Unsupervised Combinatorial Probabilistic Reasoning: Probabilistic Coin Change Problem

Zhongdi Qu[1], Yingheng Wang[1], Utku Umur Acikalin[1], Aaron M. Ferber[1], Goncalo J. Gouveia[2], Brandon Bills[3], Guohui Li[3], Joshua Kline[3], Sunandini Yedla[3], Xiao Wang[3], Frank C. Schroeder[2], Carla P. Gomes[1]

[1] Department of Computer Science, Cornell University
[2] Boyce Thompson Institute and Department of Chemistry and Chemical Biology, Cornell University
[3] Thermo Fisher Scientific
zq84@cornell.edu, yingheng@cs.cornell.edu, ua45@cornell.edu, amf272@cornell.edu, gjg84@cornell.edu, brandon.bills@thermofisher.com, guohui.li@thermofisher.com, joshua.kline@thermofisher.com, sunandini.yedla@thermofisher.com, xiao.wang2@thermofisher.com, schroeder@cornell.edu, gomes@cs.cornell.edu

Abstract

We introduce the Probabilistic Coin Change Problem (PCCP), a novel variant of the classical Combination Coin Change Problem (CCCP), motivated by a real-world scientific inverse task. The goal of CCCP is to enumerate all unordered combinations of coin denominations that sum to a given target. In PCCP, each coin type's value follows a discrete probability distribution, and the aggregate value of a combination of coins is thus stochastic. Given a set of such coin types and noisy observations of total sums, the task is to infer the most likely latent coin combination. To address the combinatorial and probabilistic complexity of PCCP, we propose DeepProReasoner (**Deep** Combinatorial **Pro**babilistic **Reason**ing with **E**mbedded **R**epresentations), an unsupervised, end-to-end, deep-learning framework that integrates combinatorial reasoning, latent-space modeling, and differentiable probabilistic reasoning. The model is trained using a reconstruction loss between the observed empirical distribution and a decoded probability mass function (PMF), enabling efficient gradient-based search over a continuous relaxation of the combinatorial space. We evaluate DeepProReasoner on two instances of PCCP: (1) a synthetic Candy Mix problem for ablation studies, and (2) a real-world task of molecular formula inference from ultrahigh resolution mass spectrometry (MS) data. Besides the two given instances, PCCP captures a wide range of inverse settings in biology, chemistry, environmental sciences, and medicine, where latent combinatorial structures give rise to noisy aggregate observations through stochastic processes. Our results show that DeepProReasoner achieves high accuracy and robustness, outperforming state-of-the-art methods.

1 Introduction

Over the past decade, AI has made remarkable strides, particularly in data-driven deep learning approaches across domains such as games (Silver et al. 2017), vision (Krizhevsky, Sutskever, and Hinton 2012), speech processing (Graves, Mohamed, and Hinton 2013), and natural language (Achiam et al. 2023). However, despite notable successes like AlphaFold (Jumper et al. 2021), scientific discovery presents fundamentally unique challenges (Wang et al. 2023). Scientific tasks are often unsupervised or weakly supervised, require adherence to established scientific principles, and demand generalization beyond the training data to enable genuine discovery. Moreover, they frequently involve exponential search spaces arising from underlying combinatorial structures (Gomes, Selman, and Gregoire 2019). A central grand challenge in scientific discovery is to develop AI systems capable of high-throughput interpretation of the scientific data being generated at rates far exceeding expert capacity (Merchant et al. 2023; Zeni et al. 2023).

We introduce a general problem capturing a broad class of scientific combinatorial inverse tasks, such as *inferring molecular formulas from mass spectrometry data*—a fundamental chemistry challenge that impacts biochemistry, medicine, and biology. We also present a novel approach for such combinatorial inverse problems.

Our contributions:

(1) We introduce the **Probabilistic Coin Change Problem (PCCP)**, a novel variant of the classical combinatorial Combination Coin Change Problem (CCCP), motivated by real-world scientific inverse tasks. In the traditional CCCP, given a set of coin denominations and a target sum, the goal is to enumerate all coin combinations (with unlimited repetitions) that sum exactly to the target. PCCP generalizes CCCP by assigning a discrete probability distribution to each coin type's value. Given coins with stochastic values and noisy observations of total sums, the task is to infer the most likely (latent) coin combination that generated the data. PCCP captures a broad class of inverse problems in biology, chemistry, environmental sciences where combinatorial latent discrete structures yield noisy aggregate observations through random processes.

(2) To address the combinatorial and probabilistic complexity of PCCP, we propose **DeepProReasoner** (**Deep** Combinatorial **Pro**babilistic **Reason**ing with **E**mbedded **R**epresentations), a novel unsupervised, end-to-end deep learning framework that integrates combinatorial reasoning, latent-space modeling, and differentiable probabilistic reasoning. DeepProReasoner uses a recurrent encoder to map the empirical (observed) discrete distribution function into a semantically structured and interpretable latent space. This latent space jointly embeds both the latent coin combination and the observed empirical distribution, capturing multiple layers of structure. It is integrated with a fully differentiable combinatorial probabilistic forward model that approximates the theoretical probability mass function (PMF) of the total sums as a function of the latent coin combination. This design enables unsupervised end-to-end training using a reconstruction loss between the observed empirical distribution and the generated PMF, allowing for efficient gradient-based search and bypassing the combinatorial explosion of the discrete search space. When prior knowledge restricts the combinatorial space of valid coin combinations, a second encoder explicitly embeds each feasible combination into the latent space, and gradient-based optimization effectively performs local search within this space.

To empirically evaluate DeepProReasoner, we consider two distinct instances of PCCP:

(3) We introduce the **Candy Mix problem**, which captures subtle variations in candy weights due to random production fluctuations, enabling controlled ablation studies and elucidating key determinants of PCCP's empirical complexity. We characterize DeepProReasoner's performance profile through systematic experiments on the Candy Mix problem, identifying regimes where it is most impactful.

(4) We consider a real-world task of molecular formula inference from mass spectrum (MS1) data, **MS1-to-MF**, addressing an urgent, unsolved problem in biomedical and environmental sciences, where millions of unknown molecules cannot yet be identified because there are no reliable approaches for determining their molecular formulae (MFs). In an MF (e.g., $C_{10}H_{18}N_2O_3$), each atomic element (e.g., C, N, O) can be represented as a coin type with stochastic mass values due to isotopic variations. For example, 98.93% of carbon (C) naturally occurs as ^{12}C (12 Dalton), while 1.07% occurs as ^{13}C (approximately 13.00335 Dalton). A molecule's mass is the sum of its atomic masses, and its mass spectrum provides intensity measurements across mass values, reflecting stochasticity due to isotopic variation. Given a set of elements and an observed spectrum, the goal is to infer a list of possible candidate MFs ranked by how likely they produced the observed spectrum. In practice, the ranked list helps narrow down the search for molecular formulas significantly and can be further refined if other data are available. Top-1 accuracy of the ranked list is a key metric, as it reflects the method's ability to automate the task. Our work leverages recent advances in ultrahigh resolution MS instruments that can resolve isotopic fine structure, i.e., individual peaks resulting from different isotopologues. For example, molecules containing one ^{13}C instead of one ^{12}C (mass difference 1.003) and molecules containing one ^{15}N instead of one ^{14}N (mass difference 0.997) would show up as separate peaks in a spectrum. Isotopic fine structure, in principle, should enable precise determination of MFs (Kingma 2014; Nagao et al. 2014). However, empirical noise and instrument-specific processing effects still pose a key challenge for automated MF determination (Claesen et al. 2025), requiring not only combinatorial reasoning but also data-driven calibration. DeepProReasoner effectively combines reasoning and learning within an unified end-to-end framework. We demonstrate its significant real-world scientific potential by applying it to the challenging task of inferring MFs solely from ultrahigh resolution MS1 data at scale, significantly advancing the state-of-the-art in this domain.

