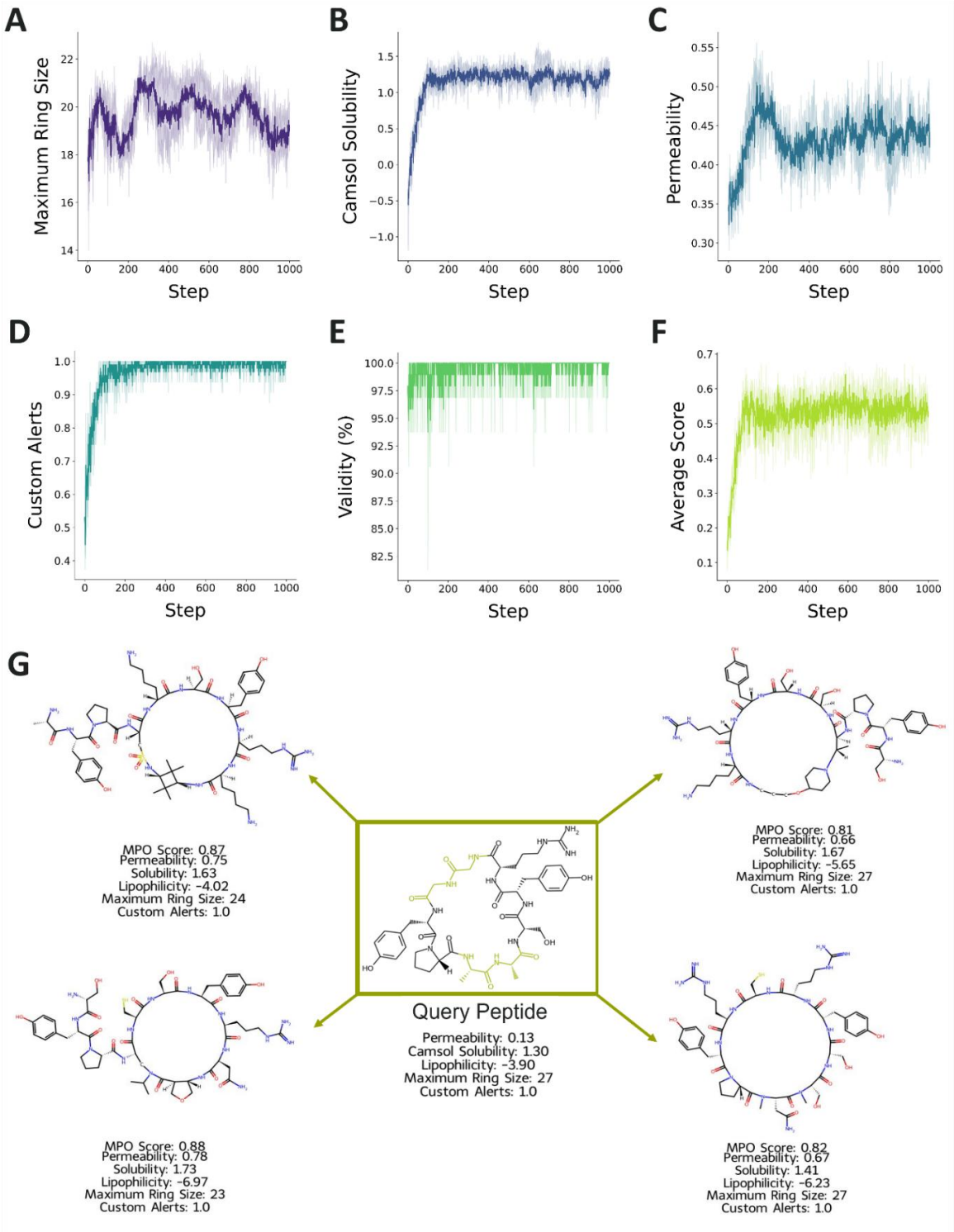


Figure 9. Learning process and generated examples for MPO aiming for permeable and soluble cyclic peptides

An MPO task to optimize a 9-mer peptide for permeable, soluble and cyclic topology was applied to demonstrate how PepINVENT tackles complex design challenges. The learning process is shown for (A)

the size of the largest ring, (B) solubility scorer, (C) passive permeability predicted by a classifier, (D) custom alerts that penalizes undesirable substructures, (E) percentage of validity for the batch generated in the step, and (F) the average of the aggregated scores, or rewards for the batches, over RL-guided learning steps. The generative run was conducted in triplicate with the provided plots showing the average of these runs. Solubility, largest ring size, and custom alerts were learned after 400 epochs. This implies that the chemical space gets constrained to designs that satisfy these properties with high chemical validity of generated strings. However, permeability remained harder to learn with scores varied around an average likelihood of about 40–45%. Regardless, PepINVENT still successfully suggested (G) cyclic peptides with high MPO scores, predicted to be permeable and soluble for the query 9-mer peptide. The figure was adapted from Paper I[78] under a Creative Commons Attribution 3.0 Unported Licence[81].

Outcome

PepINVENT represents one of the first chemistry-aware AI-driven peptide design tools that generate amino acids at atomic-level. The generator has learned peptide chemistry over diverse peptide topologies and building block compositions. It can access novel NNAAs, and modifications over various topologies that sequence-based generators cannot reach. Extending the chemical space beyond natural amino acids or enumerated spaces, our generator can effectively explore a vast chemical space. This was achieved by establishing CHUCKLES as a peptide language that uses molecular chemistry and monomer order. The generative model showed strong performance on validity, novelty, uniqueness, and diversity. However, the limitations on much longer peptides, distinct topologies such as bicycles or stapled peptides still need to be understood for further development.

When coupled with RL, generation is steered to optimize peptides by balancing trade-offs between scoring components of MPO while maintaining high validity. An inherent limitation to the system is the number of positions submitted for modification. More modifications enable more significant shifts toward optimized properties. It can be unfair to mask only one or two amino acids in long peptides and expect the generated ones to achieve multi-fold improvements in properties. Therefore, the design goal must be structured in a reasonable way.

Although generative models are promising avenues, there is still room for improvement. Generative runs can offer tens to thousands of designs, if not more, satisfying an MPO task of interest. Selecting or prioritizing peptide designs from the generative output still requires good post-processing. One thing that was not included in this work was constraining the chemical space to generate synthetically feasible amino acids and peptides. An example of this can be seen in the infeasible heterocycle incorporation to the backbone (Figure 9.G). This was addressed with a post-processing evaluation for synthesizability, explored in *Paper IV*. Regardless, complex peptide optimization tasks can be handled by PepINVENT meeting the design goals.

Since the model is not restricted to a particular task or target, it is applicable to wide range of projects. It could easily be extended with project-specific scoring schemes. Integrated into real-life applications, it can shorten the exhaustive DMTA cycle for peptide lead optimization.

Assessing Peptide Designs

This section explores how to evaluate peptide designs for prioritization in the optimization setting. Virtual designs can be selected as candidates by scoring and ranking them by calculating properties, predictive models, and/or more complex methods for structure-based approaches. Here, we explore how to predict two major characteristics needed for peptides as therapeutic agents: synthesizability and cell membrane permeability. Together, these elements contribute to a closed loop where peptide generation is driven by reliable property prediction, and synthetic feasibility assessment providing a filter for candidates to improve DMTA efficiency.

Predicting permeability with confidence

When predictive models are employed for design evaluation, scoring these designs accurately and reliably is crucial for a successful transition from *in silico* to wet lab. Predictive models that are strong predictors of intricate properties are central for the acceleration promised by AI. A predictor with high generalizability enables its applicability in practice, when we want to predict on new molecules of interest. In a generative setting, it unlocks true exploration of the chemical space where predictions are confidently made on diverse designs. Supplying algorithms with large, highly diverse data can provide a generalizability to a certain extent; however, such extensive datasets might not be available for every property or modality.[92,93] Models trained with smaller or not very diverse training data generally get evaluated on test datasets split from already small training set. This leads to an overestimation of generalizability. Publicly available permeability data for peptides is one such case where generalizability can easily be overestimated.

Peptide permeability is often treated as a binary classification for cell-penetrating or non-penetrating. Different assays examining distinct transport pathways have been generally disregarded due to lack of annotation.[94] Assay conditions as well as presence of modifications or cargoes were not utilized during data preprocessing for ML purposes.[95–97] In *Paper II* and *III*, I focus on predicting passive permeability of conjugate-free peptides. More specifically, the focus is on building peptide-specific predictive models with uncertainty quantification to address challenges presented by publicly available data. In this section, I first established the best practices for model building in *Paper II.* Later, I demonstrated how to guide generative design using these uncertainty-calibrated predictors in *Paper III.* Together, these works represent the first application of uncertainty in peptide therapeutics.

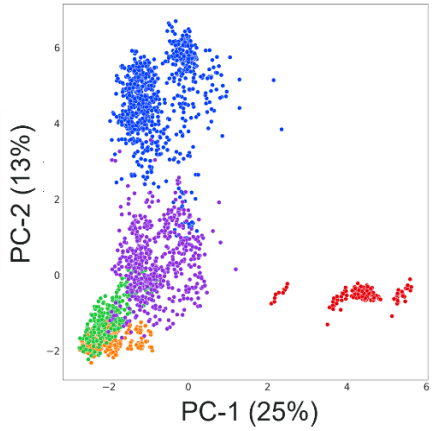
Building Permeability Prediction with Uncertainty Quantification (*Paper II*)

Cell membrane permeation is essential for peptides to access the intracellular targets as standalone drugs. Additionally, permeable peptides have been be leveraged as drug delivery agents for other modalities including small molecules, antibodies and nucleic acids.[1,14] While larger and bulky peptides traverse the cell membrane through active transport, smaller peptides can also enter passively, presenting attractive opportunities for oral bioavailability.[11] As discussed in the *Introduction*, cyclic peptides present a structural advantage over linear counterparts for permeability with reduced entropic penalty when traversing the membrane.[26] Passive diffusion has been extensively studied, with emphasis on establishing the structure-activity relationship (SAR) between chameleonicity and permeability.[98]

Passive permeability of cyclic peptides can be measured by a cheap non-cell-based assay called Parallel Artificial Membrane Permeability Assay (PAMPA)[26,40]. The assay measures the concentration difference between two compartments, separated by an artificial membrane, after peptides are placed in one compartment and incubated. A peptide with a logarithmic experimental permeability above -6 is deemed permeable.[38,40,99] Due to its simplicity, PAMPA is widely used for passive permeability studies and therefore has the greatest amount of public data available. The data provided across studies were recently collated in the Cyclic Peptide Membrane Permeability Database (CycPeptMPDB).[38] Many studies collated in the CycPeptMPDB, used positional scanning to introduce point mutations down the sequence of a wild-type non-permeable cyclic peptide. They conclude that minimal changes such as single stereochemical inversion or backbone *N*-methylation could make peptides permeable.[26] Consequently, the reported synthetic strategies are generally paired with successful outcomes, causing imbalance in the database toward permeable peptides. [26,38] While current design strategies for permeability often rely on trial-and-error, the complexity of SAR can be understood by predictive models.

From CycPeptMPDB, 6,876 cyclic peptides were extracted across 35 publications after data preprocessing. Many of these studies examined a set of highly similar analogs of a wild-type peptide, introducing minor modifications that can significantly enhance the peptide's permeability.[38] Here, data composition determined the scope of the predictive model as it needs to: i) detect minute molecular differences, and ii) classify peptide permeability as a binary task to mitigate inter-assay variability across laboratories.

Any prediction from a predictive model is associated with a degree of uncertainty. Combining data from different sources demands a strong applicability domain assessment to expose gaps and limits in the covered chemical space. Understanding the applicability domain informs us when predictions should not be trusted. The applicability domain in this work is assessed by CP,



Pre-processed Data	Number of Peptides	Data Size (%)	Permeable class (%)
Source: 2016 Furukawa	668	9.7 %	56.4 %
Source: 2013 Chugai	878	12.8 %	89.8 %
Source: 2021 Kelly	1518	22.1 %	61.3 %
Source: 2020 Townsend	3086	44.9 %	70.0 %
Remaining sources	726	10.6 %	53.9 %

Figure 10. The chemical space of the multi-source dataset for peptide permeability.
Chemical space of the multi-source dataset was visualized with principal component analysis (PCA) on Morgan Fingerprints with radius=3, using chirality. Two-dimensional principal component analysis collectively explained 38% of the variance in the dataset. Four sources contributing the largest amount of data to the combined dataset were used for leave-one-source-out evaluation. They were colored as: 2016_Furukawa [100] in orange, 2013_Chugai[101] in blue, 2021_Kelly[102] in red, and 2020_Townsend[103] in green. The number of peptides from these sources as well as permeability class balance is tabulated, with remaining data (in purple). These sources represent significant portions in the dataset representing distinct domains in the chemical space. The figure was adapted from Paper II[104] under a Creative Commons Attribution 3.0 Unported Licence[81].

with methodology described in the *Introduction*. Best practices for building peptide predictive models were explored by investigating how to balance the trade-off between conformal validity and efficiency at a given confidence level. Also, using Mondrian ICP enables each binary class to be calibrated independently. It provides stronger insights on applicability domain for imbalanced classes.

