# Generative Flow Networks for Discrete Probabilistic Modeling

Dinghuai Zhang, Nikolay Malkin, Zhen Liu, Alexandra Volokhova, Aaron Courville, Yoshua Bengio

## Energy-based models

**Energy-based models** (EBMs) specify a distribution over a space $\mathcal{X}$ by

$$p_\phi(\mathbf{x}) = \underbrace{\frac{1}{Z}}_{\text{partition function } Z=\sum_{\mathbf{x}\in\mathcal{X}}\exp(-\mathcal{E}_\phi(\mathbf{x}))\text{ (intractable to compute)}} \exp(-\overbrace{\mathcal{E}_\phi(\mathbf{x})}^{\text{function with learned parameters } \phi})$$

- Energy functions can impose structure and smoothness in distributions and be used as composable modules
- However, they pose learning challenges and cannot, in general, be sampled from exactly

### Fitting EBMs to maximize likelihood of a dataset

The gradient of the negative log-likelihood for a sample $\mathbf{x}$ is:

$$-\nabla \log p_\phi(\mathbf{x}) = \underbrace{\nabla\mathcal{E}_\phi(\mathbf{x})}_{\substack{\text{minimize energy of positive example}\\ \mathbf{x}\text{ from dataset}}} - \underbrace{\mathbb{E}_{\mathbf{x}_{\rm neg}\sim p_\phi}[\nabla\mathcal{E}_\phi(\mathbf{x}_{\rm neg})]}_{\substack{\text{maximize energy of negative example}\\ \mathbf{x}_{\rm neg}\text{ sampled from } p_\phi}}$$

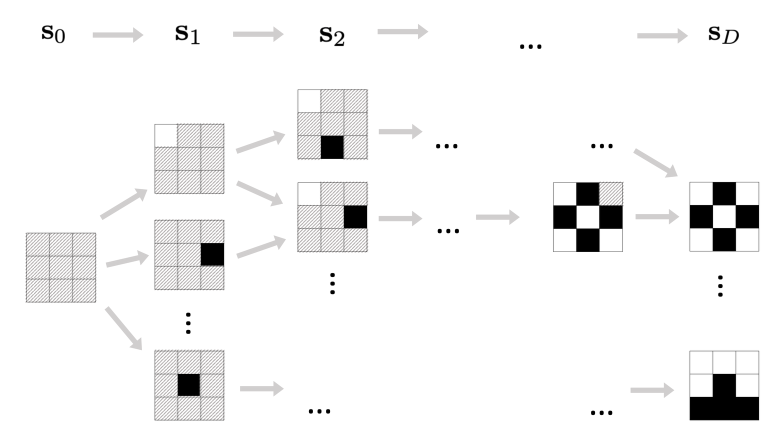
- Estimating the second term requires sampling from the distribution (intractable)
- **Contrastive divergence**-like algorithms approximate the expectation by taking $\mathbf{x}_{\rm neg}$ from a local exploration (MCMC chain) starting at $\mathbf{x}$
- **Mode-mixing problem:** Parts of $\mathcal{X}$ are poorly explored by MCMC, leading to:
  - **spurious modes** in the learned distribution, especially in high-dimensional spaces with combinatorial growth of modes
  - **difficulty of sampling** from the trained EBM by MCMC

**Motivation:** Tackle the mode-mixing problem by training an EBM jointly with a sequential sampling model (GFlowNet) that samples from $p_\phi$.

## GFlowNets for high-dimensional discrete data

**Generative flow networks** (GFlowNets, [1]): hierarchical variational inference via RL

- Stochastic sampling policies trained to generate objects $\mathbf{x}\in\mathcal{X}$ by sequences of actions
- Trained to make the marginal likelihood of sampling $\mathbf{x}$ **proportional** to a reward $R(\mathbf{x})$
  - Setting $R(\mathbf{x}) = \exp(-\mathcal{E}_\phi(\mathbf{x}))$ gives a **sampler for an EBM** $p_\phi$
  - Can be used when many sequences of actions may give the same object