2 Problem Definition

The **Probabilistic Coin Change Problem (PCCP)** is a variant of the classical Combination Coin Change Problem (CCCP). In the traditional CCCP, given a set of coin types and a target sum, the goal is to enumerate all unordered combinations (with unlimited repetitions) of the coins whose values sum exactly to the target. PCCP extends this setting by allowing each coin type to have a stochastic value, modeled as a random variable with a known discrete distribution. Given a set of coins with associated value distributions and (noisy) samples of total sums, the goal is to infer the coin combination that maximizes the likelihood of generating the observed data.

2.1 Formal Problem Definition

Let $T = \{T_1, \dots, T_n\}$ be a set of n types. Each type T_i is associated with a discrete random variable W_i, supported on a finite set $\{w_{i,1}, \dots, w_{i,m_i}\} \subset \mathbb{R}_{>0}$, with associated probabilities $\{p_{i,1}, \dots, p_{i,m_i}\} \subset \mathbb{R}_{>0}$. A composition is defined by a count vector $\mathbf{n} = (t_1, \dots, t_n) \in \mathbb{N}^n$, where t_i denotes the number of items of type T_i. Given a composition $\mathbf{n}$, the total sum S is a random variable defined as $S = \sum_{i=1}^{n} \sum_{j=1}^{t_i} W_{i,j}$, where $W_{i,j} \overset{\text{iid}}{\sim} W_i$.

Observation Model. The true values of S are not directly observable. Instead, we are given noisy measurements: $\tilde{S} = S + \epsilon, \epsilon \sim \mathcal{E}$, where $\mathcal{E}$ is an unknown noise distribution (e.g., Gaussian). These noisy samples form an empirical distribution with K supports: $\tilde{P}_{\tilde{S}} = \{(\tilde{S}_k, \tilde{I}_k)\}_{k=1}^{K}$, where $\tilde{I}_k$ is a noisy estimate of the probability mass for support $\tilde{S}_k$. We assume that K is polynomial in n. This reflects practical constraints: finite resolution and sampling limit the number of distinguishable values.

Inputs: (i) A set of n types T, where each type T_i is associated with a discrete distribution $W_i = \{(w_{i,j}, p_{i,j})\}_{j=1}^{m_i}$; and (ii) a noisy empirical distribution over the total sums $\tilde{P}_{\tilde{S}} = \{(\tilde{S}_k, \tilde{I}_k)\}_{k=1}^{K}$. **Output**: the latent composition vector that best explains the observed distribution: $\mathbf{n}^* = \arg\max_{\mathbf{n} \in \mathbb{N}^n} \mathbb{P}(\tilde{P}_{\tilde{S}} \mid \mathbf{n})$.

PCCP is motivated by the observation that many real-world inverse problems involve intrinsic randomness and measurement noise, aspects not captured by classical combinatorial formulations like CCCP. We introduce two concrete instances of PCCP to illustrate its breadth.

2.2 Candy Mix Problem

Inputs: (i) A set of n candy types T where each candy type T_i is associated with a discrete distribution on the weights $W_i = \{(w_{i,j}, p_{i,j})\}_{j=1}^{m_i}$; and (ii) a polynomial length noisy empirical distribution $\tilde{P}_{\tilde{S}} = \{(\tilde{S}_k, \tilde{I}_k)\}_{k=1}^{K}$, representing the

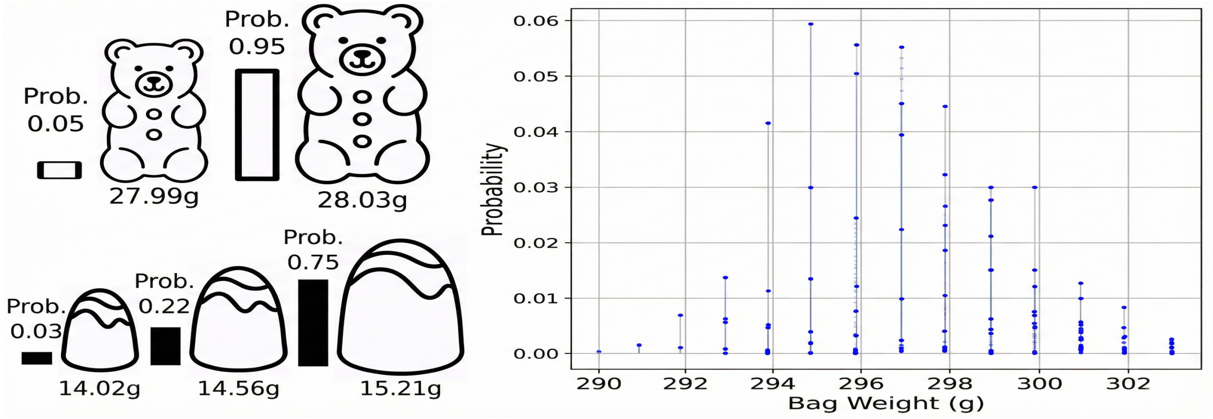

Figure 1: Example inputs to the Candy Mix problem: candy types and discrete distributions on their weights (left), and noisy empirical probability mass function (PMF) of the total bag weights (right). Bags share a fixed but unknown composition (counts per candy type). The weight of the bags varies due to random draws from the candy weight distributions and measurement noise. Given the weight distributions on the candy types, and the empirical PMF produced by sampling a collection of bags, the goal is to infer the candy composition in the bags. In this case, given the empirical PMF shown, the composition of the bags is: 1 gummy bear, 2 chocolates, and 13 caramels.

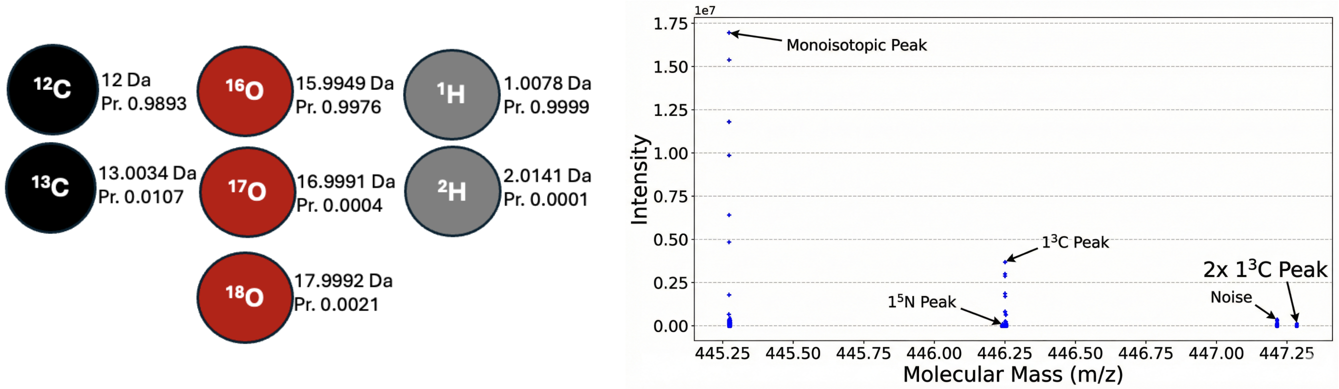

Figure 2: Example inputs to the MS1-to-MF problem: element types with isotope distributions (top), and a mass spectrum that represents the noisy empirical PMF of the isotopologues (bottom). Isotopologues are molecules with the same molecular formula but different isotopes for select elements, so their weights differ. A mass spectrum gives a noisy measurement on the empirical PMF over the masses of the isotopologues. Given the elements' isotopic distribution and a mass spectrum, the goal is to produce a list of possible candidates ranked by the likelihood of producing the spectrum. The mass spectrum shown corresponds to the formula $C_{20}H_{37}N_4O_7$.

noisy distribution over total bag weights for bags with identical but unknown composition. **Output:** The underlying bag composition as a count vector $\mathbf{n} = (t_1, \ldots, t_n) \in \mathbb{N}^n$ where t_i denotes the number of candies of type T_i per bag.