In *Paper II*, the aim was to systematically assess how to build an uncertainty-aware permeability classifier for cyclic peptides. This also meant given the state of available public data, the model had to be generalizable to new and unseen peptides. The study followed a leave-one-source-out evaluation repeated for various ML algorithms over a series of experiments.[100–103] Data from the four publications with the largest datasets, out of 35 publications, were each left out as holdout test sets. Individually, these sources correspond to approximately 10-45% of the entire dataset (Figure 10). The remaining data was split into 80% proper training and 20% calibration sets. Four baseline ML models were trained on each dataset, totaling 16 models to evaluate. The algorithms include a tree-based method, Random Forest (RF), a kernel-based method, Support Vector Machine (SVM), and two gradient boosting methods of Extreme Gradient Boosting (XGBoost), and Light Gradient Boosting Machine (LightGBM)[105]. 2048-bit chirality-aware Morgan Fingerprints[106] with a radius of 4 was used as the peptide descriptor. Building models

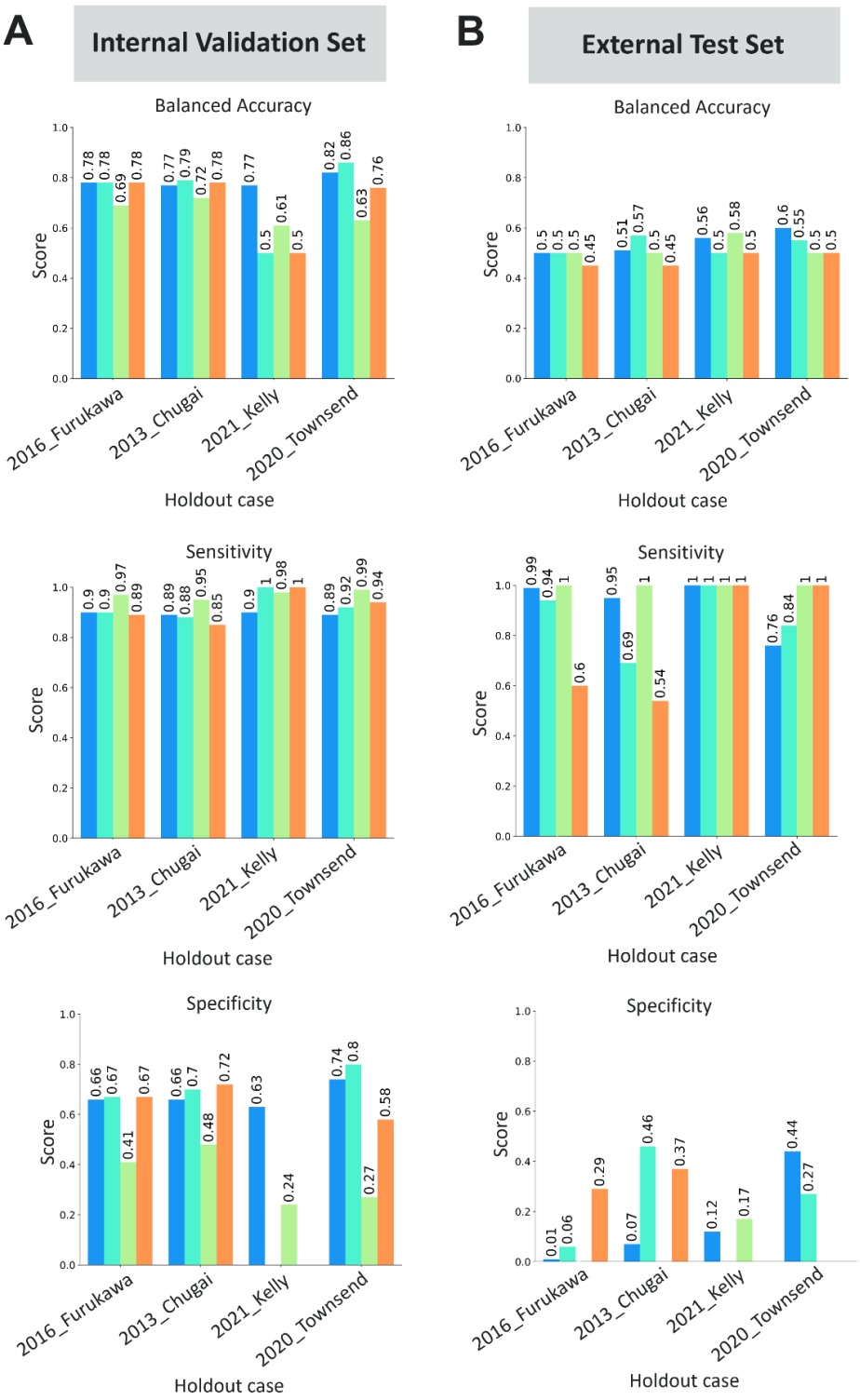


Figure 11. Model performance metrics on test sets: (A) internal validation set and (B) external test set.

(A) Internal validation set represents a test set on the chemical space predictors were trained on, also known as the source domain. (B) External validation set represents a test set on the target domain, or

the unseen chemical space. Model performance metrics on balanced accuracy, sensitivity, and specificity are shown for each hold out case and for the four baseline ML models. Models for each holdout case were trained on the same training data and tested on either internal validation or external test set (the holdout test set). The models were color-coded as: RF in blue, XGBoost in turquoise, LightGBM in green, and SVM in orange. Good model performance on the internal validation dropped for the external test set. Specificity was generally lower than sensitivity due to the imbalance of the training data towards more permeable peptides. In unseen domains, model predictions fell to the dominant class, leading to many false positives. The figure was adapted from Paper II[104] under a Creative Commons Attribution 3.0 Unported Licence[81].

on fingerprints that retain chirality was expected to capture small modifications such as backbone modifications through substructure patterns. After the models were trained on the training set, their performance was evaluated on two test sets before calibration. This was done to illustrate how chemical space influences predictive accuracy. The test sets corresponded to their respective calibration set, and holdout test set. In each case, the calibration set was split in a stratified manner from the same chemical space as the model trained on. This set represented an internal validation set while the holdout test set represented an external test set. The external test set was considered to be not necessarily similar to the learned chemical space. It was evident that models demonstrated strong performance, generally above 0.7 for balanced accuracy, on their respective internal validation sets (Figure 11.A). Sensitivity was high around 0.9, while specificity was lower around 0.6 on the internal validation sets. This indicated overprediction of the majority class, or the "Permeable" class (Figure 11.B). In contrast, these models failed to generalize to dissimilar instances from holdout test sets. Balanced accuracy dropped significantly and the gap between sensitivity and specificity increased (Figure 11). This highlighted the non-exchangeability of internal and external test sets. This did not mean that the models overfit to the training data. But the models could not be expected to provide reliable predictions for unseen peptides in the test set since calibration and holdout sets were not exchangeable.

As a solution, to extend the applicability domain of the models to our holdout test sets, three calibration strategies were compared. The comparison entailed calibrating the trained models with various calibration sets with distinct compositions. Following calibration, they were evaluated on the validity and efficiency of their predictions on the holdout test sets. The predictions were obtained at a confidence level of 80%. The three calibration strategies included calibration sets composed of:

- **Standard calibration set (split from the training set distribution):** a held-out subset from the learned chemical space, or from the source domain, was used to calibrate nonconformity scores. This set was the same set used as internal validation and represented the baseline as the standard approach for CP.
- **Standard calibration set and 20% of holdout test set:** standard calibration set was augmented with a fraction of the holdout data. This was expected to restore the exchangeability between the final calibration set and the target domain. This approach

would increase the similarity between the instances used to compute nonconformity scores and test set of interest.

- **20% of the holdout test set:** a fraction of the holdout data used as the calibration set, calibrating models only on their respective target domains. This approach removes the standard calibration set from the experiment.

Benchmarking these calibration strategies revealed how exchangeability between calibration and test sets drives the efficiency of predictive models. When the models were calibrated with the standard calibration set, thigh validity but low efficiency were observed for predictions on the holdout test set at 80% confidence (Figure 12). This indicated that the models were struggling in differentiating between permeable and non-permeable classes. The confusion led to assigning mostly the "Both" class.

Augmenting the calibration set with data from the target domain seemed like a plausible strategy since it could inform the calibration on the new instances. Although the efficiency of the predictions was improved, minor sacrifices on validity were observed over the 16 trained and calibrated predictive models (Figure 12). Therefore, more single-label predictions (permeable or non-permeable) did not correspond to the correct class assignment. Stable or reduced validity was attributed to the influence of the standard calibration set, dominating the final calibration set.

Finally, the best trade-off balance was presented when the calibration set only contained holdout test set instances. Calibrating the models solely on target domains showed higher efficiency and stable validity, in some cases even higher validity compared to the other two strategies. Validity was above 80% for all models indicating that the model performances also matched the confidence demanded (Figure 12). The conclusion was that full exchangeability was required to effectively utilize the model on new target domains. By calibrating the permeability model on an unseen domain, the applicability domain was extended without the need to retrain the underlying algorithm. Finally, data splitting strategies were investigated for how to construct cross-validation folds. Splitting the data based on the sources they were collated from or on similar scaffolds were concluded to facilitate robust model performance with high efficiency and validity.[104]

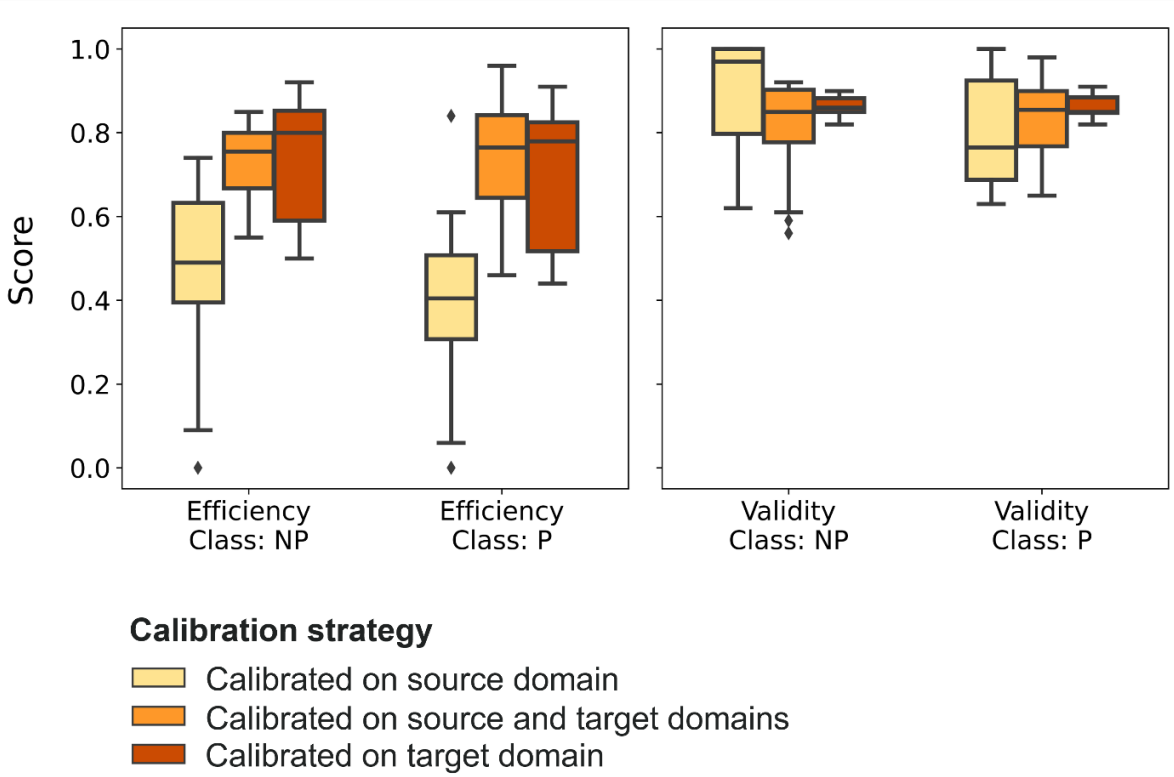


Figure 12. Box plots showing the distributions of efficiency and validity of the models with different calibration strategies, when predicting out-of-domain peptides

The difference in efficiency and validity for each permeability class of "Permeable" (P) and "Non-permeable" (NP) are shown across the models when predicting on hold out test sets. These metrics were plotted together for all predictive models built in this study to facilitate generalization of the best performing strategy. The three strategies used were: i) calibrating on source domain (yellow), ii) calibrating on both source and target domains (orange), and iii) calibrating only on the target domain (red). The latter strategy performed the best with validity at or above the demanded confidence of 80% while the efficiency was high for both classes. Therefore, the models calibrated on target domains produced more valid and efficient predictions, in other words, more reliable results. The figure was adapted from Paper II[104] *under a Creative Commons Attribution 3.0 Unported Licence*[81].

Outcome

This work laid out a methodology to maximize the utility of peptide property predictors trained on multi-source data in their true applicability boundaries. CP provides a practical approach to restore the reliability of predictions on never-before-seen peptides by calibrating the models with target domains. Since calibrating only requires computing nonconformity scores, it is more efficient and faster than retraining models. In practice, domain adaptation can be achieved by calibrating the predictors with a small representative set of the new peptides characterized experimentally. The number of samples required for calibration is typically smaller than effective retraining would need. Retraining when the new domain is underrepresented in the overall chemical space risks poor learning of the domain of interest. In a general sense, any

predictive task can be deployed and utilized in drug discovery projects reliably using uncertainty quantification to yield valid and efficient predictions. In this study, we were interested in cyclic peptide permeability. We did not consider other factors important for permeability such as peptide length and amino acid composition. A further investigation into the limits of this domain adaptation would therefore be a suitable next step. Considering significantly out of distribution domains such as larger peptides, linear peptides, and even macrocycles, could help identify the limitations of the methodology. When calibrating does not benefit in producing valid and efficient predictions anymore, it could help identify a threshold for domain dissimilarity. The threshold then can establish when to rely on calibrating and when to consider training domain-specific models or using other methods like transfer learning.

