On Divergence Measures for Training GFlowNets

Tiago da Silva **Eliezer de Souza da Silva** **Diego Mesquita**
{tiago.henrique, eliezer.silva, diego.mesquita}@fgv.br
School of Applied Mathematics
Getulio Vargas Foundation
Rio de Janeiro, Brazil

Abstract

Generative Flow Networks (GFlowNets) are amortized samplers of unnormalized distributions over compositional objects with applications to causal discovery, NLP, and drug design. Recently, it was shown that GFlowNets can be framed as a hierarchical variational inference (HVI) method for discrete distributions. Despite this equivalence, attempts to train GFlowNets using traditional divergence measures as learning objectives were unsuccessful. Instead, current approaches for training these models rely on minimizing the log-squared difference between a proposal (forward policy) and a target (backward policy) distributions. In this work, we first formally extend the relationship between GFlowNets and HVI to distributions on arbitrary measurable topological spaces. Then, we empirically show that the ineffectiveness of divergence-based learning of GFlowNets is due to large gradient variance of the corresponding stochastic objectives. To address this issue, we devise a collection of provably variance-reducing control variates for gradient estimation based on the REINFORCE leave-one-out estimator. Our experimental results suggest that the resulting algorithms often accelerate training convergence when compared against previous approaches. All in all, our work contributes by narrowing the gap between GFlowNet training and HVI, paving the way for algorithmic advancements inspired by the divergence minimization viewpoint.

1 Introduction

The approximation of intractable distributions is one of the central issues in probabilistic machine learning and modern statistics [5, 14]. Bayesian inference, for instance, relies on the assessment of intractable posterior distributions [15, 35, 42]. Variational inference (VI) methods circumvent this intractability by approximating the target distribution with a tractable parametric model. Conventionally, the problem reduces to minimizing a divergence measure, e.g., Kullback-Leibler (KL) divergence [5, 15, 39] or Renyi-α divergence [19, 30], between the variational approximation and the target.

In this context, Generative Flow Networks (GFlowNets) [3, 4, 17] were recently proposed as a family of variational approximations well-suited for distributions over compositional objects such as graphs. Notably, GFlowNets have found empirical success within various applications from causal discovery [7, 8], NLP [12], and biochemical modeling [3, 13]. In a nutshell, a GFlowNet learns an iterative generative process (IGP) [11] over an extension of the target's support that yields independent samples [3, 17] from the target distribution. Remarkably, training GFlowNets typically consists of minimizing the log-squared difference between a proposal and target distributions over the extended space via SGD [4, 23], contrasting with divergence-minimizing algorithms commonly used in VI [5, 32].

Indeed, Malkin et al. [24] suggests that trajectory balance (TB) loss training for GFlowNets leads to better approximations of the target distribution than directly minimizing the reverse and forward KL divergences, particularly in setups with sparser rewards. Nevertheless, as we highlight in Section 3, these results are a potential consequence of biased and high variance estimation of the divergence's gradients. Therefore, in Section 5, we present a comprehensive empirical investigation of the

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

minimization of well-known divergence measures (including reverse and forward KL), showing they are an effective procedure that often accelerate the training convergence of GFlowNets relative to alternatives. To achieve these results, we develop in Section 4 a collection of control variates (CVs) [26, 31] to reduce the variance without introducing bias on the estimated gradients, improving the efficiency of the optimization algorithms [34, 37]. In summary, our *main contributions* are:

1. We evaluate the performance of forward and reverse KL- [16], Renyi-α [33], and Tsallis-α [38] divergences as learning objectives for GFlowNets through an extensive empirical campaign and highlight that they frequently outperform traditional loss functions in terms of convergence speed.
2. We design CVs for the gradients of GFlowNets' divergence-based objectives that can be easily evaluated within automatic differentiation frameworks [29]. We demonstrate that these CVs drastically reduce the variance of the gradient estimators.
3. We developed a theoretical connection between GFlowNets and VI beyond the setup of finitely supported measures [24, 46], establishing results for arbitrary measurable topological spaces.

2 Revisiting the relationship between GFlowNets and VI

We start reviewing Lahlou et al. [17]'s work on continuous GFlowNets. Then, we extend Malkin et al. [24]'s results on the equivalence between GFlowNets and HVI for to the context of continuous state spaces. Finally, we describe variance reduction techniques for stochastic optimization.