2.3 MS1-to-MF

Consider a factory that produces mixed bags of candies (e.g., chocolates, gummies, caramels), where each candy type has a characteristic weight distribution due to manufacturing variability. All bags share a fixed but **unknown** composition—that is, the number of candies per candy type—but the total bag weight varies due to random draws from the weight distributions and measurement noise. The factory provides a noisy distribution over the bag weights, estimated from finite samples, together with the candy types and their weight distributions. The task is to recover the latent bag composition that generated the observed distribution.

Mass spectrometry is a core tool in chemistry, biology, and materials science to identify and characterize molecules. The process consists of four main stages: (1) *Ionization*: Molecules in a sample are converted into charged particles (ions). (2) *Separation*: These ions are separated according to their mass-to-charge ratios (m/z) using electric or magnetic fields. (3) *Detection*: The separated ions are measured, producing a mass spectrum—a plot of ion intensities versus m/z. The intensity at each m/z reflects the number of ions detected at that mass.(4) *Analysis*: The resulting spectra are analyzed to infer their molecular formulas (MF). With the advancement of ultrahigh resolution MS, the peaks in the spectrum give more precise measurements on the abundance of individual isotopologues of the molecule. Molecule assignment can thus be determined by reasoning on the relative abundances of the isotopologues.

MS1-to-MF can be modeled as an instance of the PCCP. Each atomic element is associated with a discrete isotopic mass distribution reflecting natural isotope abundances. An MF is represented by a count vector $\mathbf{n} = (n_1, \ldots, n_n) \in \mathbb{N}^n$, where n_i denotes the number of atoms of element E_i. The mass of a molecule is the sum of its atomic masses, which is stochastic due to isotopic variation. A mass spectrum provides intensity measurements across mass values, capturing the distribution of isotopologues. However, both instrument noise and sampling noise distort the empirical estimate of the true isotopic distribution. Given a set of elements and a measured spectrum (e.g., Fig. 2), the goal is to infer a list of MFs ranked by how likely they produced the observed spectrum.

Inputs: (i) A set of n atomic elements E, where each element E_i is characterized by a known isotopic mass distribution $W_i = \{(w_{i,j}, p_{i,j})\}_{j=1}^{m_i}$ and; (ii) a polynomial length observed mass spectrum $\tilde{P}_{\tilde{S}} = \{(\tilde{S}_k, \tilde{I}_k)\}_{k=1}^{K}$, representing the noisy distribution over isotopologues. **Output:** A list of MFs ranked by how likely they produced the mass spectrum.

3 Methods

3.1 Encoder: Learning an Interpretable Latent Space for Scalable Optimization

PCCP exhibits a combinatorial search space: the number of possible compositions of the count vector grows as $\mathcal{O}(B^n)$, where n is the number of types and B is the upper bound on the count of each type. This exponential growth in dimensionality renders exhaustive search intractable for large n. To overcome this, we propose a latent space reasoning framework. Instead of searching directly over discrete count vectors, we use a Gated Recurrent Unit (GRU) network (Cho et al. 2014) to encode observations into an interpretable latent space where continuous relaxations of the count vectors live. During inference, the latent vectors are rounded to obtain predictions on the counts. During training, a differentiable combinatorial probabilistic reasoning module, covered in more detail later, takes the latent vectors as inputs and approximates the probability distribution over total sums S. The model is trained end-to-end using the reconstruction loss between the empirical distribution of the observed total sums, and the reconstructed PMF, enabling efficient gradient-based optimization and circumventing the combinatorial explosion of the discrete space, facilitating scalable inference.

3.2 Differentiable Combinatorial Probabilistic Reasoner

Central to this framework is a differentiable probabilistic reasoner that approximates the probability distribution over total sums S given a latent representation of the composition vector. Consider a composition vector $\mathbf{z} = (t_1, \ldots t_n)$, specifying how many instances of each type occur. For each type T_i which appears t_i times, we can define a value allocation vector $\mathbf{c}_i = (c_{i,1}, \ldots, c_{i,m_i})$, where $c_{i,j}$ denotes the number of times weight $w_{i,j}$ is drawn. Each $\mathbf{c}_i$ satisfies $\sum_j c_{i,j} = t_i$ and follows the multinomial distribution: $\mathbb{P}(\mathbf{c}_i) = \frac{t_i!}{\prod_{j=1}^{m_i} c_{i,j}!} \prod_{j=1}^{m_i} p_{i,j}^{c_{i,j}}$. Given a configuration, $(\mathbf{c}_1, \ldots, \mathbf{c}_n)$, i.e., the value allocation vector of each type T_i, the total sum is: $s = \sum_{i=1}^{n} \sum_{j=1}^{m_i} c_{i,j} w_{i,j}$. The probability of observing s is $\mathbb{P}(S = s \mid \mathbf{z}) = \sum_{\mathcal{C}} \prod_{i=1}^{n} \mathbb{P}(\mathbf{c}_i)$, where $\mathcal{C}$ contains all the configurations $(\mathbf{c}_1, \ldots, \mathbf{c}_n)$ that result in s.

To enable efficient differentiable reasoning, we approximate the PMF over S by selectively evaluating only a tractable subset of the total sums. For each type T_i, the modal weight $w_{i,\text{mode}}$ is the most probable value in its weight distribution. We introduce a tunable hyperparameter q and say that a configuration $(\mathbf{c}_1, \ldots, \mathbf{c}_n)$ is q-admissible if the total number of non-modal weights satisfies $\sum_{i=1}^{n} \sum_{j=1, j\neq\text{mode}}^{m_i} c_{i,j} \leq q$. We compute $\mathbb{P}(S = s \mid \mathbf{z})$ only for q-admissible configurations. In practice, q can be tuned or chosen based on prior knowledge of the problem domain. The more skewed the weight distributions, the smaller q needs to be to give an accurate approximation of the PMF since configurations involving many non-modal values have negligible probability. Let $h_i = \sum_{j\neq\text{mode}} c_{i,j} \leq q$ be the number of non-modal weights in $\mathbf{c}_i$. We pre-compute the non-modal term $\mathbb{H}(\mathbf{c}_i) = \prod_{j=1, j\neq\text{mode}}^{m_i} \frac{p_{i,j}^{c_{i,j}}}{c_{i,j}!}$ and compute the full multinomial probability during training as

$$\mathbb{P}(\mathbf{c}_i) = \frac{t_i!}{(t_i - h_i)!} p_{i,\text{mode}}^{t_i - h_i} \mathbb{H}(\mathbf{c}_i) \tag{1}$$

$$= t_i(t_i - 1)\ldots(t_i - h_i + 1) p_{i,\text{mode}}^{t_i - h_i} \mathbb{H}(\mathbf{c}_i) \tag{2}$$

Equation (2) provides a workaround for differentiably approximating the multinomial distribution. Note in Equation (2), t_i no longer needs to be an integer. Thus the probabilistic reasoner can take a continuous relaxation of the composition vector from the latent space as the input. Moreover, the output of the probabilistic reasoner $\mathbb{P}(S = s \mid \mathbf{z})$ is differentiable with respect to $\mathbf{z}$. Training the model with a reconstruction loss between the predicted and the empirical PMF, gradients can be backpropagated from the loss through the probabilistic reasoner and the latent space, and guide the encoder to learn meaningful representations. This allows the model to discover latent count structures directly from observed sums, guided by a reasoning-based training signal without the use of ground-truth count vectors. When paired data is scarce, e.g., MS1-to-MF, our unsupervised framework provides a principled training strategy.

