On Divergence Measures for Training GFlowNets

Tiago da Silva, Eliezer da Silva, Diego Mesquita

Keywords — GFlowNets, Variational Bayesian inference

TL;DR
- we revisit the relationship between GFlowNets and VI in continuous spaces,
- we empirically demonstrate that the well-known difficult of training GFlowNets by minimizing traditional divergence measures arises from the large gradient variance of the corresponding stochastic learning objectives,
- we develop statistically efficient variance-reduced gradient estimators for both the α and KL family of divergences,
- we verify that the resulting learning algorithms significantly improve upon conventional log-squared losses in terms of convergence speed,
- we re-open the once-dismissed research line focused on VI-inspired algorithmic improvements of GFlowNets.

I. Background: GFlowNets

GFlowNets are amortized algorithms for sampling from distributions over compositional objects, i.e., over objects that can be sequentially constructed from an initial state through the application of simple actions (e.g., graphs via edge-addition).

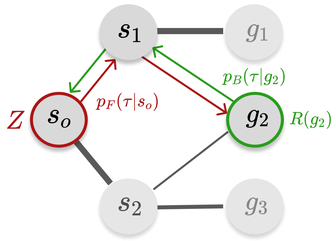

Figure 1: A GFlowNet learns a forward policy on a state graph.

In a nutshell, a GFlowNet is composed of two three ingredients.
1. An extension $\mathcal{S}$ of the target distribution support's $\mathcal{X}$.
2. A measurable pointed DAG $\mathcal{G}$ on $\mathcal{S}$ dictating how the states in $\mathcal{S}$ are connected to one another. We refer to $\mathcal{G}$ as the state graph.
3. A forward and backward policies defining the stochastic transitions within $\mathcal{G}$.

For finite state spaces, the state graph might be represented as the above DAG.

$$p_F(\tau) = \prod_{(s,s')\in\tau} p_{F(s' \mid s)} \text{ and } \sum_{\tau \rightsquigarrow g} p_F(\tau) = R(g). \tag{1}$$

To achieve this, we parameterize $p_F(\tau)$ as a neural network trained by minimizing

$$\mathcal{L}_{TB}(p_F) = \mathbb{E}\left[\left(\log \frac{p_F(\tau)Z}{p_B(\tau \mid x)R(x)}\right)^2\right]. \tag{2}$$

for a given $p_B(\tau|x)$. GFlowNets can be trained in an **off-policy** fashion and the above expectation can be under any full-support distribution over trajectories.

II. Background: Embarrassingly Parallel Inference

Reward functions can often be multiplicatively decomposed in simpler primitives,

$$R(g) = \prod_{1\leq i\leq N} R_i(g). \tag{3}$$

Each R_i may be a **subposterior** conditioned on a subsample of the data (Figure 2). Often, the R_i's cannot be disclosed due to privacy or computational constraints.

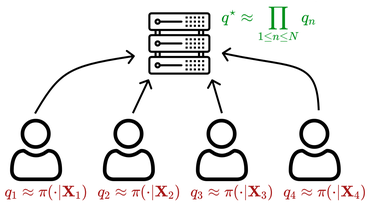

Figure 2: Approximated and embarrassingly parallel Bayesian inference.

Commonly, an approximation q_i to each R_i is locally learned and publicly shared to a centralizing server. An approximation to R, then, is obtained by approximating

$$q(g) \approx \prod_{1\leq i\leq n} q_i(g). \tag{4}$$

III. Contrastive Balance Condition

Our objective is to solve the approximation problem in Equation 4 when each q_i is a trained GFlowNet. To achieve this, we develop the **CB condition**.

Contrastive balance condition. Let p_F and p_B be the policies of a GFlowNet. Then,

$$\frac{p_F(\tau)}{R(x)p_B(\tau|x)} = \frac{p_F(\tau')}{R(x')p_B(\tau'|x')} \tag{5}$$

for all trajectories τ, τ' finishing at x, x' is a sufficient condition for ensuring that a GFlowNet samples sink nodes from $\mathcal{G}$ proportionally to R.

Differently from alternative balance conditions, the CB **does not rely** on auxiliary quantities such as Z. Clearly, enforcing CB is a sound learning objective for training GFlowNets.

