A Pareto-optimal compositional energy-based model for sampling and optimization of protein sequences

Anonymous Author(s)
Affiliation
Address
email

Abstract

Generative models are one of the most prominent machine learning methods leveraged by engineering and life sciences. Nevertheless, besides resemblance to the data we wish to sample, oftentimes our aim is to propose new data points that will satisfy, not one, but multiple properties of interest. Such aspiration becomes even more challenging when those properties are independent or orthogonal to each other. In this work, we propose a *Pareto-compositional energy-based model (pcEBM)*, a generative framework for sampling new datapoints that simultaneously adhere to various constraints and optimize distinct properties. pcEBM emerges naturally from an integrated multi-gradient energy-based sampling framework and is shown to lead to practical benefits in a series of both real world (antibody-design) empirical analysis.

1 Introduction

Generative models have recently gained increased interest within the machine learning community. Part of the excitement stems from the realistic-looking images created by generative adversarial networks [13], variational autoencoders [18], energy-based models, (EBMs) [9] or more recently and diffusion models [28] that made headlines [25, 5]. However, there are far less success stories in industry applications [6, 15]. There are multiple reasons for this fallback: not all datasets are in image format; there is a lack of evaluation protocols and metrics for synthetic data; it is still challenging to control a generative model; and finally, we are still not able to generate or imagine samples that are (very) different from what we have seen in training. Overall, the lack of constraints and control in generative modeling limits the current use cases.

This manuscript is motivated by the need to accelerate the development and discovery of new drugs, namely, therapeutic antibodies, by incorporating machine learning methods in classical computational biology pipelines. Prominent generative models have already been proposed for such purposes, [10, 12, 24] to name a few. However, generating new samples without guidance or control, does not guarantee usability or success of the proposed molecules. Each molecule has to satisfy multiple properties such as affinity, aggregation, thermal stability [1, 30], before it even qualifies as a potential candidate for a clinical trail. Not taking such fundamental properties into consideration, leads to potential complications later on during clinical trials.

Motivated by this real-world challenge, we set out to propose a new energy-based generative method targeted at protein design and discovery that simultaneously takes into account multiple properties that an antibody (protein) has to satisfy. The concurrent adherence to multiple properties is a challenge in itself, as it often involves optimizing multiple conflicting objectives. In antibodies, for instance, naive optimization of affinity can be achieved at the expense of other desirable developability properties.

Submitted to NeurIPS 2021 AI for Science Workshop.

Unfortunately, when one cares about the optimization of multiple conflicting properties, it is often impossible to find a single sample that satisfies all the objectives simultaneously. Instead, we wish to propose a set of diverse data points from the *Pareto front* [3] that correspond to different choices for the different objective functions. By doing so, one gains a global perspective the optimal trade-off between objectives and can select a molecule according to their preference. To achieve this, we rely on recent advances in multi-objective optimization (MOO)[27, 22] as well as compositional sampling with EBMs [9], to build a Pareto-optimal compositional EBM, *pcEBM*.

In what follows, we first present the related work and background on multiple gradient descent (MGD) [7, 22] and EBMs in section 2, which compose the pcEBM described in section 3. In section 4 we include empirical evaluation and discussion, before concluding in section 5.

2 Background

2.1 Problem setup

Antibodies (as any protein) are composed of 2 chains of amino acids (AA), which can be represented as sequences of characters (22 characters corresponding to each of the 22 possible amino acids), typically of total length $L \sim 250$. We will denote the sequences as $x = (x_1, ...x_L)$ where each x_l corresponds to the AA at position l. For each of those sequences, we measure m properties, such that $f_i : \mathbb{R}^L \to \mathbb{R}, i \in 1..m$. Our goal is to generate new sequences x^* with prefered values for each of the m properties. Since we can not fully satisfy all of the properties/functions simultaneously (they may be conflicting with each other), we are interested in finding samples which can not be further improved simultaneously for all the objectives, yielding the notion of *Pareto optimality* [4].