3.3 Constrained Reasoning via Dual Encoders for Molecular Formula Inference

In the MS1-to-MF problem, the space of valid count vectors is constrained by chemical principles (Kind and Fiehn 2007); the optimal count vector must not only explain the observed mass spectrum but also be chemically plausible. Prior work has introduced specialized algorithms to incorporate such chemical constraints to significantly reduce the search space (Böcker et al. 2008). Even within a narrow

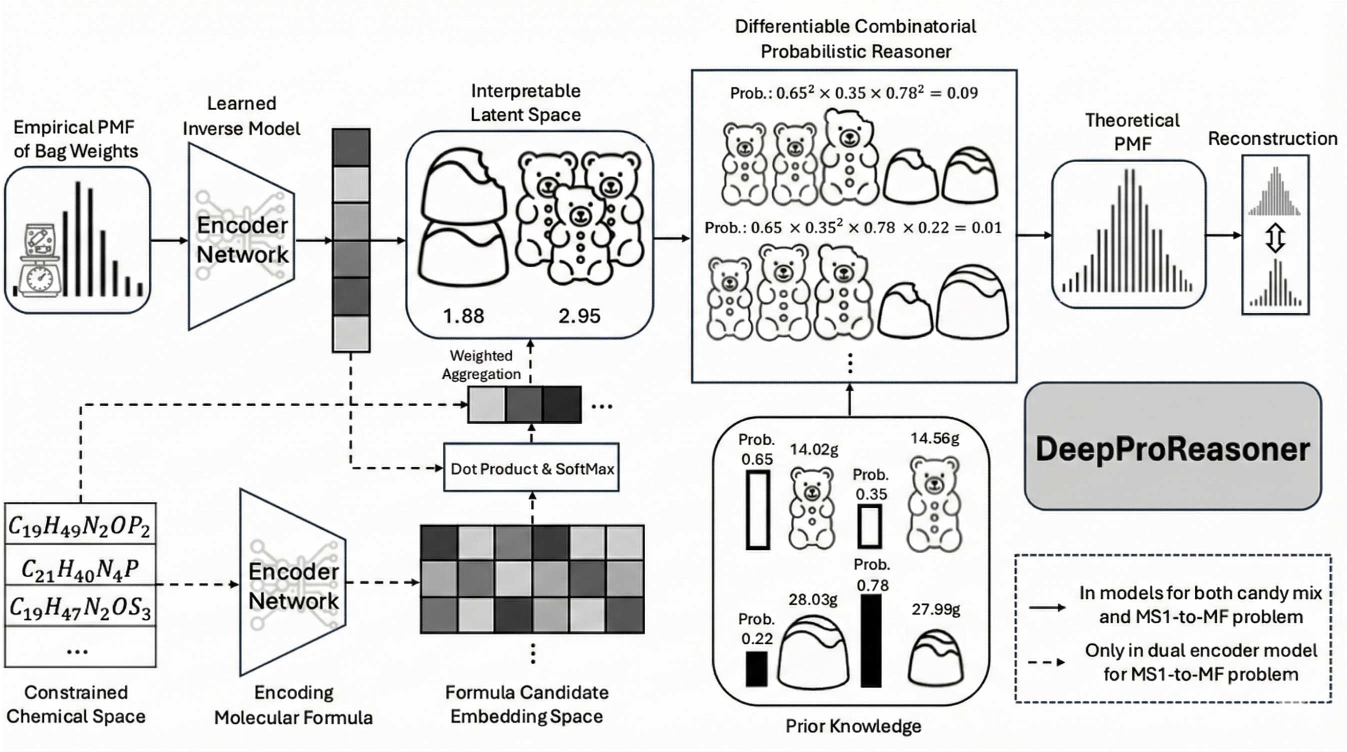
Empirical PMF of Bag Weights
Learned Inverse Model
Encoder Network
Interpretable Latent Space
1.88
2.95
Differentiable Combinatorial Probabilistic Reasoner
Prob.: 0.65² × 0.35 × 0.78² = 0.09
Prob.: 0.65 × 0.35² × 0.78 × 0.22 = 0.01
Theoretical PMF
Reconstruction
Weighted Aggregation
Dot Product & SoftMax
DeepProReasoner
$C_{19}H_{49}N_2OP_2$
$C_{21}H_{40}N_4P$
$C_{19}H_{47}N_2OS_3$
...
Constrained Chemical Space
Encoder Network
Encoding Molecular Formula
Formula Candidate Embedding Space
Prob. 0.65
14.02g
Prob. 0.35
14.56g
Prob. 0.78
Prob. 0.22
28.03g
27.99g
Prior Knowledge
In models for both candy mix and MS1-to-MF problem
Only in dual encoder model for MS1-to-MF problem

Figure 3: DeepProReasoner: An encoder embeds the empirical PMF into an interpretable latent space representing relaxed predicted composition vectors. A fully differentiable combinatorial probabilistic reasoner approximates the theoretical PMF, given the latent vector. The reconstruction loss between the reconstructed and the empirical PMFs is backpropagated through the reasoner to the encoder to guide the learning of meaningful embeddings and the prediction that best aligns with the empirical PMF. When prior knowledge limits feasible compositions to an enumerable set, a second encoder embeds them into the latent search space. The feasible candidates' embeddings are scored by similarity to the PMF embedding, and the scores define a weighted aggregation interpreted as a continuous relaxation of the predicted composition. The reconstruction loss is passed to both the PMF encoder and the candidates encoder. This joint training encourages the two encoders to align embeddings, steering the prediction toward the feasible candidate that best aligns with the empirical PMF.

mass window, the space of possible compositions remains large, but when constrained, enumeration becomes tractable.

To enable reasoning within this constrained search space, we propose a dual-encoder architecture. Given a mass spectrum $\tilde{P}_{\tilde{S}} = \{(\tilde{S}_k, \tilde{I}_k)\}_{k=1}^{K}$ sorted by increasing mass, we first apply the method of (Fiehn 2007) to find all feasible candidate molecular formulas whose monoisotopic molecular mass – the exact mass of the molecule assuming the most abundant (almost always also the lightest) isotope of each element – falls within the interval $[\tilde{S}_1 - \sigma, \tilde{S}_1 + \sigma]$, where σ is an instrument-dependent hyperparameter that accounts for mass measurement noise. Let $\mathcal{F} = \{\mathbf{n}^{(1)}, \ldots, \mathbf{n}^{(M)}\} \subset \mathbb{N}^n$ denote the resulting set of valid candidate count vectors. We then encode the spectrum and candidate formulas into a shared latent space using two encoders. A GRU-based encoder $f_{\text{spec}} : \mathbb{R}^{K\times 2} \to \mathbb{R}^d$ maps the input spectrum into a fixed-dimensional spectrum embedding $\mathbf{z}_{\text{spec}} = f_{\text{spec}}(\tilde{P}_{\tilde{S}})$. Each atomic element has a trainable embedding vector, forming an embedding matrix $E \in \mathbb{R}^{n\times d_e}$. Candidate count vectors $\mathbf{n}^{(m)}$ are multiplied by the element embeddings and then passed through a Multi-Layer Perceptron (MLP) to produce a candidate embedding: $\mathbf{z}_{\text{form}}^{(m)} = \text{MLP}(E^\top \mathbf{n}^{(m)}) \in \mathbb{R}^d$.