Notations. Let $(\mathcal{S}, \mathcal{T})$ be a topological space with topology $\mathcal{T}$ and Σ be the corresponding Borel σ-algebra. Also, let $\nu\colon \Sigma \to \mathbb{R}_+$ be a measure over Σ and $\kappa_f, \kappa_b\colon \mathcal{S} \times \Sigma \to \mathbb{R}_+$ be transition kernels over $\mathcal{S}$. For each $(B_1, B_2) \in \Sigma \times \Sigma$, we denote by $\nu \otimes \kappa(B_1, B_2) \coloneqq \int_{B_1} \nu(\mathrm{d}s)k(s, B_2)$. Likewise, we recursively define the *product kernel* as $\kappa^{\otimes 0}(s, \cdot) = \kappa(s, \cdot)$ and, for $n \geq 1$, $\kappa^{\otimes n}(s, \cdot) = \kappa^{\otimes n-1}(s, \cdot) \otimes \kappa$ for a transition kernel κ and $s \in \mathcal{S}$. Note, in particular, that $\kappa^{\otimes n}$ is a function from $\mathcal{S} \times \Sigma^{\otimes n+1}$ to $\mathbb{R}_+$, with $\Sigma^{\otimes n+1}$ representing the product σ-algebra of Σ [2, 40]. Moreover, if μ is an absolutely continuous measure relatively to ν, denoted $\mu \ll \nu$, we write $\mathrm{d}\mu/\mathrm{d}\nu$ for the corresponding density (Radom-Nikodym derivative) [2]. Readers may consult the original work [1] for a thorough analysis of the concepts, theories, proofs, and methods discussed below.

GFlowNets. A GFlowNet is, in its most general form, built upon the concept of a *measurable pointed directed acyclic graph* (MPDAG) [17, Definition 1], which extends the concept of a *flow network* to measurable topological spaces, replacing the DAG with transition kernels specifying how the states are conencted. Formally, we denote by $\mathcal{G} = (\mathcal{S}, \kappa_f, \kappa_b, \nu)$ a measurable pointed DAG with reference forward and backward kernels κ_f and κ_b and reference measure ν on the space $\mathcal{S}$. We assume the reader is familiar with these notions and refer to [17] for a more thorough discussion.

Definition 1 (GFlowNets [17])**.** A *GFlowNet* $(\mathcal{G}, P_F, P_B, \mu)$ is composed of a MPDAG $\mathcal{G}$, a measure $\mu \ll \nu$, and σ-finite Markov kernels $P_F \ll \kappa_f$ and $P_B \ll \kappa_b$, called *forward* and *backward* policies.

Training GFlowNets. In practice, we denote by $p_{F_\theta}\colon \mathcal{S} \times \mathcal{S} \to \mathbb{R}_+$ (resp. p_B) the density of P_F (resp. P_B) relative to κ_f (resp. κ_b), which we parameterize using a neural network with weigths θ. Our objective is, for a given *target measure* $R \ll \mu$ on $\mathcal{X}$ with $r = \mathrm{d}R/\mathrm{d}\mu$, estimate the θ for which the distribution over $\mathcal{X}$ induced by $P_F(s_o, \cdot)$ matches R, i.e., for every $B \in \Sigma$,

$$\sum\nolimits_{n \geq 0} \int_{\mathcal{S}^n} p_{F_\theta}^{\otimes n}(s_o, s_{1:n}, s_f) \mathbb{1}_{s_n \in B} \kappa_f^{\otimes n}(s_o, \mathrm{d}s_{1:n}) = \frac{R(B)}{R(\mathcal{X})}.$$

Importantly, the above sum contains only finitely many non-zero terms due to the finite absorption property of κ_f [17, Definition 1]. To ensure that p_{F_θ} abides by this equation, Lahlou et al. [17] showed it suffices that the *trajectory balance condition* is concomitantly satisfied by P_F and P_B.

Definition 2 (Trajectory balance condition)**.** For all $n \geq 0$ and $\mu^{\otimes n}$-almost surely $\forall s_{1:n} \in \mathcal{S}^n$, $p_{F_\theta}^{\otimes n}(s_o, s_{1:n}, s_f) = \frac{r(s_n)}{Z_\theta} p_B^{\otimes n}(s_n, s_{n:1}, s_o)$, with Z_θ denoting the target's partition function.