2.2 Multi objective optimization

A straightforward approach to solving MOO is *Linear Scalarization* [26] based on a preference vector $\lambda = [\lambda_1, ...\lambda_m]$ from the probability simplex on $[m]$, i.e., $S = \{\lambda : \sum_i = 1^m \lambda_i = 1, \lambda_i \geq 0, i \in [m]\}$. Each $\lambda \in S$ leads to a weighted objective function $f_\lambda(x) = \sum_i = 1^m \lambda_i f_i(x)$ and its minimizer $x^*_\lambda = argmin_x f_\lambda(x)$. As we evaluate on λ from a grid of S, we hope that the corresponding x^*_λ approximates the Pareto front. This simple to implement strategy suffers from problems such for example solutions that can only lie on the convex envelope of the Pareto front, consequently making this plausible approach only in cases when the Pareto front is convex [22].

A recent alternative for non-convex Pareto fronts is *Multiple Gradient Descent, (MGD)*, [7], based on the idea to iteratively update the variable x along a direction that ensures that all the objectives are decreased simultaneously, i.e. guids the optimization towards a Pareto improvement. We denote the gradient of the i-th objective as $g_i(x) = \nabla_i f_i(x)$. With gradient descent, we update the variable $x^k \leftarrow x^{k-1} - \epsilon g^*(x)$ where $g^*(x)$ is a vector to be determined and ϵ is a small step size. Following Taylor approximation $\langle g_i, g^* \rangle \approx -(f_i(x^k) - f_i(x^{k-1}))/\epsilon$ represents the decreasing rate of f_i when we update x along direction $g^*(x)$. In MGD, g^* is chosen to maximize the slowest decreasing rate among all the objectives, that is:

$$g^*(x) \propto \underset{g \in \mathbb{R}^d}{\text{argmax}} \{ \min_{i \in [m]} \langle g, g_i(x) \rangle, s.t. ||g|| \leq 1 \} \tag{1}$$

By doing so, $g^*(x)$ is pushed to have positive inner products with all $g_i(x)$. If that is impossible, $\{g_i(x)\}_{i=1}^m$ will contain conflicting directions and we will get $g^*(x) = 0$ which will terminate the procedure. Using Lagrangian duality, [7] has shown that the above objective has the optimal value $g^*(x) \approx \sum_{i=1}^m \lambda_i^* g_i(x)$ where $\{\lambda_i g_i(x)\}_{i=1^m}$ is the solution of

$$\min_{\lambda_i} || \sum_{i=1}^m \lambda_i g_i(x) || s.t. \sum_{i=1}^m \lambda_i = 1, \lambda_i \geq 0, \forall i.$$

The above optimisation problem has a closed form solution when $m = 2$, and a fast algorithm for $m > 2$ [27, 17]. By construction, when the step size is small, MGD decreases monotonically all objectives simultaneously and will terminate at local Pareto point.

2.3 Compositional energy-based models

Although EBMs have been around since many years ago [20, 19], they have recently been rediscovered in the quest of generative modeling. More concretly EBMs learn to represent data by approximating an unnormalized probability distribution across. They do so by fitting an energy function $E_\theta(x)$ for each data point, parameterized by a neural network, which outputs a scalar real energy such that the model distribution is:

$$p_\theta(x) \propto e^{-E_\theta(x)}.$$

Training an EBM on a given data distribution is done using contrastive divergence [16]. Besides approximating data distributions, EBMs have the inherent generative property and so we can use them to sample new data points x - from p_θ during both training and generation. For this we use MCMC based on Langevin dynamics [29]. In this framework samples are initialized from gaussian/uniform random noise followed by iterative refinement:

$$x = x^{k-1} - \frac{\eta}{2}\nabla_x E_\theta(x^{k-1}) + \omega^k, \omega \sim \mathcal{N}(0, \sigma^2) \tag{2}$$