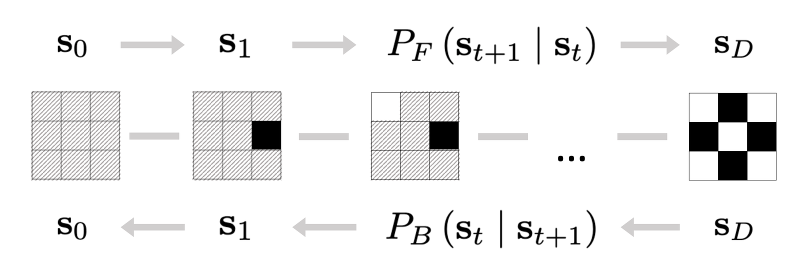

- For $D$-dimensional binary data ($\mathcal{X}=\{0,1\}^D$), actions incrementally modify a partially constructed vector (each entry is 0, 1, or void)
- Initial state is all-void; each action chooses a void entry and sets it to 0 or 1
- Stop after $D$ steps (when vector is complete)

### Training GFlowNets

- The sampling policy is given by a parametric state transition model $P_F(\mathbf{s}_{i+1}\mid\mathbf{s}_i;\theta)$
- Use the **trajectory balance (TB)** objective [4]:
  - Learn an auxiliary backward sampling model $P_B(\mathbf{s}_i\mid\mathbf{s}_{i+1};\theta)$ (erasure / choice of pixel to void) and a normalization constant $Z_\theta$
  - Optimize w.r.t. $\theta$ for trajectories $\mathbf{s}_0\to\mathbf{s}_1\to\cdots\to\mathbf{s}_D$ sampled from the policy $P_F$:

$$\mathcal{L} = \left(\log\frac{Z_\theta\prod_{i=0}^{D-1}P_F(\mathbf{s}_{i+1}\mid\mathbf{s}_i;\theta)}{R(\mathbf{s}_D)\prod_{i=0}^{D-1}P_B(\mathbf{s}_i\mid\mathbf{s}_{i+1};\theta)}\right)^2$$

- **TB training theorem:** $\mathcal{L}$ globally minimized for all trajectories $\Leftrightarrow$ marginal likelihood of reaching $\mathbf{s}_D$ proportional to $R(\mathbf{s}_D)$




## Jointly training an EBM and a GFlowNet sampler

### EB-GFN algorithm: Basic version

Algorithm for training an EBM to maximize $\sum_{\mathbf{x}_i\in\mathcal{D}}\log p_\phi(\mathbf{x}_i)$ jointly with a GFlowNet that samples from the distribution $p_\phi$:

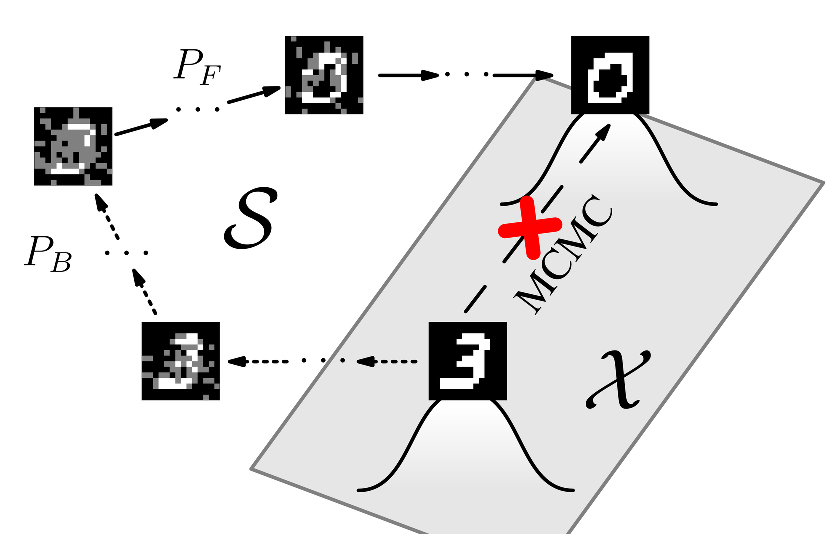
**input** Training dataset $\mathcal{D}=\{\mathbf{x}_i\}_i$.
1: Initialize GFlowNet's $P_F, P_B, Z$ with parameters $\theta$.
2: Initialize energy function $\mathcal{E}_\phi$ with parameters $\phi$.
3: **repeat**
4: Sample trajectory from GFlowNet: $\mathbf{s}_0\to\mathbf{s}_1\to\cdots\to\mathbf{s}_D$ with $\mathbf{s}_{i+1}\sim P_F(-\mid\mathbf{s}_i;\theta)$.
5: Update the GFlowNet via gradient step on $\mathcal{L}$ with reward $R(\mathbf{s}_D)=e^{-\mathcal{E}(\mathbf{s}_D;\phi)}$.
6: Uniformly sample a batch $\mathbf{x}$ from dataset.
7: Update $\phi$ with gradient of $\mathcal{E}_\phi(\mathbf{x})-\mathcal{E}_\phi(\mathbf{s}_D)$. {approximation to $-\nabla\log p_\phi(\mathbf{x})$}
8: **until** some convergence condition.