Integrating Reliable Scoring into Generative Design (*Paper III*)

Calibrated property predictors inform users how reliable and efficient model predictions are, in other words, when to take seriously or to stop trusting the model output. This is important in practice to improve the hit rates when synthesizing and testing candidates predicted to have the desired property. It is equally compelling in the generative modelling context. In PepINVENT, the model proposes amino acids to substitute one or more predefined positions in a peptide. RL fine-tunes the generation toward peptide designs with desired properties, fulfilling MPO goals.[78] Any generative design campaign is as successful as the sum of the accuracies of its scoring components, participating in the MPO.[53] The true impact of the generative methods can only be achieved when computed scores for proposed designs align with wet lab outcomes.[107]

Considering a binary classifier for permeability, permeable/non-permeable predictions reduce to a coin toss as peptides get further away from the training set of the predictive model, as demonstrated in *Paper II*. When such a predictor is used as a scoring component in the RL loop, the resulting scores can become random binary outcomes as the generator explores novel chemistry. As proposed peptides diversify with never-before-seen building blocks, distinct topologies, and sequence lengths, the predictor is expected to drift outside its applicability domain. Under these circumstances, the models cannot be expected to provide reliable predictions.[108] Therefore, determining how to score an objective for a property becomes as crucial as the choice of the property to be learned in RL. Commonly, we employ multiple predictive models to score complex properties that cannot be directly computed, such as bioactivity, stability, and permeability.[45] In such cases, the uncertainty from the predictive models becomes aggregated and too high to propagate the optimal designs to wet lab with high concordance.[53]

In *Paper III*, we benchmarked how to propagate model confidence into the scoring scheme in the RL feedback loop during generative design. We employed the generative model, PepINVENT from *Paper I*, and the uncertainty-aware permeability classifier, XGBoost, using the methodology described in *Paper II*.[78,104] By incorporating uncertainty, the predictor could report efficient and valid scores, improving the overall reliability of the proposed designs from PepINVENT (Figure 13). Additionally, we explore whether we can converge to the intended design space faster while navigating the peptide chemical space by using predictive models more effectively.

When a classification model is included in an MPO scenario, the conventional approach is to use the model probabilities.[52,109,110] These probabilities are estimated on every test instance for each class label. They are generally inferred from the predictive approach such as distance to hyperplane for SVM, the averaged probabilities across trees based on leaf class fraction for RF.[111] Using the probabilities of the desired class provides a continuous scoring rather than a binary one, when penalizing or rewarding designs in the RL loop.[53]

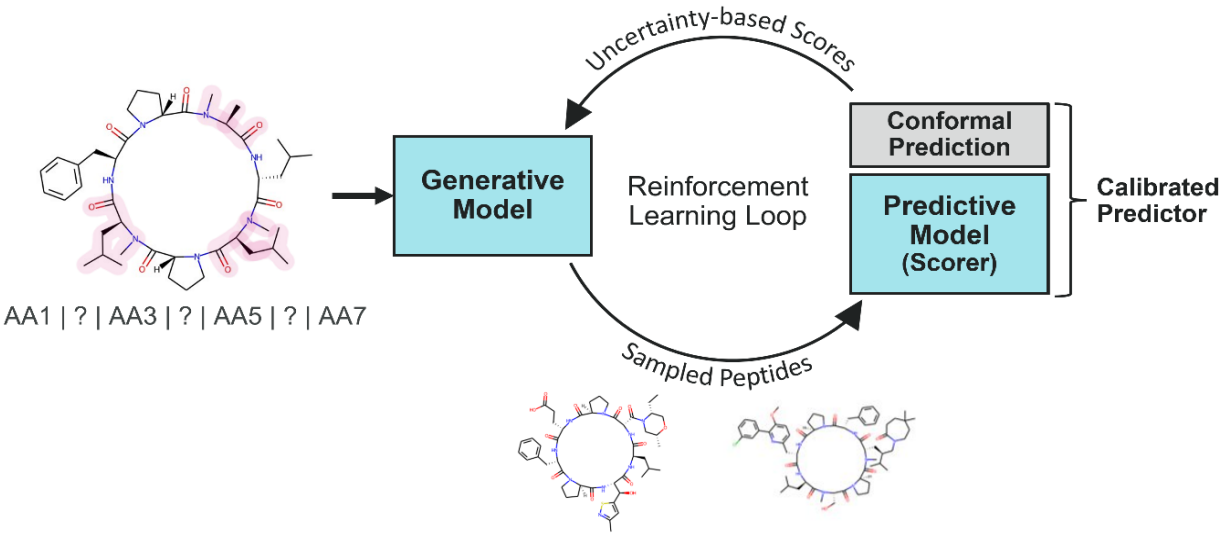


Figure 13. Schematic representation of using uncertainty-based scores from calibrated predictors in the RL loop for reliable scoring in peptide optimization.

Having a continuous scoring scale allows the generator to learn from chemistry nuances in proposed peptides. In contrast, binary scoring might prove harder to learn, thus slowing down the learning. This can be observed especially for complex properties such as permeability. In these cases, the generator should first produce enough positive samples to learn the distinction between rewarded and penalized designs.[53] With these in mind, we compared reward strategies for scoring predictive uncertainty against the standard baseline approach. The raw model predictions of the classifier served as the baseline reward strategy.

The permeability classifier used, exhibited a balanced accuracy of 78% and trained on cyclic peptides dataset from CycPeptMPDB. 150 peptides from Baker *et al.*[98] were employed as the test set. These instances were not included in the predictor's proper training or calibration sets as well as the generator's training data. The test set peptides were 6-, 7-, or 10-mer cyclic peptides and matched the training data distribution of the predictor. This ensured that the queries were within the applicable length range of the predictor.[108] Also, they were composed of natural amino acids or commonly used analogs as modified as stereoisomers or with *N*-methylated residues. The monomers constituted a subset of monomers and thus the chemical structures the predictor was trained on.[98] One to four positions of the test peptides were randomly masked and input to PepINVENT, using one of the reward strategies configured to generate permeable peptides. Learning over 350 epochs was monitored for the propagated reward, conformal p-values of P0 and P1 (representing non-permeable and permeable classes, respectively). The impact of each reward strategy on learning was evaluated for the batch of 150 peptides (Figure 14). Visualizing generation processes of the entire test population enabled investigation of the influence of the reward strategy across diverse query inputs.

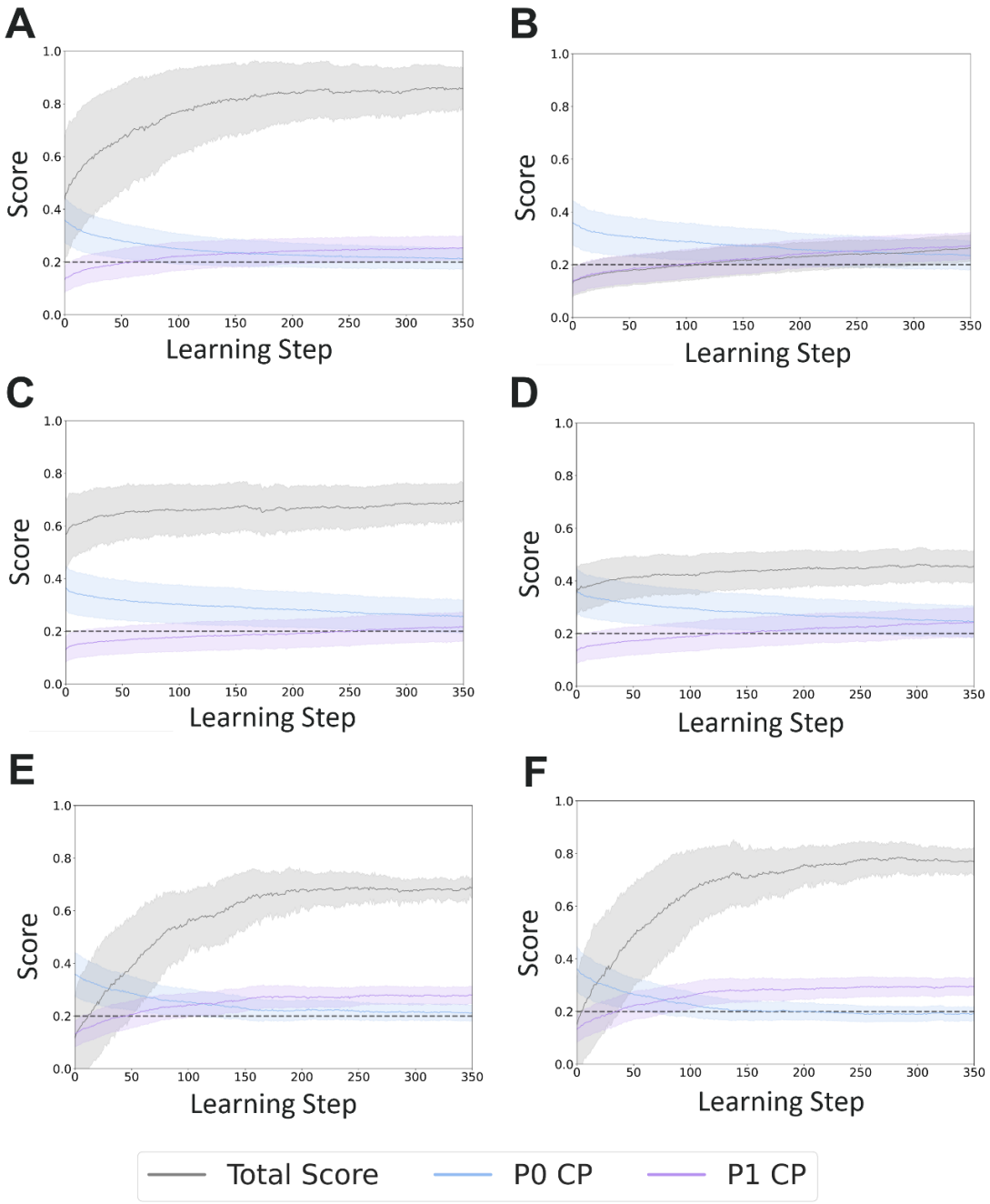


Figure 14. RL process over the strategies using predictive modelling as learning objective to steer the generation to permeable peptides

The learning objectives entailed A) maximizing baseline model using the model probability for permeable class, followed by uncertainty-based scores using conformal p-values of P0 and P1 as maximizing: B) P1, C) 1–P0, D) P1 – P0, E) Harsh Reward Function that formulates conformal efficiency, and F) Soft Reward Function that adds partial reward to Harsh function. The average and span of the aggregated scores (gray), P0 (blue), P1 (purple) for the generated batches across independent runs on 150 test peptides over the RL iterations are shown. The horizontal dashed line indicates the significance level of

0.2, or the demanded confidence of 80% where P1 above and P0 below this line indicates single-label "Permeable" class prediction for the samples. The only scoring strategy that satisfied this under 350 steps was the Soft reward. Therefore, it was determined as the best strategy for employing predictive models to enable reliable generative design for peptide permeability. The figure was adapted from Paper III.

When using raw model probabilities in the RL loop, the generation can be steered to high predicted probability for the desired class label (Figure 14.A). However, the conformal p-values on the proposed designs showed a clear failure to deliver efficient, or single-label, predictions at the 80% confidence level. This demonstrated that the exchangeability assumption between the model's calibration set and the generated peptides did not hold. Employing CP-based scoring components instead of raw predictive probabilities was hypothesized to improve the reliability of predictions.

Across the reward strategies benchmarked, relying on either p-value alone, also did not yield efficient predictions, nor did a simple combination of P0 and P1 (Figure 14.B-D). A straightforward relation between them or isolating a single p-value for scoring violated the principles of Mondrian ICP where each class is calibrated independently but still requires a collective decision-making of both classes in inference. Instead of these approaches, we converged to using conformal efficiency itself as the scoring component. Conformal efficiency was previously described as a metric to assess the size of the applicability domain.[112] This helped decouple the trade-off between efficiency and validity from reward, enabled using efficiency as reward. Since validity cannot be scored since the generated designs are unlabeled, our methodology would not be applicable. Conformal efficiency in our work was formulated as:

- Harsh reward function: A binary function that grants the maximum score of 1, for an efficient prediction that leads to a single label assignment for the "Permeable" class. In any other case, there would be no reward. 0.2 defines the significance level, or 80% confidence level.

$$Harsh\ Reward = \begin{cases} 0 & P0 > 0.2 \vee P1 < 0.2 \\ 1 & P0 \leq 0.2 \wedge P1 \geq 0.2 \end{cases}$$

- Soft reward function: A piecewise reward function that applies less stringent penalties, giving a partial reward when either of the conformal p-values meets its respective objective. Soft function follows the harsh reward conditions for maximum or minimum scores without the partial reward.

$$Soft\ Reward = \begin{cases} 0 & P0 > 0.2 \vee P1 < 0.2 \\ 0.5 & P0 \leq 0.2 \vee P1 \geq 0.2 \\ 1 & P0 \leq 0.2 \wedge P1 \geq 0.2 \end{cases}$$

To compare these reward functions to the baseline reward, a transition point was defined. The transition point was the learning step at which the average P1 value across the generated designs first exceeds the average P0 value. The transition point was not directly related to conformal efficiency. However, it was used as a signal describing the early generation of desired peptides from the starting non-permeable predictions at the given confidence. It indicates the epoch when generation steers toward designs with high P1 and low P0 values. This corresponded to maximizing the likelihood of being permeable while minimizing the likelihood of being non-permeable. The transition points for the reward functions were 163, 110, and 77 for baseline, harsh, and soft rewards, respectively (Figure 14.E,F). This implied that scoring conformal efficiency provided an advantage for the reliability of the predictions compared to using the uncalibrated model.

Formulating conformal efficiency as a single scoring component reduces the optimization of two independent and continuous conditions into a discrete reward scheme. Introducing a partial reward in the soft function made convergence to the desired permeable peptide space much faster. We observed the conditions for conformal efficiency, $P0 \leq 0.2$ and $P1 \geq 0.2$, are satisfied after 200 epochs for the soft reward. For the harsh reward, the average P0 never reached below 0.2 over 350 epochs (Figure 14.F). Therefore, the impact of the soft reward held for most, if not all, instances of our test population while the harsh reward required longer time for exploration or could be input test case dependent. The performance variation based on the input molecule could also arise from randomly masking test peptides. Masking only a small number of positions is always a bottleneck for achieving the design goals as it leaves limited room for modification to shift the property profile. Also, retaining highly charged residues in the query can increase "Non-permeable" class predictions, considering that most passively permeable peptides are hydrophobic.[91]

On top of the robust learning, using the soft reward resulted in more valid peptides than the harsh reward (Wilcoxon Test p-value < 0.05) and a similar number to our baseline reward (Table 2). We computed the pairwise Tanimoto similarities between each test peptide and its top 500 scoring generated variants. The preliminary results showed that soft reward still enabled diverse exploration compared to the baseline. This indicated that applicability of domain-informed scoring did not necessarily constrain the diversity of the chemical space explored. However, this conclusion must be investigated further for peptides significantly dissimilar to the trained chemical space and for much longer learning processes.