where k is the $k-th$ iteration step and η is the step size. This sampling procedure allows us to use the gradient of a combined distribution to generate samples from distributions composed of p_θ and other distributions. We use this ability to generate from composed distributions, consisted of m concepts c, as proposed by [8], an approach naturally named compositional EBM (cEBM). [8] propose multiple variations for executing different logical operations over properties of interest such as conjunction, negation and disjunction. In the rest of this paper we only consider conjunction, which corresponds to combining individual EBMs as product of experts:

$$p(x|c_1 \wedge c_2 \wedge ... \wedge c_m) = \prod_i p(x|c_i) \propto e^{-\sum_i E(x|c_i)}$$

Furthermore, the sampling from a composed models is correspondingly adapted:

$$x = x^{k-1} - \frac{\eta}{2}\nabla_x \sum_i E_\theta(x^{k-1}|c_i) + \omega^k, \omega \sim \mathcal{N}(0, \sigma^2) \tag{3}$$

3 Pareto compositional EBM

[22] improves MGD multi objective optimization via sampling, encouraged by the opportunity to access a diverse set of solutions i.e. Pareto front (not only a single point local Pareto estimate), and importantly, compared to existing methods, without relaying on predefined preference functions (as in Linear scaling). The main idea is to incorporate the MGD minimum norm gradient updates within sampling optimization procedures as Stein variational gradient descent [21] or Langevin dynamics [29]:

$$x^k \leftarrow x^{k-1} - \eta g^*(x) + \sqrt{2}\alpha\omega, \omega \sim \mathcal{N}(0, \sigma^2). \tag{4}$$

Where α is positive constant scaling the Gaussian noise ω.

Remembering that $g^*(x) \approx \sum_{i=1}^m \lambda_i^* g_i(x)$ and that $g_i(x) = \nabla_i f_i(x)$, and replacing the gradients in Equation 2.3 with minimum norm gradients, leads us to the pareto optimal compositional EBM, *pcEBM*. When generating new sequence x^*, weather starting form a random noise or improving on an existing one, at each iteration, if x^* is far away from the Pareto front with large multiple gradients $||g^*(x^*)||$, Equation 3 will mainly drive x^* towards the Pareto front. On the other hand, when x^* is close to the Pareto front with small $||g^*(x)||$, the noise will dominate and let x^* randomly diffuse the

set. Differently from [22] that uses MGD with sampling for multi property optimization, here, we leverage the joint framework of MGD and cEBM for multi-property sampling.

4 Empirical evaluation

Next, we empirically evaluate the four considered baselines: multiple gradient descent, (*MGD*), compositional EBM, (*cEBM*), linearly scaled cEBM (*ls-cEBM*) and pareto compositional EBM, (*pcEBM*).

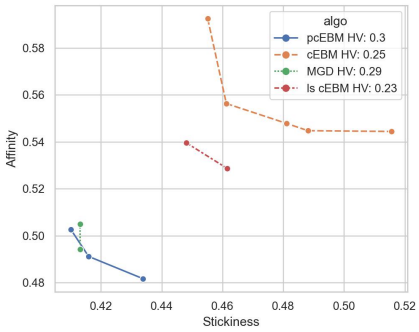

Figure 1: Pareto fronts and estimated hyper volume for all baselines with $\eta = 1e-2, n_s = 400$.

Experimental setup We wish to generate plausible antibody sequences. Antibodies are proteins constructed from a sequence of amino-acids whose functional properties depend on the contents and ordering of that sequence. Therefore, in order to sample new, or improve properties of existing antibodies, we choose to implement our baselines with sequence based convolution networks [11]. More details on the architecture can be found in Appendix.

The datasets we use are from both public and proprietary sources. We consider three significant properties for the generated sequences: (i) similarity to known human antibodies, *Ab-like*, (ii) affinity, or binding to antigen targets of interest *Aff* and (iii) stickiness or non-specificity of a the molecule *Stc*.