Contrastive balance loss. Let p_F and p_B be the policies of a GFlowNet. Define

$$\mathcal{L}_{CB}(p_F) = \mathbb{E}\left[\left(\log \frac{p_F(\tau)}{R(x)p_B(\tau|x)} - \log \frac{p_F(\tau')}{R(x')p_B(\tau'|x')}\right)^2\right]. \tag{6}$$

Then, $p_F^\star = \operatorname{argmin} \mathcal{L}_{CB}(p_F)$ samples from $\mathcal{G}$ proportionally to R.

Our empirical analysis shows that minimizing $\mathcal{L}_{CB}$, which has minimal parameterization, often leads to faster convergence relatively to previously proposed methods.

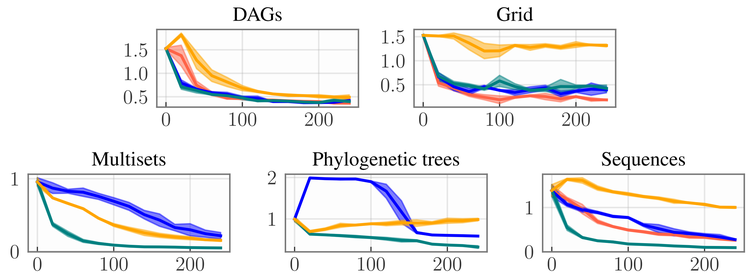

Figure 3: $\mathcal{L}_{CB}$ often outperforms $\mathcal{L}_{TB}$, $\mathcal{L}_{DB}$, and $\mathcal{L}_{DB\,\text{mod}}$ in terms of convergence speed.

IV. EP-GFlowNets and Aggregating Balance Condition

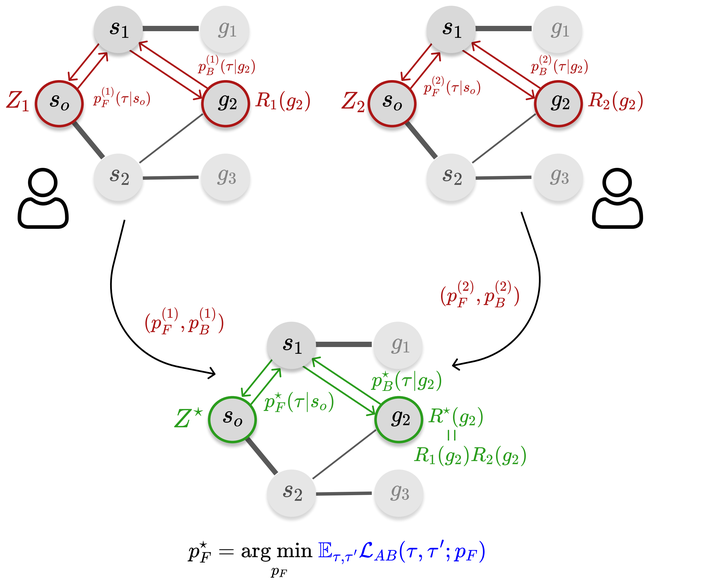

Figure 4: An overview of EP-GFlowNets for learning GFlowNets in a distributed setting. We develop a **divide-and-conquer** algorithm to train GFlowNets in a parallel. The condition below shows how to aggregate locally trained GFlowNets in a **single communication step** without directly evaluating the individual reward functions in the server.

Aggregating balance condition. Let $\left(p_F^{(1)}, p_B^{(1)}\right), \ldots, \left(p_F^{(N)}, p_B^{(N)}\right)$ be the policies of N independently trained GFlowNets. Assume each $\left(p_F^{(i)}, p_B^{(i)}\right)$ samples proportionally to R_i. If

$$\frac{\left(\prod_{1\leq i\leq N} p_F^{(i)}(\tau)\right)}{\left(\prod_{1\leq i\leq N} p_B^{(i)}(\tau \mid x)\right)} p_F(\tau')p_B(\tau \mid x) = \frac{\left(\prod_{1\leq i\leq N} p_F^{(i)}(\tau')\right)}{\left(\prod_{1\leq i\leq N} p_B^{(i)}(\tau' \mid x')\right)} p_F(\tau)p_B(\tau' \mid x'), \tag{7}$$

then the GFlowNet (p_F, p_B) samples from $\mathcal{G}$ proportionally to $\prod_{1\leq i\leq N} R_i$.