Overall, we concluded that gradual steering toward conformal efficiency facilitates getting reliable design ideas when scoring with predictive models in RL. Utilizing CP preserves the generator's ability to produce valid and diverse peptides by not drastically constraining the high reward space to the training set examples. Although the current scoring scheme yields permeable peptides with higher confidence than the uncalibrated model, input-dependent performance

variations could be reduced by introducing a more continuous scoring approach. An example approach could be where the proximity for both P0 and P1 values to the significance level is rewarded. This might produce more robust learning and faster convergence for conformally efficient designs. However, adding significant mathematical complexity could also overcomplicate learning and slow the generation of peptides that meet the design goals. Therefore, it would need exhaustive investigation for the complexity versus smoothness of the reward strategy. In *Paper III*, we provided proof of concept demonstrating the use of uncertainty-aware predictors that can guide optimization without sacrificing exploration of the relevant design space.

Table 2. Number of valid peptides generated by using the baseline reward scheme and scoring strategies using conformal efficiency

The number of chemically valid peptides generated from independent RL-guided generative runs conducted on 150 test cases is shown. Median and interquartile range are reported. In each run, RL loop iterated over 350 steps with each step producing 32 peptides. Validity over the test cases was compared with Wilcoxon Rank test to determine any changes in model performance when using uncertainty-based score versus the traditional class probability. (*) indicates a significant difference for $p<0.05$ compared to raw model predictions. The table was adapted from Paper II.

Scoring Component	Number of Valid Peptides (Median ± Interquartile range)
Raw Model Prediction (Probability of Permeable Class)	11110.5 ± 128.5
Conformal Efficiency (Harsh Reward)	11060.0 ± 386 *
Conformal Efficiency (Soft Reward)	11117.5 ± 230.5

Outcome

This work represents one of the earliest applications of uncertainty-based scoring in generative design to propose highly reliable peptide design ideas. Reliable functionalization of the predictive models is essential for a faster transition from *in silico* to wet lab, with high correlation between the computational and experimental results. Confident designs can help with prioritizing the evaluations on the right molecules. Specifically, permeable peptides can be designed by employing confident predictors in the generative process. These predictors can then facilitate prioritization of designs by learning the peptide space while bound by their applicability. This avoids overestimating their generalizability. The next step for this work is underway to validate the designs in wet lab and support the impact of this work with experimental outcomes.

During RL, we observed faster convergence without any drawbacks on diversity with the CP-based scoring in the preliminary results. However, there is a risk of generating the building blocks from the training set in longer RL runs. The most certain designs would be the training set ones seen previously by the model during training. In this study, we have not conducted an

analysis on building blocks. The generated building blocks could have limited diversity due to the pressure of producing designs with high certainty. However, employing diversity filters could overcome generating highly similar peptides. Going forward, it would be interesting to extend this methodology with other uncertainty quantification methods such as Platt scaling, additional algorithms, and diverse predictive tasks to establish uncertainty-guided RL as a generalizable approach in a survey-type study.

Predicting Synthesizability of NNAAs

Chemical synthesis of peptides has been widely available with the development of SPPS.[113] As explained in the *Introduction* section, the synthesizers can build a peptide of interest monomer-by-monomer, based on the input sequence, with the desired connectivity of amino acids.[113] Control over the connectivity is facilitated by selectively deprotecting the next residue's reactive group and coupling it to the growing chain.[114] This repeated process of selective amino acid deprotection and coupling yields the target peptide. Selectivity in this process is only achieved when different reactive groups in amino acids are protected by protection groups, deprotected via orthogonal methods.[36] Therefore, the success of SPPS is highly dependent on protecting each amino acid with an appropriate set of protection groups. This reduces the risk of creating by-products, composed of peptides with unintended monomer connectivity or sidechain reactions. This in turn leads to higher yield and purification complications.[36,115] SPPS provides an automated peptide synthesis, and it is still being developed toward improved reaction cycles with greener chemistry.[116,117] AI efforts in peptide synthesis have been centered around predicting the yields for peptides or its variants, focusing on the SPPS process itself. These approaches typically target predicting deprotection and coupling efficiency, or sequence optimization to reduce the composition of amino acids with high aggregation propensity.[118–120] Although there are only a few predictive models on this topic, these are important for prioritization of peptide designs. However, they come with their limitations. In some cases, they require synthesizing the amino acids, collecting analytical measurements on them, and then using this data as descriptors to make predictions on the peptide sequence. In other cases, they are limited to peptides composed of natural amino acids. Consequently, NNAAs fall outside the scope of these methods. We must first determine how to synthesize and protect them before the SPPS yield prediction with other tools becomes meaningful.

Synthesis Assistance for Non-Natural Amino Acids (*Paper IV*)

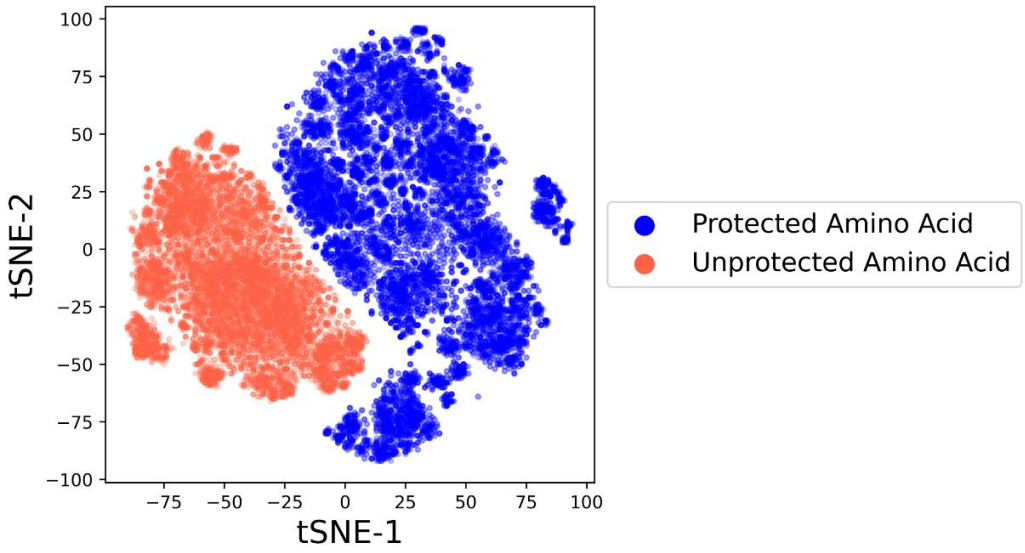
In generative design, design extends beyond a set of building blocks and explores novel NNAAs, as in PepINVENT in *Paper I.* The generated peptides can come with a synthesis overhead. When NNAAs are considered to improve therapeutic properties of peptides, they may be available already in their protected form for purchase. However, they might also need to be synthesized and later protected to become SPPS-ready. Unsuccessful synthesis of NNAAs could lead to back-and-forth between computational peptide designers and chemists, lengthening the "Design" of the DMTA cycle. Moreover, the synthetic feasibility of an NNAA itself does not guarantee an effective and orthogonal protection strategy for controlled deprotection during SPPS. Therefore, assessing synthesizability of any NNAA alone can be misleading. To bridge "Design" and "Make" stages, we explored evaluating synthesis and protection of NNAAs together as an integrated synthesis problem. Only then, we could address the true complexity of NNAA incorporation into peptides.

In *Paper IV*, we aimed to build a synthesis assistance tool, NNAA-Synth, to make individual NNAAs SPPS-ready. We achieved this by integrating orthogonal protecting group strategies, retrosynthesis planning, and DL-based route feasibility scoring. Although protection adds complexity to the synthesis challenge and expands the chemical space, NNAAs are a subclass of small molecules (Figure 15). Therefore, protected or not, their synthesis challenge is fundamentally a small molecule problem.[121] Consequently, we applied AI-based methods developed for small molecules to these building blocks. Utilizing other modality-specific methods enables repurposing them to peptides where we do not have enough publicly available data to develop modality-specific AI models on this topic. As explained before, these methods in principle are already applicable to NNAAs but have been disregarded due to the lack of *in silico* research for peptide synthesizability. This section will break down the proposed tool itself, describe its results for a large set of NNAAs and explain the case studies conducted in *Paper IV*.

NNAA-Synth is an end-to-end tool that takes a SMILES string of any NNAA as input and returns synthetic feasibility scores for all corresponding protected forms. The tool is composed of three main components: protection, retrosynthetic search, and route scoring for feasibility.

Protection as the first component, uses a custom cheminformatics workflow developed to yield corresponding protected forms of a given NNAA. The workflow uses a reference library containing SMILES Arbitrary Target Specification (SMARTS) patterns for reactive groups and detects reactive substructures in the query NNAA. Aliphatic acids and primary aliphatic or benzylic amines were encoded as distinct reactive groups, separately defining the backbone's amino and carboxyl functional groups of an amino acid.

Next, identified reactive groups are mapped to potential protection groups, with backbone functionalities handled separately to allow for orthogonal selection. In this setup, we defined, or



Metrics	Protected NNAAs	NNAAs with Optimal Protection
Total number of molecules	15508	9985
Infeasible Molecules	4435	1692
Starting Material Availability	89.26 ± 10.40	89.19 ± 11.27
Number of Reactions	8.95 ± 4.92	8.60 ± 4.98
Chemformer-Based Feasibility Score	0.05 ± 0.15	0.07 ± 0.16
Expert-Augmented Feasibility Score	8.63 ± 5.48	7.86 ± 5.40

Figure 15. Chemical space visualization for NNAA space with unprotected and protected NNAAs

Unprotected NNAAs (orange) and their protected forms (blue) are illustrated to show the diversity and the expansion of the chemical space when considering protection. The chemical space was visualized through two-component t-SNE on 512-bit count-based Morgan fingerprints with radius=3, using chirality. The protected space is also further described by metrics returned by the components of NNAA-Synth: the retrosynthesis-focused route prediction, and the two DL-based predictive models for prediction of route feasibility. The metrics described the protected library with all possible protection forms and the sub-library when only the most feasible protection was considered. While the first one showed the space NNAA-Synth helps filter, the second one was the constrained space after filtering a size equal to that of the unprotected amino acids. The figure was adapted from Paper IV[122] under a Creative Commons Attribution 3.0 Unported Licence[81].

limited, the orthogonality by capping the backbone amine and carboxyl groups with Fmoc and tert-Butyl, respectively. Fmoc is removed under basic conditions and tert-Butyl under strong acids, allowing each capped substructure to be exposed without affecting the other. These are the most common protection groups for backbone protection. Sidechain protections were chosen to be mutually orthogonal to both acid and base lability, using protection groups removable by

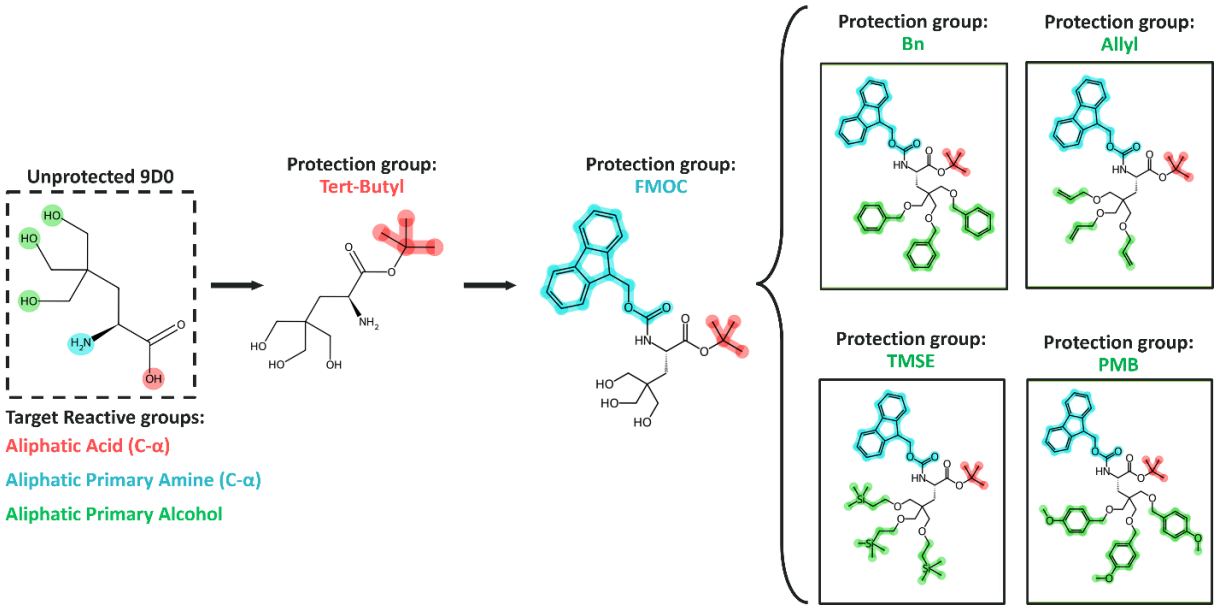


Figure 16. Illustrative example of reactive substructure detection and constructing protected forms with the custom cheminformatics workflow.
A query NNAA, 9D0[73], contains three reactive groups, the amino and carboxyl groups in the backbone, and three primary alcohols in its sidechain. By recursively passing this amino acid through the reaction templates for protection, the amino acid gets protected. Since the sidechain protection was mapped to four options, four protected forms of the amino acid were returned. The figure was extracted from Paper IV[122] under a Creative Commons Attribution 3.0 Unported Licence[81].

hydrogenation, oxidation or fluoride.[123] This facilitates selective deprotection during SPPS, exposing only the desired site for coupling, leaving other protections intact. It is also worth mentioning that some of the sidechain reactive groups can be protected by more than one protection group, leading to multiple protected versions of their NNAA. All detected reactive groups are then protected by combinatorial application of reaction templates, completing the cheminformatics workflow (Figure 16).