To compare the performance of the baselines in terms of satisfying multiple-properties, we include a metric from the multi-objective optimization literature - Hyper Volume (*HV*), [14], computed over the energy scores per property model. Additionally, to validate the similarity of the new sampled sequences (a proxy for realness) to the ones seen during the training of each property energy based model; we include edit distances (e_{dist}) [23]. The edit-distance represents the score of the best possible alignment between the two genetic sequences over all possible alignments. The computation of both metrics is included in appendix.

Next, we empirically evaluate the four considered baselines: multiple gradient descent, (*MGD*), compositional EBM, (*cEBM*), linearly scaled cEBM (*ls-cEBM*) and pareto compositional EBM, (*pcEBM*).

First, we train three separate EBMs one for each property of interest. Once they have been trained to convergence, we use the same models for sampling new sequences with each of the baselines using Langevin dynamics. The differences in implementation are as follows: *MGD* does not use any noise, at each iteration, simply optimizes the gradients, *cEBM* uses LD but has uniform weights for all properties, *ls-cEBM* is same as cEBM but including domain-informed weights, and finally *pcEBM* learns the optimal weights for each property per sequence. Each baseline has the same set of important hyper parameters, namely *step size* - η, *number of steps* - k.

Results and discussion To quantitatively evaluate the effectiveness of the proposed methods, we address the following research questions:

- **Q.1** Can we use EBMs to generate valid/reasonable antibody sequences?
- **Q.2** Does sampling with Langevin dynamics propose more Pareto optimal sequences?
- **Q.3** Does pcEBM have an advantage over cEBM?
- **Q.4** Starting from a seed sequence, can we improve a property of interest using (p)cEBM?

To tackle **Q.1**, we include Table 4. We notice that multi property EBMs have smaller distances to all properties compared to the single objective ones as expected have sequences most similar to the property they correspond for, but are less like the other objectives. Additionally from the

	$e_{dist}(generated, Ab-like)$	$e_{dist}(generated, Stc)$	$e_{dist}(generated, Aff)$
single-property EBM			
Ab-Like EBM	98.74 (±19.25)	93.84 (±19.05)	103.3 (±17.8)
Stc EBM	84.68 (±15.78)	**69.7 (± 16.42)**	88.29 (± 5.72)
Aff EBM	88.51 (±16.25)	81.4 (±15.6)	86.73 (±6.31)
$\eta = 40$			
MGD	83.51 (±16.1)	71.49 (±15.7)	85.84 (±16.3)
cEBM	92.96 (± 19.03)	86.86 (± 17.99)	97.97 (± 17.28)
ls-cEBM	92.4 (± 19.3)	87.14 (± 17.94)	97.93 (± 17.25)
pcEBM	**82.24 (±16.16)**	71.47 (± 15.82)	**85.76 (± 16.34)**
$\eta = 1e^{-2}$			
MGD	91.46 (±16.23)	84.02 (±15.43)	92.91 (±16.48)
cEBM	219.23 (±4.43)	222.47 (± 4.45)	221.99 (± 4.4)
ls-cEBM	218.78 (±4.66)	218.13 (±4.58)	218.62 (±4.73)
pcEBM	89.35 (± 14.64)	79.67 (± 13.99)	89.04 (± 14.37)

Table 1: Edit distance (mean with standard deviation in brackets) to sequences in the training set with property of interest. Results in bold are best per columns, smaller is better, considering e_{dist} as a proxy for similarity to real data with the specific property.

multi-objective baselines, pcEBM generates sequences closest to the training datasets. The difference in the results regarding different step sizes in the gradient descent, hint a faster convergence of the pcEBM variant. We confirm this in Figure 2, which presents the energy scores at each iteration step for both cEBM and pcEBM for the same step size $\eta = 1e^{-2}$. For the same starting sequence, the slope pcEBM converges much faster. This result also answers partially **Q.3**.