To enforce the trajectory balance (TB) condition, we conventionally define a stochastic optimization problem to minimize the expected log-squared difference between its left- and right-hand members under a probability measure ξ supported on an appropriate space [4, 7, 17, 18, 21, 23, 28].

GFlowNets and VI. GFlowNets can be interpreted as HVI models by framing the forward policy $p_{F_\theta}(\tau | s_o; \theta)$ as a proposal to $\frac{r(x)}{Z} p_B(\tau | x)$. Malkin et al. [24] demonstrated that, for discrete targets,

the TB loss [17, page 6] aligns with the KL divergence in terms of expected gradients. Proposition 1 establishes that this relationship also holds for distributions over measurable topological spaces.

Proposition 1 (TB loss- and KL divergence gradients for topological spaces)**.** *Let* $\mathcal{L}_{TB}(\tau;\theta) = \left(\log {Zp_{F_\theta}(\tau|s_o;\theta)}/{r(x)p_B(\tau|x)}\right)^2$ *and* $p_B(\tau) = \frac{r(x)}{Z}p_B(s_{n-1:o}|x)$ *for* $\tau = (s_o, \ldots, s_{n-1}, x, s_f)$*. Then,*

$$\nabla_\theta \mathbb{E}_{\tau\sim P_F(s_o,\cdot)}\left[\mathcal{L}_{TB}(\tau;\theta)\right] = 2\nabla_\theta \mathcal{D}_{KL}[P_F||P_B], \tag{1}$$

where $\mathcal{D}_{KL}[p_{F_\theta}||p_B] = \mathbb{E}_{\tau\sim P_F(s_o,\cdot)}\left[\log {p_{F_\theta}(\tau|s_o;\theta)}/{p_B(\tau)}\right]$ *is the KL divergence between* P_F *and* P_B*.*

Variance reduction. A naive Monte Carlo estimator for the gradient in Equation 1 has high variance [10], impacting the efficiency of stochastic gradient descent [41]. To mitigate this, we use *control variates*—random variables with zero expectation added to reduce the estimator's variance without bias [26, 31]. This method, detailed in Section 4, significantly reduces noise in gradient estimates and pragmatically improves training convergence, as shown in the experiments in Section 5.

3 Divergence measures for learning GFlowNets

We present four divergence measures to train GFlowNets along with gradient estimators for stochastic optimization. Regardless of the learning objective, recall that our goal is to estimate θ by minimizing a discrepancy measure D between P_{F_θ} and P_B that is globally minimized when $P_{F_{\theta]}} = P_B$.

3.1 Renyi-α and Tsallis-α divergences

Renyi-α [33] and Tsallis-α [38] are families of statistical divergences including, as limiting cases, the KL divergence (Section 3.2) [25]; see Definition 3. Notably, these divergences have been successfully applied to both variational inference [19] and policy search for reinforcement learning [9].

Definition 3 (Renyi-α and Tsallis-α divergences)**.** Let $\alpha \in \mathbb{R}$. Also, let p_{F_θ} and p_B be GFlowNet's forward and backward policies, respectively. Then, the *Renyi-α divergence* between P_F and P_B is

$$\mathcal{R}_\alpha(P_F||P_B) = \frac{1}{\alpha-1}\log\int_{\mathcal{P}_\mathcal{S}} p_{F_\theta}(\tau|s_o)^\alpha p_B(\tau)^{1-\alpha}\kappa_f(s_o, \mathrm{d}\tau).$$

Similarly, the *Tsallis-α divergence* between P_F and P_B is

$$\mathcal{T}_\alpha(P_F||P_B) = \frac{1}{\alpha-1}\left(\int_{\mathcal{P}_\mathcal{S}} p_{F_\theta}(\tau|s_o)^\alpha p_B(\tau)^{1-\alpha}\kappa_f(s_o, \mathrm{d}\tau) - 1\right).$$

From Definition 3, we see that both Renyi-α and Tsallis-α divergences transition from a mass-covering to a mode-seeking behavior as α ranges from $-\infty$ to ∞. Importantly, we need only the gradients of $\mathcal{R}_\alpha$ and $\mathcal{T}_\alpha$ to learn P_F. Lemma 1 provides unbiased estimators for $\nabla_\theta\mathcal{R}_\alpha$ and $\nabla_\theta\mathcal{T}_\alpha$.

