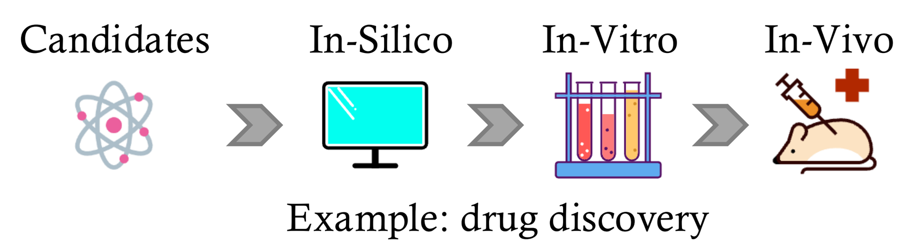
Looking Backward: Retrospective Backward Synthesis For Goal-Conditioned GFlowNets

Haoran He[1], Can Chang[2], Huazhe Xu[2], Ling Pan[1]

[1]Hong Kong University of Science and Technology [2]Tsinghua University

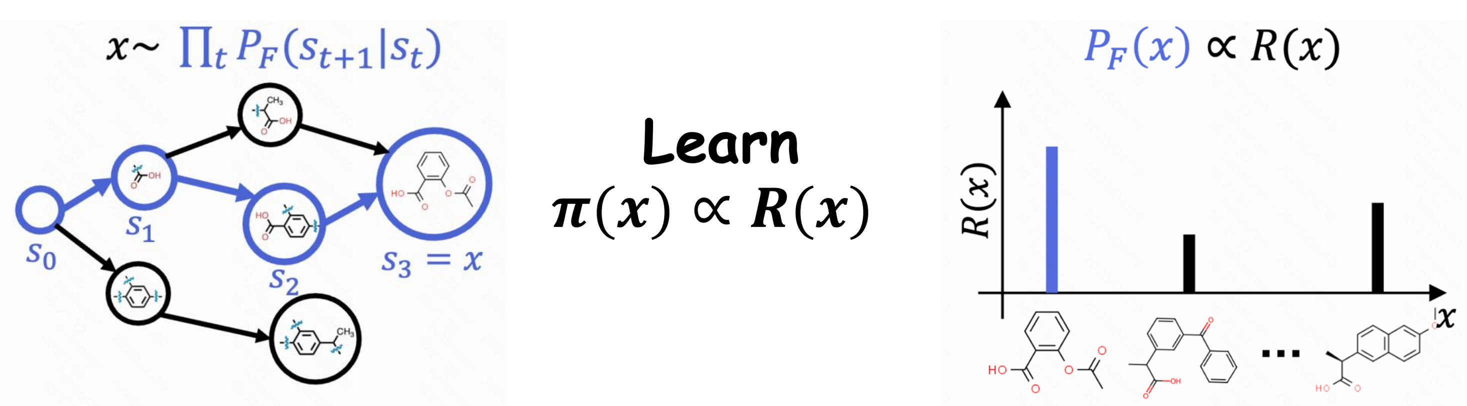
Introduction

How to learn a stochastic policy that can generate **high-reward objects** while **maintaining diversity**?



Example: drug discovery

- Generative Flow Networks (GFlowNets)



Background

- GFlowNets sample discrete objects $x \in X$ through a sequence of steps using a given set of actions $\mathcal{A}$.
- At each step of the trajectory $\tau = (s_0, s_1, \cdots, x)$, GFlowNets get a partially constructed object $s \in S$, including a starting empty state s_0 and a terminal state x

Goal-Conditioned GFlowNets

- GFlowNets [1] learn a policy π to construct x such that $\pi(x) \propto R(x)$, while Goal-conditioned GFlowNets [2] learn a policy π to construct a given goal y such that $\pi(x|y) \propto R(x,y)$.
- We formulate GC-GFlowNets as a goal-augmented DAG $\mathcal{G} = (\mathcal{S}, \mathcal{A}, \mathcal{Y}, \phi)$, where $\mathcal{Y}$ is the goal space, and $\phi: \mathcal{S} \to \mathcal{Y}$.

$$R(x,y) = \begin{cases} 1, & \|\phi(x) - y\| \le \epsilon \\ 0, & \text{otherwise} \end{cases}.$$

- Learning objective: $\forall s \to s' \in \mathcal{A}, \quad F_\theta(s|y) P_F(s'|s,y,\theta) = F_\theta(s'|y) P_B(s|s',y,\theta).$

Challenges

- Rewards are **sparse** and **binary**, as the agent only receives positive rewards upon reaching the specified goal.
- Training data is collected from interactions, which can be **limited for training** an optimal policy.