To further investigate the motivation to use multiple gradients to achieve better Pareto optimally, we calculate the hyper volume across the three properties we wish to minimize for. Here we consider the energy of each properly EBM as proxy for the objective of interest [1] The advantage of pcEBM is noticeable only at smaller steps, but also the variance in both edit distance and hypervolume for different η is much smaller for pcEBM than others. However, if we turn to Figure 1, where we plot the pareto front obtained from candidate sequences from each baseline, we see that the pcEBM front has spread than competitors. Moreover, cEBM and pcEBM fonts are closer to the origin, which in out case of minimizing objectives shows the advantage of multi properly optimization with Langevin dynamics, i.e. our **Q.2**.

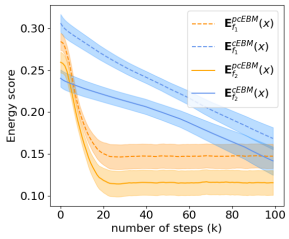

Figure 2: Energy scores per property f_i at each iteration for pcEBM and cEBM; same sequence and same step size $\eta = 1e^{-2}$.

Finally, for **Q.4** we wish to explore the possibility for improving properties of already existing antibodies by using the variants of compositional EBMs. Due to time and space limitations, in the current version we present results for the property of stickiness *Stc*. We start by screening the existing sequences to select only those with bad Stickiness scores. We have an external SeqCNN predictor (classifier) that acts as surrogate oracle for stickiness. We use that oracle to evaluate the probability for having a good Stc score, that is, the molecule is likel to not stick to anything in it's environment. We pass the selected sequences with measured (as well as predicted) bad scores, through cEBM and pcEBM respectively.Next, we use the surrogate oracle to evaluate the stickiness of the newly proposed designs. In Table 2, we see the improvements per baseline, i.e. The probability for a good score is increased by at least 30% with the EBMs. Figure 4 (a) and (b) offer an additional visual confirmation - they represents a 2D tSNE plot of all sequences with measured Stickiness, colored by their score, blue means good and red a bad

[1]This is a biased score since cEBM and ls cEBM have the advantage of sampling and evaluating wrt same model. A more fair comparison would be using external predictors as surrogates or wet-lab results to objectively evaluate.

	Stc. score
Seeds	0.59 ($\pm$ 0.26)
MGD	0.90 ($\pm$0.05)
cEBM	0.96 ($\pm$0.01)
ls-cEBM	0.96 ($\pm$0.05)
pcEBM	0.97 ($\pm$0.02)

Table 2: Average and std over 60 sequences with bad scores.

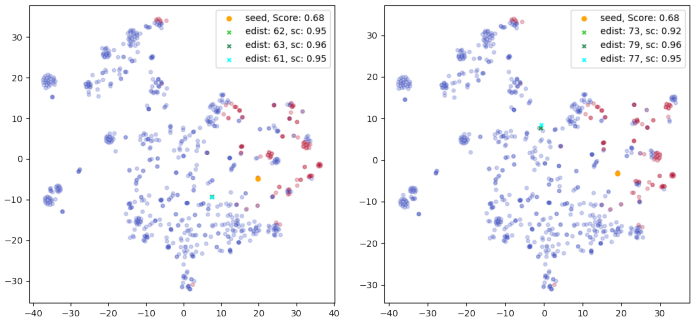

Figure 4: (a) cEBM (b) pcEBM

score. On the same plot, we overlay an example sequence (seed) with particularly bad stickiness and its corresponding proposed designs. We see that both cEBM and pcEBM improve the probability for a good stickiness score, and move the seed towards the section of the manifold covered by datapoints with good scores.

5 Conclusion

We propose a new approach to sampling data points that simultaneously satisfy multiple desired properties to the best possible extent, i.e. they are Pareto-optimal. We do so by integrating multiple gradients within compositional energy based models. In a series of experiments for generating and improving existing sequences, we validate the performance of the compositional EBMs as well as their Pareto optimality. The generality and modularity of pcEBM allow for multiple expansions we are excited to explore: (i) other data types, in particular graph structures suitable for molecules, (ii) incorporating other sampling frameworks such as Stein variational gradient descent [21], (iv) incorporating uncertainty estimates and (iv) pareto-optimal training for EBMs, to name a few.