Lemma 1 (Gradients for $\mathcal{R}_\alpha$ and $\mathcal{T}_\alpha$)**.** *Let* θ *be the parameters of* p_{F_θ} *in Definition 3 and, for* $\tau \in \mathcal{P}_\mathcal{S}$*,* $g(\tau,\theta) = \left({p_B(\tau|x)r(x)}/{p_{F_\theta}(\tau|s_o;\theta)}\right)^{1-\alpha}$*. The gradient of* $\mathcal{R}_\alpha$ *wrt* θ *is*

$$\nabla_\theta\mathcal{R}_\alpha(p_{F_\theta}||p_B) = \frac{\mathbb{E}[\nabla_\theta g(\tau,\theta) + g(\tau,\theta)\nabla_\theta \log p_{F_\theta}(\tau|s_o;\theta)]}{(\alpha-1)\mathbb{E}[g(\tau,\theta)]};$$

the expectations are computed under P_F*. Analogously, the gradient of* $\mathcal{T}_\alpha$ *wrt* θ *is*

$$\nabla_\theta\mathcal{T}_\alpha(p_{F_\theta}||p_B) \overset{C}{=} \frac{\mathbb{E}[\nabla_\theta g(\tau,\theta) + g(\tau,\theta)\nabla_\theta \log p_{F_\theta}(\tau|s_o;\theta)]}{(\alpha-1)},$$

in which $\overset{C}{=}$ *denotes equality up to a multiplicative constant.*

3.2 Kullback-Leibler divergence

As mentioned earlier, the KL divergence [16] is a limiting member of the Renyi-α and Tsallis-α families of divergences, derived when $\alpha \to 1$ [30], and is the most widely deployed divergence measure in statistics and machine learning. To conduct variational inference, one regularly considers both the *forward* and *reverse* KL divergences. Similarly to Lemma 1, Lemma 2 lays out unbiased estimators for both $\mathcal{D}_{KL}[P_B||P_F]$ and $\mathcal{D}_{KL}[P_F||P_B]$ based on the REINFORCE method [41].

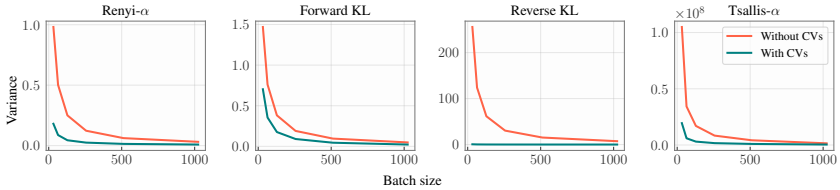

Figure 1: **Variance of the estimated gradients as a function of the trajectories' batch size.** Our control variates greatly reduce the estimator's variance, even for relatively small batch sizes.

Lemma 2 (Gradients for the KL divergence)**.** *Let θ be the parameters of P_F and $s(\tau;\theta) = \log p_{F_\theta}(\tau|s_o;\theta)$. Then, the gradient of $\mathcal{D}_{KL}[P_F||P_B]$ relatively to θ satisfies*

$$\nabla_\theta \mathcal{D}_{KL}\left[P_F||P_B\right] = \mathbb{E}_{\tau\sim P_F(s_o,\cdot)}\left[\nabla_\theta s(\tau;\theta) + \log\frac{p_{F_\theta}(\tau|s_o)}{p_B(\tau|x)r(x)}\nabla_\theta s(\tau;\theta)\right]$$

Correspondingly, the gradient of $\mathcal{D}_{KL}[P_B||P_F]$ wrt θ is

$$\nabla_\theta \mathcal{D}_{KL}[P_B||P_F] \overset{C}{=} -\mathbb{E}_{\tau\sim P_F(s_o,\cdot)}\left[\frac{p_{F_\theta}(\tau|s_o)}{p_B(\tau|x)r(x)}\nabla_\theta s(\tau;\theta)\right].$$

4 Control variates for low-variance gradient estimation

Control variates. Let $f\colon \mathcal{P}_\mathcal{S} \to \mathbb{R}^d$ be a function on the space of trajectories $\mathcal{P}_\mathcal{S}$ in $\mathcal{G}$ and π be a probability measure on the Borel σ-algebra of $\mathcal{P}_S$. We recall that a *control variate* is a function $g\colon \mathcal{P}_\mathcal{S} \to \mathbb{R}^d$ with zero mean under π and for which the Monte Carlo estimate of $\mathbb{E}_\pi[f(\tau) - a\cdot g(\tau)]$ for a *baseline* a has smaller variance than that of $\mathbb{E}_\pi[f(\tau)]$. Proposition 2 shows how to choose a.