Next, we compute similarity scores between the spectrum embedding and all associated candidate embeddings by taking the dot product, $s^{(m)} = \langle \mathbf{z}_{\text{spec}}, \mathbf{z}_{\text{form}}^{(m)} \rangle$, and normalizing via softmax to obtain a soft assignment over the candidate sets. During inference, the candidates are ranked according to these scores. During training, the scores are used to compute a weighted average of the candidate count vectors, producing a continuous composition vector interpreted as a relaxed count vector: $\bar{\mathbf{n}} = \sum_{m=1}^{M} \text{softmax}(s^{(m)}) \cdot \mathbf{n}^{(m)}$. This vector is subsequently passed to the probabilistic reasoner to approximate the PMF over the mass values. During training, the reconstruction loss between the reconstructed PMF and the observed mass spectrum is backpropagated through both encoders. The joint training encourages the spectrum and formula encoders to learn a shared latent space where spectrum embeddings and candidate formula embeddings are aligned. Because the latent space contains continuous relaxations of all chemically feasible candidates, gradient descent effectively performs local search within this constrained space, steering the solution toward the candidate count vector that best aligns with the observed spectrum.

4 Related Work

Coin Change Problems The Coin Change Problem is a well-studied combinatorial problem in theoretical computer science. Its optimization version looks for the smallest number of coins and is known to be weakly NP-hard (Hartmanis 1982). The counting version asking for all combinations of coins that sum up to the target is in $\#P$. Considerable research has been devoted to variations of the problem (Gupta, Huang, and Impagliazzo 2024; Cowen, Cowen, and Steinberg 2008), but to the best of our knowledge, the probabilistic variation introduced here has not been considered.

MS1-to-MF Inferring molecular formula from mass spectra is a fundamental problem spanning disciplines such as chemistry, biology, drug discovery and forensic sciences. Previous works (Kind and Fiehn 2007; Böcker et al. 2009; Dührkop et al. 2019; Schmid et al. 2023) have developed various computational methods for the task. These methods all generally follow the following procedures, including the state of the art, SIRIUS (Dührkop et al. 2019) and MZmine (Schmid et al. 2023): (1) compute all candidate formulas from the monoisotopic peak, that is, the peak with the smallest mass where all atoms are the most abundant isotope; (2) for every such candidate formula, simulate its theoretical isotope pattern; and (3) match and rank it against the input empirical spectrum using a similarity metric like the cosine score, or maximum likelihood estimation. For every individual example, these methods deterministically search for the candidate that best explains the spectrum, without leveraging cross-example information like instrumental noise. *DeepProReasoner fundamentally differs from the current approaches by seamlessly integrating reasoning and learning: it combines the three components within a single unified framework. These components provide mutual self-supervision and enable the model to learn global calibration directly from data.*

MS1/MS2 To improve performance on the task of molecular formula inference, previous work has also included the use of tandem mass spectra (Goldman et al. 2023; Dührkop et al. 2019; Schmid et al. 2023; Hong et al. 2024). Mass spectra on isotopologues is often called MS1 while tandem mass spectra, often called MS2, are produced by measuring ion fragments rather than full compounds. Candidate formulas ranking inferred from MS1 data can be further refined based on how well they explain the observed fragmentation pattern shown on the MS2 spectrum. Our work focuses on solving the task of molecular formula inference using only MS1 data, without MS2 data. In real applications, MS1 data are faster and easier to obtain than MS2, and MS2 data is often unavailable. In addition, molecular formulas inferred from MS1 data greatly improve the scope and accuracy of molecular formula inference from MS2 data, when it is available. Thus, a high-quality method using solely MS1 data can enable scalable automatic molecular formula inference.

Artificial Intelligence for Combinatorial Scientific Reasoning AI for scientific discovery (Wang et al. 2023) has been advancing rapidly. A recent breakthrough is AlphaFold (Jumper et al. 2021), a deep learning method that solves the protein folding problem with remarkable accuracy. While learning algorithms excel in continuous domains, they often struggle with problems that are discrete and highly combinatorial in nature. AI reasoning techniques have long been applied to such combinatorial scientific problems. One early example is Dendral, considered the first expert system, which combined mass spectrometry data with a chemistry knowledge base to infer possible chemical structures responsible for the observed spectra (Lindsay 1980; Buchanan and Feigenbaum 1981). Other notable applications include protein sequence design for inverse folding (Vucinic et al. 2020), marker genotyping incompatibility detection in complex pedigrees (Sanchez, de Givry, and Schiex 2008; O'Connell and Weeks 1999), and DNA word design (Codish, Frank, and Lagoon 2017; Frutos et al. 1997). More recently, researchers have begun integrating deep learning with scientific reasoning under uncertainty, addressing challenges such as crystal phase mapping (Chen et al. 2021), multi-species bird population modeling (Chen et al. 2016), and symbolic discovery of ordinary differential equations (Jiang, Nasim, and Xue 2025). In this vein, our study explores the synergy between deep learning and combinatorial reasoning in solving the probabilistic coin change problem and its real-world counterpart: the MS1-to-MF task.

5 Experiments

We empirically evaluate DeepProReasoner on solving the PCCP in the two applications: controlled synthetic settings of Candy Mix, and real-world MS1-to-MF instances.

5.1 Candy Mix Problem

Experimental Setup We study the PCCP in controlled synthetic settings of the Candy Mix problem by varying the number of candy types, weight distributions, and number of sampled bags. To analyze empirical hardness, we focus on scenarios with 3 and 4 candy types, which yield 17,576 and 456,976 possible compositions (with 0–25 candies per type), respectively. Each dataset is split 80-20 for training and testing. In semi-supervised settings, up to 10,000 training examples are labeled. For each setting, we conduct 5 different runs with different random seeds and report mean accuracy with standard variation.

Approaches DeepProReasoner's encoder is a 4 layer gated recurrent unit (GRU) network with hidden layer dimension of 512. We use the RNN to encode the empirical PMF into a fixed 512 length vector. We then use a linear layer to map the latent vector to a count prediction with the same size as the number of candy types. Mean absolute error is used to compare the reconstructed and the empirical PMF. Adam optimizer (Kingma 2014) with default parameters ($\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$) is used for training. For the unsupervised training, each batch contains 64 unlabelled examples. The learning rate starts at 0.0001 and is decayed by half every 20 epochs. We stop the training at 100 epochs.