References

[1] M. Bailly, C. Mieczkowski, V. Juan, E. Metwally, D. Tomazela, J. Baker, M. Uchida, E. Kofman, F. Raoufi, S. Motlagh, et al. Predicting antibody developability profiles through early stage discovery screening. In *MAbs*, volume 12, page 1743053. Taylor & Francis, 2020.

[2] J. Blank and K. Deb. pymoo: Multi-objective optimization in python. *IEEE Access*, 8:89497–89509, 2020.

[3] Y. Censor. Pareto optimality in multiobjective problems. *Applied Mathematics and Optimization*, 4(1):41–59, 1977.

[4] A. Chinchuluun and P. M. Pardalos. A survey of recent developments in multiobjective optimization. *Annals of Operations Research*, 154(1):29–50, 2007.

[5] J. Conard. How gpt-3 wrote a movie about a cockroach-ai love story. In *Wired*, 2022.

[6] D. Davis. Generative design is doomed to fail. In *Blog post*, 2020.

[7] J.-A. Désidéri. Multiple-gradient descent algorithm (mgda) for multiobjective optimization. *Comptes Rendus Mathematique*, 350(5-6):313–318, 2012.

[8] Y. Du, S. Li, and I. Mordatch. Compositional visual generation with energy based models. *Advances in Neural Information Processing Systems*, 33:6637–6647, 2020.

[9] Y. Du and I. Mordatch. Implicit generation and modeling with energy based models. *Advances in Neural Information Processing Systems*, 32, 2019.

[10] R. R. Eguchi, C. A. Choe, and P.-S. Huang. Ig-vae: Generative modeling of protein structure by direct 3d coordinate generation. *PLoS computational biology*, 18(6):e1010271, 2022.

[11] J. Gehring, M. Auli, D. Grangier, D. Yarats, and Y. N. Dauphin. Convolutional sequence to sequence learning. In *International conference on machine learning*, pages 1243–1252. PMLR, 2017.

[12] V. Gligorijevic, D. Berenberg, S. Ra, A. Watkins, S. Kelow, K. Cho, and R. Bonneau. Function-guided protein design by deep manifold sampling. *bioRxiv*, 2021.

[13] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

[14] A. P. Guerreiro, C. M. Fonseca, and L. Paquete. The hypervolume indicator: Problems and algorithms. *arXiv preprint arXiv:2005.00515*, 2020.

[15] R. Hao. Deep generative models could offer the most promising developments in ai. In *Venture Beat*, 2022.

[16] G. E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002.

[17] M. Jaggi. Revisiting frank-wolfe: Projection-free sparse convex optimization. In *International Conference on Machine Learning*, pages 427–435. PMLR, 2013.

[18] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[19] Y. LeCun, S. Chopra, R. Hadsell, M. Ranzato, and F. Huang. A tutorial on energy-based learning. *Predicting structured data*, 1(0), 2006.

[20] Y. LeCun, S. Chopra, M. Ranzato, and F.-J. Huang. Energy-based models in document recognition and computer vision. In *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, volume 1, pages 337–341. IEEE, 2007.

[21] Q. Liu. Stein variational gradient descent as gradient flow. *Advances in neural information processing systems*, 30, 2017.

[22] X. Liu, X. Tong, and Q. Liu. Profiling pareto front with multi-objective stein variational gradient descent. *Advances in Neural Information Processing Systems*, 34:14721–14733, 2021.

[23] B. Paaßen, C. Gallicchio, A. Micheli, and B. Hammer. Tree edit distance learning via adaptive symbol embeddings. In *International Conference on Machine Learning*, pages 3976–3985. PMLR, 2018.