Proposition 2 (Control variate for gradients)**.** *Let $f, g\colon \mathcal{P}_\mathcal{S} \to \mathbb{R}^d$ be vector-valued functions and π be a probability measure on $\mathcal{P}_\mathcal{S}$. Consider a* baseline *$a \in \mathbb{R}$ and assume $\mathbb{E}_\pi[g(\tau)] = 0$. Then,*

$$\arg\min_{a\in\mathbb{R}} \operatorname{Tr}\operatorname{Cov}_\pi[f(\tau) - a\cdot g(\tau)] = \frac{\mathbb{E}_\pi[g(\tau)^T(f(\tau) - \mathbb{E}_\pi[f(\tau')])]}{\mathbb{E}_\pi[g(\tau)^T g(\tau)]}. \tag{2}$$

When implementing the REINFORCE gradient estimator, the expectation we wish to estimate may be generally written as $\mathbb{E}_{P_F(s_o,\cdot)}\left[\nabla_\theta f(\tau) + f(\tau)\nabla_\theta \log p_{F_\theta}(\tau)\right]$. For the second term, we use a leave-one-out estimator [37]. For the first term, we use $\nabla_\theta \log p_{F_\theta}$ as a control variate and approximate both the numerator and denominator of Equation 2 with the delta method [36, Sec. 7.1.3]. Importantly, the resulting estimators can be efficiently implemented in an autodiff framework [6, 29] and add a negligible overhead to the GFlowNet training process. Figure 1 shows the drastic reduction in gradient variance for the problem of generating 16-sized subsets of $\{1, \ldots, 32\}$ [4, Section 5.1].

5 Training GFlowNet via divergence minimization

Experimental setup. We compare the convergence speed in terms of the rate of decrease of a measure of distributional error when using different learning objectives for a GFlowNet trained on the tasks of Bayesian phylogenetic inference [45, BPI], autoregressive sequence design [23], set generation [4], and mixture of Gaussians [17]. If the target is finitely supported, we adopt the evaluation protocols of [3, 22, 23, 27] and compute the L_1 distance between the learned p_T and target r. Otherwise, we follow [17, 43] and measure the Jensen-Shannon divergence ($\mathcal{D}_{JS}$, [20]) between p_T and r.

Results. Table 1 shows that the minimization of divergence-based objectives for GFlowNet training leads to significantly faster convergence when compared against minimizing the TB loss [23] in three-out-of-four problems. For the task of BPI, the results were not statistically distinguishable. We refer the reader to [1] for a more extensive experimental campaign encompassing a broader set of problems and a larger number of baselines [22, 44]. Remarkably, our conclusions regarding the efficacy of divergence-based objectives are preserved in this expanded analysis.

6 Conclusions

In a comprehensive range of experiments, we showed that divergence measures are effective learning objectives for training GFlowNets, achieving competitive performance to flow-based training objectives when appropriate variance reduction techniques for gradient estimation are implemented. Additionally, we established a theoretical connection between GFlowNets and HVI beyond the setting of finitely supported measures. Overall, our work highlights the potential of the once-dismissed VI-inspired schemes for training GFlowNets, paving the way for further research on the algorithmic improvement of these models based on principled approaches from the VI literature.

Table 1: Divergence minimization achieves better than or similar results relatively to minimizing TB.

	BPI	Sequences	Sets	GMs
TB	$0.22_{\pm 0.04}$	$0.28_{\pm 0.06}$	$0.07_{\pm 0.00}$	$0.31_{\pm 0.08}$
Rev. KL	$0.21_{\pm 0.04}$	$\mathbf{0.16}_{\pm 0.06}$	$\mathbf{0.03}_{\pm 0.00}$	$0.31_{\pm 0.09}$
For. KL	$0.22_{\pm 0.04}$	$0.23_{\pm 0.12}$	$\mathbf{0.03}_{\pm 0.00}$	$\mathbf{0.09}_{\pm 0.10}$
Renyi-0.5	$0.22_{\pm 0.03}$	$0.23_{\pm 0.10}$	$\mathbf{0.03}_{\pm 0.00}$	$0.19_{\pm 0.13}$
Tsallis-0.5	$0.21_{\pm 0.04}$	$0.22_{\pm 0.09}$	$\mathbf{0.03}_{\pm 0.00}$	$0.21_{\pm 0.11}$

References

[1] Anonymous. On divergence measures for training GFlownets. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[2] S. Axler. *Measure, Integration &; Real Analysis*. Springer International Publishing, 2020. ISBN 9783030331436. doi: 10.1007/978-3-030-33143-6.