Similarly to the CB condition, we enforce the condition above by minimizing the expected log-squared difference between the left- and right-hand sides.

Aggregating balance loss. Under the conditions of Equation 7, define

$$\mathcal{L}_{AB}(p_F) = \mathbb{E}\left[\left(\log \frac{\left(\prod_{1\leq i\leq N} p_F^{(i)}(\tau)\right)}{\left(\prod_{1\leq i\leq N} p_B^{(i)}(\tau \mid x)\right)} \frac{p_F(\tau')}{p_B(\tau' \mid x')} - \log \frac{\left(\prod_{1\leq i\leq N} p_F^{(i)}(\tau')\right)}{\left(\prod_{1\leq i\leq N} p_B^{(i)}(\tau' \mid x')\right)} \frac{p_F(\tau)}{p_B(\tau \mid x)}\right)^2\right]. \tag{8}$$

Then, $\mathcal{L}_{AB}$ is globally minimized at a policy p_F sampling proportionally to $\prod_{1\leq i\leq N} R_i$.

Realistically, each GFlowNet will **only partially satisfy** their local balance conditions. Yet, we show the aggregated model can be **accurate** even under such **imperfect conditions**.

Influence of local failures. Under the notations of Equation 7, assume that

$$1 - \alpha_n \leq \min_{x\in\mathcal{G},\tau\rightsquigarrow x} \frac{p_F^{(n)}(\tau)}{p_B^{(n)}(\tau \mid x)R_n(x)} \leq \max_{x\in \operatorname{cal}\{X\},\tau\rightsquigarrow x} \frac{p_F^{(n)}(\tau)}{p_B^{(n)}(\tau \mid x)R_n(x)} \leq 1 + \beta_n \tag{9}$$

for each $n \in [[1, N]]$. Also, assume that the aggregated model satisfies Equation 7. Then, the Jeffrey divergence between the learned $\hat{R}$ and target R distributions is bounded by

$$\mathcal{D}_J\left(R, \hat{R}\right) \leq \sum_{n=1}^{N} \log\left(\frac{1+\beta_n}{1-\alpha_n}\right). \tag{10}$$

V. Empirical results on benchmark tasks

We assess the performance of EP-GFlowNets in distributed versions of set and sequence generation, grid exploration, Bayesian phylogenetic inference and structure learning.

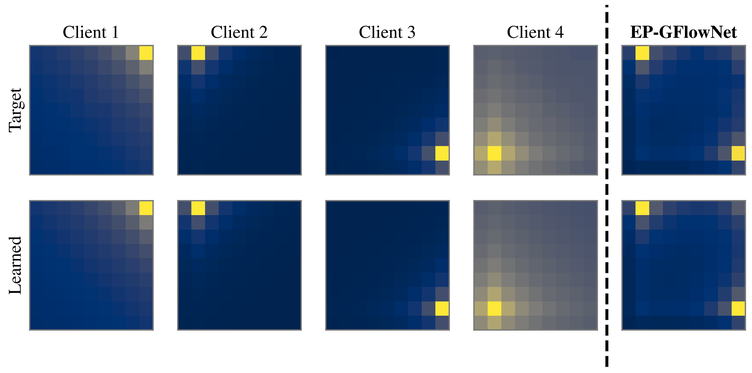

Figure 5: Results for the Grid environment showcasing the correctness of EP-GFlowNets.

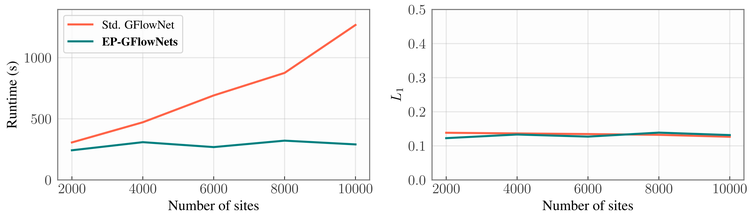

Figure 6: Results for Bayesian phylogenetic inference highlight that EP-GFlowNets can achieve a significant speed-up in learning while incurring a negligible accuracy loss.