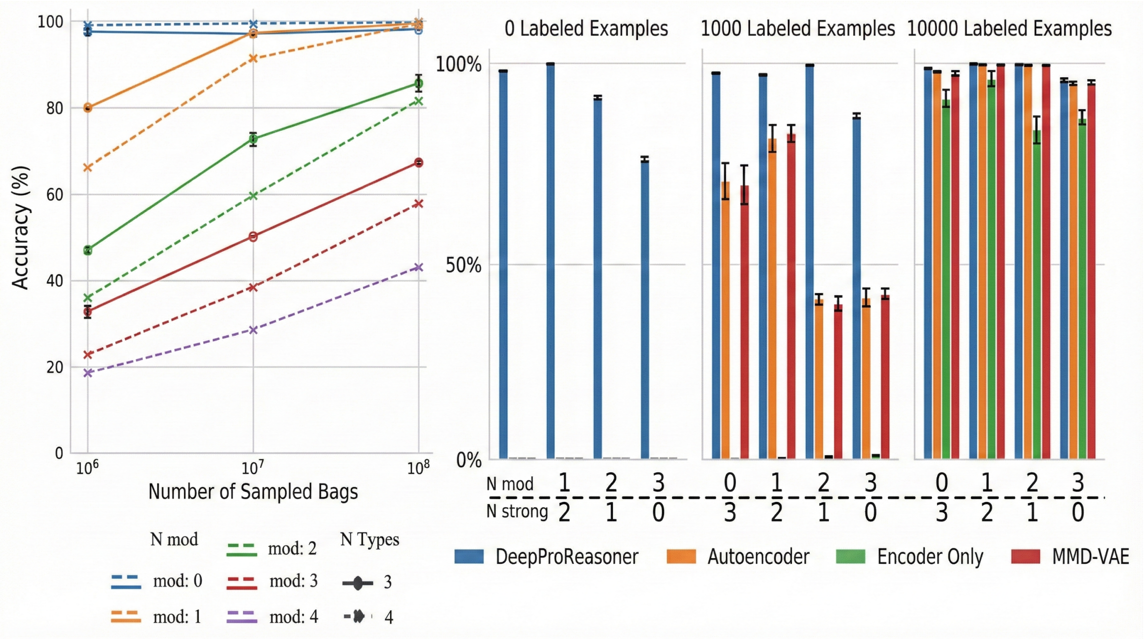

Figure 4: Empirical Candy Mix evaluation (mean ± SD). **Left**: Empirical complexity for various settings: number of candy types (N=3, solid lines; N=4 dashed lines); Candy type weight distribution skewness characterized by its modal probability ("strong" (0.95) or "mod" (0.75)). As an example, the solid blue line corresponds to bags with 3 candy types, 0 of which have "mod" weight distribution, so they all have "strong" skewness. **Right**: Performance of learning-based approaches in semi-supervised settings varying the amount of labeled data (0 left; 1,000 middle; 10,000 right), with 3 candy types and 10^8 bags. On the unsupervised setting, DeepProReasoner attains high performance whereas the baselines completely fail. Additionally, DeepProReasoner outperforms baselines in all supervision settings.

For the semi-supervised experiments, we adopt a balanced batching strategy where each minibatch consists of 32 labeled and 32 unlabeled samples. The unsupervised loss is computed only on the unlabeled data. This ensures that each training iteration includes sufficient supervision to guide model updates effectively. Furthermore, we include a warmup phase at the start of training, during which the learning rate is set to a higher value of 5×10^{-3} for the first 1% of total epochs. All models are trained using a single NVIDIA A100 GPU with 80GB of memory.

We evaluate the single-encoder variant of the DeepProReasoner against various baselines, a standard autoencoder (AE), an MMD-VAE (Louizos et al. 2015), and an encoder-only model. Each baseline is designed to isolate and assess the contributions of specific components within the DeepProReasoner framework, thus we maintain the same neural architectures where applicable. Specifically, the autoencoder and MMD-VAE evaluate the impact of using the differentiable combinatorial probabilistic reasoner over a fully learned decoder. The encoder-only approach evaluates the impact of reconstruction instead of or in addition to direct supervision. The same encoder, optimizer, learning rate scheduler, and batching strategy as the DeepProReasoner are used for all learning-based baselines.

The AE and MMD-VAE operate by taking in the latent vector as input and predicting the original observation with a learnable decoder. The decoder is a transformer that maintains embeddings for each candy type, as well as each aggregated weight of interest. The transformer has 4 layers, with hidden size of 128, and feed-forward dimension of 512. Ultimately, the transformer is used to generate a predicted PMF with the same length as the size of the input. The standard autoencoder uses the mean absolute error (MAE) between the observations and the generated PMF as the loss term. The MMD-VAE additionally uses Maximum Mean Discrepancy (MMD) to align the aggregated latent variables with a prior distribution as its regularization loss.

In the semi-supervised setting, we add the MAE between the predicted and true count vectors, across examples where the labels are present, to the loss function. The encoder only approach is trained using only this loss.

Results and Discussion Figure 4 shows results for the Candy Mix problem. On the left, we examine the performance of DeepProReasoner on 3 and 4 types of candies. Each type's weight distribution is characterized by its modal probability: "strong" (0.95) or "moderate" (0.75). All combinations are explored, simulating real-world settings like isotopic or manufacturing variations. DeepProReasoner attains near-perfect performance when the probability distributions on the candy weights are strongly skewed. (Note: for the real-world MS1-to-MF problem (Section 5.2), the element weight distributions are even more skewed than the all "strong" setting, indicating the strength of DeepProReasoner.) We vary the number of sampled bags, with more samples bringing the empirical distribution closer to the theoretical distribution, thus giving cleaner data, and higher accuracy. DeepProReasoner's performance degrades with not-strongly skewed weight distributions and noisier data (i.e., fewer sampled bags).

Figure 4 (right) compares DeepProReasoner with the baselines on problems with 3 candy types and 10^8 sampled bags. Note the baselines all have 0% unsupervised accuracy. We further explore semi-supervised settings.

5.2 MS1-to-MF

We showcase the performance of DeepProReasoner for inferring molecular formulas (MFs) accessing only **unlabeled** real-world MS1 data.

Data A single sample of the model organism *C. elegans*, and a single sample of Mouse Liver, containing thousands of unknown compounds were prepared for Liquid Chromatography – Mass Spectrometry analysis (LC-MS). Data were collected by the ThermoFisher Scientific research lab yielding 27,422 unlabeled spectra for the *C. elegans* dataset of and 17,363 for Mouse Liver. representing salient compounds. The datasets were obtained with an ultrahigh resolution of 1,067,105 full width at half maximum (FWHM) at 200 Da. Metabolomics experts from the Department of Chemistry and Chemical Biology at Cornell University labeled 381 spectra (235 unique MFs) in the *C. elegans* set and 162 spectra (113 unique MFs) in the Mouse Liver set with the correct MF, evenly split into validation and test sets. Examples were labeled over three months, underscoring the need for unsupervised methods and precluding (semi-)supervised approaches. The two datasets are different in the MFs they cover: only 17 are in common among the labels. The Mouse Liver set is typical for samples from human or animal sources, but poses significant challenges due to both the molecular mass range and elemental composition. Many compounds exhibit high molecular weights, which substantially increase the combinatorial space of candidate formulas. Moreover, most molecules are composed primarily of common elements such as C, H, N, O, and P, which yield subtle isotopic patterns that are difficult to distinguish. In contrast, elements like S or Cl produce distinctive isotopic signatures that make formula inference more tractable. Consequently, accurately identifying elemental counts becomes especially difficult when discriminative isotopic features are limited.