- The GFlowNet is continually updated to track changes in the energy function $\mathcal{E}_\phi$, thus providing good negative samples to approximate the log-likelihood gradient of the EBM
- Performance can be evaluated by likelihood of $\mathcal{D}$ under either the EBM or the GFlowNet sampler (both require Monte Carlo estimation or importance sampling)
- To improve convergence: also train the GFlowNet on reverse action sequences, starting from $\mathbf{x}_i$ and sampling from $P_B$

### GFlowNets for local exploration

- The GFlowNet sampler can also be used as a local search proposal: sample a trajectory of $K$ backward (erasure) steps from a complete vector $\mathbf{x}\in\mathcal{X}$ using $P_B$, then $K$ forward steps using $P_F$ to obtain another vector in $\mathbf{x}'\in\mathcal{X}$
  - This back-and-forth proposal approximates block Gibbs sampling
- Algorithm variant: Start from dataset example $\mathbf{x}\in\mathcal{D}$, make a local move to obtain $\mathbf{x}'\in\mathcal{X}$, and use $\mathbf{x}'$ as the negative sample in step 7
  - Number of steps $K$ can be annealed during training
  - Can also include a Metropolis-Hastings rejection step; we prove that the acceptance probability is always 1 if and only if the GFlowNet loss vanishes