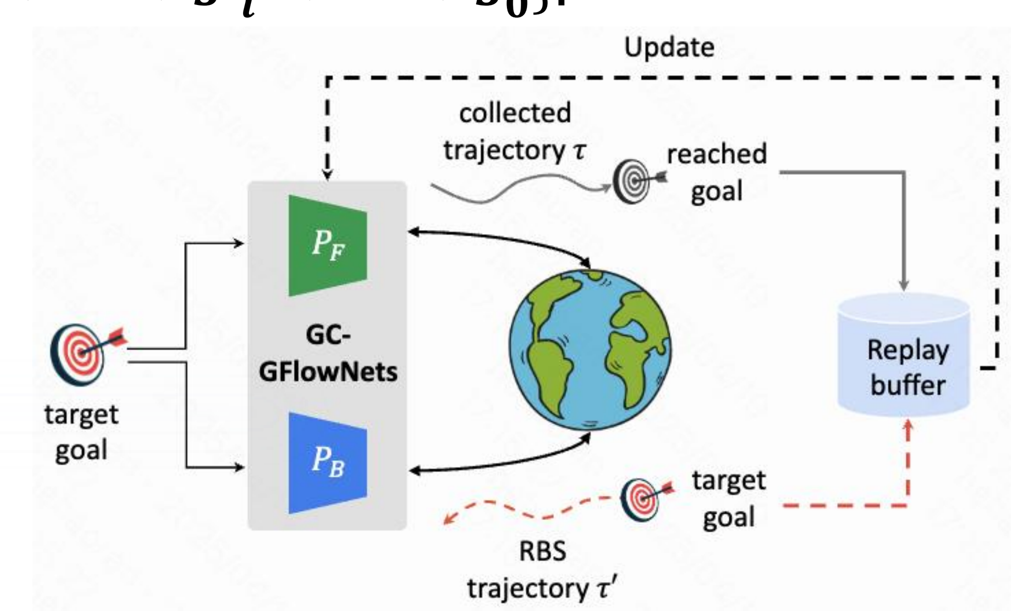
Significant issues that need to be addressed!

Proposed Method

! Synthesize backward trajectories by looking backward

! Introduce RBS: Retrospective Backward Synthesis

- Given a trajectory $\tau = \{s_0 \to \cdots \to s_i \to \cdots \to x\}$ collected by the forward policy P_F that fails to reach the goal ($x \neq y$), **RBS utilizes P_B to synthesize backward trajectory $\tau' = \{y \to \cdots \to s'_i \to \cdots \to s_0\}$.**



Advantages

✓ Leading to positive rewards as they consistently reach desired goals.

✓ Expanding data coverage.