Protected versions of the query NNAA are then passed to the second component: retrosynthetic search. Retrosynthesis is performed by AiZynthFinder, a small molecule retrosynthesis prediction algorithm.[124–126] AiZynthFinder proposes synthesis routes by iteratively breaking down the target, or the product, into reactants until purchasable starting materials are reached. The search algorithm is Monte Carlo Tree Seach and uses neural networks for decision-making. These decisions are informed by a reaction template library and a list of stock molecules, identifying potential intermediates and starting materials along the route.[124–126] The neural networks for expansion and filter policies were trained on data extracted from the United States Patent and Trademark Office (USPTO).[125–127] The stock molecules were supplied from eMolecules[74] and combined with the protection groups used in our cheminformatics workflow. The only amino acid-specific tailoring in retrosynthesis was to treat protection groups with

complex structures as intact, preventing bond decomposition that would lead to additional steps of rebuilding the protection groups from scratch. This allowed prioritization of routes that use the complex protection groups directly or in-stock amino acids that already contain them, as precursors in route prediction. The simpler protecting groups can be incorporated into the route by synthesizing them on the fly. AiZynthFinder returns multiple routes for each target molecule, or protected NNAA, for downstream feasibility scoring.

In the third and last component, proposed routes for each protected form are subjected to route scoring for feasibility. Synthetic feasibility is scored by two DL predictive models: Chemformer and expert-augmented feasibility models. Chemformer feasibility model is a pretrained Transformers model that was fine-tuned for product prediction using in-house reaction data.[128,129] Chemformer generates the top 10 products with the highest likelihood for given reactants. If the true product appears among these top 10, the step is assigned the product's likelihood as its score, otherwise the score is zero.[128] All single-step scores are then aggregated by multiplication to yield the route score.[128] Focusing on single-steps, Chemformer captures the reaction-level plausibility by looking at reaction-to-product transformations. This model represented a feasibility filter: routes with Chemformer-based scores above zero are forwarded to the second DL model. A score of zero indicates at least one step with reactants unlikely to produce the predicted product under the proposed reaction.

The second DL model is the expert-augmented feasibility model. This model uses descriptors computed on the reaction-level, the route-level and on the target molecule, a protected NNAA in our case.[130] It predicts if the route has feasibility evaluation of "Good" for scores in [0,5), "Plausible" for scores in [5,9), or "Bad" for scores in [9,20].[130] The model returns an augmented score combining distance of query route to historical data in the latent space and route length to capture the expert opinion.[130] This model provides route-level feasibility evaluation where a lower score indicates better feasibility. Dual scoring combining route-level and reaction-level predictions, is included in NNAA-Synth for a more robust evaluation using complementary yet distinct models.

NNAA-Synth was built on 9,985 NNAAs publicly available from Amarasinghe *et al.*[73]. This building block set was also used to construct the training set peptides for PepINVENT in *Paper I*. The large set ensured that diverse sidechains and reactive groups were represented in the cheminformatics workflow. Additionally, this provided domain alignment between the generative model's learned space, and the chemical space NNAA-Synth is built on. Around one third of these amino acids required sidechain protection, resulting in 35 reactive functionalities needing protection. The cheminformatics workflow for protection yielded from a single to 32 protected forms of input NNAAs. The unprotected library with 9,985 NNAAs was transformed into a protected library of 15,508 protected residues (Figure 15). The entire protected library and its subset of optimal protection per NNAA showed similar route composition, indicating that a

holistic evaluation of route quality is necessary to make the distinction between building blocks rather than looking at simple route statistics. In some cases, protection did not significantly shift the synthetic complexity of the amino acids, and the unprotected NNAA remained as the synthesis bottleneck. Overall, looking at the most feasible route for individual NNAAs, approximately 25% of routes were solved to an available precursor while 17% of the NNAAs were identified as synthetically infeasible in any protected form (Figure 15).

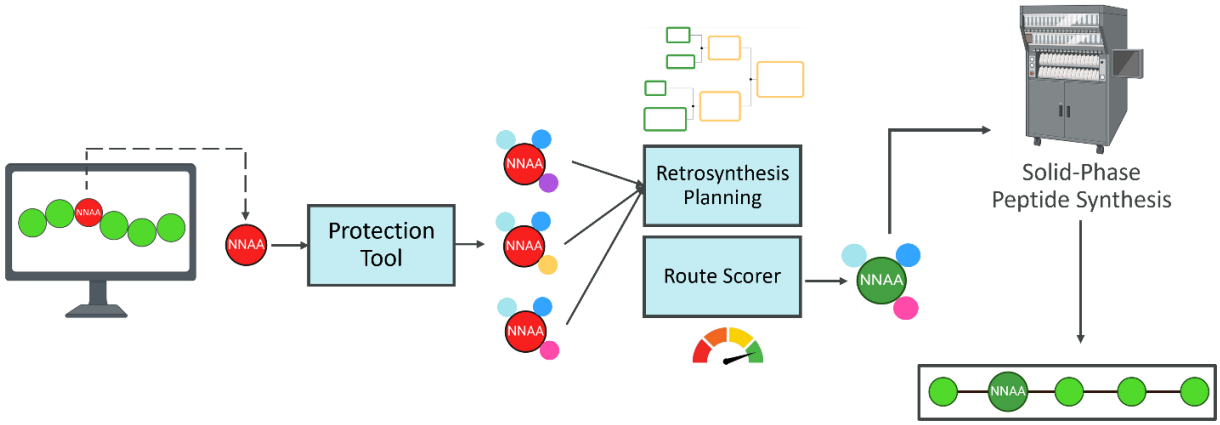
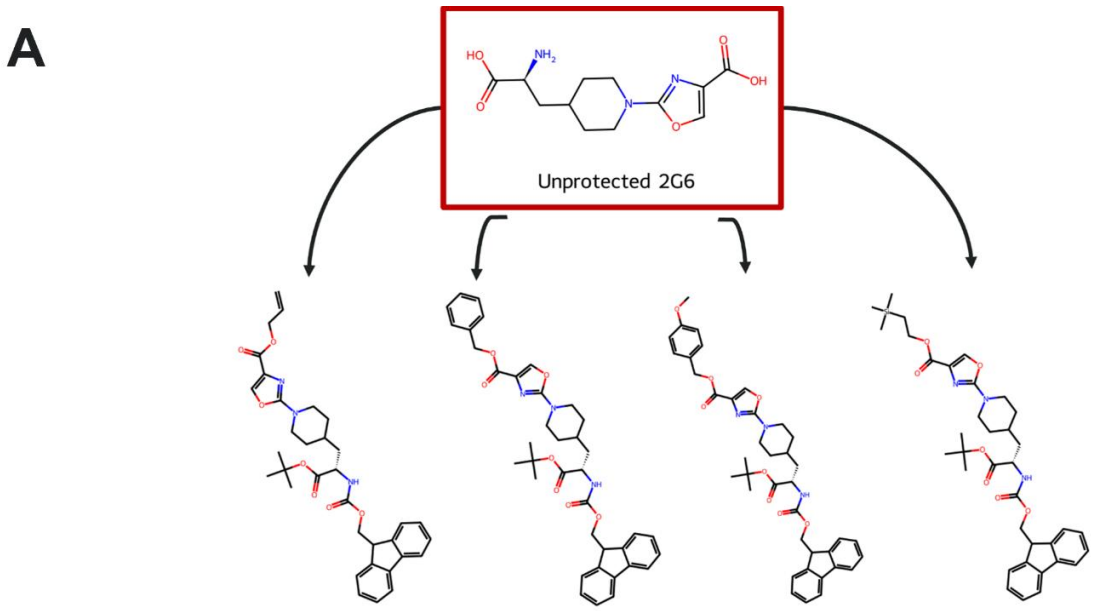


Figure 17. Schema for using NNAA-Synth for *in silico* peptide design to select the most feasible SPPS-compatible NNAA protection. *The figure was extracted from Paper IV*[122] *under a Creative Commons Attribution 3.0 Unported Licence*[81].

NNAA-Synth was intended as a post-processing tool to rank PepINVENT-generated peptides by cumulative feasibility scores over the proposed NNAAs. Nevertheless, we presented NNAA-Synth as a completely independent application to avoid narrowing the utility of the tool to a single niche topic. Instead, we demonstrated how NNAA-Synth can be applied across peptide chemistry and computational chemistry tasks. In the two case studies, our aim was to show how NNAA-Synth can facilitate *in silico* analysis of which protection strategy would be the most feasible for a protected NNAA of interest, and which NNAAs can be prioritized according to their synthetic accessibility for the peptide designs (Figure 17).

Choosing the most feasible protection strategy influences the yield of NNAAs when supplied to SPPS. Lower synthesizability direct influences wet lab cost, yield, and the effort required for reaction condition optimization. To identify an optimal protection strategy, we input a query NNAA to NNAA-Synth in the case study (Figure 18). The tool generated four protected variants, ran retrosynthesis, and scored the best route for each variant. All variants passed the Chemformer filter while the expert-augmented model rated one as infeasible, two as plausible and one as having good feasibility (Figure 18.A). We therefore selected the protection strategy associated with a good feasibility route and could follow the suggested route for synthesis (Figure 18.B).



	FMOC, tBu, Allyl	FMOC, tBu, Bn	FMOC, tBu, PMB	FMOC, tBu, TMSE
Number of Reaction Steps	3	3	5	17
Precursor Availability	100%	100%	100%	90%
Chemformer-based Feasibility Score	0.137	0.344	0.011	0.001
Expert-augmented Feasibility Score	4.31	5.12	7.13	19.91

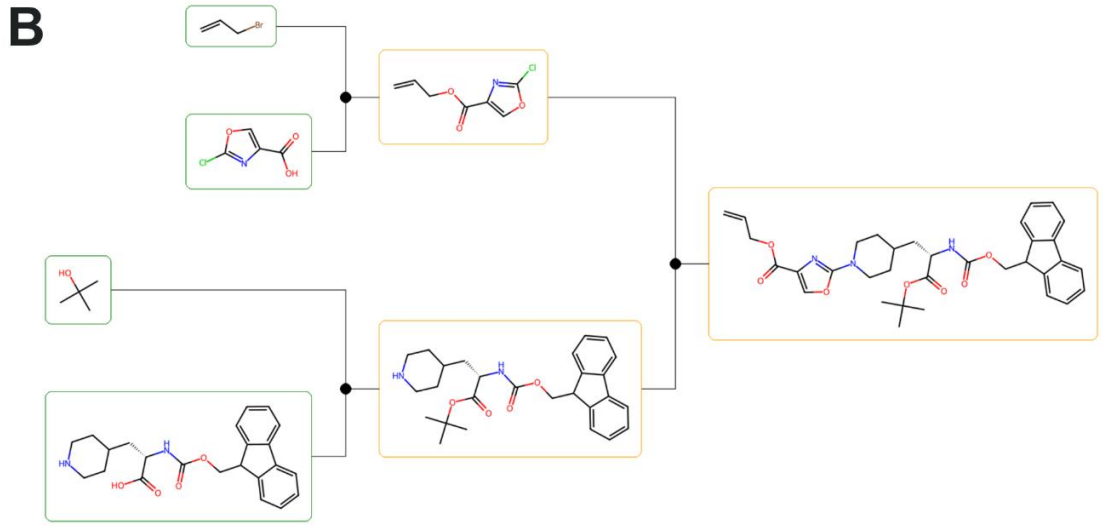


Figure 18. Selection of the most synthetically feasible protection strategy for an NNAA

(A) A query NNAA, 2G6[73] was input to NNAA-Synth that yields four orthogonally protected forms of 2G6. The tool outputs feasibility scores from two DL models, Chemformer and Expert-Augmented models, predicted on the route proposed by AiZynthFinder. Protecting the query with the set of Fmoc, tert-Butyl and allyl groups was proposed as the most synthetically feasible protection with a non-zero Chemformer score and the lowest expert-augmented score among the options. The best expert-augmented score

indicates "Good" feasibility with (B) the proposed route illustrated. The route contained four starting materials framed with green colors indicating their availability in stocks. The protected NNAA was suggested to be synthesized with three reactions. The yellow frames show any molecule that is not found in the available stocks supplied to AiZynthFinder. The figure was extracted from Paper IV[122] under a Creative Commons Attribution 3.0 Unported Licence[81].

In the second case study, we demonstrated how NNAAs can be prioritized during VS (Figure 19). We extracted structure-based screening results from Amarasinghe *et al*.[73] They applied docking and molecular mechanics generalized born surface area (MM-GBSA) (Figure 19.A). Both methods were used to predict binding affinity changes of site-specific single mutations in a peptide-protein complex. Docking scored intermolecular interactions between the peptide and its target to estimate the best ligand conformation that fits to the binding pocket.[131] MM-GBSA estimates the change in binding free energy with lower scores indicating greater stability of the peptide-target complex.[132] In this thesis, I included the results on a single position in the peptide that was mutated to the 9,985 NNAAs considered in NNAA-Synth. Lower scores on both structural methods indicate better predicted binding affinity for both methods, therefore NNAAs in the bottom-left of the plot would typically be selected for synthesis (Figure 19.A-B). However, when this plot was colored by NNAA-Synth scores, it is clear that the top-ranked candidates can be synthetically infeasible (Figure 19.B-C). Extending VS campaigns with NNAA-Synth would allow synthesizability-aware NNAA prioritization, that can reduce time and resource costs for testing peptide variants in the wet lab. In our example, 61 NNAAs with "Good" feasibility with acceptable docking and MM-GBSA scores were selected. To prioritize even a smaller subset, the most optimal routes identified by NNAA-Synth can be further examined for other qualities. These can include having available precursors as starting materials, 52 NNAAs in our case, or having a lower number of reaction steps, slicing down to 17 NNAAs (Figure 19.D).