### Why EB-GFN?



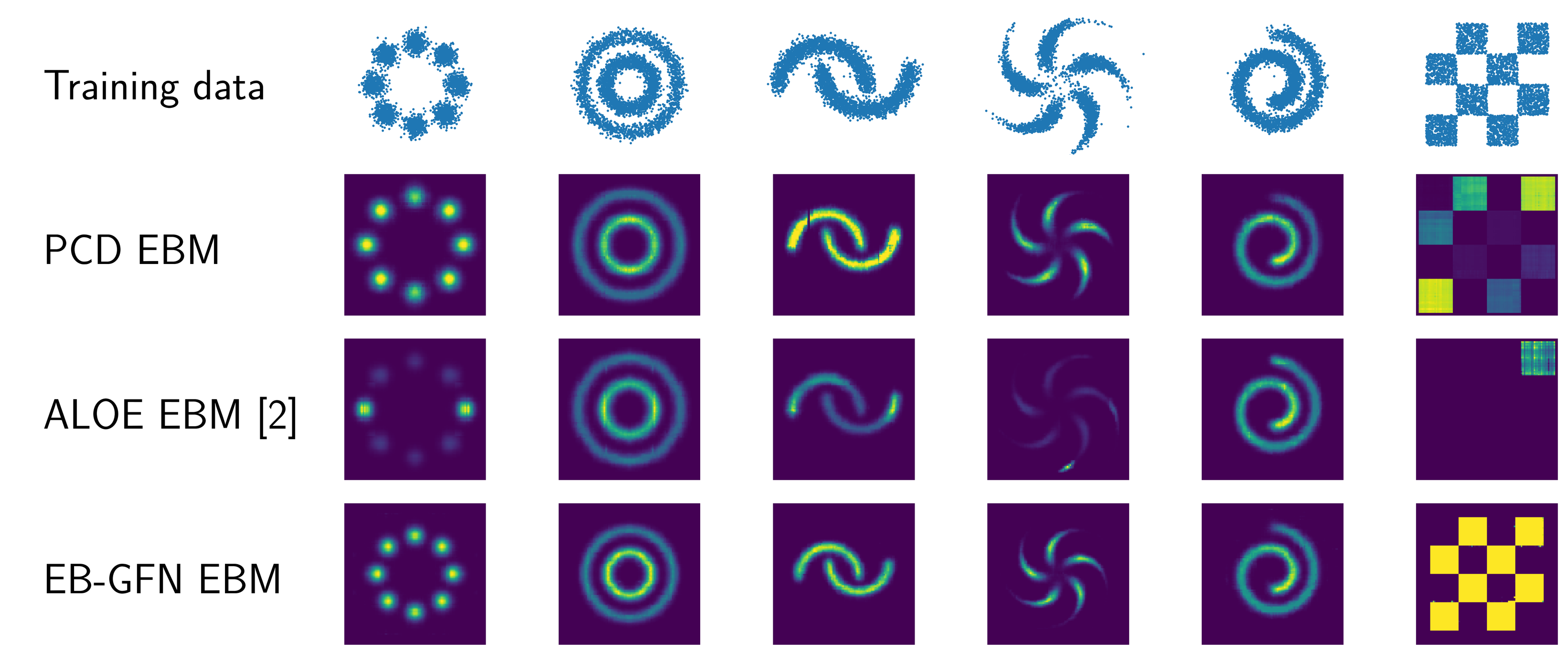

Learning a sampler jointly with the energy-based model

- **improves EBM training** due to the learned proposal distribution for producing negative samples and the auxiliary state space that encodes uncertainty
- **amortizes** the EBM: the GFlowNet can be used to sample approximately from $p_\phi$ in a fixed number of steps, without MCMC

## Selected results

### 2D synthetic data

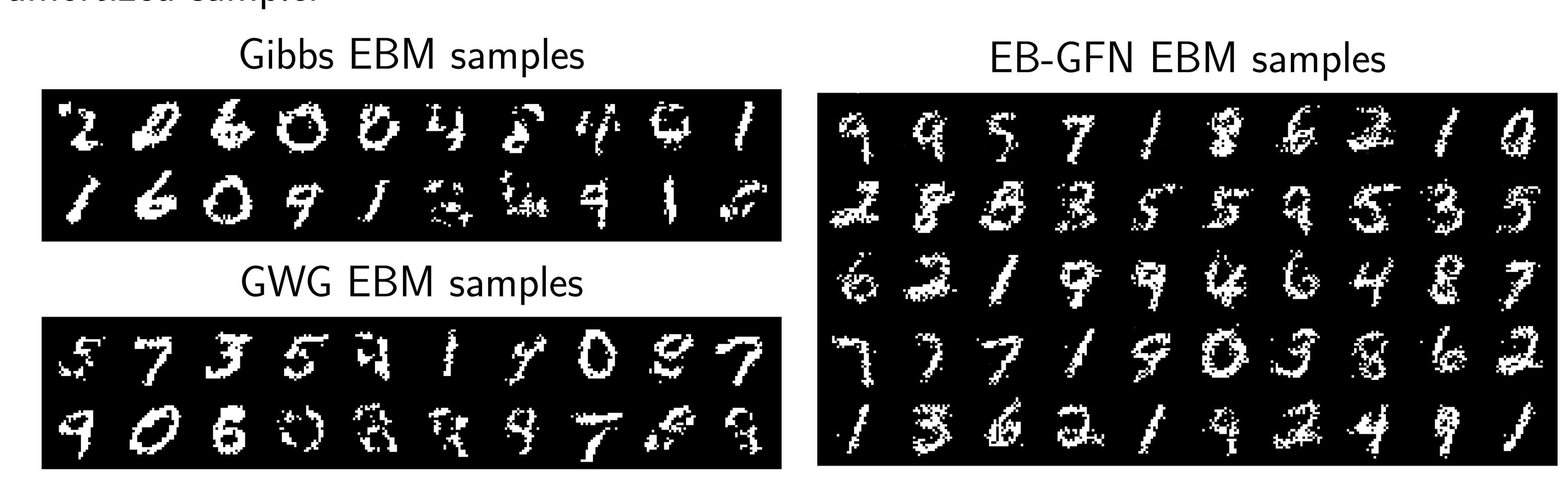
- 2-dim synthetic data quantized and represented as 32-dim binary vectors
- Energy function and GFlowNet policies are 3-hidden-layer MLPs
- EB-GFN performs well in settings with many well-separated modes