[3] E. Bengio, M. Jain, M. Korablyov, D. Precup, and Y. Bengio. Flow network based generative models for non-iterative diverse candidate generation. In *NeurIPS (NeurIPS)*, 2021.

[4] Y. Bengio, S. Lahlou, T. Deleu, E. J. Hu, M. Tiwari, and E. Bengio. Gflownet foundations. *Journal of Machine Learning Research (JMLR)*, 2023.

[5] D. M. Blei and et al. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 2017.

[6] J. Bradbury, R. Frostig, P. Hawkins, M. J. Johnson, C. Leary, D. Maclaurin, G. Necula, A. Paszke, J. VanderPlas, S. Wanderman-Milne, and Q. Zhang. JAX: composable transformations of Python+NumPy programs, 2018.

[7] T. Deleu, A. Góis, C. C. Emezue, M. Rankawat, S. Lacoste-Julien, S. Bauer, and Y. Bengio. Bayesian structure learning with generative flow networks. In *UAI*, 2022.

[8] T. Deleu, M. Nishikawa-Toomey, J. Subramanian, N. Malkin, L. Charlin, and Y. Bengio. Joint Bayesian inference of graphical structure and parameters with a single generative flow network. In *Advances in Neural Processing Systems (NeurIPS)*, 2023.

[9] S. Depeweg, J. M. Hernández-Lobato, F. Doshi-Velez, and S. Udluft. Learning and policy search in stochastic dynamical systems with bayesian neural networks. *arXiv preprint arXiv:1605.07127*, 2016.

[10] A. B. Dieng, D. Tran, R. Ranganath, J. Paisley, and D. Blei. Variational inference via χ upper bound minimization. *NeurIPS*, 2017.

[11] T. Garipov, S. D. Peuter, G. Yang, V. Garg, S. Kaski, and T. S. Jaakkola. Compositional sculpting of iterative generative processes. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[12] E. J. Hu, M. Jain, E. Elmoznino, Y. Kaddar, G. Lajoie, Y. Bengio, and N. Malkin. Amortizing intractable inference in large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

[13] M. Jain, E. Bengio, A. Hernandez-Garcia, J. Rector-Brooks, B. F. P. Dossou, C. A. Ekbote, J. Fu, T. Zhang, M. Kilgour, D. Zhang, L. Simine, P. Das, and Y. Bengio. Biological sequence design with GFlowNets. In *International Conference on Machine Learning (ICML)*, 2022.

[14] M. Järvenpää and J. Corander. On predictive inference for intractable models via approximate bayesian computation. *Statistics and Computing*, 33(2), Feb. 2023. ISSN 1573-1375.

[15] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul. An introduction to variational methods for graphical models. *Mach. Learn.*, 37(2):183–233, 1999.

[16] S. Kullback and R. A. Leibler. On Information and Sufficiency. *The Annals of Mathematical Statistics*, 1951.

[17] S. Lahlou, T. Deleu, P. Lemos, D. Zhang, A. Volokhova, A. Hernández-García, L. N. Ezzine, Y. Bengio, and N. Malkin. A theory of continuous generative flow networks. In *ICML*, volume 202 of *Proceedings of Machine Learning Research*, pages 18269–18300. PMLR, 2023.

[18] E. Lau, N. M. Vemgal, D. Precup, and E. Bengio. DGFN: Double generative flow networks. In *NeurIPS 2023 Generative AI and Biology (GenBio) Workshop*, 2023.

[19] Y. Li and R. E. Turner. Rényi divergence variational inference. *NeurIPS*, 29, 2016.

[20] J. Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information theory*, 37(1):145–151, 1991.

[21] D. Liu and et al. Gflowout: Dropout with generative flow networks. In *International Conference on Machine Learning*, ICML'23. JMLR.org, 2023.