[24] D. Repecka, V. Jauniskis, L. Karpus, E. Rembeza, I. Rokaitis, J. Zrimec, S. Poviloniene, A. Laurynenas, S. Viknander, W. Abuajwa, et al. Expanding functional protein sequence spaces using generative adversarial networks. *Nature Machine Intelligence*, 3(4):324–333, 2021.

[25] K. Roose. We need to talk about how good a.i. is getting. In *The New York Times*, 2022.

[26] S. Sayin. Kaisa m. miettinen nonlinear multiobjective optimization kluwer academic publishers, 1999 isbn 0-7923-8278-1, 320 pages eur 134.00/usd 129.95. *European Journal of Operational Research*, 148(1):229–230, 2003.

[27] O. Sener and V. Koltun. Multi-task learning as multi-objective optimization. *Advances in neural information processing systems*, 31, 2018.

[28] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015.

[29] M. Welling and Y. W. Teh. Bayesian learning via stochastic gradient langevin dynamics. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 681–688. Citeseer, 2011.

[30] A.-M. Wolf Pérez, N. Lorenzen, M. Vendruscolo, and P. Sormanni. Assessment of therapeutic antibody developability by combinations of in vitro and in silico methods. In *Therapeutic Antibodies*, pages 57–113. Springer, 2022.

Appendix

5.1 Additional details on experiments

SeqCNNs For all properties and baselines we use an identical NN architecture, that is one model per protein chain, consisted of 3 Conv1D layers with kernel size 9 and padding 1, with ReLU non-linearities and penultimate layer of size 256. All EBMs were trained with contrastive training, Adam optimizer and early stopping criterion.

Hyperparameters For each multipropery sampling baseline, a hyper parameter grid was explored with random search:

- step size - $\eta \in \{1^{e-4}, 1^{e-2}, 1, 10, 40\}$
- number of steps - $k \in \{100, 200, 300, 400\}$
- noise type - $\omega \in \{\mathcal{N}, \mathcal{U}\}$

5.2 Metrics

Hypervolume HV is an indicator for evaluating the quality of sets of solutions in multi objective optimization. For a given reference point of $r = [r_1, ..., r_m]$ we have an upper bound of the objectives, such that $supx f_i(x) \leq r_i, \forall i \in [m]$. For a given set of solutions $\mathcal{X} = \{x_l\}_{l=1}^n$, a hypervolume indicator $HV(\mathcal{X})$ is a measure of the region between all f_i objectives and r:

$$HV(\mathcal{X}) = \Lambda\{(q \in \mathbb{R}^d | \exists x \in \mathcal{X} : q \in \prod_i^m [f_i(x), r_i]\})$$

where $\Lambda(\cdot)$ denotes the Lebesgue measure.

In the context of our experiments, we are interested in minimizing the energy scores per each property, and we choose a reference point $r = (1.0, 1.0, 1.0)$. We compte HV from each Pareto front to the reference point with the implementation provided in PyMoo [2].

Edit distance

5.3 Additional results from experiments

	HV(Ab-Like, Aff, Stc)	HV(Aff, Stc)	HV(Ab-Like, Stc)	HV(Ab-Like, Aff)
$\eta = 1e^{-2}$				
MGD	0.04	0.22	0.23	0.29
ls-cEBM	0	0.24	0.24	0.23
cEBM	0	0.25	0.24	0.25
pcEBM	**0.049**	**0.3**	**0.29**	**0.30**
$\eta = 1$				
MGD	0.04	0.25	0.25	0.27
ls-cEBM	0.	0.25	0.24	0.25
cEBM	0.046	0.29	0.29	0.3
pcEBM	0.041	0.27	0.27	0.29
$\eta = 40$				
MGD	0.033	0.25	0.25	0.26
ls-cEBM	0.043	0.27	0.26	0.26
cEBM	0.046	0.29	0.29	0.29
pcEBM	0.037	0.27	0.27	0.28

Table 3: Hyper volume across multiple energy score properties at different step sizes. Bold results are best per column.