### Binary images

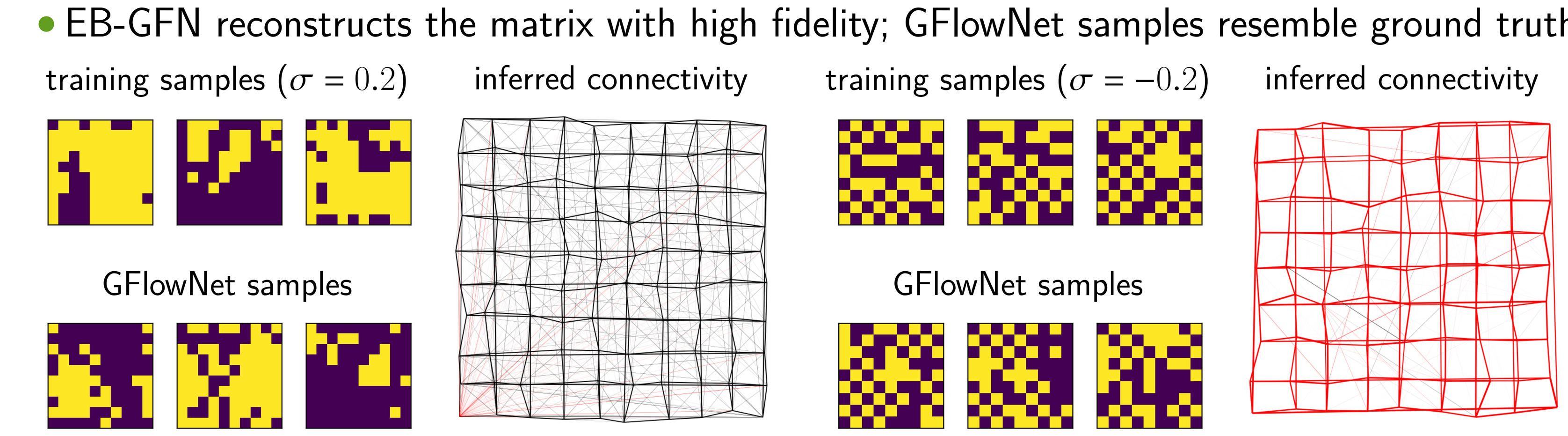
- Experiments on MNIST, Omniglot, etc.
- EB-GFN outperforms CD-based algorithms, while also yielding an amortized sampler

| | persistent CD | | |
|---|---|---|---|
| | Gibbs | GWG [3] | EB-GFN |
| MNIST bits/dim | 0.211 | 0.141 | 0.138 |




### Ising models

- Simple example of Markov random field, energy given by connectivity matrix
- Infer the connectivity matrix from 2000 ground truth samples (ground truth connectivity = adjacency matrix of grid with fixed edge weight $\sigma$)
- EB-GFN reconstructs the matrix with high fidelity; GFlowNet samples resemble ground truth




[1] Emmanuel Bengio, Moksh Jain, Maksym Korablyov, Doina Precup, and Yoshua Bengio. Flow network based generative models for non-iterative diverse candidate generation. *Neural Information Processing Systems (NeurIPS)*, 2021.

[2] Hanjun Dai, Rishabh Singh, Bo Dai, Charles Sutton, and Dale Schuurmans. Learning discrete energy-based models via auxiliary-variable local exploration. *Neural Information Processing Systems (NeurIPS)*, 2020.

[3] Will Grathwohl, Kevin Swersky, Milad Hashemi, David Kristjanson Duvenaud, and Chris J. Maddison. Oops I took a gradient: Scalable sampling for discrete distributions. *International Conference on Machine Learning (ICML)*, 2021.

[4] Nikolay Malkin, Moksh Jain, Emmanuel Bengio, Chen Sun, and Yoshua Bengio. Trajectory balance: Improved credit assignment in GFlowNets. *arXiv preprint 2201.13259*, 2022.

paper & code

GFlowNet tutorial