Approaches We train DeepProReasoner (dual encoder) separately on the two datasets and compare against SOTA methods for MS1-to-MF, with standard machine learning baselines failing to identify even a single formula correctly in this unsupervised setting. The training settings are similar to those for the Candy Mix problem as described in Section 5.1, with the same optimizer and learning rate scheduler. The PMF encoder is a GRU network with 4 layers and 1024 hidden dimension each layer. A last linear layer maps the 1024-d vector to a latent vector of dimension 64. The training of DeepProReasoner is efficient: on the larger *C. elegans* set, it took **2 hours and 7 minutes** for the model to converge, and on the smaller Mouse Liver set, it took **1 hours and 18 minutes**, both on a single NVIDIA A100 GPU with 80GB memory. The formula encoder uses a 64-d trainable embedding vector to encode each element. A candidate formula is encoded by multiplying its count vector by this embedding matrix. Then, it goes through a 2-layer MLP with hidden and output dimension of 64. We use the validation set to choose these hyperparameters and model checkpoint. Comparisons include SIRIUS 4.0 (Dührkop et al. 2019) and MZmine 3 (Schmid et al. 2023), two standard domain-specific methods, as well as a set of heavily studied algorithmic baselines. All approaches leverage a set of candidate formulas determined to be within 5ppm of the monoisotopic peak, generated using HR2 (Fiehn 2007). The candidate sets were generated with 7 elements, C, H, O, N, S, P, and Cl, the most commonly occurring elements in small organic and bio-organic molecules, including metabolites, drugs, and natural products. This selection ensures broad chemical coverage in molecular formula inference, covering $> 99\%$ of the mass of living matter (Emsley 1998). For the algorithmic baselines, the isotopic patterns of each candidate formula are computed using pyteomics (Goloborodko et al. 2013). The generated spectra are compared against the experimental spectra using various similarity metrics commonly used in the mass spectrometry literature, such as Hungarian and Shifted cosine, mean squared error, mean absolute error, and Jensen-Shannon divergence. Methods are evaluated by how often the correct formula is ranked within the top 1 to 3 candidates. For each data set, we randomly subsampled 50% of the test set without replacement five times and report mean performance with standard deviation across the five subsets.

Results and Discussion We present results on the real-world MS1 data in Figure 5. DeepProReasoner attains high performance across the board. Notably, on the *C. elegans* set, DeepProReasoner achieves a top 1 accuracy of 92.62% compared to 80.78% of the next best method of MAE similarity search. On the more challenging Mouse Liver set, DeepProReasoner maintains a top 1 accuracy of 92.84% while the next-best baseline JS struggles at 63.21%. The high top-1 accuracy means that DeepProReasoner can automatically infer a single correct molecular formula with high confidence instead of giving the user a list of examples to sift through manually. Moreover, DeepProReasoner is much more robust in the more difficult regime because it combines structured probabilistic reasoning with deep representation learning, enabling the model to disambiguate subtle isotopic differences and learn global calibration directly from data. This joint reasoning-and-learning framework allows DeepProReasoner to outperform traditional methods that rely solely on deterministic scoring or local similarity metrics.

6 Conclusion

We introduce the Probabilistic Coin Change Problem, a novel generalization of the classical CCP designed to model combinatorial inverse problems where latent discrete structures generate noisy, aggregate observations via random processes. To address PCCP, we propose DeepProReasoner, a fully differentiable, end-to-end framework for unsupervised combinatorial probabilistic reasoning. DeepProReasoner fundamentally differs from existing approaches by seamlessly integrating reasoning and learning. It unifies

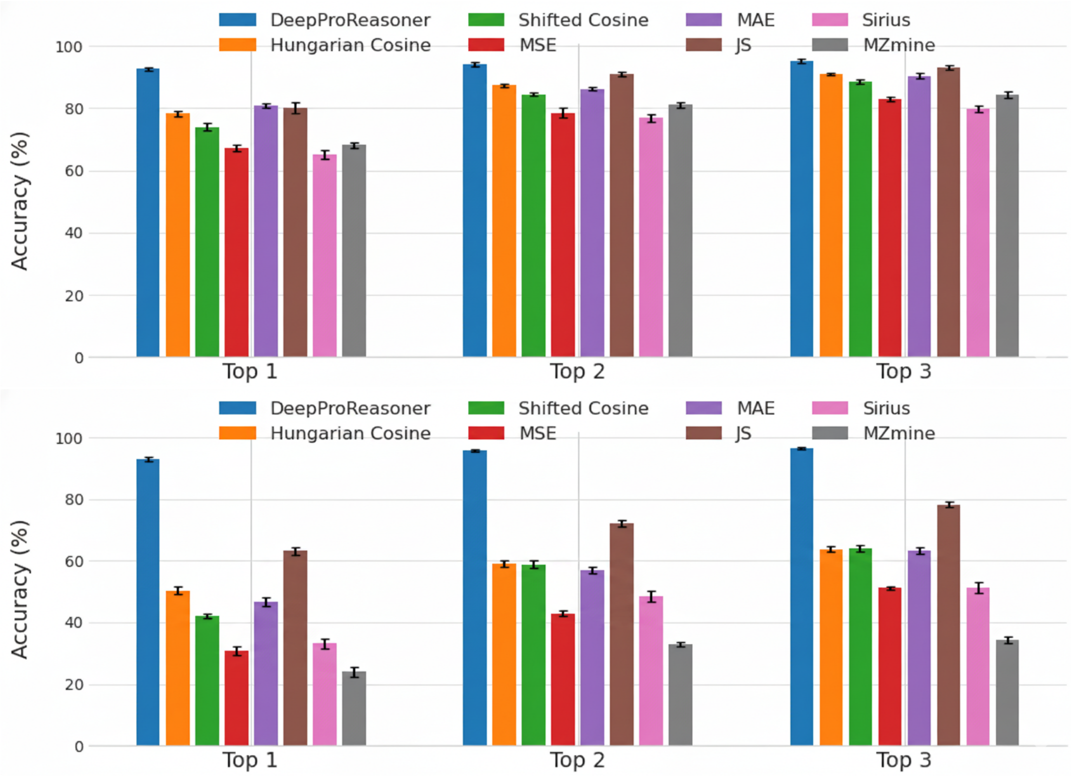

Figure 5: Top K formula accuracy (mean ± SD) on real-world MS1 data. **Top**: *C. elegans*. **Bottom**: Mouse Liver. We compare DeepProReasoner with the current state-of-the-art methods MZmine 3 and SIRIUS 4 and additional baselines to test spectrum similarity metrics that are popular in the mass spectrometry literature, Hungarian and Shifted cosine, and measures for comparing discrete probability distributions, mean absolute error (MAE), mean squared error (MSE), and Jensen-Shannon divergence (JS). DeepProReasoner attains high performance especially in top-1 accuracy, outperforming the baselines by at least 11.8 points on the *C. elegans* set. The advantage of DeepProReasoner is even more significant on the more challenging Mouse Liver dataset, outperforming the baselines by at least 29.6 points.

three core tasks, combinatorial search, differentiable probabilistic reasoning, and representation learning, within a single deep-learning framework. These components provide mutual self-supervision and enable the model to learn global calibration directly from data. More specifically, DeepProReasoner learns to infer latent structures by integrating a novel differentiable probabilistic forward model of the PMF of observed sums, enabling training via a reconstruction loss without requiring ground-truth labels. We empirically demonstrate DeepProReasoner's effectiveness on two instances of the PCCP: controlled settings of the synthetic Candy Mix problem, and molecular formula inference from real-world MS1 data. On the Candy Mix problem, DeepProReasoner successfully recovered latent compositions across various settings without supervision, elucidating the empirical complexity of PCCP in various settings. On MS1-to-MF, DeepProReasoner substantially outperformed existing state-of-the-art methods, achieving high top-1 accuracies across different datasets. This highlights its potential for high-throughput, automated molecular formula inference.

Acknowledgments

This project is partially supported by an AI2050 Senior Fellowship, a Schmidt Sciences program (to C.P.G.); the National Science Foundation (NSF) (to C.P.G.); the National Institute of Food and Agriculture (USDA/NIFA) (2023-67021-39829 to C.P.G.); the Air Force Office of Scientific Research (AFOSR) (FA9550-23-1-0322 to C.P.G); and National Institute of Health (NIH) (R35 GM131877 to F.C.S).