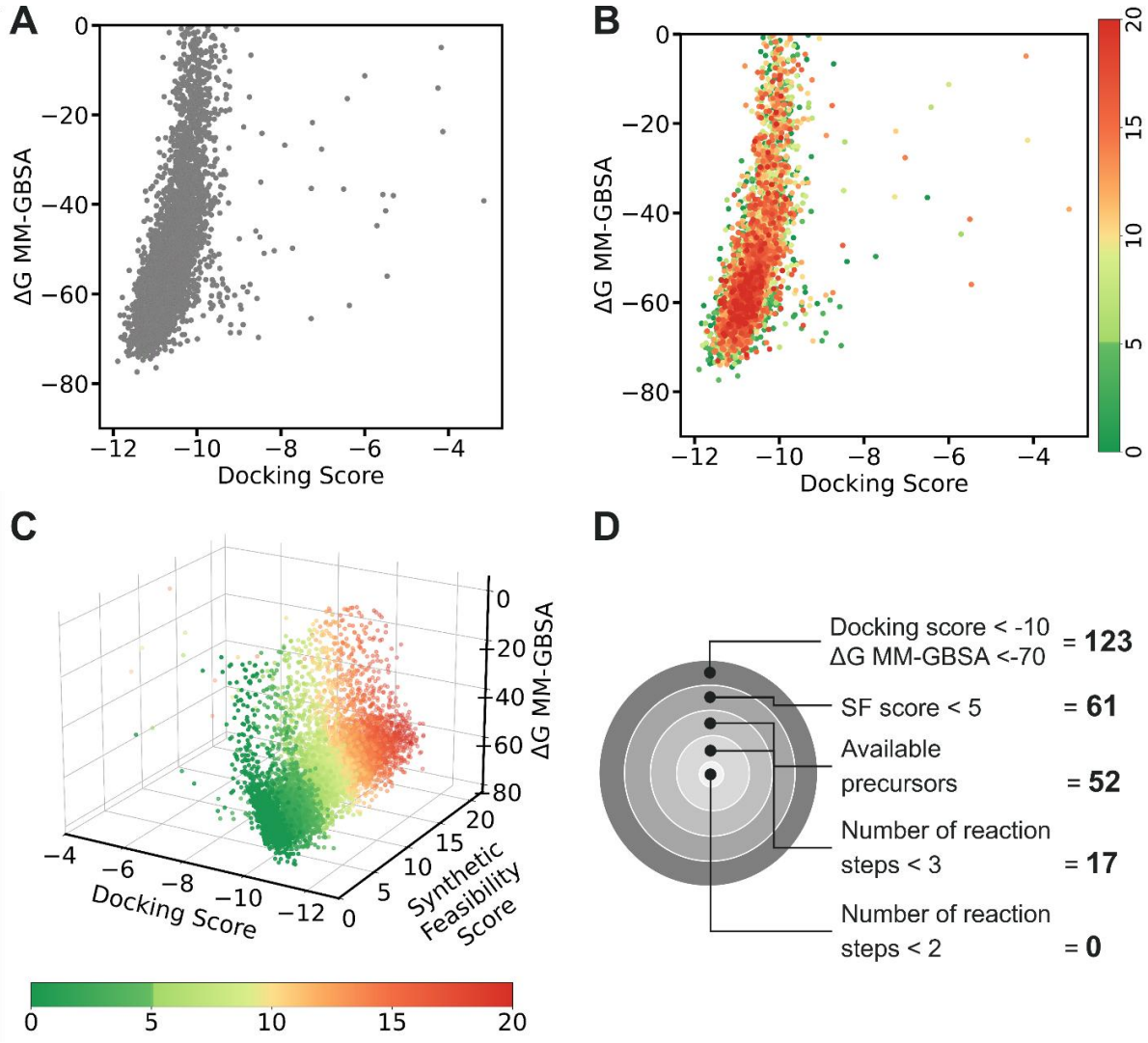


Figure 19. Prioritization of the most synthetically feasible NNAAs from VS campaign for positional mutation

One of the positions of a 9-mer peptide was mutated to approximately 10,000 NNAAs with the aim of improving the binding affinity of the peptide. (A) Docking and MM-GBSA scores of the single-mutants were calculated and ranked to identify the lowest scoring NNAAs that correspond the greatest potential for affinity enhancement.[73] (B) NNAA-Synth scores are used to color the scatter plot of the structural scores. This shows that candidate selection solely based on the structural scores would have high attrition due to a subset of the NNAAs being synthetical infeasible. A stronger selection should include (C) employing NNAA-Synth as an additional scorer in VS. In such case, (E) a targeted selection scheme including all components as well as route metrics can enable NNAA prioritization to the required size. The figure was adapted from Paper IV[122] under a Creative Commons Attribution 3.0 Unported Licence[81].

Outcome

Synthesizability is the critical bridge from *in silico*, whether it is traditional computational chemistry methods or generative model hallucinations, to bench-synthesized molecules. Navigating the vast chemical space of NNAAs must be linked to exploiting synthetically accessible building blocks and synthesizable peptide molecules. NNAA-Synth provides a synthesizability filter for peptide optimization workflows through a feasibility ranking of SPPS-ready amino acids. Additionally, it integrates SPPS-compatibility via orthogonal protection into the synthesis assessment. The tool implementation is modular with straightforward extension of reactive functionalities and mapping to new protection groups, enabling easy customization for the peptide community.

Certain aspects of peptide synthesis were beyond the scope of this work and therefore considered as limitations. First, the protection strategy allows independent exposure of substructures, but it still does not provide site-specific orthogonality when multiple copies of the same reactive substructure occur in a sidechain. In such cases, the same protection group is applied to all copies, which would then be removed together during deprotection. Though it remains a potential future improvement, this specialized case was not addressed in this general solution. Second, investigating the success of SPPS cycles including peptide yield, stability, aggregation or topology-defining intramolecular reactions was included, limiting the tool to the synthesis of building blocks only. Lastly, another limitation was constraining the tool to α-amino acids even though orthogonality would remain for other amino acid types. Next steps to address these limitations could transform the synthesizability assessment tool from a building block-specific method into a peptide-level evaluator.

Conclusions and Future Perspectives

The overarching goal of this thesis was to establish AI-driven approaches for next-generation peptide design to support therapeutic candidate development. With this focus, four studies were conducted to address three bottlenecks in peptide therapeutics: peptide design beyond the natural repertoire, reliable property prediction, and make-readiness of non-natural amino acids. Because data-driven approaches are difficult to apply to peptides containing NNAAs due to the scarcity of publicly available data, this thesis adopts the perspective that peptide design can be considered as the contextual assembly of building blocks that are small molecules themselves. Framing the design problem in this way enables extension of ML and DL methods established for small molecules to next-generation peptides, unlocking new opportunities to enhance their therapeutic potential.

In *Paper I*, we explored generative models to design peptides beyond natural repertoire by developing PepINVENT. PepINVENT uses a Transformer–based model that has learned peptide chemistry across multiple topologies, natural and non-natural amino acids, and common modifications. When coupled with RL, this space can be effectively navigated to explore and exploit high reward spaces that satisfy multi-objective property optimization. By exploring a vast peptide chemical space in atomic resolution that was inaccessible to sequence-based generators, PepINVENT can propose diverse peptide designs with novel NNAAs for lead optimization tasks. The molecular generative design is only truly promising if the scoring components provide reliable evaluations and the proposed designs can be synthesized.

In *Papers II* and *III*, we focused on reliable property prediction for cyclic peptide permeability and how to utilize it within generative modelling. *Paper II* develops data-driven approaches to build permeability predictors on multi-source datasets, a typical case for public data for peptides containing NNAAs. Introducing conformal prediction as an uncertainty quantification framework allows assessing a model's applicability domain and informs how model reliability deteriorates under domain shift when applied to unseen peptides, with new modifications or NNAAs. We demonstrate that calibrating on samples from the target domain extends model applicability to new peptides, restoring reliability by preserving its efficiency and validity on never-before-seen peptides. In *Paper III*, we employ this uncertainty-informed predictive model as a scoring component in RL loop to steer the generation toward peptides confidently predicted as permeable. After benchmarking different strategies of integrating model reliability, we found that designs converge to CP-determined single-label predictions when conformal efficiency is formulated directly as a reward. Compared to uncalibrated models, a faster convergence to our design goal was reached without compromising diversity of the generative process. This work establishes how uncertainty-focused predictions can be employed in generative design to strengthen the quality of designs.

In *Paper IV*, we addressed the key feedback we received for PepINVENT, assessing the synthetic feasibility of the generated designs to ensure a smooth transition from *in silico* design to bench. Throughout the thesis, I highlight that NNAAs can push peptides to their therapeutic potential and generative design enables exhaustive exploration of this extended space. Yet *de novo* design of NNAAs is only truly promising if they can be synthesized and incorporated into peptides. NNAA-Synth presents an end-to-end synthesis assistance tool that facilitates selection of the most feasible protection strategy, necessary for SPPS-compatibility and prioritizes NNAA candidates by synthesizability. This is achieved by yielding a synthetic accessibility score from evaluating route feasibility by employing AI-driven approaches for retrosynthesis and route quality prediction. Integrated as a post-filter in PepINVENT, NNAA-Synth triages designs based on their composition of make-ready NNAAs with optimal protection strategy. This can facilitate reducing attrition and cutting iterations in the DMTA cycle.

Collectively, this thesis presents how AI can contribute to peptide design by a set of tools combining generative modelling, predictive modelling and cheminformatics. These tools form an ideation pipeline with flexible design, reliable prediction, and feasible selection. A chemistry-informed generative model, PepINVENT, proposes peptide designs containing NNAAs by exploring a much larger chemical space than explored by traditional methods. Reinforcement learning guides the generation to suggest tailor-made peptides that meet design goals. Best practices established for reliable property prediction enable reaching design objectives with high confidence, improving the likelihood of experimental success. Finally, a post-processing filter, NNAA-Synth, prioritizes peptide designs that are synthetically accessible at the building block-level, ensuring direct inclusion in chemical synthesis with SPPS-ready amino acids. These studies represent advances on AI-driven acceleration of DMTA cycle for NNAA-containing peptides, enabling design of candidates for both intracellular and extracellular targets.

While this thesis immerses in inclusion of NNAAs in peptide design, the peptide field has ever-growing interest, with new solutions tackling wide range of problems both computationally and experimentally. To contribute to the community, all codebases, datasets, and model checkpoints have been released as open-source, manuscripts with Creative Commons Attribution 3.0 Unported License and codebases with Apache License 2.0.

A next step for this work would be to apply these methodologies in a practical application to demonstrate the impact in a real-life peptide project. Furthermore, PepINVENT can benefit from more scoring components. New property predictors such as affinity, immunogenicity, stability that are topology-agnostic when applicable and generalizable to modified peptides. Currently, there is no proxy to estimate peptide affinity. Structure prediction can enable simulation and analysis of peptide's interaction with their targets, allowing for binding affinity assessment. The current solutions are applicable to peptides with either natural amino acids or their extension with a handful of NNAAs. Structure-based scoring workflows to improve capabilities to simulate

interactions with targets would be essential for pharmacophore development. Furthermore, extending the synthesizability considerations to peptide-level from monomer-level would facilitate better transition from dry to wet lab. Synthetic feasibility is a complex problem. Amino acid composition, peptide aggregation, terminal modifications, cyclization efficiency are few of the many factors that must be considered for accurate estimation of the SPPS yield. Following the trends in recent publications, structure-based drug design, especially co-folding has been explored to predict the structures of peptide-protein complexes.[133–136] Peptide co-folding yields complexes with high prediction uncertainty and once again limited to few topologies and a small subset of available building blocks.[135,137] There is still a lot more to be done for peptides. Despite the rapidly evolving research landscape for peptides, substantial opportunities remain for AI-driven therapeutic peptide design.

Acknowledgements

First and foremost, I want to express my gratitude to my industrial supervisor, Ola Engkvist for his endless support and always helping me keep in focus. I learned a lot from his experience and solution-oriented approach. I also thank my academic supervisor, Florian David, for his support. I am very grateful to both my supervisors for giving me the freedom to explore my ideas and research topics I was genuinely curious about. I am very thankful for my examiner, Marija Cvijovic for her gentle support and critical feedback. I also would like to thank other co-supervisors/main collaborators for my PhD, Verena Siewers, Leonardo De Maria, and Andy Davis. Everyone who has been involved has shaped my scientific thinking and helped me establish what I stand for as a scientist.

During my PhD, I truly benefited from interdisciplinary collaborations and want to thank all colleagues I have worked with for all the corridor discussions and for the answers to my endless questions and emails. It would be impossible to fit all the names, but I feel very lucky to have experienced both industrial and academic PhD life. I thank my collaborators in Chalmers: Jiwei Mao, Louis Scott, Maximilian Otto. There have been few people that I had the privilege to work with and tackle problems that seemed impossible at the time, but became achievable because of them: Laura van Weesep, Sunay Chankeshwara. Samuel Genheden, Mikhail Kabeshov, Jon Paul Janet. I thank everyone in the Molecular AI for the friendly environment, for inspiring me and collaborating generously. I learned a lot from each and every one of you.

I, of course, cannot thank enough to my mentor, Ulf Norinder, for his unwavering support, his yearly visits and collaborating when I needed his support. I am honored to work with you during my PhD but even more so to have had the opportunity to simply talk with you. Your insight into science and life are always illuminating and profoundly valued.

This work would not have been possible without the constant support of my family. They have been instrumental in providing solid ground with their endless support and love. Thanks to my mom, dad, and my little, not-so-little, brother Cem for always being there for me. To my partner, thank you from the bottom of my heart for your encouragement, emotional support, and being ever-so-loving. Thank you for the endless pep talks and your patience through all my complaining. I love you to the end of the universe… and back! I also am truly lucky to have my best friends, Abir, Alaya, Begum, Busra, Ecem, Melis, Reem, Tefa and the friends I came across in AZ, Pallavi, Deniz, Elise, Lena, Vignesh. My best friends have always believed in me more than I have and made sure that I am aware of that in every late-night call we had. The support system my family and friends have provided made it impossible to stop pushing forward. Thank you, thank you, thank you! This PhD was funded by the Swedish Foundation for Strategic Research (SSF). Thank you for making this possible.

