NOTE:

- **Use consistent mathematical notation throughout**
- **Include clear figures and diagrams**
- **Provide code snippets for key implementations**
- **Reference related work appropriately**
- **Writing style: maintain an academic tone while ensuring readability. Use precise technical language but explain complex concepts clearly. Include examples and visualizations to aid understanding.**

Contents

1 Introduction

NOTE:

- **Motivation and problem statement**
- **Background and reinforcement learning challenges**
- **Research objectives and contributions**
- **Thesis structure overview**

1.1 Motivation

Many real-world applications present a fundamental challenge that current reinforcement learning (RL) methods struggle to address effectively: the problem of delayed and sparse rewards.

> **Delayed and Sparse Rewards:** Learning scenarios where meaningful feedback signals (rewards) are provided only far after a long sequence of actions, and where most actions yeild no immediate feedback.
>
> *Example: In drug discovery, the effectiveness of a designed molecule can only be evaluated after its complete synthesis, with no intermediate feedback during the design process.*

Consider, for instance, the process of drug design, where a reinforcement learning agent must make a series of molecular modifications to create an effective compound. The value of these decisions — the drug's efficacy — can only be assessed once the entire molecule is complete. Similarly, in robotics tasks like assembly or navigation, success often depends on precise sequences of actions where feedback is only available upon completing the entire task.

Traditional reinforcement learning algorithms face two critical limitations in such environments:

1. **Credit Assignment:** When rewards are delayed, the algorithm struggles to correctly attribute success or failure to specific actions in a long sequence. This is analogous to trying to improve a chess strategy when only knowing the game's outcome, without understanding which moves were actually decisive.

2. **Exploration Efficiency:** With sparse rewards, random exploration becomes highly inefficient. An agent might need to execute precisely the right sequence of actions to receive any feedback at all, making random exploration about as effective as searching for a needle in a haystack.

This thesis investigates a novel approach to addressing these challenges through the comparison of two promising methodologies: **Generative Flow Networks** (GFlowNets) [CITATION NEEDED] and **Bayesian Exploration Networks** (BEN) [CITATION NEEDED]. These approaches represent fundamentally different perspectives on handling uncertainty and exploration in reinforcement learning:

1. GFlowNets frame the learning process as a flow network, potentially offering more robust learning in situations with multiple viable solutions.

2. BENs leverages Bayesian uncertainty estimation to guide exploration more efficiently, potentially making better use of limited feedback.

By comparing these approaches, we aim to understand their relative strengths and limitations in environments with delayed and sparse rewards, ultimately contributing to the development of more efficient and practical reinforcement learning algorithms. Our investigation focuses specifically on examining these methods in carefully designed environments that capture the essential characteristics of delayed and sparse reward scenarios while remaining tractable for systematic analysis.

1.2 Research Objectives and Contributions

This thesis aims to advance our understanding of efficient learning in sparse reward environments through three primary objectives:

1. **Comparative Analysis:** Conduct a rigorous empirical comparison between GFlowNets and Bayesian Exploration Networks in standardized environments with delayed rewards.

2. **Hypothesis Testing:** Investigate whether BEN's Bayesian exploration strategy leads to more efficient learning compared to GFlowNets in highly delayed reward scenarios, particularly during early training stages.

3. **Algorithmic Understanding:** Analyze the underlying mechanisms that drive performance differences between these approaches, focusing on their handling of uncertainty and exploration.

The contributions of this work include:

- A comprehensive empirical evaluation using the n-chain environment with varying degrees of reward delay.

- Insights into the relative strengths and limitations of Bayesian and flow-based approaches to exploration.

- Implementation and analysis of both algorithms with comparisons.

1.3 Thesis and Structure

The remainder of this thesis is structured as follows:

Section 2: Background and Related Work provides the theoretical foundations of reinforcement learning and explores existing approaches to handling sparse rewards. This chapter establishes the mathematical framework and notation used throughout the thesis.

Section 3: Theoretical Framework presents our hypothesis and analytical approach. We develop the mathematical foundations for comparing GFlowNets and BEN, with particular attention to their theoretical guarantees and limitations.

Section 4: Experimental Design details our testing methodology, including environment specifications, evaluation metrics, and implementation details. This chapter ensures reproducibility and clarity in our experimental approach.

Section 5: Results and Analysis presents our findings, including both quantitative performance metrics and qualitative analysis of learning behaviors. We examine how each algorithm handles the exploration-exploitation trade-off and adapts to varying levels of reward sparsity.

Section 6: Conclusion summarizes our findings, discusses their implications for the field, and suggests directions for future research.

2 Background and Related Work

NOTE:

- **Fundamentals of reinforcement learning**
  - **Markov Decision Processes**
  - **Q-learning and temporal difference methods**
- **Sparse reward challenges**
- **Survey of existing approaches**
  - **GFlowNets**
  - **Deep exploration networks (BEN)**
  - **Comparison of methodologies**

3 Theoretical Framework

NOTE:

- **Hypothesis development**
- **Problem formulation**
  - **Mathematical notation and definitions**
  - **Assumptions and constraints**
- **Proposed solution approach**

4 Experimental Design

NOTE:

- Test environments
  - N-Chain implementation
  - GridWorld setup
- Evaluation metrics
  - Sample efficiency
  - Final performance
  - Exploration behavior
- Implementation details
  - Network architectures
  - Training procedures
  - Hyperparameter selection

5 Results and Analysis

NOTE:

- Quantitative results
  - Performance comparisons
  - Statistical analysis
- Qualitative analysis
  - Exploration patterns
  - Learning behavior
- Discussion of findings

6 Conclusion

NOTE:

- Summary of contributions
- Key insights
- Future work directions