References

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Böcker, S.; Letzel, M. C.; Lipták, Z.; and Pervukhin, A. 2009. SIRIUS: decomposing isotope patterns for metabolite identification. *Bioinformatics*, 25(2): 218–224.

Böcker, S.; Lipták, Z.; Martin, M.; Pervukhin, A.; and Sudek, H. 2008. DECOMP—from interpreting mass spectrometry peaks to solving the Money Changing Problem. *Bioinformatics*, 24(4): 591–593.

Buchanan, B. G.; and Feigenbaum, E. A. 1981. DENDRAL and Meta-DENDRAL: Their applications dimension. In *Readings in artificial intelligence*, 313–322. Elsevier.

Chen, D.; Bai, Y.; Ament, S.; Zhao, W.; Guevarra, D.; Zhou, L.; Selman, B.; van Dover, R. B.; Gregoire, J. M.; and Gomes, C. P. 2021. Automating crystal-structure phase mapping by combining deep learning with constraint reasoning. *Nature Machine Intelligence*, 3(9): 812–822.

Chen, D.; Xue, Y.; Chen, S.; Fink, D.; and Gomes, C. 2016. Deep multi-species embedding. *arXiv preprint arXiv:1609.09353*.

Cho, K.; Van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; and Bengio, Y. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Claesen, J.; Rockwood, A.; Gorshkov, M.; and Valkenborg, D. 2025. The isotope distribution: A rose with thorns. *Mass Spectrometry Reviews*, 44(1): 22–42.

Codish, M.; Frank, M.; and Lagoon, V. 2017. The DNA Word Design Problem: A New Constraint Model and New Results. In *IJCAI*, 585–591.

Cowen, L. J.; Cowen, R.; and Steinberg, A. 2008. Totally greedy coin sets and greedy obstructions. *the electronic journal of combinatorics*, R90–R90.

Dührkop, K.; Fleischauer, M.; Ludwig, M.; Aksenov, A. A.; Melnik, A. V.; Meusel, M.; Dorrestein, P. C.; Rousu, J.; and Böcker, S. 2019. SIRIUS 4: a rapid tool for turning tandem mass spectra into metabolite structure information. *Nature methods*, 16(4): 299–302.

Emsley, J. 1998. *The Elements*. Oxford: Clarendon Press, 3rd edition.

Fiehn, O. 2007. Molecular Formula Generator - Seven Golden Rules. https://fiehnlab.ucdavis.edu/projects/seven-golden-rules/molecular-formula-generator. Accessed: 2025-08-02.

Frutos, A. G.; Liu, Q.; Thiel, A. J.; Sanner, A. M. W.; Condon, A. E.; Smith, L. M.; and Corn, R. M. 1997. Demonstration of a word design strategy for DNA computing on surfaces. *Nucleic Acids Research*, 25(23): 4748–4757.

Goldman, S.; Xin, J.; Provenzano, J.; and Coley, C. W. 2023. Mist-cf: Chemical formula inference from tandem mass spectra. *Journal of Chemical Information and Modeling*, 64(7): 2421–2431.

Goloborodko, A. A.; Levitsky, L. I.; Ivanov, M. V.; and Gorshkov, M. V. 2013. Pyteomics—a Python framework for exploratory data analysis and rapid software prototyping in proteomics. *Journal of The American Society for Mass Spectrometry*, 24(2): 301–304.

Gomes, C. P.; Selman, B.; and Gregoire, J. M. 2019. Artificial intelligence for materials discovery. *MRS Bulletin*, 44(7): 538–544.

Graves, A.; Mohamed, A.-r.; and Hinton, G. 2013. Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing*, 6645–6649. Ieee.

Gupta, S.; Huang, B.; and Impagliazzo, R. 2024. The Greedy Coin Change Problem. *arXiv preprint arXiv:2411.18137*.

Hartmanis, J. 1982. Computers and intractability: a guide to the theory of np-completeness (michael r. garey and david s. johnson). *Siam Review*, 24(1): 90.

Hong, Y.; Li, S.; Ye, Y.; and Tang, H. 2024. FIDDLE: a deep learning method for chemical formulas prediction from tandem mass spectra. *bioRxiv*, 2024–11.

Jiang, N.; Nasim, M.; and Xue, Y. 2025. Active Symbolic Discovery of Ordinary Differential Equations via Phase Portrait Sketching. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 17626–17634.

Jumper, J.; Evans, R.; Pritzel, A.; Green, T.; Figurnov, M.; Ronneberger, O.; Tunyasuvunakool, K.; Bates, R.; Žídek, A.; Potapenko, A.; et al. 2021. Highly accurate protein structure prediction with AlphaFold. *nature*, 596(7873): 583–589.

Kind, T.; and Fiehn, O. 2007. Seven Golden Rules for heuristic filtering of molecular formulas obtained by accurate mass spectrometry. *BMC bioinformatics*, 8: 1–20.

Kingma, D. P. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25.

Lindsay, R. K. 1980. Applications of artificial intelligence for organic chemistry: the DENDRAL project. *(No Title)*.

Louizos, C.; Swersky, K.; Li, Y.; Welling, M.; and Zemel, R. 2015. The variational fair autoencoder. *arXiv preprint arXiv:1511.00830*.

Merchant, A.; Batzner, S.; Schoenholz, S. S.; Aykol, M.; Cheon, G.; and Cubuk, E. D. 2023. Scaling deep learning for materials discovery. *Nature*, 624(7990): 80–85.

Nagao, T.; Yukihira, D.; Fujimura, Y.; Saito, K.; Takahashi, K.; Miura, D.; and Wariishi, H. 2014. Power of isotopic fine structure for unambiguous determination of metabolite elemental compositions: in silico evaluation and metabolomic application. *Analytica chimica acta*, 813: 70–76.

O'Connell, J. R.; and Weeks, D. E. 1999. An optimal algorithm for automatic genotype elimination. *The American Journal of Human Genetics*, 65(6): 1733–1740.

Sanchez, M.; de Givry, S.; and Schiex, T. 2008. Mendelian error detection in complex pedigrees using weighted constraint satisfaction techniques. *Constraints*, 13(1): 130–154.

Schmid, R.; Heuckeroth, S.; Korf, A.; Smirnov, A.; Myers, O.; Dyrlund, T. S.; Bushuiev, R.; Murray, K. J.; Hoffmann, N.; Lu, M.; et al. 2023. Integrative analysis of multimodal mass spectrometry data in MZmine 3. *Nature biotechnology*, 41(4): 447–449.

Silver, D.; Schrittwieser, J.; Simonyan, K.; Antonoglou, I.; Huang, A.; Guez, A.; Hubert, T.; Baker, L.; Lai, M.; Bolton, A.; et al. 2017. Mastering the game of go without human knowledge. *nature*, 550(7676): 354–359.

Vucinic, J.; Simoncini, D.; Ruffini, M.; Barbe, S.; and Schiex, T. 2020. Positive multistate protein design. *Bioinformatics*, 36(1): 122–130.

Wang, H.; Fu, T.; Du, Y.; Gao, W.; Huang, K.; Liu, Z.; Chandak, P.; Liu, S.; Van Katwyk, P.; Deac, A.; et al. 2023. Scientific discovery in the age of artificial intelligence. *Nature*, 620(7972): 47–60.

Zeni, C.; Pinsler, R.; Zügner, D.; Fowler, A.; Horton, M.; Fu, X.; Shysheya, S.; Crabbé, J.; Sun, L.; Smith, J.; et al. 2023. Mattergen: a generative model for inorganic materials design. *arXiv preprint arXiv:2312.03687*.