References

1. Muttenthaler, M., King, G. F., Adams, D. J. & Alewood, P. F. Trends in peptide drug discovery. *Nat. Rev. Drug Discov.* **20**, 309–325 (2021).
2. Otvos, L. The latest trends in peptide drug discovery and future challenges. *Expert Opin. Drug Discov.* **19**, 869–872 (2024).
3. Wang, Q., Chu, X. & Liu, J. Peptide-Based Biomaterials as a Promising Tool for Cancer Radiotherapy. *Adv. Sci.* **12**, e01775 (2025).
4. Wang, L. *et al.* Therapeutic peptides: current applications and future directions. *Sig Transduct Target Ther* **7**, 48 (2022).
5. Tsomaia, N. Peptide therapeutics: Targeting the undruggable space. *Eur. J. Med. Chem.* **94**, 459–470 (2015).
6. Lipinski, C. A., Lombardo, F., Dominy, B. W. & Feeney, P. J. Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings. *Adv. Drug Deliv. Rev.* **23**, 3–25 (1997).
7. Blanco, M. J., Gardinier, K. M. & Namchuk, M. N. Advancing New Chemical Modalities into Clinical Studies. *ACS Med. Chem. Lett.* **13**, 1691–1698 (2022).
8. Zheng, B., Wang, X., Guo, M. & Tzeng, C. M. Therapeutic Peptides: Recent Advances in Discovery, Synthesis, and Clinical Translation. *Int. J. Mol. Sci.* **26**, 5131 (2025).
9. Xiao, W. *et al.* Advance in peptide-based drug development: delivery platforms, therapeutics and vaccines. *Sig Transduct Target Ther* **10**, 74 (2025).
10. Ji, X., Nielsen, A. L. & Heinis, C. Cyclic Peptides for Drug Development. *Angew. Chem., Int. Ed.* **63**, e202308251 (2024).
11. Dougherty, P. G., Sahni, A. & Pei, D. Understanding Cell Penetration of Cyclic Peptides. *Chem. Rev.* **119**, 10241–10287 (2019).
12. Merritt, H. I., Sawyer, N. & Arora, P. S. Bent Into Shape: Folded Peptides to Mimic Protein Structure and Modulate Protein Function. *Pept Sci.* **112**, e24145 (2020).
13. Lombardi, L., Genio, V. Del, Albericio, F. & Williams, D. R. Advances in Peptidomimetics for Next-Generation Therapeutics: Strategies, Modifications, and Applications. *Chem. Rev.* **125**, 7099–7166 (2025).
14. Ahn, G., Banik, S. M. & Bertozzi, C. R. Degradation from the outside in: Targeting extracellular and membrane proteins for degradation through the endolysosomal pathway. *Cell Chem. Biol.* **28**, 1072–1080 (2021).
15. Breaking Down Patent Barriers: A Guide for Compounding Pharmacies. *DrugPatentWatch* https://www.drugpatentwatch.com/blog/breaking-down-patent-barriers-a-guide-for-compounding-pharmacies/?srsltid=AfmBOoqHwwKPmzT3GsJMuu_KiaB09tJ4NfT1v6ZpEySlsteGcwAN55nE (2025).
16. Tran, B. Biopharmaceuticals: The Patent Implications of Peptide Therapeutics. *PatentPC* https://patentpc.com/blog/patent-implications-of-peptide-therapeutics (2026).
17. Coley, C. W. Defining and Exploring Chemical Spaces. *Trends Chem.* **3**, 133–145 (2021).
18. Reymond, J. L. The Chemical Space Project. *Acc. Chem. Res.* **48**, 722–730 (2015).
19. Orsi, M. & Reymond, J. L. Navigating a 1E+60 Chemical Space of Peptide/Peptoid Oligomers. *Mol. Inform.* **44**, e202400186 (2024).
20. Sharma, K. K. *et al.* Unnatural Amino Acids: Strategies, Designs, and Applications in Medicinal Chemistry and Drug Discovery. *J. Med. Chem.* **67**, 19932–19965 (2024).

21. Sergeeva, A., Kolonin, M. G., Molldrem, J. J., Pasqualini, R. & Arap, W. Display technologies: Application for the discovery of drug and gene delivery agents. *Adv. Drug Deliv. Rev.* **58**, 1622–1654 (2006).
22. Castro, T. G., Melle-Franco, M., Sousa, C. E. A., Cavaco-Paulo, A. & Marcos, J. C. Non-Canonical Amino Acids as Building Blocks for Peptidomimetics: Structure, Function, and Applications. *Biomolecules* **13**, 981 (2023).
23. Ding, Y. *et al.* Impact of non-proteinogenic amino acids in the discovery and development of peptide therapeutics. *Amino Acids* **52**, 1207–1226 (2020).
24. Mahapatra, M. K., Karuppasamy, M. & Sahoo, B. M. Semaglutide, a glucagon like peptide-1 receptor agonist with cardiovascular benefits for management of type 2 diabetes. *Rev. Endocr. Metab. Disord.* **23**, 521–539 (2022).
25. Taechalertpaisarn, J., Ono, S., Okada, O., Johnstone, T. C. & Lokey, R. S. A New Amino Acid for Improving Permeability and Solubility in Macrocyclic Peptides through Side Chain-to-Backbone Hydrogen Bonding. *J. Med. Chem.* **65**, 5072–5084 (2022).
26. Buckton, L. K., Rahimi, M. N. & McAlpine, S. R. Cyclic Peptides as Drugs for Intracellular Targets: The Next Frontier in Peptide Therapeutic Development. *Chem. Eur. J.* **27**, 1487–1513 (2021).
27. Meister, D., Taimoory, S. M. & Trant, J. F. Unnatural amino acids improve affinity and modulate immunogenicity: Developing peptides to treat MHC type II autoimmune disorders. *Pept. Sci.* **111**, e24058 (2019).
28. López-López, E., Sánchez-Castañeda, J. P., Martinez-Cortés, M. S., de la Fuente-Nunez, C. & Medina-Franco, J. L. Exploring and expanding the chemical multiverse of peptides. *Chem. Sci.* **17**, 1461–1479 (2026).
29. Swenson, C. S., Mandava, G., Thomas, D. M. & Moellering, R. E. Tackling Undruggable Targets with Designer Peptidomimetics and Synthetic Biologics. *Chem. Rev.* **124**, 13020–13093 (2024).
30. Han, Y., Zhang, Y. K., Li, H., Ma, Z. & Wang, Y. Peptide Drug: Design and Clinical Applications. *MedComm* **6**, e70287 (2025).
31. Wong, D. A. *et al.* Characterizing and engineering post-translational modifications with high-throughput cell-free expression. *Nat. Commun.* **16**, 7215 (2025).
32. Josephson, K., Hartman, M. C. T. & Szostak, J. W. Ribosomal Synthesis of Unnatural Peptides. *J. Am. Chem. Soc.* **127**, 11727–11735 (2005).
33. Zhou, K. Z. Q. & Obexer, R. Non-Canonical Amino Acids for Engineering Peptides and Proteins with new Functions. *Isr. J. Chem.* **64**, e202400006 (2024).
34. Huang, W., Toth, I. & Skwarczynski, M. Automated Synthesis and Purification of Hydrophobic Peptides. in *Methods in Molecular Biology* (eds. Hussein, W. M. & Skwarczynski, M.) vol. 2931 343–354 (Humana Press Inc., 2025).
35. Duro-Castano, A., Conejos-Sánchez, I. & Vicent, M. J. Peptide-Based Polymer Therapeutics. *Polymers* **6**, 515–551 (2014).
36. Isidro-Llobet, A., Álvarez, M. & Albericio, F. Amino acid-protecting groups. *Chem. Rev.* **109**, 2455–2504 (2009).
37. Mäde, V., Els-Heindl, S. & Beck-Sickinger, A. G. Automated solid-phase peptide synthesis to obtain therapeutic peptides. *Beilstein J. Org. Chem.* **10**, 1197–1212 (2014).
38. Li, J. *et al.* CycPeptMPDB: A Comprehensive Database of Membrane Permeability of Cyclic Peptides. *J. Chem. Inf. Model.* **63**, 2240–2250 (2023).

39. Ramelot, T. A., Palmer, J., Montelione, G. T. & Bhardwaj, G. Cell-permeable chameleonic peptides: Exploiting conformational dynamics in de novo cyclic peptide design. *Curr. Opin. Struct. Biol.* **80**, 102603 (2023).
40. Williams, J. *et al.* Using in vitro ADME data for lead compound selection: An emphasis on PAMPA pH 5 permeability and oral bioavailability. *Bioorg. Med. Chem.* **56**, 116588 (2022).
41. Linker, S. M. *et al.* Lessons for Oral Bioavailability: How Conformationally Flexible Cyclic Peptides Enter and Cross Lipid Membranes. *J. Med. Chem.* **66**, 2773–2788 (2023).
42. Gavenonis, J., Sheneman, B. A., Siegert, T. R., Eshelman, M. R. & Kritzer, J. A. Comprehensive analysis of loops at protein-protein interfaces for macrocycle design. *Nat. Chem. Biol.* **10**, 716–722 (2014).
43. DeLano, W. L. Unraveling hot spots in binding interfaces: progress and challenges. *Curr. Opin. Struct. Biol.* **12**, 14–20 (2002).
44. Huang, H., Damjanovic, J., Miao, J. & Lin, Y. S. Cyclic peptides: Backbone rigidification and capability of mimicking motifs at protein–protein interfaces. *Phys. Chem. Chem. Phys.* **23**, 607–616 (2021).
45. Goles, M. *et al.* Peptide-based drug discovery through artificial intelligence: towards an autonomous design of therapeutic peptides. *Brief. Bioinform.* **25**, bbae275 (2024).
46. Zhang, K. *et al.* Artificial intelligence in drug development. *Nat. Med.* **31**, 45–59 (2025).
47. Nissan, N., Allen, M. C., Sabatino, D. & Biggar, K. K. Future Perspective: Harnessing the Power of Artificial Intelligence in the Generation of New Peptide Drugs. *Biomolecules* **14**, 1303 (2024).
48. Xu, X. *et al.* HELM-GPT: de novo macrocyclic peptide design using generative pre-trained transformer. *Bioinformatics* **40**, btae364 (2024).
49. Li, G., Iyer, B., Prasath, V. B. S., Ni, Y. & Salomonis, N. DeepImmuno: deep learning-empowered prediction and generation of immunogenic peptides for T-cell immunity. *Brief. Bioinform.* **22**, bbab160 (2021).
50. Wu, Z. *et al.* Signal Peptides Generated by Attention-Based Neural Networks. *ACS Synth. Biol.* **9**, 2154–2161 (2020).
51. Chen, S. *et al.* Design of target specific peptide inhibitors using generative deep learning and molecular dynamics simulations. *Nat. Commun.* **15**, 1611 (2024).
52. Olivecrona, M., Blaschke, T., Engkvist, O. & Chen, H. Molecular de-novo design through deep reinforcement learning. *J. Cheminform.* **9**, 48 (2017).
53. Blaschke, T. *et al.* REINVENT 2.0: An AI Tool for de Novo Drug Design. *J. Chem. Inf. Model.* **60**, 5918–5922 (2020).
54. Loeffler, H. H. *et al.* Reinvent 4: Modern AI–driven generative molecule design. *J. Cheminform.* **16**, 20 (2024).
55. David, L., Thakkar, A., Mercado, R. & Engkvist, O. Molecular representations in AI-driven drug discovery: a review and practical guide. *J. Cheminform.* **12**, 56 (2020).
56. Fialková, V. *et al.* LibINVENT: Reaction-based Generative Scaffold Decoration for in Silico Library Design. *J. Chem. Inf. Model.* **62**, 2046–2063 (2021).
57. Pascanu, R., Mikolov, T. & Bengio, Y. On the difficulty of training recurrent neural networks. in *Proceedings of the 30th International Conference on Machine Learning* 1310–1318 (PMLR, Atlanta, 2013). doi:10.5555/3042817.3043083.

58. Vaswani, A. *et al.* Attention is All you Need. in *Proceedings of the 31st International Conference on Neural Information Processing Systems* vol. 30 6000–6010 (Curran Associates Inc., New York, USA, 2017).
59. Pandey, P., Patel, V., George, N. V. & Mallajosyula, S. S. KELM-CPPpred: Kernel Extreme Learning Machine Based Prediction Model for Cell-Penetrating Peptides. *J. Proteome Res.* **17**, 3214–3222 (2018).
60. Fernández-Díaz, R., Ochoa, R., Hoang, T. L., Lopez, V. & Shields, D. C. How to build machine learning models able to extrapolate from standard to modified peptides. *J. Cheminform.* **17**, 185 (2025).
61. Shafer, G. & Vovk, V. A tutorial on conformal prediction. *J. Mach. Learn. Res.* **9**, 371–421 (2008).
62. Alvarsson, J., Arvidsson McShane, S., Norinder, U. & Spjuth, O. Predicting With Confidence: Using Conformal Prediction in Drug Discovery. *J. Pharm. Sci.* **110**, 42–49 (2021).
63. Morger, A. *et al.* Studying and mitigating the effects of data drifts on ML model performance at the example of chemical toxicity data. *Sci. Rep.* **12**, 7244 (2022).
64. Norinder, U., Carlsson, L., Boyer, S. & Eklund, M. Introducing conformal prediction in predictive modeling. A transparent and flexible alternative to applicability domain determination. *J. Chem. Inf. Model.* **54**, 1596–1603 (2014).
65. Norinder, U., Myatt, G. & Ahlberg, E. Predicting Aromatic Amine Mutagenicity with Confidence: A Case Study Using Conformal Prediction. *Biomolecules* **8**, 85 (2018).
66. Geylan, G. Training Machine Learning-based QSAR models with Conformal Prediction on Experimental Data from DNA-Encoded Chemical Libraries. (Uppsala University, 2021).
67. Angelopoulos, A. N. & Bates, S. *A Gentle Introduction to Conformal Prediction and Distribution-Free Uncertainty Quantification*. *ArXiv* (2022).
68. Gare, C. L., White, A. M. & Malins, L. R. From lead to market: chemical approaches to transform peptides into therapeutics. *Trends Biochem. Sci.* **50**, 467–480 (2025).
69. Shrestha, R., Fajardo, J. E. & Fiser, A. Residue-based pharmacophore approaches to study protein-protein interactions. *Curr. Opin. Struct. Biol.* **67**, 205–211 (2021).
70. Watanabe-Takahashi, M. *et al.* A unique peptide-based pharmacophore identifies an inhibitory compound against the A-subunit of Shiga toxin. *Sci. Rep.* **12**, 11443 (2022).
71. Yoshida, S. *et al.* Peptide-to-Small Molecule: A Pharmacophore-Guided Small Molecule Lead Generation Strategy from High-Affinity Macrocyclic Peptides. *J. Med. Chem.* **65**, 10655–10673 (2022).
72. Vincenzi, M., Mercurio, F. A. & Leone, M. Virtual Screening of Peptide Libraries: The Search for Peptide-Based Therapeutics Using Computational Tools. *Int. J. Mol. Sci.* **25**, 1798 (2024).
73. Amarasinghe, K. N. *et al.* Virtual Screening Expands the Non-Natural Amino Acid Palette for Peptide Optimization. *J. Chem. Inf. Model.* **62**, 2999–3007 (2022).
74. eMolecules. https://www.emolecules.com/.
75. Arús-Pous, J. *et al.* Exploring the GDB-13 chemical space using deep generative models. *J. Cheminform.* **11**, 20 (2019).
76. Mardikoraem, M., Wang, Z., Pascual, N. & Woldring, D. Generative models for protein sequence modeling: recent advances and future directions. *Brief. Bioinform.* **24**, bbad358 (2023).