[22] K. Madan, J. Rector-Brooks, M. Korablyov, E. Bengio, M. Jain, A. C. Nica, T. Bosc, Y. Bengio, and N. Malkin. Learning gflownets from partial episodes for improved convergence and stability. In *International Conference on Machine Learning*, 2022.

[23] N. Malkin, M. Jain, E. Bengio, C. Sun, and Y. Bengio. Trajectory balance: Improved credit assignment in GFlownets. In *NeurIPS (NeurIPS)*, 2022.

[24] N. Malkin, S. Lahlou, T. Deleu, X. Ji, E. Hu, K. Everett, D. Zhang, and Y. Bengio. GFlowNets and variational inference. *International Conference on Learning Representations (ICLR)*, 2023.

[25] T. Minka. Divergence measures and message passing. Technical report, Research, Microsoft, 2005. URL https://www.seas.harvard.edu/courses/cs281/papers/minka-divergence.pdf.

[26] A. B. Owen. *Monte Carlo theory, methods and examples*. 2013.

[27] L. Pan, D. Zhang, A. Courville, L. Huang, and Y. Bengio. Generative augmented flow networks. In *International Conference on Learning Representations (ICLR)*, 2023.

[28] L. Pan, D. Zhang, M. Jain, L. Huang, and Y. Bengio. Stochastic generative flow networks. In *UAI*, volume 216 of *Proceedings of Machine Learning Research*, pages 1628–1638. PMLR, 2023.

[29] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. Pytorch: An imperative style, high-performance deep learning library, 2019.

[30] B. Poczos and J. Schneider. On the estimation of α-divergences. In *AISTATS*. PMLR, 2011.

[31] R. Ranganath, S. Gerrish, and D. M. Blei. Black box variational inference. In *AISTATS*, volume 33 of *JMLR Workshop and Conference Proceedings*, pages 814–822. JMLR.org, 2014.

[32] R. Ranganath, D. Tran, J. Altosaar, and D. M. Blei. Operator variational inference. In *NeurIPS*, pages 496–504, 2016.

[33] A. Rényi. On measures of entropy and information. In *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics*. University of California Press, 1961.

[34] L. Richter, A. Boustati, N. Nüsken, F. J. R. Ruiz, and Ö. D. Akyildiz. Vargrad: A low-variance gradient estimator for variational inference. 2020.

[35] C. P. Robert et al. *The Bayesian choice: from decision-theoretic foundations to computational implementation*, volume 2. Springer, 2007.

[36] M. J. Schervish. *Theory of statistics*. Springer Science & Business Media, 2012.

[37] J. Shi, Y. Zhou, J. Hwang, M. Titsias, and L. Mackey. Gradient estimation with discrete stein operators. *NeurIPS*, 35, 2022.

[38] C. Tsallis. Possible generalization of boltzmann-gibbs statistics. *Journal of statistical physics*, 52:479–487, 1988.

[39] M. J. Wainwright and M. I. Jordan. Graphical models, exponential families, and variational inference. *Found. Trends Mach. Learn.*, 1(1-2):1–305, 2008.

[40] D. Williams. *Probability with Martingales.* Cambridge University Press, 1991.

[41] R. J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 1992.

[42] J. S. Yedidia, W. T. Freeman, and Y. Weiss. Bethe free energy, kikuchi approximations, and belief propagation algorithms. *Advances in neural information processing systems*, 13(24), 2001.

[43] D. Zhang, R. T. Q. Chen, C.-H. Liu, A. Courville, and Y. Bengio. Diffusion generative flow samplers: Improving learning signals through partial trajectory optimization. In *The Twelfth International Conference on Learning Representations*, 2024.

[44] D. W. Zhang, C. Rainone, M. Peschl, and R. Bondesan. Robust scheduling with gflownets. In *International Conference on Learning Representations (ICLR)*, 2023.

[45] M. Y. Zhou, Z. Yan, E. Layne, N. Malkin, D. Zhang, M. Jain, M. Blanchette, and Y. Bengio. PhyloGFN: Phylogenetic inference with generative flow networks. In *The Twelfth International Conference on Learning Representations*, 2024.

[46] H. Zimmermann, F. Lindsten, J. van de Meent, and C. A. Naesseth. A variational perspective on generative flow networks. *Trans. Mach. Learn. Res.*, 2023, 2023.