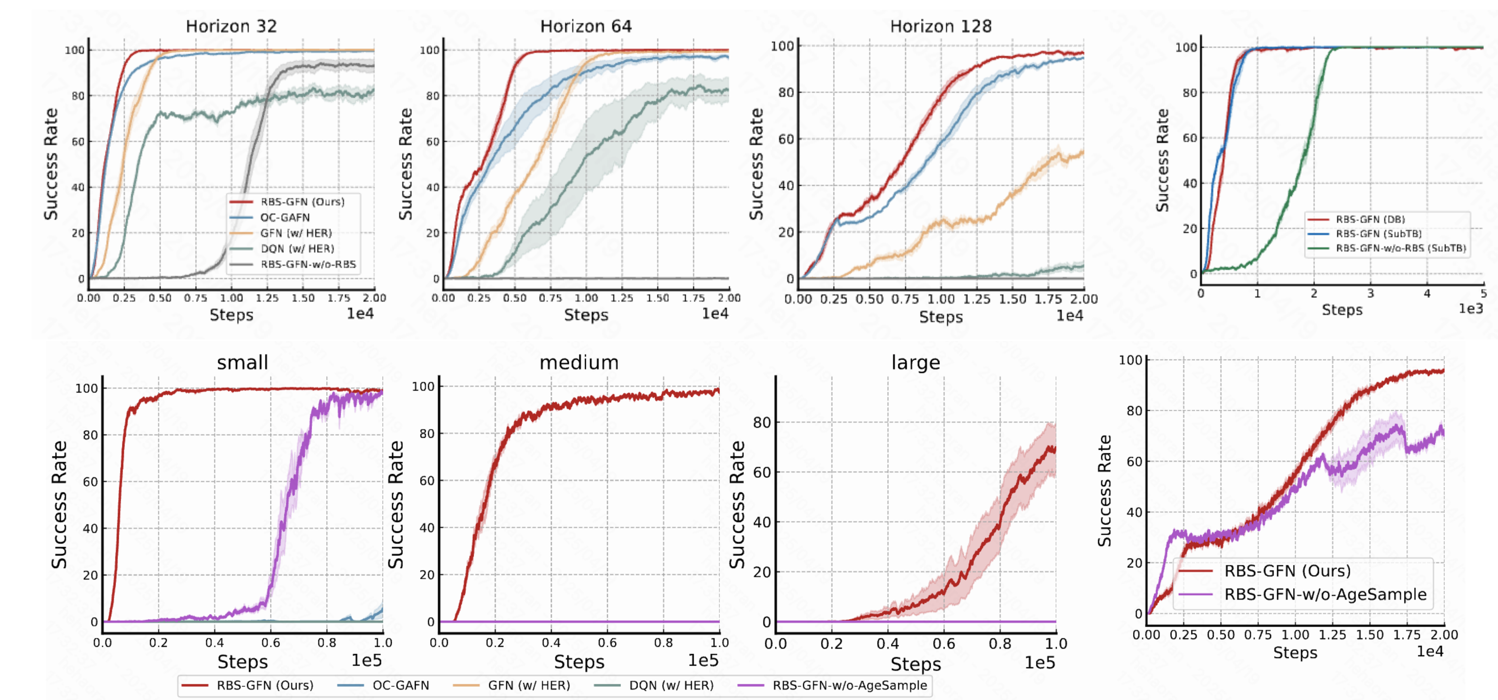
Efficient training techniques

- Age-based sampling: Assign high priority to newly collected experiences
- Backward Policy Regularization: Encourage PB to resemble a uniform distribution

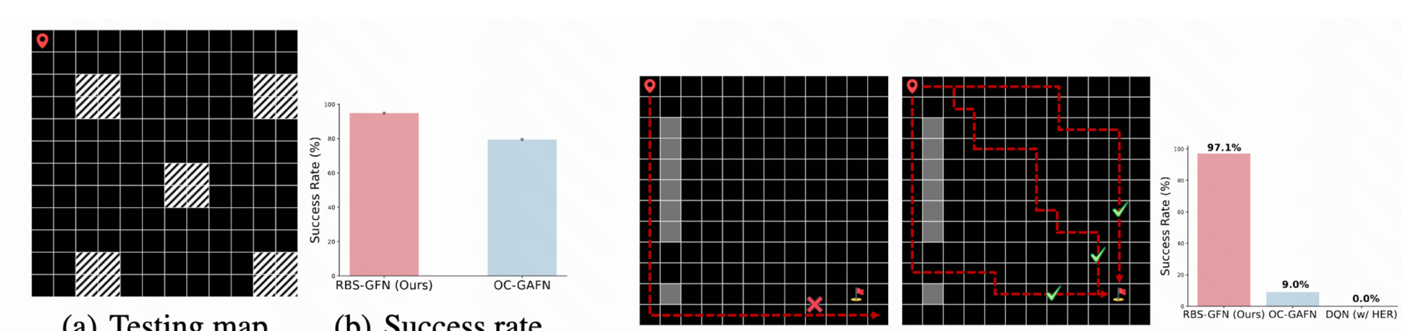
$$\longrightarrow \mathcal{L}_{\text{RBS-GFN}} = \mathcal{L}_{\text{GC-GFN}} + \gamma \times D_{\text{KL}}(P_B(\cdot|s', y, \theta) \| \mathcal{U}).$$

Selected Experiments

- RBS-GFN achieves the best goal-reaching performance across different domains (e.g., GridWorld, sequence generation).



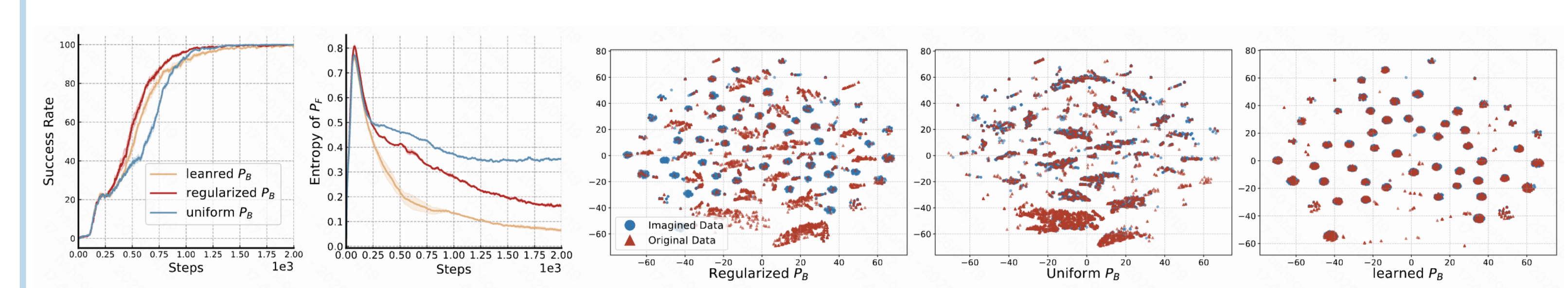
- RBS-GFN can generalize to unseen goal and unseen environments with satisfactory goal-reaching success rates



(a) Testing map (b) Success rate (a) DQN (b) RBS-GFN (c) Success rate

Visualization

- **Q:** Why does RBS enable the learning of goal-conditioned GFlowNets *effectively* and *efficiently*?
- **A:** Expand the training data with high-quality and high-diversity synthetic experiences.



(a) Quantitative results (b) t-SNE visualization

[1] Flow Network based Generative Models for Non-Iterative Diverse Candidate Generation, NeurIPS 2021 [2] Pre-Training and Fine-Tuning Generative Flow Networks, ICLR 2024