77. Wan, F., Kontogiorgos-Heintz, D. & de la Fuente-Nunez, C. Deep generative models for peptide design. *Digital Discovery* **1**, 195–208 (2022).
78. Geylan, G. *et al.* PepINVENT: generative peptide design beyond natural amino acids. *Chem. Sci.* **16**, 8682–8696 (2025).
79. Lewis, M. *et al.* BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. *Proceedings of the Annual Meeting of the Association for Computational Linguistics* 7871–7880 (2020) doi:10.18653/V1/2020.ACL-MAIN.703.
80. Tibo, A., He, J., Janet, J. P., Nittinger, E. & Engkvist, O. Exhaustive local chemical space exploration using a transformer model. *Nat. Commun.* **15**, 7315 (2024).
81. Deed - Attribution 3.0 Unported - Creative Commons. https://creativecommons.org/licenses/by/3.0/.
82. Siani, M. A., Weininger, D. & Blaney, J. M. CHUCKLES: A Method for Representing and Searching Peptide and Peptoid Sequences on Both Monomer and Atomic Levels. *J. Chem. Inf. Comput. Sci.* **34**, 588–593 (1994).
83. Arús-Pous, J. *et al.* Randomized SMILES strings improve the quality of molecular generative models. *J. Cheminform.* **11**, 71 (2019).
84. Davis, C. S. The computer generation of multinomial random variates. *Comput. Stat. Data Anal.* **16**, 205–217 (1993).
85. Sutskever, I., Vinyals, O. & Le, Q. V. Sequence to sequence learning with neural networks. in *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2* 3104–3112 (MIT Press, Cambridge, MA, USA, 2014). doi:10.5555/2969033.2969173.
86. Polykovskiy, D. *et al.* Molecular Sets (MOSES): A Benchmarking Platform for Molecular Generation Models. *Front. Pharmacol.* **11**, (2020).
87. Landrum, G. RDKit: Open-source cheminformatics. http://www.rdkit.org/.
88. Wu, H., Mousseau, G., Mediouni, S., Valente, S. T. & Kodadek, T. Cell-Permeable Peptides Containing Cycloalanine Residues. *Angew. Chem., Int. Ed.* **55**, 12637–12642 (2016).
89. Stahl, S. J. *et al.* Generation and Characterization of a Chimeric Rabbit/Human Fab for Co-Crystallization of HIV-1 Rev. *J. Mol. Biol.* **397**, 697–708 (2010).
90. Oeller, M. *et al.* Sequence-based prediction of the intrinsic solubility of peptides containing non-natural amino acids. *Nat. Commun.* **14**, 7475 (2023).
91. Sugita, M. *et al.* Large-Scale Membrane Permeability Prediction of Cyclic Peptides Crossing a Lipid Bilayer Based on Enhanced Sampling Molecular Dynamics Simulations. *J. Chem. Inf. Model.* **61**, 3681–3695 (2021).
92. Parrondo-Pizarro, R., Menestrina, L., Garcia-Serna, R., Fernández-Torras, A. & Mestres, J. Enhancing molecular property prediction through data integration and consistency assessment. *J. Cheminform.* **17**, 163 (2025).
93. Nigam, A. K. *et al.* Assigning Confidence to Molecular Property Prediction. *Expert Opin. Drug Discov.* **16**, 1009–1023 (2021).
94. Agrawal, P. *et al.* CPPsite 2.0: a repository of experimentally validated cell-penetrating peptides. *Nucleic Acids Res.* **44**, D1098–D1103 (2016).
95. de Oliveira, E. C. L., Santana, K., Josino, L., Lima e Lima, A. H. & de Souza de Sales Júnior, C. Predicting cell-penetrating peptides using machine learning algorithms and navigating in their chemical space. *Sci. Rep.* **11**, 7628 (2021).

96. Schissel, C. K. *et al.* Deep learning to design nuclear-targeting abiotic miniproteins. *Nat. Chem.* **13**, 992–1000 (2021).
97. Bernardes-Loch, R. M. *et al.* PerseuCPP: a machine learning strategy to predict cell-penetrating peptides and their uptake efficiency. *Bioinformatics Advances* **5**, vbaf213 (2025).
98. Bhardwaj, G. *et al.* Accurate de novo design of membrane-traversing macrocycles. *Cell* **185**, 3520-3532.e26 (2022).
99. Ottaviani, G., Martel, S. & Carrupt, P.-A. Parallel Artificial Membrane Permeability Assay: A New Membrane for the Fast Prediction of Passive Human Skin Permeability. *J. Med. Chem.* **49**, 3948–3954 (2006).
100. Furukawa, A. *et al.* Passive Membrane Permeability in Cyclic Peptomer Scaffolds Is Robust to Extensive Variation in Side Chain Functionality and Backbone Geometry. *J. Med. Chem.* **59**, 9503–9512 (2016).
101. Kariyuki, S. *et al.* WO2013100132A1. Peptide-compound cyclization method. (2013).
102. Kelly, C. N. *et al.* Geometrically Diverse Lariat Peptide Scaffolds Reveal an Untapped Chemical Space of High Membrane Permeability. *J. Am. Chem. Soc.* **143**, 705–714 (2021).
103. Townsend, C. *et al. The Passive Permeability Landscape Around Geometrically Diverse Hexa- and Heptapeptide Macrocycles*. *ChemRxiv* (2020) doi:10.26434/CHEMRXIV.13335941.V1.
104. Geylan, G., De Maria, L., Engkvist, O., David, F. & Norinder, U. A methodology to correctly assess the applicability domain of cell membrane permeability predictors for cyclic peptides. *Digital Discovery* **3**, 1761–1775 (2024).
105. Ke, G. *et al.* LightGBM: A Highly Efficient Gradient Boosting Decision Tree. in *Proceedings of the 31st International Conference on Neural Information Processing Systems* 3149–3157 (Curran Associates Inc., California, USA, 2017). doi:10.5555/3294996.
106. Rogers, D. & Hahn, M. Extended-connectivity fingerprints. *J. Chem. Inf. Model.* **50**, 742–754 (2010).
107. van den Broek, R. L., Patel, S., van Westen, G. J. P., Jespers, W. & Sherman, W. In Search of Beautiful Molecules: A Perspective on Generative Modeling for Drug Design. *J. Chem. Inf. Model.* **65**, 9383–9397 (2025).
108. Liu, W., Li, J., Verma, C. S. & Lee, H. K. Systematic benchmarking of 13 AI methods for predicting cyclic peptide membrane permeability. *J. Cheminform.* **17**, 129 (2025).
109. Zervou, M. A., Doutsi, E., Pantazis, Y. & Tsakalides, P. Classifier-driven generative adversarial networks for enhanced antimicrobial peptide design. *Brief. Bioinform.* **26**, bbaf500 (2025).
110. Das, P. *et al.* Accelerated antimicrobial discovery via deep generative models and molecular dynamics simulations. *Nat. Biomed. Eng.* **5**, 613–623 (2021).
111. Pedregosa, F. *et al.* Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).
112. Heyndrickx, W. *et al.* Conformal efficiency as a metric for comparative model assessment befitting federated learning. *Artif. Intell. Life Sci.* **3**, 100070 (2023).
113. Masui, H. & Fuse, S. Recent Advances in the Solid-and Solution-Phase Synthesis of Peptides and Proteins Using Microflow Technology. *Org. Process Res. Dev.* **26**, 1751–1765 (2022).

114. Coin, I., Beyermann, M. & Bienert, M. Solid-phase peptide synthesis: from standard procedures to the synthesis of difficult sequences. *Nat. Protoc.* **2**, 3247–3256 (2007).
115. Albericio, F. Orthogonal Protecting Groups for N-Amino and C-Terminal Carboxyl Functions in Solid-Phase Peptide Synthesis. *Pept Sci.* **55**, 123–139 (2000).
116. Jad, Y. E., Kumar, A., El-Faham, A., De La Torre, B. G. & Albericio, F. Green Transformation of Solid-Phase Peptide Synthesis. *ACS Sustain. Chem. Eng.* **7**, 3671–3683 (2019).
117. Rossino, G. *et al.* Peptides as Therapeutic Agents: Challenges and Opportunities in the Green Transition Era. *Molecules* **28**, 7165 (2023).
118. Li, C. *et al.* Machine Learning Guides Peptide Nucleic Acid Flow Synthesis and Sequence Design. *Adv. Sci.* **9**, 2201988 (2022).
119. Tamás, B., Alberts, M., Laino, T. & Hartrampf, N. *Amino Acid Composition Drives Peptide Aggregation: Predicting Aggregation for Improved Synthesis*. *ChemRxiv* (2025) doi:10.26434/CHEMRXIV-2025-WJBMV.
120. Mohapatra, S. *et al.* Deep Learning for Prediction and Optimization of Fast-Flow Peptide Synthesis. *ACS Cent. Sci.* **6**, 2277–2286 (2020).
121. Scitable. amino acid. *Nature Education* https://www.nature.com/scitable/definition/amino-acid-115/ (2014).
122. Geylan, G. *et al.* From concept to chemistry: integrating protection group strategy and reaction feasibility into non-natural amino acid synthesis planning. *Chem. Sci.* **16**, 17927–17938 (2025).
123. Wuts, P. G. M. & Greene, T. W. *Greene's Protective Groups in Organic Synthesis*. (John Wiley & Sons, Inc., 2006). doi:10.1002/0470053488.
124. Genheden, S. *et al.* AiZynthFinder: a fast, robust and flexible open-source software for retrosynthetic planning. *J. Cheminform.* **12**, 70 (2020).
125. Saigiridharan, L. *et al.* AiZynthFinder 4.0: developments based on learnings from 3 years of industrial application. *J. Cheminform.* **16**, 57 (2024).
126. Segler, M. H. S., Preuss, M. & Waller, M. P. Planning chemical syntheses with deep neural networks and symbolic AI. *Nature* **555**, 604–610 (2018).
127. Lowe, D. Chemical reactions from US patents (1976-Sep2016). *figshare* https://doi.org/10.6084/m9.figshare.5104873 (2017).
128. Westerlund, A. M. *et al.* Do Chemformers Dream of Organic Matter? Evaluating a Transformer Model for Multistep Retrosynthesis. *J. Chem. Inf. Model.* **64**, 3021–3033 (2024).
129. Irwin, R., Dimitriadis, S., He, J. & Bjerrum, E. J. Chemformer: a pre-trained transformer for computational chemistry. *Mach. Learn.: Sci. Technol.* **3**, 015022 (2022).
130. Guo, Y. *et al. An Expert-Augmented Deep Learning Approach for Synthesis Route Evaluation*. *ChemRxiv* (2025) doi:10.26434/CHEMRXIV-2024-TP7RH-V2.
131. Fischer, A., Smieško, M., Sellner, M. & Lill, M. A. Decision Making in Structure-Based Drug Discovery: Visual Inspection of Docking Results. *J. Med. Chem.* **64**, 2489–2500 (2021).
132. Genheden, S. & Ryde, U. The MM/PBSA and MM/GBSA methods to estimate ligand-binding affinities. *Expert Opin. Drug Discov.* **10**, 449–461 (2015).
133. Nittinger, E., Yoluk, Ö., Tibo, A., Olanders, G. & Tyrchan, C. Co-folding, the future of docking – prediction of allosteric and orthosteric ligands. *Artif. Intell. Life Sci.* **8**, 100136 (2025).

134. Rettie, S. A. *et al.* Accurate de novo design of high-affinity protein-binding macrocycles using deep learning. *Nat. Chem. Biol.* **21**, 1948–1956 (2025).
135. Li, Q. *et al. RareFold: Structure Prediction and Design of Proteins with Noncanonical Amino Acids*. *bioRxiv* (2025) doi:10.1101/2025.05.19.654846.
136. Miao, J., Descoteaux, M. L. & Lin, Y. S. Structure prediction of cyclic peptides by molecular dynamics + machine learning. *Chem. Sci.* **12**, 14927–14936 (2021).
137. Chang, L., Mondal, A., Singh, B., Martínez-Noa, Y. & Perez, A. Revolutionizing Peptide-Based Drug Discovery: Advances in the Post-AlphaFold Era. *Wiley Interdiscip. Rev. Comput. Mol. Sci.* **14**, e1693 (2023).